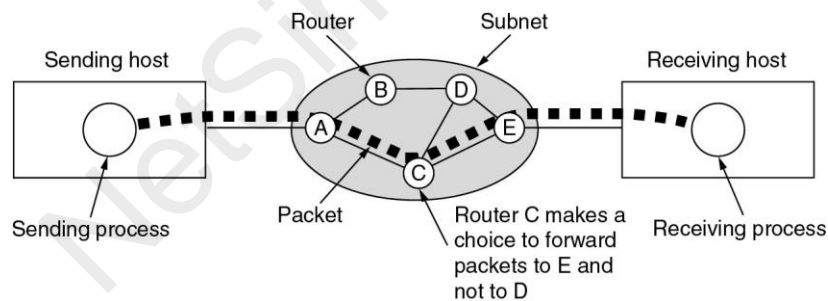


مسیریابی در شبکه های ادهاک

۱-۳ مقدمه

فرایند انتقال اطلاعات از یک منبع به یک مقصد معین را مسیریابی^۱ بسته های اطلاعات در شبکه می نامند. این عمل وظیفه اصلی لایه شبکه است. ایجاد این مسیر در مهمترین حالت به الگوریتم مسیریابی مورد استفاده بستگی دارد. در همه شبکه ها - اعم از شبکه های ثابت و سیار - مسیریابی یکی از مهمترین مسائل آن شبکه است و پروتکل های مختلفی برای این کار تهیه شده است. پروتکل های مسیریابی دو وظیفه عمده دارند، اول انتخاب مسیرهای بهینه بین گره های مبدأ و مقصد و دوم تحویل پیام ها از مبدأ به مقصد. در شبکه های ادهاک مسیریابی نهایتاً منتهی به حل مسئله یافتن کوتاهترین مسیر می شود [1]. هر مسیریاب قادر به ارائه خدمات دیگری نیز می باشد که برخی از مهمترین آنها عبارتند از: مرتب سازی ترافیک، فیلترینگ امنیتی، مدیریت شبکه و نگهداری منابع شبکه [1].



شکل ۱-۳: ارسال یک بسته و مسیریابی و رسیدن بسته به مقصد

^۱. Routing

۳- ۲ اجزای مسیریابی

مسیریابی با دو فعالیت اساسی سر و کار دارد: تخمین بهینه مسیرهای مسیریابی و حمل کردن بسته‌های اطلاعاتی [2]. الگوریتم‌های مسیریابی آن بخش از نرم افزار لایه شبکه‌اند که مسئولیت تصمیم‌گیری در خصوص خط خروجی که یک بسته ورودی باید بر روی آن ارسال شود، را دارا می‌باشند. انواع مختلفی از الگوریتم‌های مسیریابی وجود دارد. هر الگوریتم یک فشردگی متفاوتی در شبکه و منابع مسیریابی دارد. در نتیجه، الگوریتم‌های مسیریابی از معیارهای گوناگونی که بر محاسبه مسیرهای بهینه تاثیر می‌گذارد، استفاده می‌کنند [1]. در شبکه‌های ادهاک یک ارتباط مخابراتی مستقیم بین هر دو گره از شبکه با شرایط انتشار رادیویی و محدودیت‌های توان ارسال گره‌ها صورت می‌گیرد. اگر هیچ مسیر مستقیمی بین مبدأ و مقصد وجود نداشته باشد، از مسیریابی چندگذره استفاده می‌شود و در صورتیکه هیچ گونه ساختار ثابت شبکه‌ای موجود نباشد، هر گره در کنار ارسال و دریافت داده، تعدادی از گره‌های همسایه خود را جهت مسیریابی بر مبدأ خود، به کار گرفته و فعال می‌نماید، هر چند کاربرهای آن این موضوع را متوجه نمی‌شوند. یک بسته تا هنگامی که به مقصد خود نرسیده باشد، گره‌های مسیر را به عنوان مسیر یاب استخدام می‌نماید. مسیریابی در این شبکه از طرف خود گره‌ها حمایت می‌شود [1]. اما مساله این است که مسیر گره مبدأ و مقصد چگونه معلوم می‌شود. مسیریابی در شبکه‌های ادهاک بدلیل اینکه شکل شبکه در حال تغییر است، از طریق گره‌ها صورت می‌گیرد. گره‌ها سیار هستند و می‌توانند شبکه را ترک کرده و یا وارد آن شوند، بنابراین شکل شبکه همواره با سرعت زیادی در حال تغییر است و استفاده از روش‌های جدولی مسیریابی متداول، در چنین شرایطی مناسب نمی‌باشد. از آنجایی که در این نوع شبکه هیچ کنترل مرکزی جهت نگهداری اطلاعات به هنگام شکل شبکه وجود ندارد، نیاز به استفاده از الگوریتم‌های توزیع شده وفقی کاملاً حس می‌شود. اطلاعات کنترلی جهت شناسایی حالت جاری شبکه، بین گره‌ها مبادله می‌شود [1].

از مشکلات دیگر این نوع ارتباط در مسیریابی که ناشی از بی‌سیسم بودن آن می‌باشد می‌توان به پهنای باند محدود، تداخل و توان مصرفی اشاره کرد. تداخل باعث می‌شود که مسیر یاب اطلاعات اشتباه را به مسیرهای غیر منتظره و بلااستفاده بفرستد. همانطور که قبلاً ذکر شد، به دلیل اینکه انرژی گره‌ها محدود است، پروتکل اجرایی با مدنظر گرفتن این موضوع، باید توان مصرفی گره‌ها را به حداقل ممکن برساند [1].

پروتکل‌های متنوع مسیریابی برای استفاده در شبکه‌های ادهاک وجود دارد. مهمترین این پروتکل‌ها از سوی گروه شبکه‌های ادهاک موبایل وابسته به سازمان IETF معرفی شده است. این پروتکل‌ها برای شبکه‌های متجانس و سیار ادهاک که براساس IP بنا شده‌اند، طراحی گردیده است. هر گره‌ای توانایی تشخیص هویت را دارد. فرض

می‌شود هر گره دارای یک آدرس یکتا است. در این پروتکل‌ها تنها معیار در مسیریابی حداقل گذر بین مبدأ و مقصد معرفی می‌شود. بنابراین پارامترهای دیگر نظیر تأخیر مسیر، انرژی مصرفی، توزیع مناسب توان بین پایانه‌ها، تعدیل بار شبکه و کیفیت سرویس در نظر گرفته نشده‌اند. این پروتکل‌ها تنها بر یک مسیریابی سریع و نگهداری مسیر تحت حداقل هزینه تأکید دارند.

۳-۳ اهداف طراحی

الگوریتم‌های مسیریابی اغلب یک یا چندین مورد از هدفهای طراحی زیر را دارند:

- بهینگی^۲
- سادگی و سرباره کم^۳
- نیرومندی و پایداری^۴
- همگرایی سریع^۵
- انعطاف پذیری^۶ [2]

۱-۳-۳ بهینگی

حداکثر کردن بازدهی شبکه و قابلیت الگوریتم مسیریابی برای انتخاب بهترین مسیر را که به استانداردهای مورد استفاده برای محاسبه بستگی دارد، بهینگی گویند. به طور طبیعی، پروتکل‌های مسیریابی باید دقیقاً الگوریتم-های محاسبه استاندارد خود را تعریف کنند.

۲-۳-۳ سادگی و سرباره کم

². optimality
³. simplicity and low overhead
⁴. robustness and stability
⁵. rapid convergence
⁶. flexibility

الگوریتم‌های مسیریابی تا حد ممکن باید طوری طراحی شوند که ساده باشند. به عبارت دیگر، الگوریتم مسیریابی باید کارایی‌اش را بطور موثر با کمترین نرم افزار و سرباره عرضه کند. بهره‌وری مخصوصا زمانی مهم است که نرم افزار الگوریتم مسیریابی باید در یک گره با منابع فیزیکی محدود اجرا شود.

۳-۳-۳ نیرومندی و پایداری

الگوریتم‌های مسیریابی باید نیرومند باشند، این به معنی آن است که آنها باید در رویارویی غیر عادی یا اوضاع پیش‌بینی نشده، از قبیل خرابی‌های سخت افزاری، شرایط بار زیاد و پیاده‌سازی معیوب کار خود را به طور صحیح انجام دهند. چون مسیر یاب‌ها در نقاط پیوند شبکه قرار داده می‌شوند، وقتی آنها خراب شوند، می‌توانند باعث مشکلات قابل توجهی شوند. بهترین الگوریتم‌های مسیریابی اغلب آنهایی هستند که پایداری خود را در برابر آزمون گذشت زمان ثابت کرده‌اند و در شرایط متنوع در شبکه همچنان پایدار مانده‌اند.[1]

۴-۳-۳ همگرایی سریع

الگوریتم‌های مسیریابی باید به سرعت همگرا شوند. همگرایی فرآیند توافق همه مسیر یاب‌ها روی مسیر بهینه است. وقتی که یک رویداد باعث خرابی یا غیر قابل دسترس بودن مسیرها می‌شود، مسیر یاب‌ها پیام‌های به روز کننده مسیریابی را برای بررسی مجدد مسیرهای شبکه پخش می‌کنند. در نتیجه باعث تجدید محاسبه مسیرهای بهینه و مطابقت همه مسیر یاب‌ها با این مسیرها می‌شود [1].

۴-۳-۳ انعطاف پذیری

الگوریتم‌های مسیریابی باید قابل انعطاف باشند بدین معنی که آنها باید به سرعت و با دقت خود را با تنوع در وضعیت شبکه وفق دهند. برای مثال فرض کنید که یک قسمت شبکه خراب شده است، بطوریکه الگوریتم‌های مسیریابی زیادی از این مشکل آگاه شده‌اند، آنها به سرعت می‌خواهند بهترین مسیر بعدی را بجای مسیرهایی که به صورت عادی از آن استفاده می‌کردند انتخاب کنند. الگوریتم‌های مسیریابی می‌توانند برای وفق دادن تغییرات در پهنای باند شبکه، اندازه صف مسیریابی، تاخیر شبکه و تغییرات دیگر برنامه‌ریزی کنند [2].

۴-۳ (۴-۳) ارسال بسته‌ها در شبکه ادهاک

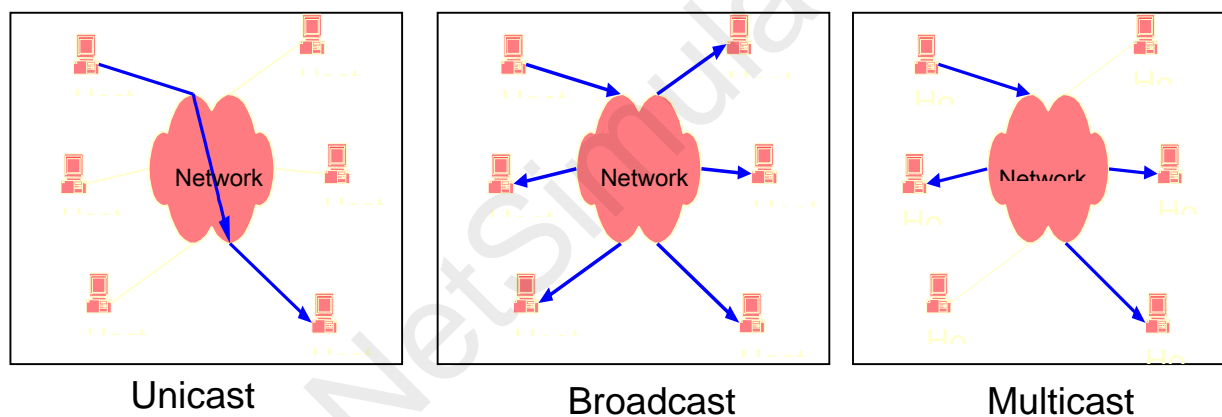
به سه روش زیر ارسال بسته‌ها در شبکه ادهاک انجام می‌شود:

۱. تک پخشی^۷

۲. چند پخشی^۸

۳. انتشاری^۹

در مدل تک پخشی یک گره بسته اطلاعات را فقط به یک گره ارسال می‌کند. به عبارت دیگر مقصد نهایی دریافت پیام تنها یک گره است. در مدل چند پخشی یک گره به چند گره از پیش تعیین شده ارسال می‌کند و در مدل انتشاری یک گره بسته را به تمام گره‌های شبکه ارسال می‌کند. شکل ۱-۳ انواع روش‌های ارسال بسته‌ها را در شبکه‌های ادهاک را نشان می‌دهد. از مهمترین معیارهای طبقه‌بندی پروتکل‌های مسیریابی ادهاک زمان‌بندی می‌باشد. بر این اساس پروتکل‌ها به دو دسته جدولی و نیازی طبقه‌بندی می‌شوند. [2]



⁷Uni cast

⁸ multicast

⁹ broadcast

شکل ۳-۲: انواع روش‌های ارسال بسته‌ها در شبکه‌های ادهاک

۳-۴) پروتکل‌های جدولی^{۱۰}

به آن دسته از پروتکل‌هایی گفته می‌شود که به صورت پیوسته شکل شبکه را با استفاده از تبادل اطلاعات گره‌های شبکه، به‌روزرسانی می‌نماید. بنابراین در زمان نیاز، مقاصد همواره مشخص هستند [۱]. خانواده پروتکل‌های بردار فاصله نمونه‌هایی از این نوع مسیریابی می‌باشند. اولین روشهایی که در بحث مسیریابی شبکه‌های ادهاک مورد توجه قرار گرفتند، از این نوع پروتکل بوده است. بزرگترین مشکلی که ضمن بکارگیری پروتکل‌های مسیریابی جدولی در شبکه‌های ادهاک بوجود می‌آید، ناشی از تغییر پیوسته شکل شبکه است. بنابراین اطلاعات شکل شبکه باید مرتباً انتشار یابد که این خود هزینه سنگینی را در منابعی چون پهنای باند، توان باتری‌ها و زمان پردازش، ایجاد می‌کند. همچنین به دلیل الگوهای مختلفی که برای یک شبکه می‌تواند وجود داشته باشد، گاهی تنها درصدی از مسیریاب‌ها استفاده می‌شود و بقیه بی‌دلیل، اطلاعات خود را در شبکه منتشر می‌کنند. با این حال این پروتکل‌ها برای آن دسته از شبکه‌های ادهاک که سرعت تغییراتشان به اندازه کافی کم باشد، قابل استفاده هستند. این پروتکل‌های مسیریابی روشهای متفاوتی برای انتشار اطلاعات تغییر یافته توپولوژی شبکه و تعداد جداول لازم مرتبط با مسیریابی دارند. بعضی از مثال‌های این پروتکل‌ها DSDV، CGSR^{۱۱}، WRP^{۱۲}، FSR^{۱۳}، GSR^{۱۴} و HSR^{۱۵} می‌باشند. در الگوریتم‌های جدولی همیشه تعداد مشخصی انتقال اطلاعات کنترلی در حال انجام است، حتی اگر برای مدتی اصلاً هیچ انتقال داده‌ای در شبکه انجام نگیرد. مشخص است که چنین شیوه‌ای هزینه زیادی را متحمل خواهد شد. [3][4][6]

۳-۶) پروتکل‌های نیازی^{۱۶}

مسیریابی بر حسب تقاضا از جدیدترین طرحهای مسیریابی‌های بیسیم راه یافته به کلاس مقیاس پذیر است. که مبنی بر یک خط‌مشی پرسش و پاسخ می‌باشد [۱]. برخلاف گروه قبلی، گره‌ها در چنین پروتکل‌هایی به

¹⁰. Proactive protocol

¹¹. Cluster Gateway Switch Routing

¹². Wireless Routing Protocol

¹³. Fisheye State Routing

¹⁴. Global State Routing

¹⁵. Hierarchical State Routing

¹⁶. Reactive protocol

صورت پیوسته تغییر شکل شبکه را متذکر نمی‌شوند و شکل شبکه تنها به کاربرانی که قصد استفاده یا مسیریابی را دارند معرفی می‌شود. کشف مسیر معمولاً با جریان درخواست مسیر^{۱۷} در شبکه انجام می‌شود و گره‌ای که در مسیر مقصد قرار دارد، با ارسال پاسخ مسیر^{۱۸} به گره منبع با استفاده از لینک معکوس، رسیدن درخواست را اطلاع می‌دهد و هنگامی که یک مسیریاب در شبکه استفاده نمی‌شود، از جدول مسیریابی کنار گذاشته می‌شود. هنگامی که استفاده از یک مسیریاب در شبکه مورد نیاز باشد، پیام "درخواست مسیر" در شبکه منتشر می‌شود و این پیام توسط مبدأ یا گره‌ای در نزدیکی آن دریافت می‌شود. آن گره نیز یک پیام "پاسخ مسیر" به سوی گره اولیه روانه می‌کند. AODV^{۱۹}، DSR^{۲۰}، TORA^{۲۱}، LMR^{۲۲} و ABR^{۲۳} مثالهایی از پروتکل‌های نیازی^{۲۴} هستند. هدف این پروتکل‌ها این است که با کشف تنها مسیرهای درخواست شده و مورد نیاز، هزینه را حداقل کنند. این دسته از پروتکل‌ها باعث کاهش چشمگیری در مصرف منابع شبکه و هزینه‌های آن خواهند شد ولی در عوض تأخیر در هنگام برقراری اتصال را افزایش می‌دهند.

در واقع برای کاهش سرباره نگهداری اطلاعات پروتکل‌های جدولی، از این پروتکل‌ها استفاده می‌شود. سرباره کشف مسیر هنگامی که لینک معکوس داشته باشیم با $O(N+M)$ رشد کرده و در لینک‌های تک جهتی $O(2N)$ می‌باشد^{۲۵}. روشهای متفاوتی برای کشف مسیرها در الگوریتم‌های نیازی وجود دارد. بیشتر الگوریتم‌ها از یک طرح مشتق شده از مسیریابی پل (LAN) یعنی کشف مسیر بوسیله یادگرفتن وارونه استفاده می‌کنند. در حالت کلی پروتکل‌های نیازی به دو دسته مسیریابی منبع و پرش به پرش تقسیم‌بندی می‌شوند [3][4][5][7]

۷-۳) مسیریابی پرش به پرش

در مسیریابی پرش به پرش هر بسته داده تنها آدرس مقصد و آدرس پرش بعدی را حمل می‌کند. از این رو هر گره واسط در مسیر مقصد از جدول مسیریابی خود برای انتقال به جلو بسته داده در جهت مقصد استفاده می‌کند. منبع در جستجوی مسیر، یک سوال در شبکه شناور می‌کند. گره‌های عبوری به محض دریافت سوال،

17. Route request

18. Route reply

19. *ad hoc* On-demand Distance Vector

20. Dynamic Source Routing

21. Temporally Ordered Routing Algorithm

22. Lightweight Mobile Routing

23. Associativity Based Routing

24. On demand

25. تعداد گره‌ها در مسیر پاسخ M تعداد گره‌های شبکه N،

مسیر را تا منبع یاد می‌گیرند و مسیر را در جدول ارسال وارد می‌کنند. مقصد مورد نظر سرانجام سوال را دریافت می‌کند و بنابراین می‌تواند با استفاده از ردیابی مسیر سوال، پاسخ بدهد. این کار برقراری یک مسیر دو طرفه را امکان پذیر می‌سازد.

برای کاهش سرباره مسیر جدید، بسته سوال اگر با یک گره که قبلاً مسیر برای چنین مقصدی را داشت، برخورد کند، در آن مسیر برای مقصد رها می‌شود. بعد از محاسبه مسیر، تا زمانی که منبع از آن استفاده می‌کند بصورت به‌روز نگهداری می‌شود. برای مثال، یک ارتباط ناموفق ممکن است باعث رهاسازی سوال و پاسخ دیگری شود تا مسیر همیشه به‌روز نگهداری شود. مزیت این روش در این است که تطابق بهتری با محیط‌های دینامیکی شبکه‌های ادهاک موبایل دارد و اطلاعات مسیریابی تازه‌تری دارد. استفاده از اطلاعات مسیره‌تازه‌تر به معنای محاسبات کمتری برای انتقال داده می‌باشد. عیب این روش در آن است که هر گره واسطه‌ای بایستی اطلاعات جدول مسیریابی را برای هر مسیر فعال ذخیره کند و هر گره‌ای بایستی از گره‌های همسایه مطلع باشد [۱][6].

۳-۸) مسیریابی منبع

یک طرح دیگر برای ردیابی مسیرهای نیازی، مسیریابی منبع می‌باشد. DSR در غالب یک چنین پروتکلی پیشنهاد شده است. اگر دیتا برای فرستادن موجود باشد اما هیچ مسیری برای مقصدش معلوم نشده باشد، منبع یک تقاضای مسیر^{۲۶} شناور را ارسال می‌کند. زمانی که این بسته بوسیله مقصد یا یک گره که مسیر تا مقصد را می‌داند دریافت شود، یک پاسخ مسیر^{۲۷} برای منبع بوسیله مسیر انتخاب شده فرستاده می‌شود. در مسیریابی منبع، بسته‌های داده آدرس کامل منبع به مقصد را حمل می‌کنند. هر گره در شبکه، یک منبع ذخیره‌سازی مسیرهایی را که یاد گرفته است، دارد. هر یک از گره‌های واسط این بسته‌ها را طبق اطلاعاتی که در ابتدای بسته وجود دارد انتقال می‌دهد و نیازی به نگهداری اطلاعات مسیریابی به‌روز شده در گره‌های واسط نمی‌باشد. اگر یک گره مسیری را که بعلت تغییرات توپولوژی منسوخ شده، تشخیص دهد، یک خطای مسیر^{۲۸} برای منبع می‌فرستد. منبع سپس یک فرآیند کشف مسیر برای ایجاد کردن یک مسیر جدید طلب می‌کند. یکی از معایب مسیریابی منبع در این است که برای شبکه‌های وسیع عملکرد خوبی ندارد. این امر به دو دلیل اصلی است: اول اینکه تعداد گره‌های واسط

²⁶. route request

²⁷. route reply

²⁸. route error

در مسیر افزایش یافته و احتمال خطای مسیر افزایش می‌یابد. دوم اینکه با ازدیاد گره‌های واسط، میزان سرباره‌ای که توسط هر سربند بسته داده ایجاد می‌شود، رشد می‌یابد [۱].

آزمایش‌های اولیه که برای تعداد کمی از گره‌های ثابت (مثلاً ۱۰ عدد) انجام گرفته است. تأخیر حدود ۲ میلی ثانیه برای دو گذر و ۷۰ میلی ثانیه برای چهار گذر را نتیجه داده است که در بسیاری از موارد عملی قابل قبول می‌باشد. در شرایط کنونی بازده شبکه با افزایش تعداد گذرها پایین می‌آید. به عنوان مثال در حالی که یک مسیریان گذر از نرخ ۲ الی ۳mbit/s برخوردار است، این سرعت در مسیری با چهارگذر به ۱mbit/s نزول می‌کند [5].

۳-۹) جمع‌بندی

از آنجایی که بیشتر دستگاه‌ها در شبکه‌های ادهاکبا باتری راه‌اندازی می‌شوند، مصرف انرژی سیار مهم بوده و مسیریابی باید برای بیشینه نمودن عمر باتری، بهینه شود. در میان عملیاتی که انرژی مصرف می‌نمایند، انتقال بیسیم داده رتبه اول را دارد. چندین پروتکل مسیریابی که در آنها توان سیستم در نظر گرفته می‌شود، در فصل بعد معرفی شده است.

نتایج، این حقیقت را روشن می‌کند که هر دو گروه پروتکل‌های مسیریابی جدولی و نیازی نمی‌توانند در شبکه‌هایی که بسیار پویا می‌باشند، خوب عمل کنند. اگر چه پروتکل‌های جدولی خیلی سریع می‌توانند مسیر مورد نیاز را فراهم کنند، در عوض برای اینکه همیشه بتوانند توپولوژی به‌روز شده شبکه را نگهداری کنند، منابع شبکه را به هدر می‌دهند. از طرف دیگر پروتکل‌های نیازی، می‌توانند میزان استفاده از منابع شبکه را کاهش دهند، اما امکان دارد که با درخواست‌های مسیر در الگوریتم‌های مبتنی بر flooding، با تأخیر زیادی همراه باشد [3][4][5][6][7].

روش دیگر مسیریابی، استفاده از پروتکل‌های دورگه می‌باشد که مقاصد پروتکل‌های جدولی و نیازی را جمع‌بندی می‌کند. در این پروتکل‌ها هر گره فقط توپولوژی همسایه نزدیک به خود را نگهداری می‌کند بنابراین میزان ترافیکی که در پروتکل‌های جدولی موجود می‌باشد را کاهش می‌دهد. پروتکل‌های ZRP^{۲۹} و ZHLS^{۳۰} نمونه‌هایی از این دسته می‌باشند [4].

²⁹. Zone Routing Protocol

³⁰. Zone based Hierarchical Link State Routing Protocol

مسیریابی در ادهاک

۳-۱۰) مقدمه

پروتکل‌های مسیریابی موجود برای شبکه‌های بی‌سیم ادهاک را به گونه‌های متفاوتی تقسیم‌بندی می‌کنند. نمونه‌ای از تقسیم‌بندی را که بر اساس زمان بود در انتهای فصل قبل مورد بررسی قرار دادیم. در حالت کلی طرح‌های مسیریابی موجود را می‌توان در چهار گروه پهن‌آور دسته‌بندی کرد:

الف- global, precomputed routing

مسیرها از قبل برای همه مقصدها محاسبه می‌شوند و بوسیله فرآیند به‌روزکننده به‌صورت دوره‌ای، نگهداری می‌شوند. بیشترین مسیریابی مرسوم شامل DV^{۳۱} و LS^{۳۲} در این گروه قرار می‌گیرند.

ب- on-demand routing

در این روش همانطور که توضیح داده شد مسیر برای یک مقصد ویژه تنها زمانی که نیاز باشد محاسبه می‌شود.

ج- location base routing

محاسبه مسیر با اطلاع از مکان جغرافیایی مقصد، معمولاً بوسیله GPS بدست می‌آید.

د- flooding

فرض می‌شود کمترین یا بدون اطلاع از ساختار شبکه هستیم. هیچ مسیری بر حسب تقاضا محاسبه نمی‌شود. در روش flooding یک بسته به تمامی گره‌هایی که در منطقه تحت پوشش هستند منتشر می‌شود. این گره‌ها نیز به انتشار بسته‌ها پرداخته و عملیات فوروارد تا هنگامی که بسته به مقصد نهایی^{۳۳} برسد ادامه پیدا می‌کند. مزیت این روش مقاومت در برابر تغییرات پیکربندی و سرباره کم پروتکل مسیریابی^{۳۴} می‌باشد. درحقیقت، در شبکه‌هایی که حرکت زیادی دارند، این روش تنها راه حل ممکن برای مسیریابی است. کپی‌های چندگانه که از یک بسته در شبکه حرکت می‌کند اشکال آن می‌باشد که عرض باند و توان باتری گره‌های ارسال کننده را تلف می‌کند [۱].

۳-۱۲) طرح‌های مسیریابی

مسیریابی Global, Precomputed خود می‌تواند به دو گروه دیگر تقسیم‌بندی شود: مسطح^{۳۵} و سلسله مراتبی^{۳۶}. البته در حالت کلی نیز می‌توان بر اساس ساختار، پروتکل‌های مسیریابی ادهاک را به این دو

^{۳۱}. Distance Vectore

^{۳۲}. Link State

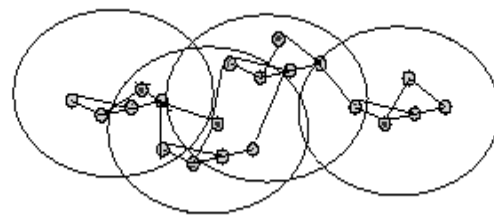
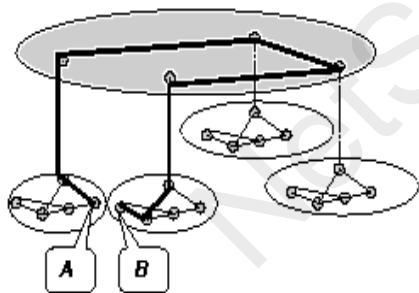
^{۳۳}. Ultimate destination

^{۳۴}. Routing protocol overhead

^{۳۵}. flat

^{۳۶}. hierarchical

دسته تقسیم کرد. در روشهای مسیریابی مسطح (تخت) هر گره یک جدول مسیریابی با ورودی‌هایی برای تمام گره‌های شبکه ایجاد می‌کند. تمام گره‌های شبکه نقش یکسانی در عملیات مسیریابی دارند و هیچ گره‌ای نقش متمایزی از سایر گره‌ها به عهده ندارد. در گروه مسیریابی مسطح، پروتکل‌های زیادی برای پشتیبانی مسیریابی ادهاک بی سیم سیار پیشنهاد شده است. تعدادی از طرح‌ها در ادامه طرح‌های شبکه‌های سیمی سنتی پیشرفت کرده‌اند. برای مثال، اساس DSDV³⁷ پرکینز مبنی بر DBF³⁸ است و WRP³⁹ گارسیا نیز مبنی بر الگوریتم کشف مسیر استوار است که هر دوی این‌ها مسیریابی بدون حلقه⁴⁰ می‌باشند و غیره [۱] [8]. روش مسیریابی تخت در صورتی قابل قبول است که تعداد کاربران کم باشد و با افزایش تعداد کاربران، هزینه‌ها افزایش می‌یابد. بنابراین الگوریتم‌های مسیریابی تخت برای شبکه‌های بزرگ مناسب نیستند. در شکل ۴-۱ ساختارهای تخت و سلسله مراتبی نشان داده شده است.



ساختار سلسله مراتبی

ساختار تخت

شکل ۳-۱: نمایش ساختارهای تخت و سلسله مراتبی

در طرح‌های مسیریابی مسطح هر گره یک جدول مسیریابی دارد که همه ورودی‌های گره‌ها در شبکه را در آن نگهداری می‌کند. این روش اگر تعداد استفاده کننده کم باشد قابل قبول است. به هر حال، به محض اینکه

³⁷. Destination Sequenced Distance Vector

³⁸. Distributed Bellman Ford

³⁹. Wireless Routing Protocol

⁴⁰. Loop-Free

تعداد میزبان‌های سیار افزایش یابد، سرباره هم افزایش می‌یابد. بنابراین الگوریتم مسیریابی مسطح مقیاس خوبی برای شبکه‌های بزرگ نیست. برای مقیاس پذیر کردن یک شبکه، از تکنیک‌های سلسله مراتبی می‌تواند استفاده شود. در این تکنیک‌ها شبکه دارای دو نوع گره، یکی گره انتهایی و دیگری گره سوئیچ می‌باشد. فقط گره‌های انتهایی می‌توانند مبدأ و مقصد ترافیک داده کاربر باشند و فقط گره‌های سوئیچ می‌توانند توابع مسیریابی را اجرا کنند. برای تشکیل پایین‌ترین سطح گروه در روش سلسله مراتبی، نقاط انتهایی مناسب‌ترین سوئیچی را انتخاب می‌کنند که با بررسی کیفیت اتصال رادیویی با آنها همکاری می‌کند [9].

سرخوشه‌ها^{۴۱} بصورت خودگردان نقاط انتهایی خود را در سلولهای اطراف این سوئیچ‌ها گروه‌بندی می‌کنند. این فرآیند تشکیل سلول نامیده می‌شود. سوئیچ‌ها به نوبت و به صورت سلسله مراتبی خود را در خوشه‌ها سازمان‌دهی می‌کنند. سرگروه اولین سطح، گروه‌های سطح بالاتر را سازمان می‌دهد و به همین ترتیب ادامه پیدا می‌کند. این فرآیند دسته‌بندی سلسله مراتبی نام دارد. یک سوئیچ در سطح صفر خوشه (بالاترین سطح) قرار دارد. وقتی گره‌ها حرکت می‌کنند، ممکن است خوشه‌ها تکه تکه شوند و یا دسته بندی جدیدی تشکیل دهند. مهمترین مزیت مسیریابی سلسله مراتبی، استفاده مناسب از منابع کانال رادیویی و کاهش شدید در ذخیره جدول مسیریابی، هزینه‌های انتقال و پردازش کردن می‌باشد [10].

۳-۱۲-۱) مسیریابی بردار فاصله^{۴۲}

الگوریتم مسیریابی بردار فاصله بدین نحوه کار می‌کند که هر مسیریاب جدولی را در خود نگه می‌دارد که در آن بهترین فاصله تا هر مسیریاب مقصد و خطی که برای رسیدن به آن مقصد باید مورد استفاده قرار بگیرد، درج شده است. این جدول با مبادله اطلاعات بین مسیریاب‌های همسایه، بهنگام سازی می‌شود. الگوریتمهای مسیریابی بردار فاصله گاهی با نام‌های دیگری معرفی می‌شوند که اهم این اسامی عبارتند از: ((الگوریتم مسیریابی بلمن-فورد توزیع شده))^{۴۳} و ((الگوریتم فورد-فوکرسون))^{۴۴}. در مسیریابی بردار فاصله هر مسیریاب یک جدولی مسیریابی دارد که به ازای هر مسیریاب موجود در زیرشبکه، یک درایه در آن درج شده است.

هر درایه دارای دو بخش است:

۱- خط خروجی مناسب برای رسیدن به مقصد مورد نظر

۲- تخمینی از زمان یا فاصله رسیدن بدان مقصد [۱].

۳-۱۲-۲) مسیریابی حالت لینک^{۴۵}

⁴¹. Cluster Head

⁴². Distance Vectore Routing

⁴³. Distributed Bellman-Ford

⁴⁴. Ford-Fulkerson

⁴⁵. Link State Routing

دو مشکل اساسی منجر به زوال مسیریابی بردار فاصله و جایگزینی با مسیریابی حالت لینک شد: اول اینکه در این الگوریتم معیار تأخیر، طول صف (تعداد بسته های به صف شده منتظر ارسال بر روی یکی از خطوط خروجی مسیریاب) در نظر گرفته می شد و پهنای باند هر یک از خطوط در محاسبات انتخاب بهترین مسیر، دخالت داده نمی شد. در ابتدا تمام خطوط 56kbps بودند لذا پهنای باند خط، مورد مهمی نبود، ولی پس از آنکه برخی از خطوط به سرعت 230kbps و بیشتر ارتقاء یافتند، به حساب نیاوردن پهنای باند، مشکلی عمده به حساب می آمد، البته این امکان وجود داشت که بتوان معیار تأخیر را به پهنای باند تغییر داد و لیکن مشکل دیگری ایجاد می شد و آن هم اینکه، همگرایی الگوریتم بسیار طولانی می شد. به همین دلیل با الگوریتم مسیریابی حالت لینک تعویض شد. امروزه الگوریتمهای متفاوتی از مسیریابی حالت لینک مورد استفاده قرار می گیرد. ایده اصلی و زیربنای مسیریابی حالت لینک ساده است و می توان آن را در پنج جمله بیان کرد. هر مسیریاب باید به ترتیب زیر عمل کند:

- همسایه های خود را شناسایی کرده و آدرس های شبکه را بدست بیاورد.
- تأخیر هر یک از همسایه های خود را اندازه گیری نماید.
- بسته ای بسازد و اطلاعاتی که از همسایه کسب کرده، در آن جاسازی کند.
- این بسته ها را برای تمام مسیریاب های دیگر بفرستد.
- کوتاه ترین مسیر رسیدن به دیگر مسیریاب ها را محاسبه نماید [۱].

۳-۱۳) پروتکل های مسیریابی تک پخشی

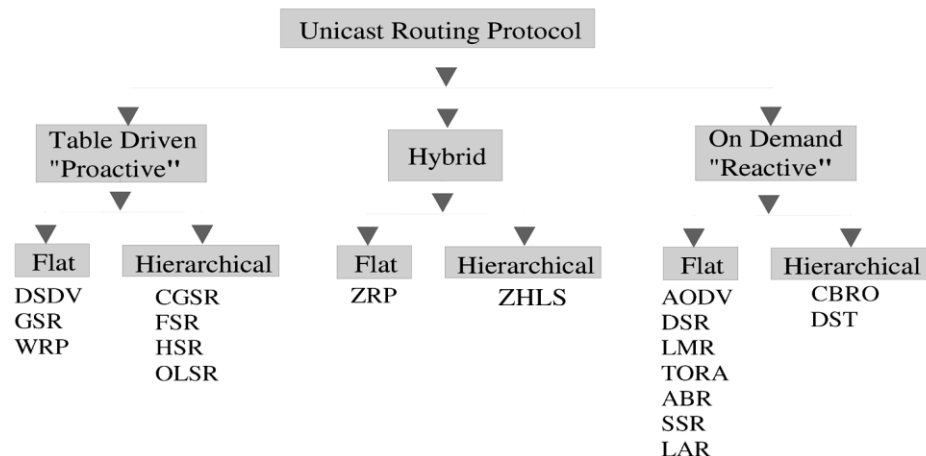
شکل ۳-۳ پروتکل های مسیریابی تک پخشی موجود و طبقه بندی آنها را نشان می دهد [۱]. سردسته پروتکل مسیریابی تک پخشی وجود دارد:

- ۱- پروتکل های مبتنی بر جدول^{۴۶} (پروتکل های جدولی)
- ۲- پروتکل های بنابه درخواست^{۴۷} (پروتکل های نیازی)
- ۳- پروتکل های دورگه^{۴۸} که تلفیقی از دو نوع قبلی هستند.

⁴⁶. Table-Driven

⁴⁷. On-Demand

⁴⁸. Hybrid



شکل ۳-۲: دسته بندی پروتکل های مسیریابی تک پخشی

۳-۱۳-۱ پروتکل های مسیریابی تک پخشی مبتنی بر جدول

۳-۱۳-۱-۱ پروتکل DSDV

DSDV یک پروتکل مناسب برای شبکه های ادهاک می باشد. این پروتکل گسترش یافته الگوریتم بردار فاصله کلاسیک یا DBF^{۴۹} است. در **DSDV** به منظور جلوگیری از مشکل حلقه که در این الگوریتم **DBF** وجود دارد، کارهایی انجام گرفته است. با ترکیب ورودی هر جدول مسیریابی با یک شماره ترتیبی^{۵۰} برای تنظیم اطلاعات مسیریابی، می توان از تشکیل حلقه ها جلوگیری کرد. در پروتکل **DSDV** با استفاده از جداول مسیریابی که در هر گره شبکه **AdHoc** ذخیره شده است، عملیات مسیریابی بین گره ها انجام می شود. جداول مسیریابی در هر گره، همه مقصدهایی که در دسترس هستند را مشخص می کند و تعداد گام های فاصله تا آنها را نیز معلوم می کند.

ورودی جدول مسیریابی با یک شماره ترتیبی همراه است که توسط گره مقصد ایجاد می شود. برای اینکه اطلاعات جداول مسیریابی مناسب و صحیح باقی بمانند، **DSDV** هم از بروزرسانی دوره ای و هم از بروزرسانی مسیر فعال شده استفاده می کند. بروزرسانی مسیر فعال شده با بروزرسانی دوره ای انجام می شود تا بتواند در زمانی که هیچ تغییری در توپولوژی بوجود نیامده است، اطلاعات مسیریابی را با حداکثر سرعت ممکن منتشر کند. بسته های بروزکننده شامل مقصدهای قابل دسترس در هر گره و تعداد پرش های لازم به هر مقصد و یک شماره ترتیبی می باشد که به آن مسیر مربوط می شود. داده ها در طول فاصله زمانی بین اولین دریافت و دریافت بهترین مسیر برای هر مقصد مشخص، نگهداری می شود. ممکن است ساختار داده طوری تنظیم شود که در اعلام مسیریابی

⁴⁹. Distributed Bellman-Ford

⁵⁰. Sequence Number

که احتمالاً تغییر می‌کند، تأخیر ایجاد شود تا نوسانات جداول مسیریابی از بین برود و از تعداد پخش‌های تکراری مسیرهایی که با یک شماره ترتیبی وارد می‌شوند، کاسته شود [12][11].

اعلام مسیر^{۵۱}

در پروتکل DSDV لازم است که هر گره سیار جدول مسیریابی خود را به تمام همسایه‌هایش^{۵۲} در آن لحظه اعلام کند. اعلان باید طوری انجام شود که مطمئن باشیم که هر گره سیار همیشه می‌تواند موقعیت مکانی هر گره سیار دیگری در مجموعه خود را تشخیص دهد، چون بندرت ممکن است که ورودی‌های موجود در لیست به طور پویا تغییر کنند. همچنین هر گره سیار آماده است که بسته‌های داده را طبق درخواست باز هم به سایر گره‌ها پخش کند. این توافق باعث می‌شود که بتوان کمترین تعداد گام برای مسیریابی به مقصد را تشخیص داد. در این روش ممکن است یک گره سیار داده را با هر گره سیار دیگری در همان گروه مبادله کند حتی اگر مقصد داده در داخل منطقه‌ای نباشد که بتوان به طور مستقیم ارسال را انجام داد. تمام گره‌های یک گروه برای ایجاد مسیر داده در داخل گروه، داده‌های لازم را به طور دوره‌ای هر چند ثانیه یک‌بار منتشر می‌کنند.

ورودی‌های جدول مسیریابی و معیارهای انتخاب مسیر

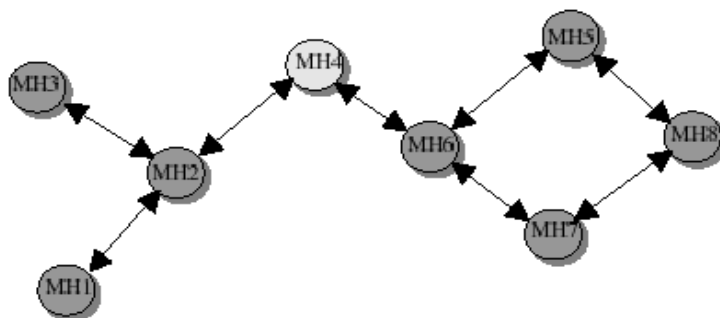
هر بسته مسیریابی که توسط گره‌ها منتشر می‌شود شامل شماره ترتیبی جدید خود و اطلاعاتی برای مسیرهای جدید می‌باشد. این اطلاعات عبارتند از: آدرس مقصد، تعداد پرش‌های لازم برای رسیدن به مقصد، شماره ترتیبی اطلاعات رسیده از آن مقصد که در ابتدا توسط مقصد ایجاد و ضمیمه شده است. وقتی که گره‌ای وارد شبکه می‌شود، یک پیام ردیابی ارسال می‌کند که یک شماره ترتیبی محلی را به آن چسبانیده است. گره‌های همسایه‌اش آن پیام را می‌شنوند و اطلاعات را برای این گره بروزرسانی می‌کنند. اگر سایر گره‌ها از قبل هیچ ورودی برای آن گره نداشته باشند، همسایه‌ها به راحتی آدرس آن را در جدول مسیریابی خود وارد می‌کنند. اگر گره‌ها قبلاً یک ورودی برای آن گره داشته باشند، شماره ترتیبی اطلاعات ارسال شده با شماره ترتیبی ذخیره شده برای آن گره مقایسه می‌شود. اگر پیام دریافت شده شماره ترتیبی بزرگ‌تری داشته باشد، یعنی گره مذکور اطلاعات جدیدتری در مورد موقعیت خود ارسال کرده است و ورودی‌ها باید متناسب با اطلاعات دریافت شده جدید بروز شوند، چراکه اطلاعات با شماره ترتیبی بزرگ‌تر قطعاً جدیدتر هستند. اطلاعات جدیدی که یک گره دریافت می‌کند، برای ارسال به همسایه‌هایش به منظور بروزرسانی فعال می‌شوند تا همسایه‌ها نیز بتوانند از تغییرات توپولوژی مطلع شوند. گره‌های همسایه نیز از همین روش پیروی

⁵¹. Route Advertisement

⁵². Neighbor Nodes

می‌کنند. در هر پرش، معیار انتخاب مسیر (معمولاً تعداد پرش‌ها تا مقصد)، هر بار یک واحد اضافه می‌شود. بدین ترتیب اطلاعات جدید به تدریج در تمام گره‌ها به‌روز می‌شود و از این پس گره‌ها مسیر صحیح رسیدن به مقصد و تعداد پرش‌های لازم را می‌دانند. وقتی گره سیار نقل مکان می‌کند، باعث می‌شود که بعضی از اتصال‌ها از بین بروند. وقتی اتصال به گام بعدی^{۵۳} برای یک مسیر از بین برود، به هر مسیری که از آن اتصال استفاده می‌کند، یک معیار بینهایت و یک شماره ترتیبی جدید نسبت داده می‌شود. این تنها حالتی است که شماره ترتیبی به جای اینکه توسط مقصد مشخص شود، توسط گره‌های دیگر نسبت داده می‌شود. وقتی گره‌ای یک معیار بینهایت دریافت می‌کند درحالی‌که در جدول خود شماره ترتیبی مساوی یا بزرگ‌تری برای آن مقصد ولی با معیار غیر بینهایت دارد، بلافاصله یک انتشار بروزرسانی را فعال می‌کند. بنابراین مسیرهای با معیار بینهایت خیلی زود توسط مسیرهای با معیار محدود، جایگزین می‌شوند. برای کاهش هزینه‌های بروزرسانی اطلاعات توپولوژی شبکه، یک گره می‌تواند دو نوع بسته را انتخاب کند: Full-dump و افزایشی^{۵۴}. پیام‌های Full-dump تمام اطلاعات مسیریابی را از جدول مسیریابی فرستنده به همراه دارد. معمولاً این اطلاعات حتی برای شبکه‌های کوچک نیز نیاز به چندین بسته دارند. در عوض پیام‌های افزایشی فقط اطلاعاتی که بعد از آخرین پیام Full-dump تغییر کرده است را با خود دارند و بر طبق قرارداد باید فقط توسط یک بسته ارسال شوند. وقتی گره‌ها به ندرت حرکت می‌کنند، پیام‌های افزایشی به یک بسته ساده محدود بوده و برای بروزرسانی اطلاعات مسیریابی کافی و مناسب هستند. در این حالت پیام‌های Full-dump می‌توانند با فرکانس کمتری ارسال شوند. وقتی گره‌ها دائماً در حال حرکت هستند، اندازه پیام‌های بروزکننده افزایشی به پیام‌های Full-dump نزدیک می‌شود. در این حالت پیام‌های Full-dump باید با فرکانس بیشتری ارسال شوند تا بتوان اندازه پیام‌های بروزکننده افزایشی را کاهش داد.

پروتکل DSDV برای پخش اطلاعات جداول مسیریابی، هم از بروزرسانی دوره‌ای برای پخش کلیه اطلاعات توپولوژی به صورت پیام‌های Full-dump و افزایشی و هم از بروز سازی فعال شده برای پخش تغییرات مهم در توپولوژی استفاده می‌کند. در شکل ۳-۳ توپولوژی یک شبکه فرضی نشان داده شده است.



⁵³. Next hop

⁵⁴. Incremental

شکل ۳-۳: مثالی از یک شبکهٔ ادهاک

هر گره دو نوع جدول مسیریابی را نگهداری می‌کند. یکی جدول هدایت کننده^{۵۵} که از آن برای یافتن مسیر به سمت مقصد و به جلوراندن بسته‌ها استفاده می‌کند و دیگری جدول اعلام شده^{۵۶} که ورودی‌های آن به ترتیب اولویت از بالا به پایین یک انتشار بروزکننده را فعال می‌کنند.

جدول ۳-۱، یک ساختار فرضی برای جدول هدایت کننده نگهداری شده در MH4 را نشان می‌دهد.

جدول ۳-۱) جدول هدایت کننده در MH4

Destination	Next Hop	Metric	Seq. Number	Install-time	Stable Data
MH ₁	MH ₂	2	S406_MH ₁	T001_MH ₄	Ptr1_MH ₁
MH ₂	MH ₂	1	S128_MH ₂	T001_MH ₄	Ptr1_MH ₁
MH ₃	MH ₂	2	S564_MH ₃	T001_MH ₄	Ptr1_MH ₁
MH ₄	MH ₄	0	S710_MH ₄	T001_MH ₄	Ptr1_MH ₁
MH ₅	MH ₆	2	S392_MH ₅	T002_MH ₄	Ptr1_MH ₁
MH ₆	MH ₆	1	S076_MH ₆	T001_MH ₄	Ptr1_MH ₁
MH ₇	MH ₆	2	S128_MH ₇	T002_MH ₄	Ptr1_MH ₁
MH ₈	MH ₆	3	S050_MH ₈	T002_MH ₄	Ptr1_MH ₁

با فرض اینکه آدرس هر گره MH_i باشد، کلیهٔ اعداد ترتیبی به صورت SXXX-MH_i نمایش داده می‌شوند. زمان نصب کمک می‌کند تا مشخص کنیم که چه زمانی مسیرهای قدیمی^{۵۷} را باید حذف کنیم. در

⁵⁵. Forwarding Table

⁵⁶. Advertised Table

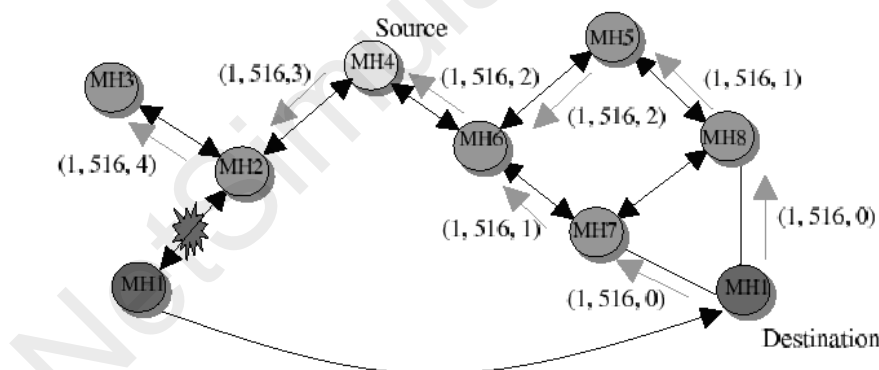
⁵⁷. Stale Routes

جدول ۱-۳ می‌بینیم که کلیه گره‌ها تقریباً در یک زمان برای MH4 در دسترس می‌باشند چون برای بیشتر آنها Install-Time یکسان می‌باشد. جدول ۲-۳، ساختار جدول اعلام شده برای MH4 را نشان می‌دهد.

جدول ۲-۳: جدول مسیر اعلام شده در MH4

Destination	Metric	Seq. Number
MH ₁	2	S406_MH ₁
MH ₂	1	S128_MH ₂
MH ₃	2	S564_MH ₃
MH ₄	0	S710_MH ₄
MH ₅	2	S392_MH ₅
MH ₆	1	S076_MH ₆
MH ₇	2	S128_MH ₇
MH ₈	3	S050_MH ₈

حالا فرض می‌کنیم که MH1 حرکت می‌کند و در همسایگی MH8 و MH7 قرار می‌گیرد و از سایر گره‌ها (خصوصاً MH2) دور می‌شود. اعلان مسیر در گره مقصد آغاز می‌شود و در کل شبکه منتشر می‌شود. شکل ۳-۴، انتشار منطقی را از یک ورودی مسیر، از یک گره مقصد به سایر گره‌های شبکه را نشان می‌دهد.



شکل ۳-۴: نقل مکان در شبکه ادهاک

طبق شکل فوق، مقصد یک شماره ترتیبی را به اعلان مسیر جدید ضمیمه می‌کند. شکل فوق فقط اطلاعات مربوط به مقصد MH1 را نشان می‌دهد، اگرچه در حقیقت هر گره باید جدول مسیر کامل و نه فقط اطلاعات MH1 را ارسال کند. طبق جدول ۳-۴، جدول هدایت کننده داخلی ممکن است در MH4 ظاهر شود.

جدول ۳-۳: جدول هدایت کننده در MH4 (بروز شده)

Destination	Next Hop	Metric	Seq. Number	Install	Stable Data
MH ₁	MH ₆	3	S516_MH ₁	T810_MH ₁	Ptr1_MH ₁
MH ₂	MH ₂	1	S238_MH ₂	T001_MH ₄	Ptr1_MH ₁
MH ₃	MH ₂	2	S674_MH ₃	T001_MH ₄	Ptr1_MH ₁
MH ₄	MH ₄	0	S820_MH ₄	T001_MH ₄	Ptr1_MH ₁
MH ₅	MH ₆	2	S502_MH ₅	T002_MH ₄	Ptr1_MH ₁
MH ₆	MH ₆	1	S186_MH ₆	T001_MH ₄	Ptr1_MH ₁
MH ₇	MH ₆	2	S238_MH ₇	T002_MH ₄	Ptr1_MH ₁
MH ₈	MH ₆	3	S160_MH ₈	T002_MH ₄	Ptr1_MH ₁

جدول ۳-۳ فقط ورودی های MH1 را نشان می دهد و معیار جدیدی را بکار می برد، اما در فاصله زمانی مداخله^{۵۸}، رشته های ترتیبی زیادی دریافت شده اند. این تغییر بندرت بدلیل بزرگ تر بودن شماره ترتیبی اتفاق می افتد. پیام بروز کننده مشابه (مثلاً همان مقصد و با همان شماره ترتیبی) از MH5 و MH7 به ترتیب به MH6 می رسد. MH6 پیام را از MH7 می پذیرد زیرا که معیار بهتری دارد. بروز شدن مسیر اعلام شده افزایشی در MH4 در جدول ۳-۴ نشان داده شده است. در این اعلان، اطلاعات ابتدا برای MH4 می رسد، بنابراین MH4 اعلان را انجام می دهد. سپس اطلاعات مربوط به MH1 می رسد چون تنها گره ای است که هر تغییر مهم در مسیر روی این گره اثر می گذارد.

جدول ۳-۴) مسیرهای اعلام شده در MH4 (بروز شده)

Destination	Metric	Seq. Number
MH ₄	0	S820_MH ₁
MH ₁	3	S516_MH ₂
MH ₂	1	S238_MH ₃
MH ₃	2	S674_MH ₄
MH ₅	2	S502_MH ₅
MH ₆	1	S186_MH ₆
MH ₇	2	S238_MH ₇
MH ₈	3	S160_MH ₈

میرا کردن نوسانات^{۵۹}

DSDV مکانیزمی را بکار می گیرد تا بتواند نوسانات موجود در بروز کردن جداول مسیریابی را میرا کند. در یک محیط جایی که گره های مستقل زیادی بطور نامنظم اطلاعات مسیریابی را پخش می کنند، ممکن است که بعضی از نوسانات گسترش پیدا کنند. برای مثال ممکن است گره ای دو مسیر با یک شماره ترتیبی برای یک مقصد خاص داشته باشد، اگرچه همیشه مسیر با کمترین معیار زودتر می رسد.

DSDV این مشکل را با استفاده از ایده زمان نشست^{۶۰} برطرف می کند. زمان نشست برای یک مسیر، مدت زمانی است که پس از آن مسیر پایدار و ثابت می شود. به عبارت دیگر برای اینکه تصمیم بگیریم که چه مدتی قبل از اعلان مسیر جدید باید منتظر بمانیم، از زمان نشست استفاده می کنیم. با ایجاد تأخیر در اعلان مسیرهای

⁵⁸. Intervening

⁵⁹. Damping Fluctuations

⁶⁰. Settling Time

ناپایدار به اندازه زمان نشست، نوسانات جدول مسیریابی میرا می‌شود و در نتیجه تعداد انتشارهای بروز کننده مسیرها کاهش می‌یابد.

از مزیت‌های پروتکل DSDV می‌توان از: بدون حلقه بودن در کلیه لحظات، نیاز به ملزومات کم برای حافظه، همگرایی سریع از طریق بروزرسانی‌های فعال شده، در دسترس بودن مسیرها برای کلیه گره‌های مقصد، زمان کوتاه پردازش، حجم بار مناسب شبکه و حداقل تعداد مسیرهای زائد^{۶۱} نام برد.

از معایب این پروتکل می‌توان به مواردی اشاره کرد از جمله تشخیص بهترین مقدار برای پارامترهایی مثل حداکثر زمان نشست برای یک مقصد مشخص، بسیار سخت و پیچیده می‌باشد و این نکته باعث نوسانات مسیر و اعلان‌های نادرست می‌شود و در نتیجه باعث می‌شود که پهنای باند به هدر برود. همچنین پروتکل DSDV در کنار بروزرسانی‌های فعال شده، از بروزرسانی‌های دوره‌ای نیز استفاده می‌کند که باعث به هدر رفتن شدید پهنای باند و افزایش هزینه‌های ارتباط می‌شود. در پروتکل DSDV هر گره باید تا بروز شدن بعدی مسیر که از مقصد ناشی می‌شود، صبر کند و بعد از آن می‌تواند ورودی‌های جدول مسیریابی خود را برای مقصد مربوطه بروز کند. عیب دیگر آن که پروتکل DSDV مسیریابی چند مسیر و ارسال چندپخشی را پشتیبانی نمی‌کند.

۳-۱۳-۲) پروتکل WRP^{۶۲}

این پروتکل براساس جدول است که هدف آن حفظ اطلاعات مسیریابی در تمامی گره‌های موجود در شبکه است. هر گره باید چهار جدول را در خود نگه دارد که این جداول شامل جدول فاصله، جدول مسیریابی، جدول هزینه، جدول فهرست انتقال مجدد پیام (MRL) می‌باشند، هر ورودی MRL در بر گیرنده شماره به روز شده پیام ورودی، یک شمارش‌گر انتقال مجدد و یک بردار تایید برای هر کدام از این انتقال‌های مجدد و اطلاعات یک ورودی برای هر گره همسایه می‌باشد و همچنین شامل داده‌های به‌روز شده‌ای که در خلال پیام‌های به روزکننده فرستاده شده‌اند، می‌باشد. داده‌های جدول MRL که به روز شده‌اند، باید به اطلاع همسایگان برسند و تایید رسیدن آن نیز از همسایگان باید دریافت شود. جدول MH با استفاده از پیام بروز کننده گره‌ها را از تغییرات ارتباطی شبکه آگاه می‌کنند. یک پیام بروز کننده فقط بین دو همسایه فرستاده می‌شود و شامل داده‌های به روز شده است که خود آن نیز شامل مقصد و فاصله تا مقصد می‌باشد. بعد از بررسی پیام‌های بروز کننده از همسایه یا مشاهده تغییری در ارتباط گره‌ها، پیام‌های بروز کننده خود را به همسایه‌ها می‌فرستند. در نتیجه مسیرهای جدید رد غالب پیام‌های جدید به گره‌های اولیه باز گرداننده می‌شوند و جداول ارتباط بر اساس آن به روز می‌شوند. اگر MH هیچ پیامی را برای دوره زمانی ویژه نفرستد بایستی پیام HELLO را برای اطمینان از اتصال بفرستد، در غیر این صورت نیامدن پیام از MH دلالت به برهم خوردن اتصال دارد و منجر به پیغام خطا می‌شود. هر وقت MH یک پیام HELLO از یک MH جدید دریافت کند، آن MH جدید را به جدول مسیریابی خود اضافه می‌کند و یک کپی از اطلاعات جدول مسیریابی خود را به این جدول MH می‌فرستد [13].

⁶¹. Trash

⁶². Wireless routing protocol

ارزیابی پروتکل

یکی از مزایای این پروتکل نداشتن حلقه مسیریابی است. ولی همانطور که ذکر شد هر گره‌ای بایستی چهار جدول مسیریابی را نگهداری کند. این موضوع مقدار زیادی سرباره حافظه^{۶۳} را در هر گره ایجاد می‌کند. اشکال دیگر WRP این است که از اتصال‌پذیری شبکه با ارسال بسته‌های hello اطمینان حاصل می‌کند. وقتی که بسته داده‌ای ارسال نمی‌شود این بسته‌ها بین گره‌های همسایه تبادل می‌یابد که مقدار قابل توجهی توان و عرض باند را در هر گره مصرف می‌کند که لازم است گره‌ها در همه زمان‌ها فعال بمانند و گره‌ها نمی‌توانند به مد استراحت برای ذخیره انرژی منتقل شوند.

۴-۱۳-۳ پروتکل GSR^{۶۴}

پروتکل GSR بر مبنای الگوریتم‌های رایج حالت لینک است. با محدود ساختن پیام‌های به هنگام رسانی بین گره‌های واسطه این پروتکل انتشار اطلاعات را در الگوریتم حالت لینک بهبود داده است. هر گره‌ای، جدول حالت لینک را بر مبنای اطلاعات به هنگام دریافتی از گره‌های همسایه را نگهداری کرده و به طور متناوب اطلاعات حالت لینک خود را تنها با گره‌های همسایه مبادله می‌کند. این امر به کاهش پیام‌های کنترلی در شبکه منجر می‌شود. علاوه بر این سایر پیام‌های به هنگام رسانی نسبتاً بزرگ است و مقدار زیادی پهنای باند برای این پیام‌ها مصرف می‌شود [14].

۴-۱۳-۴ مسیریابی FSR^{۶۵}

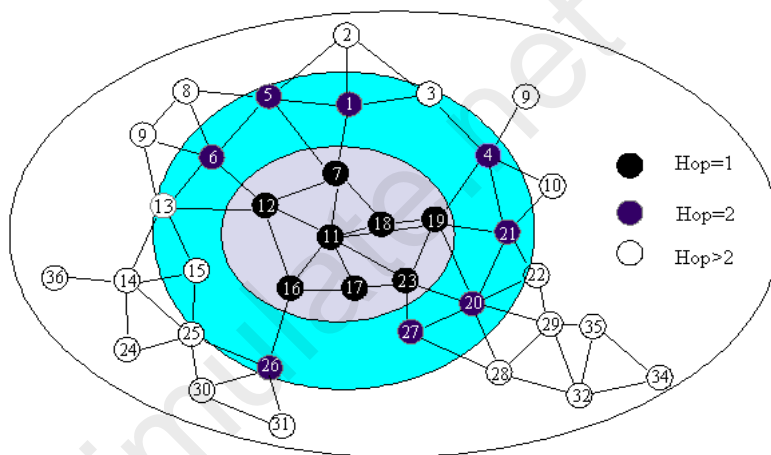
FSR یک پروتکل مسیریابی سلسله مراتبی تصریحی است. در آن از تکنیک چشم ماهی که بوسیله کلین‌راک و استونس پیشنهاد شده است، استفاده می‌کند. چشم ماهی پیکسل‌های نزدیک نقطه کانونی را با جزئیات بالا ضبط می‌کند. به محض اینکه فاصله از نقطه کانونی افزایش پیدا کند جزئیات کاهش پیدا می‌کند. در مسیریابی، روش fisheye که برای نگهداری فاصله دقیق و اطلاعات کیفیت مسیر درباره همسایه مستقیم یک گره استفاده می‌شود، به محض اینکه فاصله افزایش می‌یابد، جزئیات به طور تصاعدی کاهش پیدا می‌کند. SR تابعی مشابه مسیریابی LS است که در هر گره یک نقشه توپولوژی نگهداری می‌کند. اختلاف کلیدی در روشی که اطلاعات مسیریابی منتشر می‌شود است. در LS هر وقت که یک گره یک تغییر توپولوژی پیدا کرد، بسته‌های حالت لینک تولید می‌شوند و در شبکه شناور می‌شوند [15]. در FSR بسته‌های حالت لینک شناور

⁶³. Memory overhead

⁶⁴. Global state routing

⁶⁵. Fisheye State Routing flat

نیستند. در عوض گره ها یک جدول حالت لینک بر مبنای اطلاعات به روز شده ای که از گره های مجاور دریافت شده، را نگهداری می کنند و به طور دوره ای آنرا فقط با همسایه های محلی شان مبادله می کنند (نه شناور). در FSR شبیه LS بسته های حالت لینک پخش می شود و یک نقشه توپولوژی کامل در هر گره نگهداری می شود و کوتاه ترین مسیرها با استفاده از این نقشه ها محاسبه می شوند. در یک محیط بی سیم یک ارتباط رادیویی بین گره های سیار ممکن است به طور مکرر قطع و وصل شدن مجدد را تجربه کند، پروتکل حالت لینک یک پیام به روز کننده برای هر تغییر اینچنینی آزاد می کند که در شبکه شناور می شود و باعث سرباره بیش از اندازه می شود. FSR بوسیله استفاده از مبادله دوره ای محتوی نقشه توپولوژی از این مشکل دوری می کند و به طور زیاد، باعث کاهش سرباره پیغام کنترل می شود.

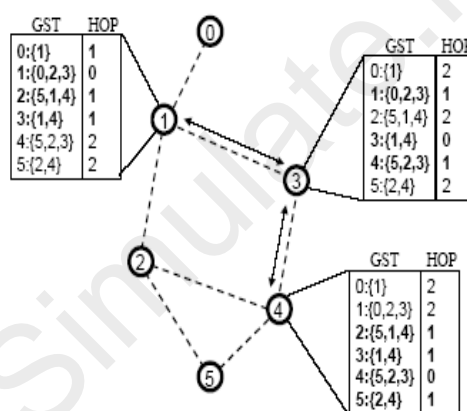


شکل ۳-۴: Fisheye و نمونه ای از درجه بندی گره ها در شبکه بی سیم سیار

وقتی که اندازه شبکه بزرگتر می شود، پیام به روز کننده، مقدار قابل توجهی از پهنای باند را مصرف می کند که به پررود این پیام بستگی دارد. به منظور کاهش اندازه پیام به روز کننده بدون اثر جدی روی دقت مسیریابی، FSR از تکنیک Fisheye استفاده می کند. شکل ۴-۴، Fisheye را در شبکه های بی سیم سیار را نشان می دهد. دایره ها با درجه رنگهای خاکستری مختلف قلمروهای Fisheye نسبت به گره مرکزی (گره ۱۱) را نشان می دهند. قلمرو عنوان دسته ای از گره ها که می توانند با تعداد معینی از جهش ها به یکدیگر برسند تعریف می شود. در شکل، سه قلمرو برای جهش های ۱ و ۲ و ۳ به ترتیب نشان داده شده است. گره ها به رنگهای سیاه و خاکستری و سفید هستند.

کاهش اندازه پیغام به روز کننده، بوسیله استفاده از دوره های تبادل مختلف برای ورودیهای مختلف در جدول بدست می آید. به طور دقیقتر، ورودیها مطابق با گره ها در میان قلمرو کوچکتر برای گره های نزدیک با بیشترین تناوب پخش می شوند.

به شکل ۳-۶ توجه کنید، ورودیها با فونت **bold** با تناوب بیشتر مبادله می‌شوند. باقیمانده ورودیها با تناوب پایین‌تر بیرون فرستاده می‌شوند. در نتیجه یک قسمت قابل توجهی از ورودیهای حالت لینک متوقف می‌شوند. بنابراین اندازه پیام کاهش می‌یابد. این استراتژیک باعث به روز شدن به موقع از ایستگاههای نزدیک و اما تاخیر بیشتر از ایستگاه دورتر می‌شود. به هر حال اطلاع نادرست از بهترین مسیر به یک مقصد دور بوسیله این حقیقت که مسیر به‌طور تصاعدی به محض اینکه بسته به مقصد نزدیک‌تر می‌شود، دقیقتر می‌شود، جبران می‌شود. **FSR** مقیاس خوبی برای شبکه‌های بزرگ، بوسیله پایین نگه داشتن مبادله حالت لینک **O/H**، است. بوسیله نگه داشتن اطلاعات مسیریابی برای هر مقصد، **FSR** از پیدا کردن مقصد به مانند مسیریابی نیازی جلوگیری می‌کند و بنابر این تاخیر انتقال بسته تکی را پایین نگه می‌دارد. به محض اینکه تحرک افزایش یابد دقت مسیرها برای مقصدهای دور دست کمتر می‌شود. به هر حال، زمانی که یک بسته به مقصدش نزدیک می‌شود، به محض اینکه در بخش‌هایی با یک نسبت تازه شدن^{۶۶} بالاتر وارد شد، به طور افزاینده ساختارهای مسیریابی دقیق را پیدا می‌کند [۴۵][۱].



شکل ۳-۶: کاهش پیغام با استفاده از Fisheye

۳-۱۳-۴) مسیریابی HSR^{۶۷}

تفکیک و دسته‌بندی کار معمولی در شبکه‌های بی سیم چند جهشی در هر دو لایه **MAC** و لایه شبکه است. دسته بندی می‌تواند کارایی شبکه را زیاد کند. برای مثال در لایه **MAC** بوسیله استفاده از توزیع مختلف کدها در بین دسته‌ها تداخل کاهش داده می‌شود و استفاده مجدد از فضا زیاد می‌شود. چنانچه تعداد گره‌ها زیاد شوند، انگیزه بیشتری برای به کار انداختن اجزای لایه شبکه به‌منظور انجام مسیریابی سلسله مراتبی است در

^{۶۶}. refresh

^{۶۷}. Hierarchical State Routing

نتیجه سرباره مسیریابی کاهش می‌یابد. در یک شبکه سیار مشکل اصلی مسیریابی سلسله مراتبی تحرک و مدیریت مکان است. برای غلبه کردن بر این مشکل در این بخش ما طرح سلسله مراتبی HSR که پویایی و دسته بندی سلسله مراتبی چند سطحی توزیع شده با یک مدیریت مکان کارآمد را با هم ترکیب می‌کند، توضیح می‌دهیم. هدف دسته‌بندی بهره‌برداری مفید از منابع کانالهای رادیویی و کاهش سرباره مسیریابی لایه شبکه است (یعنی ذخیره سازی جدول مسیریابی، پردازش و انتقال).

الگوریتم رایج حالت لینک اساس این پروتکل است. الگوریتم خوشه بندی برای سازماندهی گره‌ها به کار می‌رود. هر خوشه دارای سه نوع گره است. گره سرخوشه به عنوان هماهنگ کننده محلی گره‌ها، گره دروازه که در آن گره‌ها در دو خوشه متفاوت قرار گرفته‌اند و گره‌های داخلی که گره‌های دیگر داخل خوشه می‌باشند [۱].

دروازه^{۶۸}

گره‌ای دروازه نامیده می‌شود که در محدوده ارتباطی بیش از دو سرخوشه واقع شده باشد. هر گره‌ای که با بیشتر از یک سرخوشه در ارتباط باشد، دروازه کاندید شده برای اتصال این خوشه‌ها می‌باشد. اگر در همسایگی یک گره دو سرخوشه وجود داشته باشد، آن گره یک دروازه کاندید برای ارتباط بین دو خوشه همپوشان می‌باشد. اگر گره‌ای در همسایگی خود یک سرخوشه داشته باشد و بتواند با دو گام به دومین سرخوشه همسایه خود نیز برسد، پس این گره یک دروازه کاندید شده است که به دروازه کاندید شده در خوشه دیگر، متصل شده است و این دو دروازه کاندید شده دو خوشه جداگانه را به یکدیگر وصل می‌کنند. برای خوشه‌های همپوشان، دروازه انتخاب شده آن گره‌ای است که کوچکترین عدد شناسه را دارا می‌باشد. برای خوشه‌هایی که بهم متصل نیستند، دو دروازه انتخاب شده، جفت اتصال شده‌ای هستند که در این زوج، یکی کمترین و یا بیشترین عدد شناسه را در بین کلیه کاندیدهای متصل به این دو خوشه دارا می‌باشد [۵].

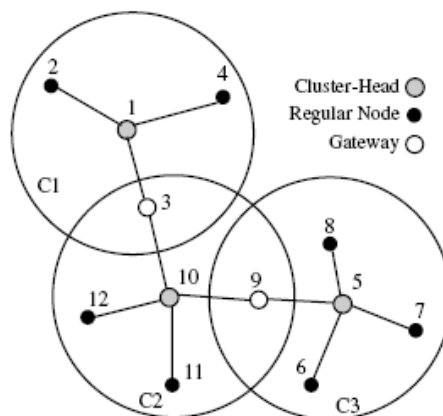
خوشه بندی^{۶۹}

گره‌های سیار در خوشه‌ها جمع‌بندی شده‌اند. خوشه‌ها توسط یک سرخوشه کنترل می‌شود که قالب مناسبی را برای گسترش ایده‌هایی مثل جداسازی کد میان خوشه‌ها، کانال دستیابی مسیر و تخصیص پهنای باند ایجاد می‌کند. با بکارگیری یک الگوریتم توزیع شده در داخل یک خوشه، یک گره به عنوان سرخوشه انتخاب می‌شود. کلیه گره‌های موجود در ناحیه ارتباطی یک سرخوشه به آن خوشه تعلق دارند. مثلاً تمام گره‌ها در یک خوشه می‌توانند با یک سرخوشه و یا احتمالاً با یکدیگر، ارتباط داشته باشند [۱]. شکل ۴-۷ تصویری از نوعی شبکه مبتنی بر خوشه را نشان می‌دهد.

⁶⁸. Gateway

⁶⁹. Clustering

مهمترین معیار پایداری^{۷۰} می‌باشد. مشکل بکارگیری ایده سرخوشه این است که گره‌ها در مقایسه با انتشار بسته‌ها، بیشتر مشغول انتخاب سرخوشه می‌باشند، پس تغییرات متناوب سرخوشه اثر بدی روی عملکرد پروتکل مسیریابی خواهد داشت. بنابراین به‌جای اینکه بعد از هر تغییر در اعضای گروه یک خوشه نیاز به انتخاب مجدد سرخوشه داشته باشیم، استفاده از الگوریتم کم‌ترین تغییرات خوشه یا LCC_{71} پیشنهاد می‌شود. در LCC ، تغییر سرخوشه فقط وقتی اتفاق می‌افتد که تغییری در شبکه باعث شود که دو سرخوشه وارد یک خوشه شوند یا یکی از گره‌ها طوری حرکت کند که از محدوده کلیه سرخوشه‌ها خارج شود. الگوریتم فوق، گسترش یافته دو الگوریتم قبلی یعنی، الگوریتم کمترین شناسه^{۷۲} و الگوریتم بیشترین اتصال^{۷۳} می‌باشد که در این دو الگوریتم ذکر شده هر زمان که اعضای خوشه تغییر کنند، دوباره سرخوشه انتخاب می‌شود. الگوریتم LCC برای تأسیس و نگهداری مسیر هم از کمترین شناسه و هم از بیشترین اتصال استفاده می‌کند. گره‌های داخل هر خوشه اطلاعات لینک خود را به گره‌های دیگر منتشر می‌کنند. هر گره دارای ID سلسله مراتبی (HID) است که رشته‌ای از آدرس‌های MAC از بالا به گره منبع است. روند خوشه‌بندی منطقی توسط ارتباط منطقی بین سرخوشه‌ها صورت می‌پذیرد و از این رو گره‌ها دارای آدرس تخصیص یافته $\langle subnet, host \rangle$ می‌باشند. گره‌های منطقی به لینک‌های منطقی متصل می‌شوند و به خوشه‌های سطح پایین‌تر تونل می‌زنند. اطلاعات حالت لینک منطقی به سطح پایین‌تر جریان می‌یابد. گره‌های فیزیکی در پایین‌ترین سطح دارای پیکربندی سلسله



۷-۳: تصویری از نوعی شبکه مبتنی بر خوشه

⁷⁰. Stability

⁷¹. Least Cluster Change

⁷². Lowest-ID

⁷³. Highest-Connectivity

مراتبی از شبکه می‌باشد. جدایی مدیریت تحرک از سطح فیزیکی مزیت HSR نسبت به دیگر پروتکل‌های مسیریابی سلسله مراتبی مانند MMWN است که این عمل توسط عامل خانگی^{۷۴} انجام می‌شود. این پروتکل دارای سرباره کنترلی کمتری می‌باشد. مانند پروتکل‌های مبتنی بر خوشه، سرباره اضافه‌ای برای آرایش و نگهداری خوشه‌ها دارد. شکل ۴-۴ یک مثال از دسته بندی فیزیکی را نشان می‌دهد. در سطح ۰ ما چهار سطح دسته بندی فیزیکی C0-1، C0-2، C0-3 و C0-4 داریم. دسته های سطح ۱ و سطح ۲ بوسیله انتخاب بازگشتی سرهای دسته تولید می‌شوند. الگوریتمهای دسته‌بندی مختلفی می‌تواند برای ایجاد دسته‌های دینامیک و انتخاب سرهای دسته‌ها استفاده شود. در دسته بندی سطح ۰، رادیوهای طیف گسترده و CDMA^{۷۵} برای استفاده مجدد از فضا در بین دسته‌ها می‌تواند مطرح شود. در میان یک دسته سطح ۰، لایه MAC^{۷۶} می‌تواند بوسیله استفاده از طرحهای مختلف متنوع (نمونه برداری، MACA، CSMA، TDMA و غیره) اجراء بشود.

به طور کلی، سه نوع از گره‌ها در یک دسته وجود دارند، بنام‌های گره سر دسته (مثل گره های ۱ و ۲ و ۳ و ۴)، گره دروازه (مثل گره های ۶ و ۷ و ۸ و ۱۱) و گره داخلی (مثل ۵ و ۹ و ۱۰). شکل ۳-۸ را مشاهده کنید. گره سر دسته مانند یک هماهنگ کننده محلی انتقال در میان دسته، رفتار می‌کند. ID های نشان داده شده در شکل ۳-۸ (در سطح صفر) آدرسهای فیزیکی هستند (مثل لایه MAC). آنها منحصر به فرد هستند. در HSR، آدرس سلسله مراتبی HID^{۷۷} یک گره، از سلسله مراتب بالا تا گره خودش تعریف می‌شود.

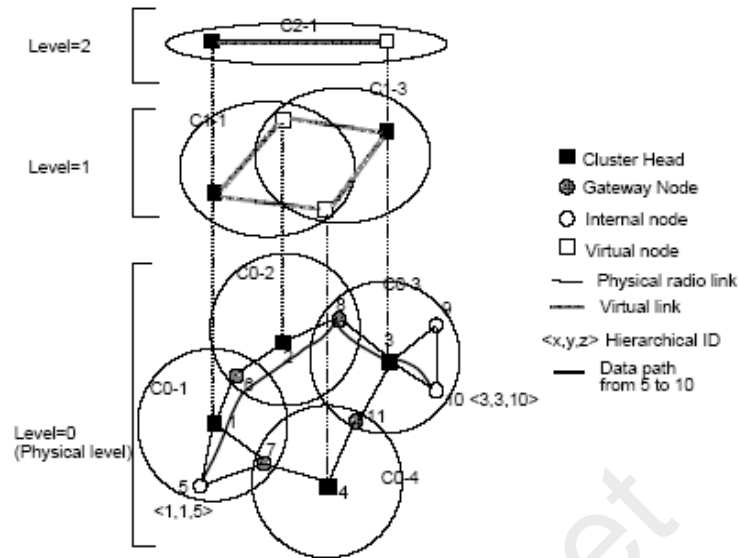
در شکل ۳-۸ آدرس سلسله مراتب گره ۶، HID(6)، ۶ و ۲ و ۳ < است. در این مثال، گره ۳، عضوی از دسته سلسله مراتب بالا (سطح ۲) و همچنین سر دسته از C1-3 است. گره ۲ عضو C1-3 و یک سر دسته از C0-2 است. گره ۶ یک عضو از C0-2 است و می‌تواند مستقیم از گره ۲ رسیده بشود. برتری این طرح آن است که هر گره می‌تواند به طور دینامیکی و محلی HID خود را به محض دریافت به‌روزکننده‌های مسیریابی از گره‌های بالاتر در سلسله مراتب، به‌روز کند.

⁷⁴. Home agent

⁷⁵. code division multiple access

⁷⁶. Medium Access Control

⁷⁷. Hierarchical ID



شکل ۳-۸: مثالی از دسته بندی فیزیکی در الگوریتم سلسله مراتبی

آدرس سلسله مراتبی برای تحویل یک بسته به مقصدش از هر جایی در شبکه‌ای که از جدولهای HSR استفاده می‌کند، کافی است. در شکل ۴-۸، برای مثال تحویل یک بسته از گره ۵ به گره ۱۰ را ملاحظه کنید. توجه کنید که $HID(5) = \langle 1, 1, 5 \rangle$ و $HID(10) = \langle 3, 3, 10 \rangle$ است. بسته بوسیله گره ۵ به سلسله مراتب بالا ارسال می‌شود (یعنی به گره ۱). گره ۱ بسته را به گره ۳ تحویل می‌دهد، که یک گره سلسله مراتبی بالا برای مقصد ۱۰ هست. گره ۱ یک "ارتباط مجازی"، یعنی یک تونل، به گره ۳ دارد برای مثال (۳ و ۸ و ۲ و ۶ و ۱). بنابراین آن بسته را به گره ۳ در امتداد این مسیر تحویل می‌دهد. بالاخره سرانجام، گره ۳ بسته را به گره ۱۰ در امتداد مسیر سلسله مراتبی رو به پایین می‌رساند که در این حالت به یک جهش تنها کاهش می‌یابد سوال قابل توجه آن است که وقتی یک شبکه واحد، بشدت رشد می‌کند، سلسله مراتب باید چند سطحی باشد. به عنوان مثال زیر شبکه ای با ۷۲۰ مسیریاب را در نظر بگیرید. اگر سلسله مراتب وجود نداشته باشد هر مسیریاب در جدول مسیریابی خود به ۷۲۰ درایه نیاز خواهد داشت. اگر زیر شبکه به ۲۴ منطقه و ۳۰ مسیریاب در هر منطقه تقسیم شود، هر مسیریاب به ۳۰ درایه برای مسیرهای محلی خود و ۲۳ درایه برای مناطق خارجی نیاز دارد و بدین ترتیب جدول جمعا ۵۳ درایه خواهد داشت. اگر سلسله مراتب، سه سطحی انتخاب شده و جمعا هشت دسته، هر دسته شامل ۹ ناحیه و در هر ناحیه ۱۰ مسیریاب تعریف شده باشد، هر مسیریاب در جدول خود به ۱۰ درایه برای مسیرهای محلی خود، ۸ درایه برای مسیریابی در نواحی داخل دسته خودش و ۷ درایه برای دسته های خارجی یعنی جمعا ۲۵ درایه نیاز دارد. دو پژوهشگر بنام‌های کامون و کلین‌راک^{۷۸} (۱۹۷۹) پی بردند که بهترین

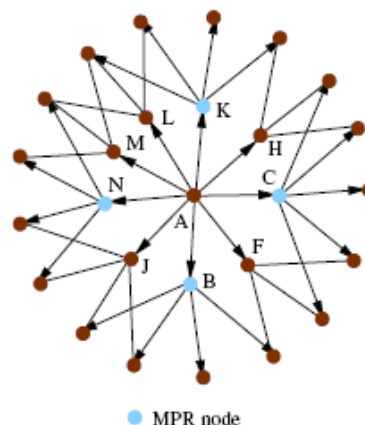
تعداد سطوح سلسله مراتب در زیر شبکه‌ای با N مسیریاب، معادل $\ln(N)$ است و هر مسیریاب به $e \cdot \ln(N)$ عدد درایه نیاز خواهد داشت [10][9].

۷۹-۳-۱-۶ پروتکل OLSR

OLSR یک پروتکل مسیریابی جدولی نقطه به نقطه است و بر اساس الگوریتم حالت لینک می‌باشد. در یک پروتکل حالت لینک صرف، تمام مسیرهای ارتباطی بین گره‌های همسایه مشخص می‌شود و در سرتاسر شبکه پخش می‌شود [۱]. پروتکل **OLSR** بهینه شده یک پروتکل حالت لینک برای شبکه‌های ادهاک موبایل است. ابتدا اندازه بسته‌های کنترل را کاهش می‌دهد و به جای تمام لینک‌ها فقط زیر مجموعه‌ای از لینک‌ها را در میان همسایگانش مشخص می‌کند که در واقع نقش انتخاب گره‌های چند نقطه‌ای آنرا ایفا می‌کنند. ثانیاً از طریق کاربرد گره‌های انتخاب شده که این مجموعه نقاط رله‌های چند نقطه نامیده می‌شوند، حجم ترافیک کنترل را به حداقل می‌رسانند. در این استراتژی هر گره ای اطلاعات پیکربندی را با تبادل متناوب پیام‌های حالت لینک نگهداری می‌کند. تازگی **OLSR** در کاهش پیام‌های کنترلی و تعداد گره‌های دوباره ارسال کننده در هر به هنگام رسانی مسیر با استفاده از استراتژی پاسخ چند نقطه‌ای (MPR) ۸۰ است. در خلال به هنگام رسانی پیکربندی هر گره‌ای در شبکه مجموعه ای از گره‌های همسایه را برای ارسال مجدد بسته‌ها انتخاب می‌کند. این مجموعه نقاط رله‌های چند نقطه‌ای نامیده شده و هر گره‌ای که داخل مجموعه نباشد می‌تواند بسته را خوانده یا آن را پردازش کند، ولی قادر به ارسال مجدد آن نمی‌باشد. برای انتخاب **MPR** ها هر گره ای به صورت متناوب لیستی از همسایه‌های اولین پرس خود را با پیام **hello** منتشر می‌کند. از میان لیستی که پیام **hello** را دریافت می‌کنند، هر گره‌ای مجموعه‌ای از همسایه‌های تک پرشی که قادر به کشف همسایه با دو پرس است را انتخاب می‌کنند. در شکل ۴-۹ گره **A** می‌تواند گره‌های **K, C, B** و **N** را به عنوان گره‌های **MPR** انتخاب کند و این گره‌ها تمامی گره‌های شبکه را کشف می‌کنند. هر گره‌ای مسیر بهینه‌ای بر حسب پرس‌ها را به سمت مقصد بر اساس اطلاعات پیکربندی مشخص می‌کنند و این اطلاعات را در جدول مسیریابی ذخیره می‌کنند. به محض اینکه انتقال داده آغاز می‌شود مسیرها به سوی هر مقصدی بی درنگ در دسترس می‌باشد [1].

⁷⁹ Optimized link state routing

⁸⁰ Multipoint replying



شکل ۳-۹- رله‌های چند نقطه‌ای

۳-۱۳-۷) پروتکل CGSR

CGSR پروتکل سلسله مراتبی دیگری است که گره‌ها با یکدیگر گروه‌بندی شده و تشکیل خوشه می‌دهند و پروتکل **DSDV** را به عنوان یک طرح پایه‌ای مسیریابی بکار می‌برد و بنابراین مانند **DSDV** هزینه‌های زیادی دارد. پروتکل **CGSR** در نوع آدرس‌دهی و طرح ساختاری بکار گرفته با پروتکل مسیریابی **DSDV** فرق دارد. به جای یک ساختار تخت، **CGSR** یک شبکه بی سیم سیار سلسله مراتبی چند پرشی با تعدادی طرح‌های مسیریابی ابتکاری می‌باشد، اگرچه **CGSR**، **DSDV** را به منظور مسیریابی ترافیک از مبدأ به مقصد، با بکارگیری مسیریابی سلسله مراتبی **Head-to-Gateway**، تصحیح می‌کند. در **CGSR** برخلاف **MMWN** نیازی به نگهداری سلسله مراتب خوشه نیست. به جای آن، هر خوشه با سرخوشه نگهداری می‌شود و گره سیاری است که به عنوان مدیر برای گره‌های سیار دیگر در خوشه انتخاب می‌شود. این گره رسانه انتقال و تمام ارتباطات داخل خوشه‌ای که در این گره اتفاق می‌افتد را کنترل می‌کند [۱].

تشکیل خوشه و انتخاب سرخوشه‌ها

الگوریتم‌های انتخاب سرخوشه، خیلی قبل‌تر پیشنهاد شده بوده است: تشکیل خوشه بر مبنای شناسه ۸۱ و تشکیل خوشه بر مبنای اتصال^{۸۲}. هر دوی این الگوریتم‌ها، هم پیاده‌سازی متمرکز و هم توزیع شده را می‌پذیرد. در الگوریتم‌های متمرکز، گره با کمترین عدد شناسه و یا گره با بیشترین تعداد همسایه به عنوان سرخوشه، انتخاب می‌شود. بنابراین تمام گره‌هایی که در خوشه هستند، از توجه بیشتر محروم می‌شوند و فرآیند انتخاب

⁸¹. Identified-based Clustering

⁸². Connectivity-based clustering

سرخوشه و تشکیل خوشه با گره‌های باقیمانده در شبکه تکرار می‌شود و این کار تا زمانی ادامه پیدا می‌کند که تمام گره‌ها تحت پوشش چند خوشه قرار بگیرند.

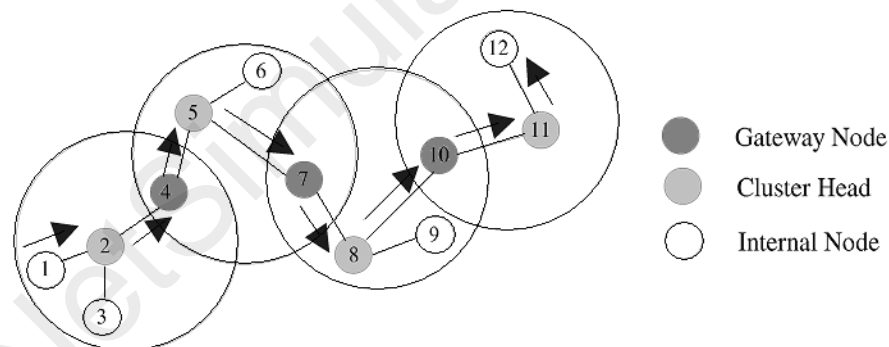
در الگوریتم توزیع شده خوشه‌بندی بر مبنای شناسه، گره‌ای که کمترین عدد شناسه را در آن همسایگی دارا باشد، خود را به عنوان سرخوشه انتخاب می‌کند. در غیر این صورت گره همسایه‌ای را کمترین شناسه را دارا می‌باشد را انتخاب می‌کند، مگر اینکه آن گره از وضعیت سرخوشه بودن خارج شده باشد.

در الگوریتم توزیع شده خوشه‌بندی بر مبنای اتصال، گره‌ای سرخوشه می‌شود که بیشترین اتصال را با همسایه‌های پوشش داده نشده خود داشته باشد و البته گره‌ای پوشش داده نشده محسوب می‌شود که سرخوشه‌ای برای خود انتخاب نکرده باشد.

اگر گره‌ای گره‌ی دیگری را به عنوان سرخوشه خود انتخاب کند، دیگر خودش نمی‌تواند سرخوشه باشد. الزامات سرخوشه طبق گره‌ای که کمترین عدد شناسه را داشته باشد، انتخاب می‌شود.

مسیریابی در CGSR^{۸۳}

الگوریتم در حالت کلی به صورت زیر عمل می‌کند که ابتدا مبدأ بسته، بسته را به سرخوشه خود ارسال می‌کند. از این سرخوشه بسته به گره دروازه فرستاده می‌شود که گره دروازه مذکور این سرخوشه و سرخوشه بعدی که



شکل ۳-۱۰: مثالی از مسیریابی CGSR از گره ۱ تا گره ۱۲

در طول مسیر به مقصد وجود دارد را بهم متصل می‌کند. دروازه، بسته را به سرخوشه می‌فرستد و به همین ترتیب ادامه پیدا می‌کند تا سرخوشه مقصد در این روش بدست آورده شود. سپس سرخوشه مقصد بسته‌ای که در داخل خوشه می‌باشد را به مقصد ارسال می‌کند. شکل ۴-۱۰ یک طرح مسیریابی CGSR را برای خوشه‌های همپوشان نشان می‌دهد. هر گره، جدولی مانند جدول ۴-۴ که در آن اعضای یک خوشه وجود دارد را نگهداری می‌کند. جدول اعضای خوشه، از هر گره نگاشتی به سرخوشه مربوط به آن گره دارد. گره‌ها، جدول اعضای خوشه را بطور دوره‌ای ارسال می‌کنند و پس از اینکه سایر گره‌ها آن را دریافت کردند، با استفاده از الگوریتم DSDV

⁸³. Cluster-head gateway switch routing

جداول خود را بروز می‌کند. علاوه بر این، هر گره یک جدول مسیریابی مانند جدول ۴-۶ را هم نگهداری می‌کند که گام بعدی برای رسیدن به خوشه هر مقصد و معیار آن مسیر در آن مشخص شده است.

جدول ۳-۴ نمونه ای از یک ورودی جدول اعضای خوشه

Node	Cluster-head	Seq#
12	11	100

جدول ۳-۶ نمونه ای از یک ورودی جدول مسیریابی در گره ۱

Cluster-head	Next Hop	Metric	Seq#
11	2	7	71

هر دو جدول فوق، برای پاک کردن مسیرهای قدیمی و جلوگیری از تشکیل حلقه‌ها، شماره ترتیبی دارند. در دریافت یک بسته، هر گره طبق جدول اعضای خوشه و جدول مسیریابی، نزدیکترین سرخوشه را در طول مسیر رسیدن به مقصد، پیدا می‌کند. سپس گره از جدول مسیریابی خود برای پیدا کردن گام بعدی کمک می‌گیرد تا بتواند به سرخوشه انتخاب شده در گام اول برسد و بسته را به آن ارسال کند [۱].

مهمترین مزیت CGSR این است که فقط مسیرهای به سمت سرخوشه نگه داشته می‌شوند و این موضوع بدلیل سلسله مراتبی بودن مسیریابی می‌باشد و سرباره کمتری درمقایسه با flooding اطلاعات مسیریابی داریم ولی سرباره دیگری برای نگهداری خوشه‌ها داریم. از نقاط ضعف این پروتکل می‌توان به دو مورد زیر اشاره کرد. ۱. بعضی گره‌ها مثل سرخوشه و گره‌های دروازه، نسبت به سایر گره‌ها محاسبات بیشتر و مسئولیت بیشتری در ارسال اطلاعات دارند.

۲. مشکل دیگر CGSR هزینه‌های بالای ناشی از نگهداری خوشه می‌باشد. مخصوصاً اینکه هر گره نیاز دارد که بطور متناوب جدول اعضای خوشه را ارسال کند و بر اساس تغییرات پیش آمده، آن را بروز کند [۱].

۳-۱۳-۸ پروتکل STAR^{۸۴}

این پروتکل نیز بر مبنای الگوریتم حالت لینک استوار است. هر مسیریاب یک درخت منبع^{۸۵} - مجموعه‌ای از لینک‌هایی که مسیرهای ترجیحی به مقصدها را دارند را نگهداری می‌کند. برخلاف اغلب روش‌های مسیریابی جدولی، برای پروتکل STAR از پیام‌های پرIODیک، جهت به‌روز کردن همسایگانش استفاده نمی‌کند. این پروتکل

⁸⁴. Source-tree adaptive routing

⁸⁵. Source tree

مقدار سرباره انتشاری به شبکه برای تبادل اطلاعات مسیریابی را، با استفاده از LORA^{۸۶} کاهش می‌دهد. علاوه بر این از ORA^{۸۷} در صورت نیاز پشتیبانی می‌کند. این عمل از روند به هنگام‌رسانی متناوب که در الگوریتم حالت لینک وجود دارد جلوگیری می‌کند. در نتیجه به هنگام‌رسانی‌های حالت لینک تنها در صورتی که اتفاق خاصی روی دهد صورت می‌پذیرد روش STAR کوششی است برای ایجاد همان عملیات مسیریابی که سایر پروتکل‌های بر اساس جدول انجام می‌دهند. علاوه بر این بازده پهنای باند آن نیز، به اندازه پروتکل‌های جدولی یا بیشتر از آن می‌باشد برای آنکه بتوان چنین کاری را انجام داد.

بهینه‌کردن مسیر مورد نیاز کنار گذاشته شده است و مسیرها اجازه دادند که برای حفظ پهنای باند غیر بهینه باشند. با وجود این STAR منوط به یک پروتکل در حال انجام است که MH‌های همسایه را به خوبی و با اطمینان دنبال می‌کند. آنرا می‌توان به همراه پیغام‌های پررودیک به کار برد، اما برای اینکار هیچ الزامی وجود ندارد. علاوه بر این لایه ارتباطی باید یک همه پخشی قابل اعتمادی را ایجاد کند در غیر این صورت از این خاصیت فقط به همراه یک قانون مسیریابی اضافی استفاده کرد.

این پروتکل به خوبی برای شبکه‌های گسترده استفاده می‌شود که کاهش قابل توجهی در عرض باند مصرفی را ایجاد می‌کند. این پروتکل حافظه و سرباره پردازشی را در شبکه‌های با تحرک زیاد لازم دارد، چرا که هر گره بخشی از گراف پیکربندی شبکه (که از درخت منبع به دست می‌آید) را مورد استفاده قرار می‌دهد [1].

۳-۱-۹ پروتکل MMWN^{۸۸}

شبکه از طریق سلسله مراتب خوشه بندی نگهداری می‌شود. هر خوشه دو نوع گره سیار دارد: سویچ‌ها و نقطه‌های پایانی^{۸۹}. هر خوشه دارای مدیر موقعیت (LM)^{۹۰} است. شکل ۴-۱۱ را ببینید. تمامی اطلاعات در پایگاه‌های داده توزیع شده دینامیکی ذخیره می‌شود. مزیت این پروتکل در این است که تنها LM‌ها به هنگام‌رسانی موقعیت و پیدا کردن مسیر را انجام می‌دهند که سرباره مسیریابی در مقایسه با الگوریتم‌های مبتنی بر جدول رایج (مانند DSDV و WRP) کاهش می‌یابد. مدیریت موقعیت مرتبط با ساختار سلسله مراتبی شبکه، یافتن موقعیت و به‌هنگام‌رسانی آن پیچیده است. به این علت که در یافتن موقعیت و فرآیند به هنگام‌رسانی، پیام‌ها بایستی از درخت سلسله مراتبی عبور کنند [4][۱].

۳-۱-۱۰ پروتکل TBRPF^{۹۱}

^{۸۶}. Least overhead routing approach

^{۸۷}. Optimum routing approach

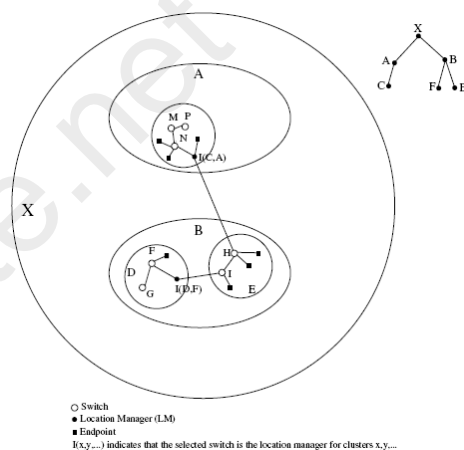
^{۸۸}. Multimedia support in mobile wireless networks

^{۸۹}. endpoints

^{۹۰}. Location manager

^{۹۱}. Topology broadcast reverse path forwarding

یک پروتکل مسیریابی مبتنی بر حالت لینک است که از مسیریابی پرش به پرش استفاده می‌کند و از مفهوم RPF^{۹۲} برای انتشار بسته‌های به‌هنگام رسانی در مسیر معکوس استفاده کرده و حداقل تعداد پرش‌ها را برای رسیدن به مقصد مشخص می‌کند. پروتکل TBRPF مشکل اطلاعات توپولوژی همه‌پخشی را که شامل هزینه‌های لینک و حالت up/down است را برای همه گروه‌های شبکه ارتباطی در نظر می‌گیرد. این اطلاعات همراه با الگوریتم انتقال مسیر می‌تواند به وسیله هر گره مورد استفاده قرار گیرد تا کار محاسبه مسیرها برای تمامی جهات محاسبه شد. در TBRPF هر گره‌ای سرباره را با گزارش درخت منبع به گره‌های همسایه کمینه می‌کند. قسمت‌های قابل گزارش هر درخت منبع با گره‌های همسایه از طریق پیام‌های hello متناوب تبادل می‌یابد. پیام‌های hello تفاضلی تنها تغییرات حالت گره‌های همسایه را گزارش می‌کنند. لذا پیام‌های hello در TBRPF کوچک‌تر از پروتکل‌هایی است که اطلاعات کامل حالت لینک را گزارش می‌دهند [۱].



شکل ۳-۱۱: مثالی از روند خوشه بندی سلسله مراتبی در MMWN

به عبارت دیگر این امر، هدف مسیریابی بر پایه و حالت لینک می‌باشد. اکثر پروتکل‌های مسیریابی حالت لینک که شامل (OSPF) می‌باشند بر پایه Flooding هستند در این پروتکل‌ها هر به روز حالت لینک به هر گروه از شبکه فرستاده می‌شود. اگرچه Flooding در شبکه‌های با اتصالات پهنای باند بالا مفید است اما درصد قابل ملاحظه‌ای از پهنای باند لینک را در شبکه‌های ادهاک موبایل در جایی که شبکه دارای لینک‌هایی بالایی با پهنای باند نسبتاً کم دارد، استفاده می‌نماید. هزینه ارتباطی اطلاعات توپولوژی همه پخش می‌تواند در صورتی که پیام به‌روزکننده در طول درخت‌های گسترده فرستاده شود، کاهش یابد، با این وجود هزینه‌های ارتباطی اضافی برای نگهداری این درخت‌ها وجود خواهد داشت.

پروتکل TBRPF بر پایه الگوریتم ERPF می‌باشد که پیام‌های تولیدی به وسیله منبع موردنظر، در مسیر معکوسی، در طول درخت گسترده‌ای که دارای جهت است، به صورت همه پخشی فرستاده می‌شود که خود این درخت بوسیله کوتاه‌ترین مسیرها از تمام گره‌ها به منبع تشکیل می‌گردد.

CRPF به وسیله الگوریتم مسیریابی برای هر گره i ، $pi(v)$ را براساس کوتاه‌ترین مسیر تامقصد (یا منبع همه پخشی) v را بدست می‌آورد. در نتیجه گره $pi(v)$ به عنوان والد گره i در درخت همه پخشی که منبع آن v است شناخته می‌شود. هر گره والد خود را از این انتخاب مسیر آگاه می‌کند در نتیجه هر والد از وضعیت فرزندانش آگاه می‌شود. گروه i ، پیام همه پخشی که از v منشاء شده است را از طریق والد $pi(v)$ دریافت می‌کند و در ادامه آن را به فرزندانش می‌فرستد.

ERPF در هنگامی که کوتاه‌ترین مسیرها به صورت دینامیکی عوض می‌شوند، قابل اطمینان نیست. در حقیقت به خاطر قابل اطمینان نبودن **ERPF** الگوریتم مسیریابی به کار رفته برای توپولوژی همه پخشی نباید به **ERPF** وابسته باشد. **TBRPF** از مفهوم **ERPF** به همراه شماره‌دادن متوالی برای بدست‌آوردن قابلیت اطمینان تخمین کوتاه‌ترین مسیرها براساس اطلاعات توپولوژی شبکه، که از درخت همه پخشی گرفته شده است، استفاده می‌کند. هنگامی که کوتاه‌ترین مسیرها تخمین زده شد، هر گره منبع، وضعیت ارتباطات به‌روز شده خود را بر تمام مسیرهایی که از آن خارج می‌شوند، با در نظر گرفتن کوتاه‌ترین مسیرهای مورد انتظار می‌فرستد، بنابراین یک درخت همه پخشی جداگانه، برای هر منبع شکل می‌گیرد، استفاده از کوتاه‌ترین مسیرهای مورد انتظار در عوض الگوریتم درختهای کوتاه‌ترین مسیر که براساس قیمت ارتباط نهاده شده‌اند، تغییرات کمتری را در درختهای همه پخشی و تغییرات کمتری در قیمت ارتباط را نتیجه می‌دهد.

TBRPF از تضاد مرغ و تخم‌مرغ رنج می‌برد. این الگوریتم درخت‌های همه‌پخشی را براساس اطلاعات دریافت شده از خود درختها محاسبه می‌کند. بنابراین صحت **TBRPF** زیر سؤال می‌رود. با این حال نشان داده شده است که هر **MH** توپولوژی صحیح را در یک زمان محدود که از **TBRPF** استفاده می‌کند، می‌داند. نتیجه می‌شود که **TBRPF** یک الگوریتم ساده است که ترافیک کنترلی و بروز رسانی کمتر را در مقایسه با **Flooding** ایجاد می‌کند و بنابراین به طور خاص در شبکه‌هایی که دارای تغییرات توپولوژی زیاد و دارای پهنای کم هستند، مفید است [16].

۳-۱۳-۱۱ پروتکل DREAM^{۹۳}

پروتکل مسیریابی **DREAM** نسبت به پروتکل‌هایی که تا به حال توضیح داده شده‌اند از روش متفاوتی استفاده می‌کند. در پروتکل **DREAM** هر گره مختصات جغرافیایی^{۹۴} خود را از راه **GPS** می‌داند. این مختصات به صورت متناوب بین گره‌ها تبادل می‌یابد و در جدول مسیریابی که جدول موقعیت^{۹۵} نامیده شده ذخیره می‌شود. در این پروتکل گره اگر فرستنده را **S** و گره مقصد را **D** بنامیم، گره **S** بسته را به تمامی همسایگان تک‌پرشه خود وارد می‌کند که این همسایگان در مسیر بین **S** و **D** قرار دارند. به منظور تعیین این مسیر، یک گره، منطقه‌ای را که احتمالاً شامل گره **D** است و منطقه مورد انتظار نامیده می‌شود، را محاسبه می‌کند. منطقه مورد انتظار به

^{۹۳}. Distance routing effect algorithm for mobility

^{۹۴}. Geographical coordinates

^{۹۵}. Location table

صورت یک دایره در اطراف گره D است که توسط گره S شناسایی می‌شود. از آنجا که ممکن است اطلاعات در مورد این موقعیت کهنه شود. شعاع R برای منطقه مورد انتظار به صورت معادله $V_{MAX}(t_1 - t_0)$ محاسبه سرعتی است که یک گره می‌تواند در شبکه های ادهاک موبایل حرکت می‌کند. با داشتن منطقه مورد انتظار، مسیر منتهی به D به وسیله خطی که از S به D کشیده می‌شود و همچنین زاویه Φ بدست می‌آید. گره‌های همسایه نیز همان فرایندها را با توجه به اطلاعاتی که از گره D دارند انجام می‌دهند. اگر یک گره، یک همسایه تک پرشه در مسیر مورد نداشته باشد، یک فرایند بازیافت I باید انجام شود، البته این فرایند بازیافت جزئی از ویژگی‌های DREAM نمی‌باشد. از مزیت‌هایی که DREMA به خاطر ویژگی‌های خود دارد یکی از مزیت تبادل اطلاعات موقعیت نسبت به تبادل کامل حالت لینک و اطلاعات بردار فاصله در این حالت که پهنای باند کمتری را مصرف می‌کند. در این پروتکل، سرباره مسیریابی کاهش می‌یابد، چرا که پیام‌های به هنگام‌رسانی طبق تحرک و اثر فاصله منتشر می‌شود این بدان معنی است که گره‌های اینان نیازی به ارسال هیچ پیام به‌روزرسانی ندارند [۱].

۳-۱۳-۲) پروتکل‌های مسیریابی تخت تک‌پخشی بر اساس نیاز

۳-۱۳-۲-۱) پروتکل مسیریابی منبع پویا DSR^{۹۶}

این پروتکل به کلاس پروتکل‌های مبتنی بر درخواست نیز تعلق داشته و به گره این اجازه را می‌دهد که به صورت دینامیک و پویا یک مسیر را در بین جهش‌های متعدد شبکه شناسایی کند و از طریق آن به مقصد برسد و مسیر رسیدن به برخی نقاط مقصد، فقط وقتی تعیین می‌شود که کسی درخواست ارسال اطلاعات به مسیر مذکور و مقصد مورد نظر را داشته باشد [۱].

در مسیریابی هر بسته لیست کامل و منظمی از گره‌هایی که باید از آنها عبور کند را دارا می‌باشد. DSR از هیچ پیام پررودیک مسیریابی استفاده نمی‌کند و از این طریق سرباره پهنای باند شبکه را کاهش داده، انرژی باتری را حفظ کرده و مانع از ایجاد تعداد زیادی به روزکننده‌های مسیریابی در سرتاسر شبکه ادهاک می‌شود. در عوض DSR مبتنی بر حمایت صورت گرفته از سوی لایه MAC می‌باشد. هر بسته‌ای بایستی آدرس کاملی (هر پرش درمسیر) از منبع به مقصد را حمل کند که در شبکه‌های وسیع کارآمد نمی‌باشد. این امر به این علت است که مقدار سرباره و متناسب با آن مصرف پهنای باند با افزایش قطر شبکه افزایش می‌یابد [4][17-18]. دومدل اصلی کاربری تحت DSR عبارتند از کشف مسیر و نگهداری مسیر.

کشف مسیر

در حالت کلی کشف مسیر هنگامی رخ می‌دهد که یک گره نیازمند ارسال پیام به گره دیگری باشد که از طریق مسیرهای شناخته شده موجود در دسترس نمی‌باشد. برای این منظور گره مبدأ یک بسته درخواست مسیر که دارای آدرس مقصد، شماره ترتیبی می‌باشد را به گره‌های همسایه خود ارسال می‌کند. گره‌های همسایه با دریافت بسته مذکور، شماره ترتیبی و آدرس مبدأ را ذخیره می‌کند و آدرس خود را به مسیر آینده اضافه می‌کند و بسته را دوباره ارسال می‌کند و این کار تا زمانی که بسته مذکور به مقصد برسد ادامه خواهد یافت. کشف مسیر مکانیزمی است که به وسیله آن یک گره به نام **S** که می‌خواهد بسته‌ای را به یک مقصد به نام **D** بفرستد، یک مسیر مرجع را به **D** دریافت می‌دارد. برای برقراری یک مسیر مرجع **S** یک درخواست مسیریابی **PREQ** را به همراه یک **ID** منحصر به فرد درخواست همه‌پخش می‌کند که ممکن است به وسیله میزبانهایش در یک رنج انتقال بی‌سیم دریافت شود.

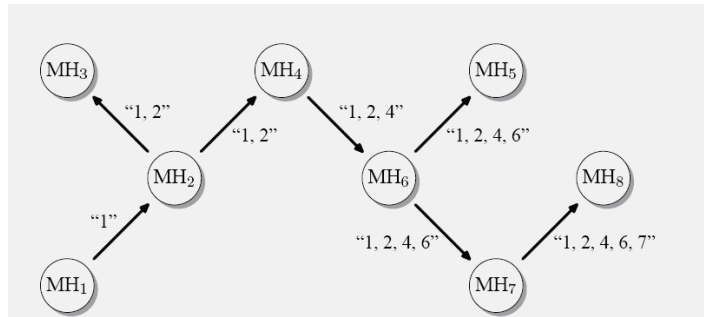
وقتی این پیام درخواست به مقصد یا گره‌ای برسد که اطلاعات مسیریابی به سمت آن مقصد را داراست آن گره یک پاسخ مسیر **PREP** را ارسال می‌دارد که این پاسخ دارای اطلاعات مربوط به مسیر می‌باشد. حافظه پنهان مسیر که در هر گره نگهداری می‌شود، مسیرهایی که آن گره شناخته یا در طول زمان تولید سرباره کرده است را ثبت می‌کند تا بدین وسیله سرباره ایجاد شده در مرحله کف مسیر را کاهش دهد. ثبت شماره ترتیبی در هر گره به این منظور می‌باشد که اگر بسته مشابه دیگری دریافت شد، آن را به عنوان یک نسخه کپی محسوب کرده و از ارسال آن جلوگیری کند، در ضمن این گره‌ها شماره گام را هم ذخیره می‌کنند تا از آن در مسیر برگشت و ارسال پاسخ مقصد به مبدأ استفاده کنند. وقتی که یک گره یک بسته درخواست مسیر را دریافت کند براساس مراحل ذیل درخواست را مورد پردازش قرار می‌دهد.

۱- اگر هر جفت آنها (آدرس آغازگر و **ID** درخواست) برای این درخواست مسیر در لیست میزبانانی که اخیراً درخواست‌هایی دریافت داشته‌اند، وجود داشته باشد، در این صورت، این بسته درخواست مورد پردازش قرار نگرفته و دور انداخته می‌شود.

۲- اگر آدرس میزبان‌ها در بخش ثبت مسیر آن پیام درخواست، وجود داشته باشد، پردازش نمی‌شود و دور انداخته می‌شود.

۳- اگر غیر از موارد بالا باشد و اگر گره، همان هدف موردنظر باشد، ثبت مسیر کامل شده و پاسخ مسیر ارسال می‌شود.

۴- در غیر این صورت گره به ثبت مسیر موجود در بسته درخواست مسیر، ضمیمه شده و دوباره همه پخش می‌شود. در شکل ۴-۱۲ نمونه‌ای از این کشف مسیر آورده شده است [۱].



شکل ۳-۱۲: نمونه‌ای از کشف مسیر در پروتکل DSR

نگهداری مسیر

بخش دوم یعنی نگهداری مسیر پس از کشف مسیر آغاز می‌شود. در خلال این بخش، همه اطلاعات ارسالی از طریق مسیرهای کشف شده قبلی به مقاصد خود فرستاده می‌شود. در واقع نگهداری مسیر مکانیزمی است که به وسیله آن یک ارسال کننده بسته (S)، تشخیص می‌دهد که آیا توپولوژی شبکه طوری تغییر کرده است که دیگر نمی‌تواند از آن مسیر به سمت مقصد D استفاده کند. وقتی آن که قسمت که وظیفه نگهداری مسیر را دارد، مشکل دار بودن یکی از مسیرها را تشخیص دهد، مثلاً اگر در طی مدت ارسال یک مسیر قطع شود، پیغام خطای مسیر RERR را به گره مبدأ می‌فرستد. در نتیجه تمام مسیرهایی که شامل آن مسیر می‌شوند نادیده گرفته می‌شوند و یک مسیر جدید را درخواست می‌کند. این امر به این دلیل است که یک میزبان که در یک مسیر مرجع قرار گرفته از منطقه تحت پوشش بی‌سیم خارج شده و یا ارتباط آن قطع می‌شود. بنابراین آن مسیر دیگر قابل استفاده نمی‌باشد و این قطع ارتباط به وسیله کنترل فعالانه تأییدات یا به صورت غیرفعال از طریق حرکت به مد شناسایی، انجام می‌شود.

در DSR گره‌ها می‌توانند مسیر چندگانه را در مخزن مسیر^{۹۷} ذخیره کنند. گره منبع مخزن مسیر را برای مسیر صحیح قبل از آغاز کشف مسیر چک کرده و در صورتی که مسیر صحیحی یافت شود نیازی به کشف مسیر نمی‌باشد که این عمل برای شبکه‌هایی که دارای تحرک کم می‌باشند بسیار مفید است و چون مسیرها در مخزن مسیر ذخیره می‌شوند برای مدتی طولانی معتبر است.

برای بالابردن کارایی و کاهش سرباره، چند روش برای بهینه‌سازی الگوریتم DSR به کار می‌رود. یکی از این روش‌های بهینه‌سازی استفاده کامل از مسیر پنهان است. داده‌های موجود در هر گره میزبان ممکن است به هر ترتیبی ذخیره شده باشند ولی در مسیرهای فعال هر مسیر دارای درختی از مسیرهاست که از خود آن ریشه می‌گیرد. یک host هنگامی که به وجود یک مسیر جدید پی ببرد می‌تواند به معلومات خود، در حافظه پنهان مسیر خود، اضافه کند.

⁹⁷. Route cache

اگر یک host داده دریافتی را به host های دیگر بفرستد این host منتظر دریافت داده‌های مسیر کامل داده تا مقصد می‌ماند. اگر یک host این مسیر را به host موردنظر برگرداند، این host مسیر را به گره‌های قبلی اطلاع می‌دهد و در نتیجه آن گره‌ها مسیر را در حافظه پنهان خود اضافه می‌کنند، بنابراین همه اطلاعات مسیرهای شبکه به صورت ذاتی همه پخش می‌شوند.

از روش‌های دیگر بهینه‌سازی بازتاب مسیرهای کوتاه‌تر است. در این وضعیت هنگامی که دو گره به اندازه کافی به هم نزدیک‌تر می‌شوند ممکن است مسیر کوتاه‌تری، بین آنها به وجود آید، بنابراین بهتر است که ارتباط از طریق مسیر کوتاه‌تر ایجاد شود.

روش دیگر بهینه‌سازی، اداره کردن تکه‌های مختلف شبکه است. یک خطای معمول که در شبکه‌های ادهاک اتفاق می‌افتد این است که هنگامی که بین دو گره که از هم دورند نیاز به برقراری ارتباط باشد و تعداد گره کافی برای ایجاد یک رشته ارتباطی بین آنها نباشد، نیاز به یک الگوریتم کشف مسیر پیدا می‌شود، به علت دوربودن دو گره، داده‌های درخواست مسیر بسیاری در شبکه پخش می‌شود که خیلی از آنها غیرسازنده است. برای حل این مشکل می‌توان شبکه را به قسمت‌های مختلف تقسیم کرد و الگوریتم‌های مسیریابی را در داخل این قسمت‌ها اداره کرد.

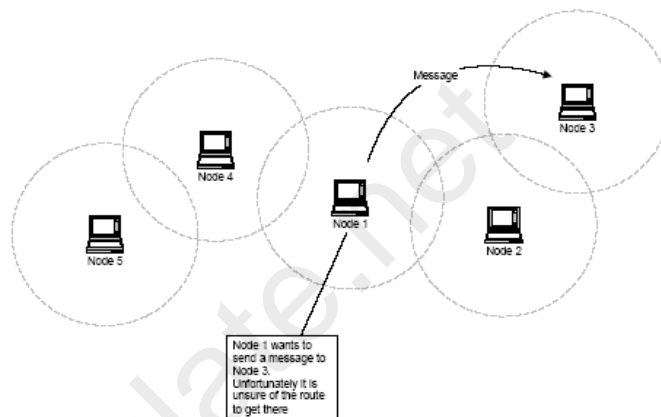
DSR خطوط متقارن را پشتیبانی می‌کند. (مقارن به مفهوم این است که هزینه و زمان رفت و برگشت یکی است). همچنین در **DSR** هنگامی که یک مسیر غیرقابل دسترس تشخیص داده شود و در همان زمان مسیر جایگزینی دیگری برای آن پیدا می‌شود، مسیر جایگزین می‌تواند جایگزین مسیر قبلی در جداول مسیریابی شود. این پروتکل برای شبکه‌هایی که کمتر از ۱۰۰ گره دارند مناسب است. مزیت **DSR** این است که گره‌ها نیاز به تبادل‌های متناوب یا پیام‌های hello ندارند که گره‌ها می‌توانند به مد استراحت جهت ذخیره توان وارد شوند. این امر باعث ذخیره مقدار قابل توجهی پهنای باند در شبکه می‌شود [۱] 17-18

۳-۲-۳-۲) مسیریابی^{۹۸} AODV

یک پروتکل مسیریابی بنابه درخواست است که به کامپیوترهای سیار، یا گره‌ها، برای انتقال پیام‌ها در میان همسایه‌هایشان اجازه می‌دهد تا با هم به طور مستقیم ارتباط برقرار کنند. **AODV** مانند تمام پروتکل‌های نیازی راکتیو تنها وقتی مسیری مورد نیاز می‌باشد عملیات یافتن مسیر انجام می‌گیرد و البته وقتی که مسیر یافته شد، تا زمانی که مورد احتیاج باشد آن مسیر ذخیره شده، باقی می‌ماند و پس از آن دیگر نگه داشته نشده و برای دفعه بعد باید دوباره یافته شود. **AODV** مطمئن می‌سازد که مسیرها شامل حلقه نباشند و برای پیدا کردن کوتاه‌ترین مسیر ممکن سعی می‌کند. همچنین **AODV** قادر به تغییرات دستی در مسیرها است و اگر یک خطا وجود داشت می‌تواند مسیرهای جدید خلق کند.

⁹⁸. Ad-hoc On Demand Distance

این پروتکل سعی می‌کند به سرعت با شرایط خطوط پویا هماهنگ شود و حجم کم پردازش و حافظه مورد نیاز از خواص این الگوریتم می‌باشد. این پروتکل از شماره ترتیبی مقصد برای اطمینان از عدم ایجاد حلقه‌ها استفاده می‌کند و همچنین مشکل شمردن تا بینهایت را که در پروتکل کلاسیک بردار فاصله موجود بود، حل می‌کند. در AODV هر گره دارای شماره ترتیبی مخصوص به خود است که به طور یکنواخت افزایش می‌یابد. این شماره وقتی افزایش می‌یابد که گره مربوطه متوجه تغییری در توپولوژی شبکه گردد. AODV قابلیت مهم دیگری نیز داراست و آن قابل استفاده بودن این پروتکل در هر سه نوع ارتباطات تک‌پخشی، چندپخشی و انتشاری می‌باشد [19].



شکل ۳-۱۳: یک شبکه پنج گره‌ای در یک شبکه ادهاک

شکل ۳-۱۳ برپا کردن پنج گره در یک شبکه بی‌سیم را نشان می‌دهد. دایره‌ها میزان ارتباط برای هر گره را نشان می‌دهد. علت اندازه محدود این است که هر گره تنها می‌تواند با گره‌های بعدی در آن ارتباط برقرار کند

کشف ۹۹ و برقراری مسیر ۱۰۰

برقراری و کشف مسیر با استفاده از AODV به صورت نیازی می‌باشد و این کار با چرخه‌های درخواست و پاسخ انجام می‌گیرد. هرگاه گره‌ای بخواهد بسته‌ای به گره مقصد دیگری بفرستد، در حال حاضر مسیری بین این دو گره وجود ندارد، عملیات کشف مسیر را آغاز می‌کند. برای پیدا موقعیت، گره مبدأ یک بسته خاص بنام تقاضای مسیر (RREQ)^{۱۰۱} ساخته و آن را به طور فراگیر منتشر می‌کنند.

در این بسته درخواست موارد ذیل موجود است: آدرس IP ای گره مبدأ، شماره ترتیبی جاری مبدأ، Broadcast ID جاری مبدأ، تعداد پرش‌ها تا مبدأ، آخرین شماره ترتیبی گره مقصد که گره مبدأ از اطلاع دارد. قالب بسته تقاضای مسیر در شکل ۴-۱۴ نشان داده شده است. این بسته مشخص می‌کند چه کسی در

⁹⁹. Route Discovery

¹⁰⁰. Route Establishment

¹⁰¹. ROUTE REQUEST

جستجوی چه کسی است. بعد از اینکه گره مبدأ بسته RREQ را ساخت، آنرا به تمام همسایگانش می‌فرستد و آنها نیز هر یک بسته را به همسایگان خود می‌فرستند و تا جایی این روند ادامه می‌یابد که بسته یا به گره مقصد برسد و یا به گره دیگری برسد که در حال حاضر مسیری تا گره مقصد دارد. این بسته همچنین شامل یک شماره تقاضا ID است. این شناسه در واقع یک شمارنده محلی است که در هر گره وجود دارد و هرگاه یک بسته تقاضای مسیر منتشر گردید یک واحد به آن اضافه می‌شود.

Route Request		Route Reply	
1	$S \rightarrow * : \langle RREQ, S, D, 0, S \rangle$		$D \rightarrow C : \langle RREP, S, D, 0, D \rangle$
2	$A : d[S] = 1, n[S] = S$ $A \rightarrow * : \langle RREQ, S, D, 1, A \rangle$	6	$C : d[D] = 1, n[D] = D$ $C \rightarrow B : \langle RREP, S, D, 1, C \rangle$
3	$B : d[S] = 2, n[S] = A$ $B \rightarrow * : \langle RREQ, S, D, 2, B \rangle$	7	$B : d[D] = 2, n[D] = C$ $B \rightarrow A : \langle RREP, S, D, 2, B \rangle$
4	$C : d[S] = 3, n[S] = B$ $C \rightarrow * : \langle RREQ, S, D, 3, C \rangle$	8	$A : d[D] = 3, n[D] = B$ $A \rightarrow S : \langle RREP, S, D, 3, A \rangle$
5	$D : d[S] = 4, n[S] = C$	9	$S : d[D] = 4, n[D] = A$

جدول ۳-۷: جدول کشف مسیر در مسیریابی AODV برای شبکه شکل ۳-۱۳

ترکیب آدرس مبدأ و فیلد شناسه تقاضا، هویت بسته تقاضای مسیر را به صورت یکتا و منحصر به فرد تعیین کرده و بدین ترتیب گره‌ها قادرند بسته‌هایی که احتمالا به صورت تکراری دریافت می‌شوند را تشخیص داده و حذف کنند. هر گره شمارنده‌های دیگری نیز دارد (source sequence , destination sequence) که هرگاه بسته تقاضای مسیر ارسال شود یک واحد بدان اضافه می‌شود و کاربرد این شمارنده‌ها تشخیص مسیر جدید از مسیر قدیمی است یعنی وقتی دو بسته یکسان در مورد مشخصات یک مسیر دریافت می‌شود این شمارنده مشخص می‌کند که کدام جدیدتر از دیگری است. آخرین فیلد یعنی شمارنده پرش^{۱۰۲} مشخص می‌کند که بسته تاکنون چند گام را طی کرده است. مقدار اولیه این فیلد صفر است و به ازای عبور از هر گره یک واحد به آن اضافه می‌شود.

Source address	Request ID	Destination address	Source Sequence	Dest sequence	Hop count
----------------	------------	---------------------	-----------------	---------------	-----------

شکل ۳-۱۴: قالب بسته ROUTE REQUEST

¹⁰². Hop Counter

بعد از اینکه گره مقصد بسته RREQ را دریافت کرد یک ورودی مسیر برگشت^{۱۰۳} برای گره مبدأ در جدول مسیر خود می‌سازد. این ورودی مسیر برگشت حاوی آدرس IP ی گره مبدأ و شماره ترتیبی و همچنین تعداد جهش‌های لازم برای رسیدن به گره مبدأ و آدرس IP ی گره همسایه‌ای است که RREQ از آن گره به مقصد رسیده است.

باید توجه کرد که هر گره‌ای که RREQ را تنها هدایت کرده است نیز در جدول مسیر خود ورودی جدیدی برای مسیر برگشت اضافه می‌کند و البته با برپا شدن هر مسیر در جدول، یک تایمر شروع به کار خواهد کرد و اگر از آن مسیر تا مدت زمان خاصی^{۱۰۴} استفاده نشد، آن ورودی خود به خود حذف خواهد شد. گره‌ای که RREQ را دریافت می‌کند، اگر گره مقصد باشد و یا اینکه مسیری با شماره ترتیبی بزرگتر یا مساوی شماره ترتیبی درون RREQ به سمت گره مقصد داشته باشد، با پیام RREP به درخواست گره مبدأ پاسخ می‌دهد در پاسخ به تقاضای مسیر گره یک بسته باز پاسخ تقاضا (RREP)^{۱۰۵} مطابق شکل ۴-۱۵ ایجاد می‌کند. فیلدهای آدرس مبدأ و مقصد دقیقاً از بسته تقاضا استخراج و در بسته کپی می‌شود. فیلد گام نیز به صفر تنظیم می‌شود توجه شود که گره‌هایی که تنها پیام RREQ را به جلو هدایت می‌کنند، اگر چند RREQ به آنها برسد فقط اولین پیام را هدایت می‌کنند و بقیه پیام‌ها تنها در صورتی به حساب می‌آید که شماره ترتیبی مقصد بزرگتر از RREQ قبلی باشد یا اینکه شماره ترتیبی مقصد مساوی قبلی باشد ولی تعداد جهش‌های کمتری تا مبدأ نیاز داشته باشد[1]

Source address	Destination address	Destination sequence	Hop count	Lifetime

شکل ۳-۱۵: قالب بسته ROUTE REPLY

شماره‌های ترتیبی:

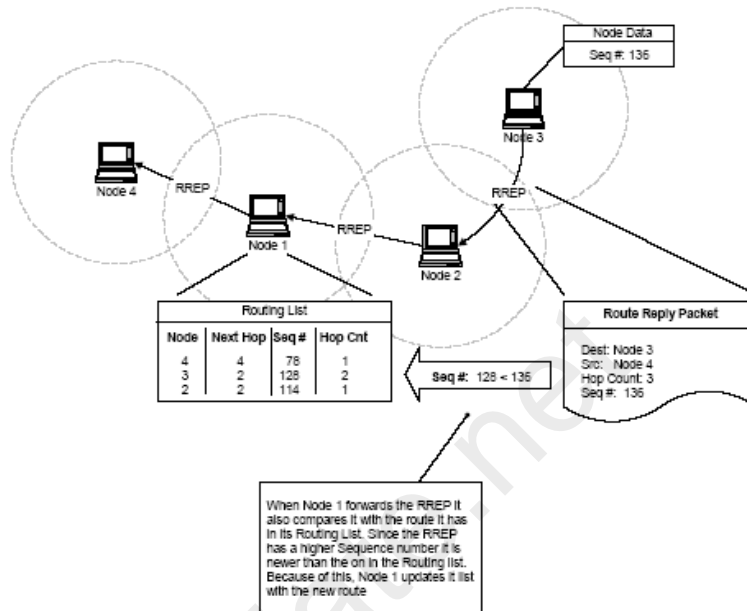
شماره‌های ترتیب مانند نشانه‌های زمانی رفتار می‌کنند. هر زمانی که هر گره پیامی از هر نوع را بیرون می‌فرستد، شماره ترتیبی‌اش را افزایش می‌دهد. هر گره شماره ترتیبی همه گره‌های دیگر را که با آنها صحبت کرده است را ضبط می‌کند. یک شماره ترتیبی بزرگتر بر یک مسیر تازه‌تر (fresher) دلالت می‌کند. این کار

¹⁰³. Reverse Route Entry

¹⁰⁴. Life time

¹⁰⁵. ROUTE REPLY

برای گره‌های دیگر کشف این که کدام اطلاعات دقیق‌تر است ممکن می‌سازد. در شکل ۴-۱۶، گره ۱ در حال ارسال یک RREP برای گرهی ۴ است. گره ۱ توجه می‌کند که مسیر در RREP یک شماره ترتیبی بزرگ‌تری نسبت به مسیر داخل لیست مسیریابی‌اش دارد. سپس گرهی ۱ لیستش را با مسیر جدید به‌روز می‌کند.



شکل ۳-۱۶: گره ۱ در حال ارسال یک RREP برای گرهی ۴ است

مدیریت جداول مسیر^{۱۰۶}

پروتکل AODV نیاز دارد که اطلاعات ذیل را درون هر یک از ورودی‌های جداول مسیر داشته باشد: آدرس IP مقصد، شماره ترتیبی مقصد، تعداد جهش‌ها تا مقصد، آدرس IP ی گره مجاور برای جهش بعدی در مسیر مورد نظر، مدت زمانی که مسیر معتبر است، لیستی از بقیه همسایه‌هایی که از این مسیر استفاده می‌کنند^{۱۰۷}، بافری برای درخواست‌ها که اطمینان می‌دهد، که هر درخواست تنها یک‌بار بررسی می‌شود [۱].

نگهداری مسیر^{۱۰۸}

وقتی مسیری بین یک گره و یک مقصد یافته می‌شود، تا زمانی که لازم باشد این مسیر نگه داشته می‌شود. حرکت گره‌ها در شبکه ادهاک باعث اثرگذاری بر روی مسیرهایی می‌شود که آن گره‌ها در آن مسیر قرار دارند. به

¹⁰⁶. Route Tables Management

¹⁰⁷. Precursor List

¹⁰⁸. Route Maintenance

چنین مسیرهایی، مسیرهای فعال^{۱۰۹} گفته می‌شود. اگر گره مبدأ در طی یک نشست^{۱۱۰} یا ارتباط جابه‌جا شود ممکن است عملیات یافتن و کشف مسیر را دوباره آغاز کند ولی وقتی گره مقصد یا گره‌های میانی جابه‌جا می‌گردند، یک پیام RERR^{۱۱۱} به سمت مبدأ فرستاده می‌شود. این پیام RERR توسط گره‌ای که خط ارتباطی مجاورش شکسته شده و البته خود گره به مبدأ نزدیک‌تر است فرستاده می‌شود. پیام RERR لیست گره‌هایی که در اثر شکست خط ارتباطی دیگر قابل دستیابی نیستند را به مبدأ گزارش می‌دهد. وقتی یک گره می‌خواهد دوباره شروع به کار^{۱۱۲} کند، شماره ترتیبی قبلی خود و همچنین شماره ترتیبی که از دیگر گره‌ها در اختیار دارد، را از دست داده است. اما با توجه به اینکه دیگر گره‌ها ممکن است این گره را بعنوان یک گره فعال بشناسند، از آن بعنوان جهش بعدی خود استفاده می‌کنند و در این صورت امکان بوجود آمدن حلقه‌ها ایجاد می‌شود.

برای جلوگیری از این مشکل هر گره در ابتدای ورود به شبکه، حداکثر به اندازه مدت زمان delete-period صبر می‌کند تا یک بسته RREQ دریافت کند و از روی آن آخرین شماره ترتیبی خود را بروز کند. گرهی مذکور در طی این مدت به بسته‌های مسیریابی هیچ پاسخی نمی‌دهد و اگر بسته داده به آن رسید یک پیام RERR به فرستنده می‌دهد.

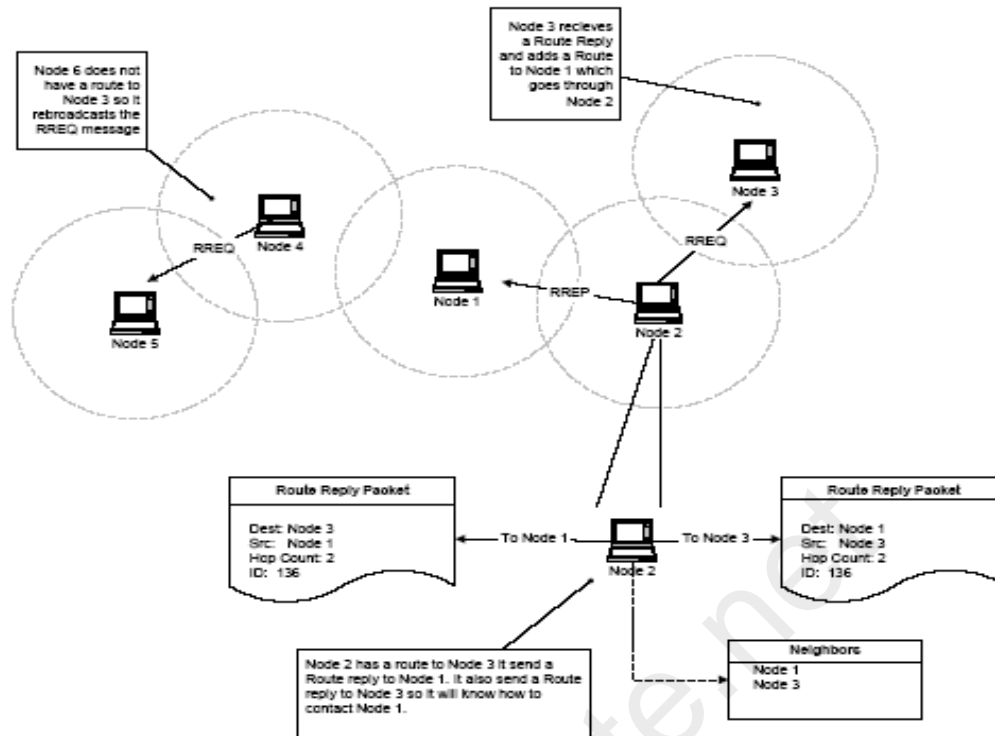
تنها با گره‌هایی می‌توان با آنها ارتباط مستقیم برقرار کرد که همسایه باشند. یک گرهی ردیابی همسایه‌هایش را بوسیله لیست کردن یک پیام hello که هر گره در فاصله معین آن را پخش می‌کند، نگه می‌دارد. وقتی که یک گره احتیاج به فرستادن یک پیام برای گره دیگری که همسایه‌اش نیست را دارد و یک پیام تقاضای مسیر منتشر می‌کند. در مثال شکل ۴-۱۷، گره ۱ قصد ارسال یک پیام برای گره ۳ را دارد. همسایه‌های گره ۱ گره‌های ۲ و ۴ هستند. از آنجایی که گره ۱ نمی‌تواند مستقیماً با گره ۳ ارتباط برقرار کند، گره یک RREQ بیرون می‌فرستد. RREQ بوسیله گره‌های ۲ و ۴ شنیده می‌شود.

¹⁰⁹. Active Path

¹¹⁰. Session

¹¹¹. Route Error Message

¹¹². Actions after Reboot

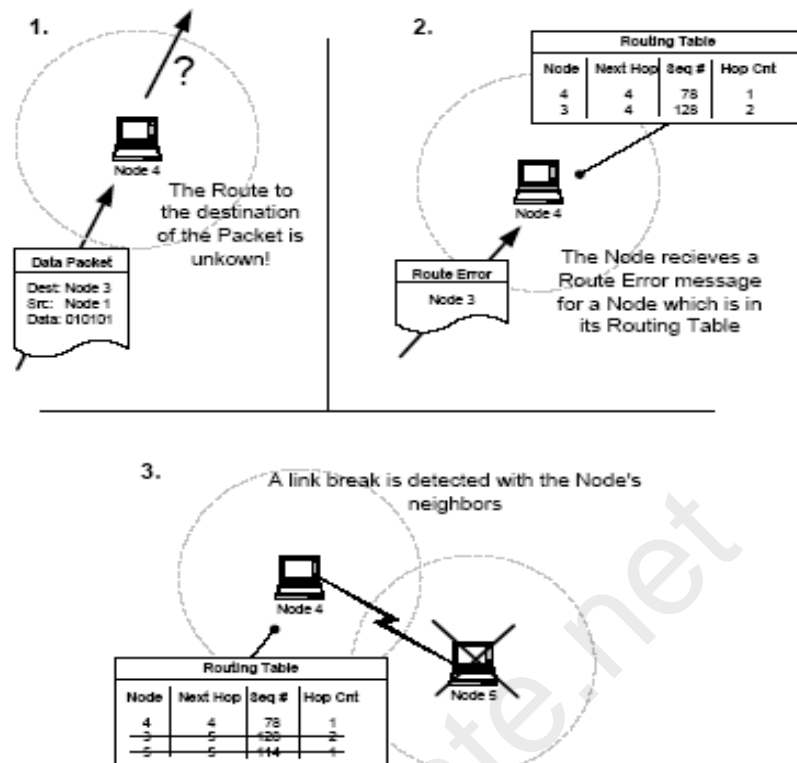


شکل ۳-۱۷: گره ۱ قصد ارسال یک پیام برای گره ۳ را دارد.

زمانی که همسایه‌های گرهی ۱ پیام RREQ را دریافت کردند، آنها ۲ انتخاب دارند، اگر یکی از آنها مقصد هست یک پیام باز پاسخ برای گره ۱ بفرستد یا اگر آنها یک مسیر تا مقصد را می‌دانند، باید دوباره پیام RREQ را برای گروه همسایه‌های خودشان منتشر کنند. اگر گره ۱، یک پاسخ در یک مدت زمان تنظیم شده از آن زمان دریافت نکرد، آن دوباره سوال را با یک طول عمر طولانی‌تر و یک شماره ID جدید خواهد فرستاد. همه گره‌ها از شماره ترتیبی در RREQ برای ضمانت کردن اینکه یک RREQ را دوباره انتشار نمی‌دهند استفاده می‌کنند. در مثال شکل ۴-۱۷ گره ۲ یک مسیر تا گره ۳ دارد و بوسیله فرستادن یک RREP به RREQ پاسخ می‌دهد، گرهی چهار از طرف دیگر مسیری تا گره ۳ ندارد بنابراین دوباره RREQ را منتشر می‌کند.

پیغام های خطا:

هر وقت که یک گره RERR را دریافت می‌کند به جدول مسیریابی توجه می‌کند و همه‌ی مسیرهایی را که شامل گره خراب هستند پاک می‌کند. شکل ۴-۱۸ سه رویداد که تحت آن گره شاید یک پیام RERR برای همسایه‌هایش منتشر کند نشان می‌دهد. در قسمت ۱، گره یک بسته دیتا که در آن ارسال به جلو فرض شده است دریافت می‌کند، اما آن هیچ مسیری برای مقصدی ندارد. مشکل واقعی این نیست که گره مسیر ندارد، بلکه مشکل آن است که تعدادی از بقیه گره‌ها فکر می‌کنند که مسیر درست برای مقصد در آن گره است.



شکل ۴-۱۸: سه رویداد که تحت آن گره شاید یک پیغام RERR برای همسایه‌هایش منتشر کند

در قسمت ۲ گره، یک RERR که باعث می‌شود دست کم یکی از مسیرهایش ناتوان بشود دریافت می‌کند. اگر این اتفاق بیفتد، گره یک RERR شامل همه گره‌هایی که غیر قابل دسترس هستند بیرون می‌فرستد. در قسمت ۳، گره متوجه می‌شود که با یکی از همسایگانش نمی‌تواند ارتباط برقرار کند. زمانی که این اتفاق می‌افتد آن گره به جدول مسیری که همسایه‌ها برای یک جهش بعدی از آن استفاده می‌کنند، توجه می‌کند و آنها را بعنوان غیرمعتبر نشانه می‌گذارد. سپس یک RERR شامل همسایه و مسیرهای غیرمعتبر بیرون می‌فرستد.

۳-۱۳-۲-۳ پروتکل TORA^{۱۱۳}

این الگوریتم مسیریابی بسیار سازگار و بدون حلقه است که بر پایه مفهوم لینک معکوس بنا شده است. این الگوریتم برای به حداقل رساندن عکس‌العمل به تغییرات توپولوژیکی طراحی شده است. نکته کلیدی در طراحی TORA این است که تولید پیام‌های کنترلی بالقوه و دور از دسترس را، از تغییرات توپولوژیکی تفکیک می‌نماید. چنین پیام‌هایی به طور موضعی در دسته کوچکی از گره‌ها نزدیک تغییر بدون نیاز داشتن به ردیف کردن دوباره راه‌حل‌های مسیریابی دینامیکی و مرتبه‌ای پیچیده قرار دارند. بهینه نمودن مسیر (کوتاه‌ترین مسیر) در درجه دوم اهمیت قرار دارد و مسیرهای طولانی در صورتی که از کشف مسیرهای تازه‌تر چشم‌پوشی شود، اغلب مورد استفاده

¹¹³. Temporally ordered routing algorithm

قرار می‌گیرد. همچنین TORA دارای قابلیت چند مسیریابی می‌باشد. هر گره با توجه به مقصد، ارتفاعی دارد که به وسیله پروتکل مسیریابی محاسبه می‌گردد. TORA برای کار در محیط‌های شبکه‌ای بسیار متحرک پیشنهاد شده است [18].

TORA از منبع سرچشمه گرفته و مسیر چندگانه‌ای را برای هر جفت از منبع و مقصد موردنظر فراهم می‌آورد. کامل کردن آن گره‌ها نیاز به نگهداری اطلاعات مسیریابی در مورد گره‌های کناریشان دارد. این پروتکل سه کار پایه‌ای را انجام می‌دهد. خلق مسیر، نگهداری مسیر، پاک کردن مسیر.

برای هر گره به هر هدف در شبکه گراف جهت‌دار بودن (DAG) چرخه مجزایی نگهداری می‌شود. وقتی یک گره نیازمند یک مسیر برای مقصد ویژه است که مسیرش خواسته شده است. این بسته در سرتاسر شبکه پخش می‌شود تا وقتی که یا به یک مقصد بر سه یا به یک مقصد برسد یا یک گره بینابینی مسیری را به همان مقصد طلب کند.

دستورالعمل (UNCQUERY) یک بسته به روزکننده را همه پخش می‌کند و ارتفاع آن را متناسب با مقصد لیست می‌کند. به مجرد اینکه بسته در طول شبکه تکثیر پیدا کرد. هر گره‌ای که به روزکننده را دریافت کرد، ارتفاع خود را تا میزانی بیش از ارتفاع آن همسایه‌ای که به روزکننده را دریافت کرده، بالا می‌برد. این امر باعث ایجاد مجموعه‌ای از سینک‌های جهت‌دار از نخستین ارسال‌کننده (QUERY) به سمت آن گره‌ای که در ابتدا اولین به روزکننده یا ایجاد کرده است، می‌شود.

وقتی یک گره متوجه شود مسیری به سمت یک مقصد دیگر معتبر نیست طوری که ارتفاعش را تصحیح می‌کند که این ارتفاع نسبت به همسایگان آن گره به یک ماکزیمم موضعی می‌رسد و یک بسته به روزکننده را نیز انتقال می‌دهد. وقتی که یک گروه هیچ همسایه‌ای با ارتفاع معین نسبت به این مسیر نداشته باشد، در این صورت MH سعی می‌کند که به همان شکلی که در بالا توضیح داده شد، یک مسیر جدید را کشف کند. وقتی که یک گره فضایی را در یک شبکه پیدا کند، یک بسته‌ای Clear ایجاد می‌کند که حالت مسیریابی را حفظ نموده و مسیرهای نامعتبر را از شبکه حذف می‌کند.

TORA به روی IMEP لایه‌گذاری می‌شود که لازم است تا انتقال مطمئن و منظم تمامی پیام‌های کنترل مسیریابی را، از یک گره به تمامی همسایگانش میسر سازد و همچنین برای هشدار دادن به پروتکل‌های مسیریابی در هر جایی که ارتباط آن گره به یکی از همسایگانش ایجاد یا قطع می‌شود، لازم است. جهت به حداقل رساندن سه باره IMEP^{۱۱۴} بسیاری از پیام‌های کنترل TORA و IMEP را قبل از اینکه ارسال شوند در درون یک بسته منفرد جمع‌آوری می‌کند. هر قفل دارای یک شماره ردیف و یک لیست پاسخ سایر همسایگانی است که از آنها هیچ ACK دریافت شده است و فقط این گره‌ها با دریافت آن، قفل مذکور را شناسایی می‌کنند. IMEP هر قفل را با دوره‌های خاصی انتقال می‌دهد و اگر لازم باشد این انتقال را برای چند دوره ادامه می‌دهد، که پس از آن، TORA از هر لینکی که قطع شده، شناسایی می‌شود و گره‌ها نیز آن را تأیید می‌کنند.

¹¹⁴. Internet MANET Encapsulation

متریک TORA شامل ۴ عنصر که عبارتند از:

۱. زمان منطقی قطع ارتباط
۲. ID منحصر به فرد یک گره که سطح مرجع جدیدی را یک بیت شناسایی انعکاس
۳. پارامتر تنظیم تکثیر
۴. ID خاص و منحصر به فرد یک گره

ارزیابی:

همانطور که عنوان شد پروتکل مسیریابی TORA بر مبنای پروتکل LMR می‌باشد و از روند لینک معکوس و تعمیر لینک استفاده می‌کند. این پروتکل DAG ایجاد می‌کند که مشابه فرایند درخواست / پاسخ مورد استفاده LMR می‌باشد [۱].

۳-۱۳-۴ پروتکل LAR^{۱۱۵}

پروتکل LAR براساس مسیریابی منطقه‌ای تعریف نشده است بلکه یک الگوریتم مسیریابی است که با استفاده از اطلاعات موقعیت برای بهبود کشف مسیر بر حسب نیاز می‌باشد. این پروتکل بر مبنای Flooding عمل می‌کند. با فرض اینکه گره‌ها، اطلاعات موقعیت دیگر گره‌ها را داشته باشند، LAR این اطلاعات موقعیت را برای محدود کردن Flooding برای یک محدود مشخص، استفاده می‌کند. در LAR طرفی به این است که اطلاعات موقعیت گره‌ها از طریق GPS بدست می‌آید. در LAR سعی می‌شود که سرباره مسیریابی کم شود، برای آگاهی از این عملکرد باید دو مفهوم منطقه انتظار (مقصد) و منطقه تقاضا را بدانیم. منطقه انتظار:

برای بیان مفهوم اول، منطقه انتظار به منطقه مقصد هم نامیده می‌شود، گره S که نیاز به یافتن یک مسیر به گروه D دارد، فرض می‌شود که گره S می‌داند که گره D در زمان t_0 در موقعیت L بوده است. یعنی منطقه انتظار گره D از نظر گره S در زمان جای t_1 ، همان محدوده‌ای است که انتظار می‌رود. شامل D باشد، برای مثال اگر گره S بداند که گره D، با سرعت متوسط V حرکت کند، با این تفاسیر S می‌تواند فرض کند که منطقه انتظار یک منطقه دایره‌ای به شعاع $V(t_1 - t_0)$ به مرکز موقعیت L که از قبل آنرا می‌دانیم. اگر سرعت واقعی بیشتر از میانگین باشد، مقصد در زمان t_1 ممکن است خارج از منطقه انتظار قرار بگیرد. بنابراین منطقه انتظار فقط یک تخمینی است که توسط گره S برای مشخص کردن منطقه‌ای که گره D در زمان t_1 در آن محدوده با

¹¹⁵. Location-aided routing

احتمال زیاد، در آن محدوده قرار دارد. چنانچه گره S موقعیت قبلی گره D را نداند نمی تواند موقعیت منطقه انتظار را به طور قابل قبولی مشخص کند.
منطقه تقاضا:

منطقه تقاضا براساس منطقه انتظار تعریف می شود. گره S برای تقاضای مسیر یک منطقه تقاضا تعریف می کند. اگر آن گره متعلق به منطقه تقاضا باشد و گره درخواست مسیر دارد، یک تقاضای مسیر می فرستد. برای افزایش احتمال اینکه تقاضای مسیر به گره D برسد، منطقه تقاضا باید شامل منطقه انتظار شود. یعنی منطقه انتظار زیرمجموعه ای از منطقه تقاضا باشد، به عبارتی منطقه تقاضا ممکن است خیلی بزرگتر از منطقه انتظار باشد. الگوریتم تشریح شده در بالا طرح LAR^1 بوده که در بستر درخواست مسیر خود، مختصات مقصد را ذخیره می کند و می فرستد ولی LAR^2 که اصلاح شده طرح اول است شامل دو مشخصه است یعنی علاوه به مختصات، فاصله را هم در خود دارد.

فرض می کنیم که S موقعیت D را در لحظه t_0 می داند (XD, YD) و زمانی که کشف مسیر به وسیله گره S شروع می شود، t_1 باشد، که در اینجا $t_1 \geq t_0$ است. گره S فاصله خود را از (XD, YD) محاسبه می کند و این فاصله را به عنوان یک اطلاعات بیشتر به همراه مختصات در بسته درخواست مسیر خود ارسال می کند. هر دو طرح سه باره ارسال شده کنترلی را محدود کرده و پهنای باند را ذخیره می کنند و در اغلب موارد کوتاه ترین مسیر را به مقصد مشخص می کنند. عیب آنها این است که گره ها باید GPS مجهز باشند [20][1].

۳-۱۳-۲-۴ پروتکل ABR^{۱۱۶}

ABR یک پروتکل مسیریابی منبع آغاز شونده است که در تکنیک درخواست پاسخ برای مشخص سازی مسیریابی که به سمت مقصد لازم است، استفاده می کند. این روش یک ایده کاملاً متفاوت در مسیریابی موبایل است. ABR شکل حلقه دار بودن، قفل شدن های بی نهایت و بسته های اضافی را ندارد. در نظر علمی ABR مسیری با عمر زیاد را برای شبکه ADHOC ایجاد می کند.

در ABR هر مسیر بر اساس اندازه گیری درجه انجمنی (بستگی) انتخاب می شود. هرگز، به صورت مکرر، یک پیام کنترلی که وجود آن گره را تولید می کند. وقتی که این پیام بر گره های همسایه می رسد، گره های همسایه جداول انجمنی خود را به زور می کنند. برای هر پیام کنترلی دریافت شده، سمبل انجمنی بین گره کنونی و گره ای که پیام را صادر کرده افزایش می یابد. درجه زیاد (کم) پایدار انجمنی وضعیت تحرک کم (زیاد) گره را مشخص می کند. هنگامی که یک گره از نزدیکی گره دیگر خارج می شود، سمبل های بستگی آنها دوباره بازسازی می شود. ABR دارای ۳ فاز است. کشف مسیر، بازسازی مسیر و حذف مسیر [۱].

کشف مسیر:

¹¹⁶. Associatively-based routing

کشف مسیر توسط یک جستجوی همه پخشی و منتظر ماندن برای بازپاسخ آن انجام می‌شود (BQ – Replay). گره‌ای که دنبال مسیر است، یک پیام BQ را به MH‌هایی که به گره مقصد مسیر دارند، همه پخشی می‌کند. همه گره‌هایی که BQ را دریافت می‌کنند، اگر گره مقصد نباشد آدرس سمبل‌های انجمنی خود را نسبت به همسایگانش به همراه داده‌های QOS به انتهای بسته جستجو اضافه می‌کند. گره جانشین (گره‌ای که بسته جستجو را از گره بالایی خود دریافت کرده است) با دریافت بسته جستجو، تمام داده‌های سمبل انجمنی آنرا حذف می‌کند و فقط داده‌های مربوط به خود گره بالایی خودش را نگه می‌دارد. در نتیجه هر گره‌ای که به مقصد می‌رسد، سمبل‌های انجمنی گره‌هایی که در مسیر، از مبداء تا مقصد قرار دارند، را در خود دارند. اگر چند مسیر دارای درجه پایداری انجمنی باشند، مسیری که دارای کمترین پرش است، انتخاب می‌شود.

در نهایت گره مقصد یک بسته REPLAY به گره مبداء، در همین مسیر، برمی‌گرداند. گره‌هایی که REPLAY را منتشر می‌کنند به مسیرهایشان، یک برچسب اعتبار اضافه می‌کنند. بقیه مسیرها غیر فعال می‌مانند و احتمال اینکه بسته‌های اضافی به مقصد برسند، بسیار کم می‌شود.

بازسازی مسیر:

ممکن است قسمتی از کشف مسیر، حذف مسیر معتبر، به روزرسانی مسیرهای معتبر و کشف مسیر جدید بر اساس جابجایی گره‌ها را شامل شود. جابجایی گره مبداء یک روند BQ – REPLAY را سبب می‌شود. هنگامی که گره مقصد حرکت می‌کند، گره‌ای که دقیقاً بالای آن قرار دارد، مسیر خود را پاک می‌کند و توسط یک جستجوی محلی (LQ) مشخص می‌کند که آیا مقصد هنوز قابل دسترسی هست یا خیر. اگر گره مقصد بسته LQ را دریافت کند، به آن توسط بهترین مسیر باز پاسخ می‌دهد، در غیر این صورت با نادیده گرفتن اولین گره، به سراغ گره بالایی آن می‌رود و از طریق آن گره جستجوی محلی برای پیدا کردن آن گره مقصد صورت می‌گیرد. این روند تا زمانی که یک مسیر به گره مقصد پیدا شود، صورت می‌گیرد.

در اینجا یک پیام RN به گره بالایی فرستاده می‌شود تا مسیر غیر معتبر را حذف کرده و جستجوی محلی را آغاز کند. اگر این روند از نصف مسیر تا مبداء تجاوز کند، جستجوی محلی ادامه پیدا نمی‌کند و یک جستجوی همه پخشی جدید از گره مبداء آغاز می‌شود. این روند انتخاب مسیر توسط ABR ممکن است که منجر به کوتاهترین مسیر نشود ولی نیاز به بازسازی کمتری دارد، که این امر عرض باند بیشتری را برای انتقال داده در اختیار شبکه می‌گذارد [۱].

۳-۱۳-۲-۶ پروتکل SSA^{۱۱۷}

پروتکل SSA از نسل ABR می‌باشد. بر خلاف الگوریتم‌های که تاکنون توضیح داده شد، یک پروتکل مسیریابی افقی مبتنی بر پایداری سیگنال، که یک الگوریتم نیازی است، مسیرها را بر اساس قدرت سیگنال انتخاب می‌کند. این انتخاب مسیر، ارتباطات قوی‌تر و پایدارتر را بر می‌گزیند.

¹¹⁷. Signal stability based adaptive routing

SSA از ترکیب دو پروتکل DRP و SRP استفاده می‌کند. کاربرد DRP برای دستیابی به جدول پایداری سیگنال (SST) و جدول مسیریابی (RT) منطقی است. بعد از پردازش بسته و بروزرسانی جداول مربوط DRP بسته را به SRP تحویل می‌دهد. SRP مربوط به هرگره اگر در گیرنده خود سیگنال قوی احساس کند، بسته را در حافظه stack ذخیره می‌کند و در غیر این صورت در جدول مسیریابی خود دنبال گره مقصد می‌گردد و بسته را با آن می‌فرستد. اگر در جدول خود مسیری پیدا نکند، یک روند جستجوی مسیر را آغاز می‌کند یک تفاوت بین الگوریتم کشف مسیر در SRP در مقایسه با AODV این است که درخواست مسیرها در SSA تنها به گره‌هایی فرستاده می‌شوند که بین دو گره کانال ارتباطی قوی موجود باشد.

اگر در طی یک زمان مشخص هیچ بازپاسخی به گره مبدا داده نشود، گره مبدا PREF خود را در Header بسته خود تغییر می‌دهد تا نشان دهد که کانال‌های ضعیف هم قابل قبول هستند. وقتی که یک ارتباط قطع شده در شبکه تشخیص داده می‌شود پیام‌های قطع ارتباط مسیر فرستاده می‌شوند و روند جستجوی مسیر دیگری شروع می‌شود و همچنین مبدا یک پیام پاک‌کننده به تمام گره‌ها می‌فرستد تا آنها را از وضعیت گره قطع شده آگاه نماید.

در SSA مانند ABR مسیرهایی که با این پروتکل انتخاب می‌شود، ممکن است کوتاه‌ترین مسیرها به مقصد نباشند و اشکال آن عدم پاسخ گره‌های واسطه به درخواست‌های مسیر است که تأخیر زیادی را قبل از کشف مسیر ایجاد می‌کند، چرا که گره مقصد مسوول انتخاب مسیر برای انتقال داده است. اشکال دیگر آن عدم تعمیر لینک‌های تخریب شده است، با توجه به اینکه گره منبع بایستی لینک‌های خراب را در نظر بگیرد، منجر به تأخیر زیادی می‌شود [۱].

۳-۱۳-۲-۷) پروتکل LMR^{۱۱۸}

برای مشخص‌سازی مسیرها از روش flooding استفاده می‌کند و گره‌ها در LMR مسیرهای چندگانه‌ای را به مقصد نگهداری می‌کنند. مزیت آن در این است که هر گره‌ای تنها اطلاعات مسیریابی گره‌های همسایه را نگهداری می‌کند که به معنای جلوگیری از تأخیر زیاد و سرباره‌ای که متناسب نگهداری مسیرهای کامل است، می‌باشد. علاوه بر این LMR ممکن است مسیرهای ناصحیح موقتی ایجاد کند که تأخیرهای زیادی را ایجاد می‌کنند [۱].

۳-۱۳-۳-۸) پروتکل CBRP^{۱۱۹}

علی‌رغم پروتکل‌های مسیریابی راکتیو، در CBRP گره‌ها به صورت سلسله مراتبی سازمان‌دهی می‌شوند. گره‌ها در این پروتکل بر خوشه‌هایی گروه‌بندی می‌شوند. در الگوریتم مسیریابی توسط خوشه‌بندی فرض بر این است که مسیرهای کلی یک طرفه‌اند و مسیرهایی که داخل خوشه‌ها تعریف می‌شوند، دو طرفه‌اند. هر خوشه دارای

¹¹⁸. Light-weight mobile routing

¹¹⁹. Cluster-based routing protocol

یک سرخوشه^{۱۲۰} است که انتقال داده را درخوشه و دیگر خوشه‌ها هماهنگ می‌کند. هر خوشه به وسیله یک خط دو طرفه با سرخوشه ارتباط دارد. خوشه‌ها ممکن است همپوشانی داشته باشد و یا کاملاً جدا باشند ولی سرخوشه‌ها نمی‌توانند کنار یکدیگر باشند و انتقال داده نیز از طریق این سرخوشه‌ها و هماهنگی آنها با هم، انجام می‌پذیرد.

برای تبادل اطلاعات بین خوشه‌های مجاور سرخوشه‌ها باید از وضعیت گره‌های دروازه مطلع باشند. بنابراین هر سرخوشه از وضعیت کلی همه خوشه‌ها با خبر است. وقتی یک گره در یک خوشه مسیری را به مقصد پیدا نمی‌کند، نشان‌دهنده این است که مقصد در خوشه دیگری واقع شده است و یک پیام درخواست مسیر را به سرخوشه ارسال می‌کند. سرخوشه با دریافت این پیام شماره شناسایی خود را به آن اضافه می‌کند و آن را در میان خوشه‌های مجاور هم پخش می‌کند. اگر خوشه همسایه یک گره دروازه باشد، درخواست سرخوشه را به سرخوشه مقصد می‌فرستد.

هنگامی که درخواست به مقصد می‌رسد، گره مقصد آنرا به گره مبدا باز می‌گرداند. هر سرخوشه میانی، در این پیام باز گردانده شده، مسیر خود را اضافه می‌کند، که این خود باعث بهینه‌سازی مسیر یابی می‌شود. برخلاف DSR در این الگوریتم، هنگام اضافه‌شدن مسیرهای جدید دیگر نیاز به ذخیره‌سازی اطلاعات توپولوژی در سطح شبکه نیست.

مزیت CBRP این است که تنها سرخوشه‌ها اطلاعات مسیریابی را تبادل می‌کنند و از این رو سرباره به شبکه تحمیل می‌شود. ولی سرباره کنترلی ارسالی بسیار کمتر از روشهای رایج flooding است، ولی شبیه تمام طرح های مبتنی بر خوشه بندی نگهداری اطلاعات خوشه ها باعث تحمیل سرباره‌ای به شبکه می‌شود. مشکل دیگری که وجود دارد، حلقه‌های مسیر موقتی است. این امر به این خاطر است که برخی گره‌ها اطلاعات پیکربندی متناقضی را به علت تاخیر انتشاری طولانی حمل می‌کند [1].

۳-۱۳-۲-۹ پروتکل RDMAR^{۱۲۱}

یک الگوریتم وفقی و با قابلیت تغییر اندازه شبکه است و در شبکه‌های بزرگ موبایلی که دارای تغییرات توپولوژیکی متوسط هستند به کار برده می‌شود. کلید طراحی آن در واکنش محلی به خرابی در ارتباط در یک محدوده کوچک در شبکه است. این رفتار که مورد دلخواه ما است با استفاده از یک مکانیزم کشف مسیر flooding که کشف ریزمسیرهای مرتبط (RDM) نامیده می‌شود، بدست می‌آید.

برای تحقق بخشیدن این کار، یک الگوریتم تکرار شونده، فاصله نسبی گره‌ها را با توجه به فاصله نسبی قبلی و متوسط میزان حرکت گره اطلاعاتی در مورد زمان باقیمانده ارتباط، تخمین می‌زند. بر اساس این فاصله نسبی محاسبه شده یک جستجوی سنگین در منطقه محلی محدود شده، که مرکز آن گره مبدا است، آغاز می‌شود. ماکزیمم شعاع انتشار این جستجو برابر، فاصله نسبی تخمین‌زده شده می‌باشد. در RDMAR بسته‌ها بین

¹²⁰. Cluster-head

¹²¹. Relative distance micro-discovery *ad hoc* routing

جایگاه‌هایی در شبکه مسیریابی می‌شوند، که هر کدام از این جایگاه‌ها دارای جداول مسیریابی مختص به خود می‌باشد. هر جدول مسیریابی تمام مقصدهای قابل دستیابی را در خود دارد.

به عنوان مثال برای هر مقصد L این جدول شامل یک مسیریاب پیش فرض است که گره بعدی که انتظار می‌رود بتواند با گره L ارتباط برقرار کند، را مشخص می‌کند. قسمت RD که فاصله نسبی تخمین زده شده بین گره L و گره کنونی را در خود دارد همچنین قسمت TLU نشان‌دهنده زمانی است که آن گره برای آخرین بار اطلاعات مسیریابی به سمت مقصد L را دریافت کرده است. قسمت RT ، زمان باقی مانده پیش از آنکه مسیر نامعتبر شناخته شود را ثبت می‌کند و یک بخش نشانده مسیر، که نشان می‌دهد که آیا مسیر منتهی به L ، فعال است یا خیر. $RDMAR$ از دو الگوریتم اصلی تشکیل شده است. اولین الگوریتم اکتشاف مسیر بدین صورت عمل می‌کند که وقتی که یک پیام بدست آمده برای گره مقصد L ، وارد گره A می‌شود و هیچ مسیری در دسترس نیست، گره A یک مرحله کشف مسیر را شروع می‌کند. در این جا گره A دو راه پیش دارد.

یا با یک درخواست مسیر که درون شبکه به صورت وسیع پخش شود که در این حالت بسته‌های درخواست مسیر در سراسر شبکه همه پخش می‌شود. یا اینکه اکتشاف مسیر را به ناحیه کوچک‌تری از شبکه محدود کند. البته به این شرط که نوعی مدل پیش‌بینی مکان برای مقصد L بتواند ایجاد شود. در مورد حالت دوم مبداء اکتشاف مسیر که همان گره L است، به جدول مسیریابی خودش مراجعه می‌کند تا اطلاعات مربوط به فاصله نسبی قبلی‌اش با L و همچنین زمان هدررفته از وقتی که برای آخرین بار اطلاعات مسیریابی را از L گرفته بود، را بدست آورد.

این زمان را زمان حرکت^{۱۲۲} می‌نامیم. براساس این اطلاعات و با فرض یک سرعت متوسط که آن را $(MICRO-VELOCITY)$ می‌نامند و یک رنج انتقال متوسط (که به آن $(MICRO-VANJE)$ می‌گویند) در آن- صورت گره A قادر خواهد بود که فاصله نسبی جدیدش تا مقصد L را بر حسب تعداد واقعی جهش‌ها تخمین بزند بدین منظور گره A انحراف مسیرش از DST را در طول زمان T_{motion} محاسبه کرده و نتیجه را با فاصله نسبی قبلی‌اش تصحیح می‌کند.

نگهداری مسیر:

یک گره حد واسط به نام A براساس بسته داده‌هایی که دریافت کرده ابتدا سرپیام^{۱۲۳} مسیریابی را پردازش نموده و سپس بسته را به پرش بعدی خود وارد می‌کند علاوه بر این گره A یک پیام مربع را می‌فرستد تا بیازماید که آیا لیک دو جهته می‌تواند با گره قبلی ایجاد شود یا خیر.

بنابراین $RDMAR$ مسیرهای دو جهته را فرض نمی‌کند اما در مقابل گره‌ها امکان برقراری ارتباط دوجهته را بررسی می‌کنند. اگر گره را به علت عدم دسترسی به هیچ مسیر یا یک خطای فوروادسازی بر اثر قطع ارتباط در مسیر داده‌ها قادر به فوردوارد کردن بسته نباشد، در این صورت گره A اقدام به بازانتقال همان بسته داده‌ها می‌کند و این کار را تا حداکثر دفعات ممکن انجام می‌دهد. با این حال اگر قطع ارتباط همچنان ادامه داشته باشد، گره L

¹²². Time motion

¹²³. Header

یک فرایند کشف مسیر را آغاز می‌کند. در RDMAR محاسبه فاصله بین منبع و مقصد به کاهش سرباره مسیریابی منجر می‌شود و از این رو بسته‌های درخواست مسیره تعداد مشخصی پرش محدود می‌شود. به عبارت بهتر روند کشف مسیر روی منطقه‌ای محلی متمرکز می‌شود. مزیت RDMAR عدم نیاز به تکنولوژی موقعیت‌یابی مانند GPS برای مشخص‌سازی الگوی مسیر است. در صورتی که از قبل منبع و مقصد ارتباط داشته باشند، فرآیند فاصله نسبی کشف مسیر روی می‌دهد و درغیراین صورت پروتکل مانند flooding عمل می‌کند [۱].

۳-۱۳-۲-۱۰ پروتکل ARA^{۱۲۴}

تمام سعی این پروتکل کاهش سرباره مسیریابی با تطبیق بر جستجوی غذا در رفتار مورچه‌ها است. به این صورت که از لانه خود شروع کرده و به سمت غذا حرکت کرده و درپشت سر خود اثری که فرومون^{۱۲۵} نامیده می‌شود را به جا می‌گذارند. بدین وسیله مسیری که مورچه طی کرده مشخص می‌شود و دیگران نیز می‌توانند این مسیر را دنبال کنند. این روند تا هنگامی که فرومون از بین نرفته وجود دارد. همانند DSR و AODV این پروتکل نیز دارای دو فاز کشف مسیر و نگهداری مسیر است. در روند کشف مسیر یک FANT^{۱۲۶} در شبکه منتشر می‌شود (مشابه RREQ). در هر پرش هر گره‌ای مقدار فرومون را با توجه به تعداد گره‌هایی که FANT به آنها رسیده است محاسبه می‌کند. سپس گره‌ها FANT را به گره‌های همسایه فوروارد می‌کنند و در صورتی که به مقصد برسد BANT^{۱۲۷} تولید می‌کند که به منبع برگشت داده می‌شود. هنگامی که گره منبع این بسته تولیدی توسط مقصد را دریافت کرد مسیری ایجاد شده و انتقال داده آغاز می‌شود. برای نگهداری مسیرها هنگامی که بسته داده بین گره‌های واسطه منتقل می‌شود مقدار فرومون افزایش می‌یابد. برای تعمیر یک لینک تخریب شده، گره‌ها ابتدا جدول مسیریابی را چک می‌کنند و در صورتی که مسیری پیدا نشد گره‌های همسایه را برای مسیر دیگری مطلع می‌کنند. مزیت این استراتژی در این است که سایز هریک از FANT و BANT ها کوچک است که میزان سرباره به ازای بسته کنترلی کمینه است. فرآیند کشف مسیر که بر مبنای flooding است و به این معناست که پروتکل قابلیت مقیاس پذیری را در صورتی که تعداد گره‌ها و جریان‌ها زیاد شود ندارد [۱][۶۵].

۳-۱۳-۲-۱۱ پروتکل FORP^{۱۲۸}

این پروتکل سعی در کاهش اثر تخریب لینک به علت حرکت گره‌ها دارد، به این صورت که با پیش بینی لینکی که در حال تخریب است لینک جایگزینی را انتخاب می‌کند. هنگامی که مسیری به مقصد مشخصی نیاز باشد و مسیر در دسترس نباشد، پیام Flow_REQ مشابه RREQ در DSR در شبکه منتشر می‌شود. هر گره‌ای که این پیام را دریافت می‌کند با استفاده از GPS به محاسبه زمان انقضای لینک (LET)^{۱۲۹} همراه پرش

¹²⁴. Ant-colony-based routing algorithm

¹²⁵. Pheromone

¹²⁶. Forwarding ANT

¹²⁷. Backward ANT

¹²⁸. Flow oriented routing protocol

¹²⁹. Link Expiration Time

قبلی می‌پردازد و این مقدار را به بسته Flow_REQ که دوباره منتشر می‌شود الصاق می‌کند. هنگامی که بسته Flow_REQ به مقصد رسید، زمان انقضای مسیر (RET)^{۱۳۰} با استفاده از کوتاه‌ترین LET برای هر گره محاسبه شده و بسته Flow_SETUP به منبع برگشت داده می‌شود. درحین انتقال داده نیز هر گره‌ای میزان LET خود را به بسته الصاق می‌کند که این امر به مقصد امکان پیش بینی زمان خرابی لینک را می‌دهد. هنگامی که انقضای مسیر توسط مقصد مشخص شود، پیام Flow_HANDOFF با flooding تولید و منتشر می‌شود و هنگامی که منبع این پیام را دریافت می‌کند بهترین مسیر را در بسته Flow_HANDOFF برای تبادل ۱۳۱ جریان بر مبنای اطلاعاتی چون RET و شماره پرش‌ها مشخص می‌کند و منبع پیام Flow_SETUP را در مسیرهای جدید می‌فرستد. مزیت این مسیریابی نسبت به دیگر پروتکل‌های مسیریابی نیازی توضیح داده شده در این است که تخریب مسیرها را به علت تحرک با نگهداری جریان ثابت داده کمینه می‌کند. از طرف دیگر چون بر مبنای flooding عمل می‌کند مشکلات مقیاس پذیری در شبکه‌های بزرگ را دارد [4].

۳-۱۳-۲ (۱۲-۲-۱۳-۳) پروتکل ROAM^{۱۳۲}

این پروتکل از هماهنگی بین گره‌ها استفاده می‌کند و عملکرد انتشاری^{۱۳۳} دارد و از مشکل جستجوی بی‌کران^{۱۳۴} در برخی پروتکل‌های واکنش دار با توقف جریان جستجوی مقصدهایی که در دسترس نمی‌باشد جلوگیری می‌کند. مزیت دیگر این است که هر مسیریاب در جدول مسیریابی خود برای هر مقصد بسته‌های داده را به آنها جریان می‌دهد. این امر باعث بهبود پهنای باند و جدول مسیریابی می‌شود. در صورتی که فاصله مسیریاب از مقصد بیشتر از حد آستانه‌ای شود پیام‌های به هنگام رسانی را به گره‌های همسایه انتشار می‌دهد. این امر باعث افزایش اتصال پذیری شبکه می‌شود و در شبکه‌هایی که تغییرات زیادی دارند به گره‌ها اجازه ورود به مد استراحت برای کنترل توان را نمی‌دهد [4].

۳-۱۳-۳ (۳-۱۳-۳) پروتکل‌های مسیریابی دورگه

این پروتکل‌ها طبیعت جدولی و نیازی دارند. برای گره‌های نزدیک از نگهداری مسیر، براساس جدول و برای گره‌های دور از استراتژی کشف مسیر استفاده می‌کند. بیشتر پروتکل‌های دورگه بر مبنای موقعیت می‌باشند که به معنای بخش‌بندی شبکه می‌باشد. دسته دیگری گره‌ها را به درخت‌ها و خوشه‌ها تقسیم می‌کنند [۱].

¹³⁰. Route Expiration Time

¹³¹. handoff

¹³². Routing on-demand acyclic multi-path

¹³³. Diffusing operation

¹³⁴. Search to infinity

۳-۱۳-۱) پروتکل^{۱۳۵} ZRP

مثالی است از یک هیبرید تشکیل شده از طرح‌های نیازی و جدولی.

ZRP میدان دید روش براساس جدول را فقط تا همسایگی موضعی گره محدود می‌سازد در حالی که جستجوی انجام شده در سرتاسر شبکه را می‌توان با استفاده از گره‌های انتخاب شده در شبکه با راندمان بالا انجام داد که این روش برخلاف انتخاب تمامی گره‌های موجود در شبکه است.

می‌توان گفت که **ZRP** انتخاب یک همسایه براساس پروتکل است. یک گره که به صورت پرواکتیو از **ZRP** استفاده می‌کند مسیرهای منتج به مقصد را در میان همسایگان حفظ می‌کند که به آن یک ناحیه مسیریابی گویند و به عنوان مجموعه‌ای از گره‌ها که حداقل فاصله آنها در پرش از گره موردنظر کمتر از پارامتری است که شعاع منطقه نامیده می‌شود.

هر گره شعاع منطقه خود را حفظ کرده و بین نواحی (مناطق) همسایه همپوشانی وجود دارد. ساخت یک منطقه مسیریابی نیازمند این است که اول از همه یک گره همسایگانش را بشناسد. همسایه به عنوان گره‌ای تعریف می‌شود که مستقیماً با گره موردنظر بتواند ارتباط برقرار کند و توسط **NDP** قابل کشف شدن **ZRP** مناطق مسیریابی را در یک جزء جدولی که پروتکل مسیریابی بدون منطقه‌ای (**IERP**) نامیده می‌شود، حفظ می‌کند. گره‌ها دارای منطقه مسیریابی می‌باشند که برد را براساس تعداد پرش‌ها تعریف می‌کند و هر گره‌ای بایستی به صورت جدولی اتصالات شبکه را نگه دارد. از این رو گره‌هایی که داخل منطقه مسیریابی می‌باشند، مسیر دردسترس است و مسیر گره‌های خارج از منطقه هستند به صورت نیازی مشخص شده و سرباره کمتری نسبت به پروتکل های جدولی خالص دارد و تاخیری که در پروتکل‌های نیازی خالص وجود دارد را با کشف سریعتر مسیر نیز کاهش می‌دهد. برای مسیریابی گره‌ای که خارج از منطقه قرار دارد، مسیریابی بایستی تنها با گره‌هایی که درلبه‌های منطقه قرار دارند انجام شود و گره‌های لبه‌ای بر اساس جدول مسیرها را به سمت مقصد نگهداری می‌کنند

IERP به عنوان یک طرح تصحیح شده بردار فاصله در نظر گرفته می‌شود. از سویی دیگر پروتکل مسیریابی بین منطقه‌ای **IERP** مسئول بدست آوردن مسیرهایی به سمت مقصدهایی است که در خارج از منطقه مسیریابی قرار گرفته‌اند. **IERP** از یک مکانیزم درخواست و پاسخ استفاده می‌کند تا مسیرهای خواسته شده را کشف کند تفاوت **IERP** با الگوریتم **flooding** استاندارد آن است که **IERP** از ساختار منطقه مسیریابی با استفاده از فرآیندی که همه پخشی نامیده می‌شود، استفاده می‌کند.

ZRP این سرویس را با استفاده از جزئی که **BRP**^{۱۳۶} نامیده می‌شود انجام می‌دهد. انتشار به صورت **Border cast** بسیار گرانتر از همه پخشی **flooding** است که در سایر پروتکل‌های راکتیو استفاده می‌شود. گره‌ها معمولاً، گره‌های مرزی بیشتری از همسایگان نشان دادند علاوه بر این هر پیام **Border Cast** مجبور است که از پرشهای شعاع ناحیه به سمت مرزها حرکت کند.

¹³⁵. Zone routing protocol

¹³⁶. Border Resalution Protocol

ZRP مکانیزم‌هایی دارد که هزینه مسیریابی Border Cast را کاهش می‌دهد همچنین Border Cast انتخابی مشابه با انتخاب MRP براساس OLSR است به این ترتیب که هر گره زیر مجموعه‌ای از گره‌های مرزی را که پوشش یکسانی را دریافت می‌کنند، انتخاب می‌کند. عیب ZRP در این است که برای منطقه مسیریابی گسترده مانند پروتکل‌های مسیریابی پیشگیرانه خالص عمل می‌کند [21].

۳-۱۳-۲) پروتکل ZHLS^{۱۳۷}

این پروتکل دارای ساختار سلسله مراتبی است. در ZHLS شبکه به مناطق غیرهمپوشانی^{۱۳۸} تقسیم بندی می‌شود و هر گره‌ای دارای ID گره و ID منطقه است که محاسبات آن از طریق GPS انجام می‌شود. ساختار سلسله مراتبی از دو سطح تشکیل شده است: پیکربندی سطح گره و پیکربندی سطح منطقه. از آنجایی که هیچ گونه سرخوشه‌ای برای هماهنگی جریان داده وجود ندارد، درمقایسه با HSR، MWMN و CGSR هیچ سرباره پردازشی برای سرخوشه و مدیر موقعیت تحمیل نمی‌شود و درمقایسه با پروتکل‌های واکنش دار دارای سرباره کمتری است. هنگامی که مقصدی در منطقه دیگری مورد نیاز باشد، گره منبع درخواست موقعیت سطح منطقه را دیگر مناطق می‌فرستد که سرباره کمتری را ایجاد می‌کند. از آنجایی که ID گره و ID منطقه برای مسیریابی لازم است، ZHLS دارای وفق‌پذیری در برابر تغییرات پیکربندی می‌باشد. بدین معنی که در صورتی که مقصد به منطقه دیگری برود نیازی به اطلاعات موقعیت جدید وجود ندارد. اشکال آن این است که تمامی گره‌ها دارای نقشه منطقه استاتیکی باشند و برای کاربردهایی که محدوده جغرافیایی شبکه دینامیکی است کاربرد ندارد [۱].

۳-۱۳-۳) پروتکل DDR^{۱۳۹}

یک پروتکل مسیریابی مبتنی بر درخت است و درخت‌ها از راه پیام‌های راهنما که تنها با گره‌های همسایه تبادل می‌یابد ساخته می‌شوند و از طریق گره‌های دروازه -گره‌هایی که در برد ارسال هستند، ولی به درخت‌های متفاوت تعلق دارند- تشکیل جنگل می‌دهند. هر درختی در جنگل تشکیل منطقه‌ای می‌دهند که با zone ID مشخص می‌شود. از آنجایی که هر گره تنها به یک منطقه (یا درخت) تعلق دارد، شبکه به صورت مناطق غیرهمپوشان تشکیل می‌شود. الگوریتم DDR شش فاز دارد: انتخاب همسایه‌های مطلوب، ساخت جنگل، خوشه-بندی داخل درخت‌ها، خوشه‌بندی بین گره‌ها، اسم دهی مناطق و تقسیم‌بندی مناطق. تمام این فازها بر مبنای اطلاعات دریافتی از پیام راهنما^{۱۴۰} است. همسایه مطلوب دارای بیشترین تعداد همسایه‌ها است، و سپس با اتصال هر گره به همسایه مطلوب خود جنگل تشکیل می‌دهد. برای مشخص سازی مسیرها HARP^{۱۴۱} در بالای DDR

¹³⁷. Zone-based hierarchical link state

¹³⁸. Non-overlapping zones

¹³⁹. Distributed dynamic routing

¹⁴⁰. Beacon message

¹⁴¹. Hybrid ad hoc routing protocol

کار می‌کند و از جدول‌های مسیریابی داخل منطقه ۱۴۲ و بین منطقه‌ای ۱۴۳ تولید شده توسط DDR برای مشخص‌سازی مسیر فعال استفاده می‌کند. برخلاف ZHLS این مسیریابی با نقشه مناطق استاتیکی کار نمی‌کند و نیازی به سرخوشه و گره ریشه برای هماهنگی انتقال نیست. از آنجایی که گره‌های همسایه نسبت به سایر گره‌ها دارای حجم ارسال بیشتری هستند زمان کمتری برای مد استراحت و ذخیره انرژی دارند، علاوه بر این باید به رقابت دستیابی به کانال در کنار همسایه مطلوب دقت شود که تاخیر زیادی دارد. در شبکه‌هایی با ترافیک بالا به کاهش توان عملیاتی می‌انجامد، زیرا برخی بسته‌ها وقتی بافر پر است حذف می‌شود [21]

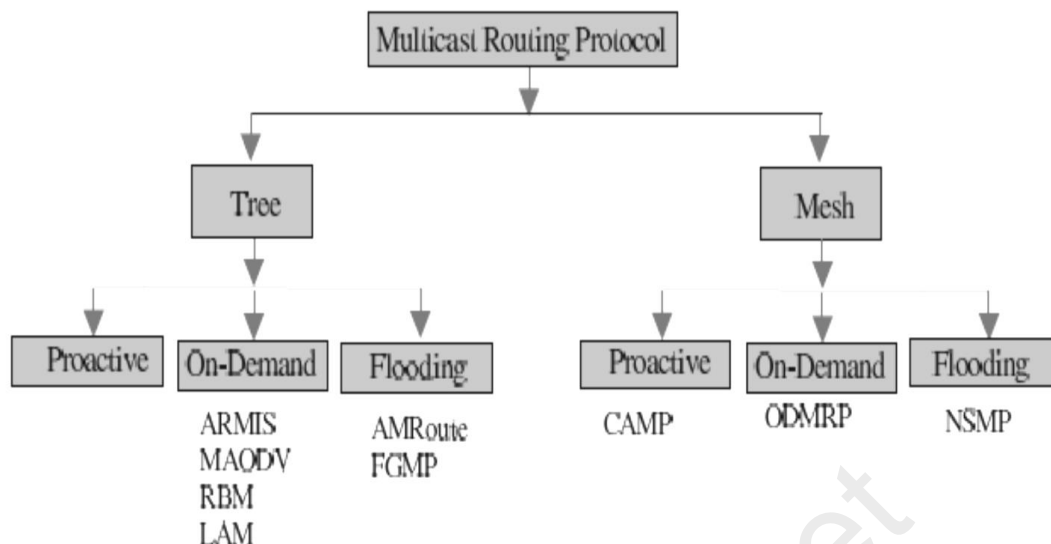
۱۴-۳ پروتکل‌های مسیریابی چند پخشی

۱-۱۴-۳ مقدمه

به الگوریتم‌های مسیریابی که در ارسال‌های چندپخشی مورد استفاده قرار می‌گیرد، مسیریابی چندپخشی گویند. در شبکه‌های ADHOC به دلیل چالش‌های نگهداری مسیر ارتباطی بین منبع و مقصد در سطح وسیعی مورد بررسی قرار می‌گیرند. حتی زمانی که برخی از گره‌های عادی قادر به ادامه مشارکت در ارسال بسته‌ها رو به جلو نیستند، این بسته باید به وسیله گره‌های مسیر دیگری جابجا شوند. پس نتیجه می‌شود که نگهداری مسیرها بین منبع منفرد و چندین مقصد تا اندازه‌ای مشکل‌زاست. با توجه به اهمیت روز افزون چندپخشی به عنوان وسیله‌ای برای کاهش استفاده از پهنای باند جهت توزیع حرمی داده‌ها و نیاز مبرم به ذخیره پهنای باند در طول میدان‌های بی‌سیم، طبیعی است که مسیریابی چندپخشی باید تا اندازه‌ای مورد توجه شبکه‌های ADHOC قرار گیرد [21].

¹⁴². Intra-zone

¹⁴³. Inter-zone



۴

شکل ۳-۱۹: دسته بندی پروتکل های مسیریابی چندپخشی

۳-۱۴-۲ طبقه بندی مسیریابی چندپخشی

همانطور که در شکل ۳-۱۹ دیده میشود سه طبقه بندی از پروتکل های مسیریابی چندپخشی وجود دارد که هر کدام از این سه دسته خود در یکی از دسته های درختی و یا بر اساس مش قرار می گیرد. پروتکل جدولی همانطور که می دانیم مسیرها را برای همه جهت های ممکن از قبل محاسبه می کند و این اطلاعات را در جدول های مسیریابی ذخیره می نماید. برای نگهداری به روز پایگاه اطلاعاتی، مسیریابی به طور پریودیک در امتداد شبکه توزیع می شود. یک مثال از این مورد CAMP می باشد. پروتکل نیازی یا بر پایه نیاز پروتکل هایی اند که مسیرهایی را برای میزبانهای دیگر بر حسب تقاضا خلق می نمایند. این ایده بر پایه مکانیزم درخواستی است که به مقصد می رسد و سپس فاز پاسخ وارد شده و مسیر را می سازد. ODMRP و MAODV مثالهایی از آن می باشند. پروتکل آخر نیز، جریان ساده است. در شبکه هر گره دریافت کننده پیام آنرا به لیستی از همسایه های خود ارسال می کند AMRoute مثالی از آنست [۱].

۳-۱۴-۲-۱ پروتکل AMRoute^{۱۴۴}

^{۱۴۴}. Adhoc multicast routing protocols

از فرضیات استفاده شده در **AMRoute** وجود الگوریتم مسیریابی تک‌پخشی می‌باشد که از ارتباط IP بین همسایه‌ها در درخت مسیریابی استفاده می‌کند. مسیرهای واقعی که یک ارتباط دو طرفه بین دو گروه برقرار می‌کنند ممکن است در واقعیت یکی نباشد، همچنین لازم نیست تمام گره‌های شبکه پروتوکل **AMRoute** را پشتیبانی کنند و همچنین وجود مسیرهای چندپخشی نیز الزامی نیست. در این الگوریتم هم گیرنده و هم فرستنده لازم است در درخت چندپخشی وارد شوند، که این درخت به میزان کمی با مدل چندپخشی معمولی متفاوت است. حداقل یکی از اعضای گروه چندپخشی باید الگوریتم **AMRoute** را پشتیبانی کند ولی در حالت ایده‌آل بهتر است، تمام گره‌ها **AMRoute** را پشتیبانی کنند.

AMRoute برای هر گروه چندپخشی یک درخت توزیع به وسیلهٔ تونلهای تک‌پخشی می‌سازد. پروتوکل دارای دو قسمت کلیدی است، ساختن مش و ساختن درخت. در پروتکل **AMRoute** برای هر گروه حداقل یک هسته منطقی وجود دارد. که این هسته برای ساختن مش و درخت و کنترل سیگنالینگ استفاده می‌شود. هسته‌ها با سیگنالینگ بین خودشان ماندن یا نماندن خود را تعیین می‌کنند. همچنین منظور از یک مش در **AMRoute** گرافی است که هر عضو گروه یک گره از آن را تشکیل می‌دهد و هر بال گراف یک تونل دوطرفه تک‌پخشی است. بعد از ساخته شدن هسته، هسته می‌تواند به صورت دینامیکی بر اساس عضویت گروه در شبکه حرکت کند. یکی از معایب استفاده از درخت چندپخشی در این الگوریتم کاهش بازدهی است زیرا مسیرهایابی که عضو این گروه نیستند نمی‌توانند در ارسال داده دخالت داشته باشند که در نتیجه آن تأخیر افزایش و پهنای باند شبکه کاهش می‌یابد.

هنگامی که مش در حال ارتباط، خود را برای تمام اعضاء پایدار می‌کند، یک درخت برای ارسال داده لازم است. بدست آوردن یک مش ساده‌تر از یک درخت است، زیرا به عنوان مثال هنگامی که مش‌های اضافی در کنار هم تشکیل شوند، پایداری ارتباط را زیاده‌تر می‌کنند، ولی چند درخت که هنوز با هم هماهنگ نشده‌اند، در کنار هم می‌توانند یک حلقه را ایجاد کنند که این سبب ناپایداری می‌شود. یک مکانیزم جستجو با استفاده از یک حلقهٔ گسترده، به اعضاء اجازه می‌دهد که یکدیگر را کشف کنند. همهٔ هسته‌ها به صورت متناوب یک پیام درخواست ارتباط **JOIN REQ** را همه‌پخشی می‌کنند. هر یک از اعضاء با دریافت این پیام خود را به هسته می‌شناساند، که در نتیجه آن یک تونل دوطرفه بین هسته و گره مربوطه برقرار می‌شود. هنگامی که یکی از اعضاء بخواهد گره را ترک کند یک پیام **JOIN-NAK** می‌فرستد. اگر این گره پیام دیگری از این گروه دریافت کرد پیام **JOIN-NAK** دیگری را می‌فرستد. هنگامی که تعداد ارتباط‌های یک گره از حد مشخص تجاوز کند، آن گره باید یکی از مسیرهای خود را قطع کند. به هر مسیر وزنی داده می‌شود که متناسب با فاصلهٔ بین آن گره تا همسایگانش است. برای دستیابی به مش‌های بهینه به هنگام حرکت اعضاء یک روند بازسازی مش متناوب نیاز است. برای ساختن درخت هسته یک پیام متناوب **TREE-CREATE** در تمام شاخه‌های مش می‌فرستد. پررود فرستادن این پیام به اندازهٔ مش و حرکت گره‌های مش بستگی دارد. هنگامی که یک گره پیام **TREE-CREATE** را به صورت غیر تکراری دریافت کند، آنرا به تمام گره‌های مش ارسال می‌کند و در آن تمام مسیرهای ورودی و خروجی خود را به عنوان شاخه‌های درخت قرار می‌دهد. در این الگوریتم پایداری و پهنای باند با هم مصالحه

دارند. هنگامی که یکی از گره‌ها داده‌ای را از یکی از شاخه‌های درخت دریافت می‌کند، آنرا به تمام شاخه‌های دیگر درخت ارسال می‌کند. اگر گره‌ای داده‌ای را از یک شاخه غیر درخت دریافت کند، یک پیام **TREE-CREATE-NAK** را در آن شاخه ارسال می‌کند که به معنی حذف آن شاخه است.

در این الگوریتم هر عضو گروه دو دسته جدول را در خود دارد. جدول اول شامل همسایگانی است که عضو مش هستند و دسته دوم همسایگانی که عضو درخت چندپخشی همچنین هر عضو تعداد پرش تا هر شاخه مش را در خود نگه می‌دارد. با فرض اینکه بین تمام گره‌ها یک مسیر توسط شاخه‌های مش وجود دارد درخت چندپخشی نیازی به تغییر در هنگام تغییرات شبکه ندارد. که این سبب پایداری و کم شدن سیگنالینگ می‌شود. که گره‌هایی که عضو گروه نیستند نیازی به بازپاسخ به سیگنالینگ‌های پروتوکل **IP** چندپخشی ندارند. در **AMRoute** از درختهای اشتراکی استفاده می‌شود بدین معنی که برای هر گروه یک درخت استفاده می‌شود. **AMRoute** از تونل‌های تک‌پخشی بین همسایگان در یک درخت استفاده می‌کند که این خود نیاز گره‌ها به پشتیبانی الگوریتم چندپخشی را مرتفع می‌سازد.

استفاده از درخت چندپخشی سبب کاهش بازدهی می‌شود، زیرا مسیرهای غیر عضو نمی‌توانند داده‌های گروه را ارسال کنند. همچنین در این الگوریتم تأخیر غالباً زیاد می‌شود، با بیشتر شدن حرکت شبکه پیدا کردن یک مسیر بهینه سخت‌تر می‌شود و نیاز به سیگنالینگ بیشتری دارد [22].

۳-۱۴-۲ پروتکل $MAODV^{145}$

در الگوریتم **MAODV** مسیرها به هنگام نیاز کشف می‌شوند و کشف مسیر نیز از طریق انتشاری (همه‌پخشی) صورت می‌گیرد. این مکانیزم منجر به تطبیق سریع با تغییر شرایط می‌شود و به پردازش و حافظه کمتری نیاز دارد. در نتیجه از شبکه استفاده کمتری می‌شود. این الگوریتم درختهای چندپخشی اشتراک دوطرفه‌ای را می‌سازد که به وسیله آن بین گیرنده‌ها و منابع می‌توان ارتباط چند پخشی برقرار نمود. در جداول مسیریابی اطلاعات زیر ذخیره می‌شود: آدرس **IP** گروه چندپخشی، آدرس **IP** سرگروه چندپخشی، شماره طول عمر و پرش‌های بعدی گروه چندپخشی، تعداد پرش نسبت به گروه چندپخشی دیگر و تعداد پرش تا سرگروه چندپخشی، کشف مسیر در این الگوریتم به این صورت است که گره‌ای که می‌خواهد با یک گره چندپخشی ارتباط برقرار کند یک پیام ارتباط **PREQ** را همه‌پخشی می‌کند که آدرس **IP** گروه مقصد را داراست و در این پیام نشانه "**J**"^{۱۴۶} فعال می‌شود که نشان‌دهنده درخواست ارتباط با گره است. اگر یک گره درخواست **PREQ** را دریافت کند و آن گره عضو چند پخشی مقصد نباشد یا اگر **PREQ** را دریافت کند ولی مسیری به گره چند پخشی نداشته باشد، آن درخواست را به همه اطرافیان همه‌پخشی می‌کند. اگر گره‌ای با گروهی ارتباط برقرار کند آن گره سرگروه می‌شود. همچنین اگر گره‌ای درخواست **PREQ** را برای چندپخشی به یک گره دریافت کند و اگر آن گره یک مسیرپاب باشد که در درخت آن گروه قرار داشته

¹⁴⁵. multicast adhoc on demand distance vector protocol

¹⁴⁶. join

باشد و یا شماره ذخیره مربوط به آن گروه از شماره موجود در PREQ بزرگتر باشد، آن گره به پیام بازپاسخ می‌دهد. در طول دوره کشف مسیر گره منبع مسیری را با گروه همه‌پخشی که دارای بیشترین شماره است و کمترین تعداد پرش در درخت چندپخشی را داراست، را پیدا می‌کند. در پایان دوره کشف مسیر، گره مبداء مسیر را به وسیله فرستادن یک پیام فعال‌سازی MACT و فعال کردن نشانه آن در جدول مسیریابی چندپخشی، فعال می‌کند.

اگر هدف غیرفعال کردن تنها یک گره در گروهی باشد، آن گره یک پیام MACT با نشانه غیر فعال‌سازی به گره بعدی می‌فرستد و اطلاعات گروه چندپخشی را از جدول مسیریابی خود حذف می‌کند. وقتی که گره بعدی پیام غیرفعال‌سازی را دریافت کند و با حذف آن تنها خود نیز یک گره تنها شود و در صورتی که عضو گروه چندپخشی نباشد، خود را از درخت مسیریابی گروه توسط ارسال یک پیام غیرفعال‌سازی به گره بعدی حذف می‌کند. نگهداری مسیر در این الگوریتم با این هدف انجام می‌شود که تمام اعضای درخت همه‌پخشی در دسترس باشند به همین دلیل نگهداری درخت به دو شکل صورت می‌گیرد، ترمیم درختی که یک شاخه آن شکسته شده و یا چسباندن دوباره درخت بعد از تغییرات شبکه.

در این الگوریتم قطع شدن یکبار ارتباط به این صورت شناخته می‌شود که هیچ داده‌ای از آن دریافت نشود. هر گره باید یک پیام HELLO را در یک دوره مشخص همه‌پخشی کند و وقتی تعداد گرفتن‌های پیام HELLO از تعداد مشخصی تجاوز کند قطع یک ارتباط تشخیص داده می‌شود. هنگامی که یک ارتباط شکسته می‌شود گره پائینی آن مسئول ترمیم آن می‌شود گره پائینی عملیات ترمیم را به وسیله فرستادن یک پیام ارتباط RRFQ به اعضای گره چندپخشی آغاز می‌کند. این پیام شامل فیلدی است که فاصله گره‌ها تا سرگروه را مشخص می‌کند. تنها گره‌هایی که دقیقاً به سرگروه نزدیک‌اند اجازه بازپاسخ به این پیام را دارند.

اگر در زمان مشخص بازپاسخی دریافت نشود، یک پیام RREQ در تمام شبکه همه‌پخشی می‌شود. هر گره‌ای که عضو درخت چندپخشی باشد و دارای شماره چندپخشی کافی باشد و تعداد پرش آن تا سرگروه بعدی از حد معینی کمتر باشد می‌تواند به این پیام بازپاسخ دهد.

هنگامی که بر اثر تغییراتی در شبکه، دو تکه از شبکه می‌توانند به هم رابطه برقرار کنند. گره دارای درخت همه پخشی یک پیام GRPH^{۱۴۷} از سرگروه^{۱۴۷} گروه دیگر دریافت می‌کند. سرگروهی که دارای آدرس IP کوچکتری است یک پیام PREQ به سرگروه دیگر می‌فرستد. این پیام RREQ دارای یک نشانه ترمیم^{۱۴۸} است. هر گره‌ای در درخت گروه دوم که این پیام را دریافت می‌کنند، این پیام را به تمام اعضای گروه، چندپخشی می‌کند. هنگامی که سرگروه دوم، این پیام را دریافت می‌کند. شماره گروه چندپخشی خود را یک واحد افزایش می‌دهد و یک پیام RREP به سرگروه اول می‌فرستد و این RREP دارای نشانه تعمیر است.

هر گره‌ای از گروه چندپخشی اول که این پیام را دریافت کند. آنرا به سرگروه، گروه خود می‌فرستد. هنگامی که سرگروه اول این پیام را دریافت کند، درخت آن‌دو به هم متصل می‌شوند.

¹⁴⁷. Group hello

¹⁴⁸. repair

در این مسیریابی همه مسیرها به وسیله استفاده از شماره مقصد از ایجاد حلقه مصون می‌مانند و به علت برحسب نیاز بودن مسیریابی در آن پیام‌های کنترلی و پهنای باند مورد استفاده در هنگام تغییرات کم شبکه، کمینه می‌شود. شبیه‌سازی‌های مختلف نشان می‌دهد، هنگامی که تحرک یک گره افزایش یابد، نیاز به ترمیم‌های بیشتر و بیشتری می‌باشد. شبکه‌های بزرگ به ترمیم‌های بیشتری نسبت به شبکه‌های کوچک نیاز دارند. افزایش تعداد ترمیم‌ها منجر به کاهش نسبت رسید داده و افزایش سرباره می‌شود. [22]

۳-۱۴-۳ پروتکل CAMP^{۱۴۹}

پروتکل CAMP یک پروتکل مسیریابی چند پخشی اشتراکی بر اساس مش است. هدف اصلی استفاده از چنین مش‌هایی حفظ ارتباط گروه‌های چند پخشی، هنگامی که مسیرهای پیاپی در حال حرکت هستند، می‌باشد. از خصوصیات CAMP حفظ شبکه چند پخشی و بدون حلقه، بدون مسیریابی بر روی این مش‌ها می‌باشد. در این الگوریتم هر گروه دارای یک مش است. برای هر دو گره روی این مش یک مسیر وجود دارد. در الگوریتم CAMP تضمین می‌شود که در یک زمان محدود، هر گیرنده عضو یک گروه چندپخشی کوتاه‌ترین مسیر را با هر منبع آن گروه داشته باشد.

این الگوریتم از هسته‌هایی برای محدود کردن ترافیک مسیریاب یک گروه چند پخشی استفاده می‌کند. در صورت رخ دادن خطا در هسته‌ها در ارسال داده‌ها و نگهداری مش‌ها مشکلی پیش نمی‌آید. هسته‌ها برای ایجاد محدودیت بر روی ترافیک، بسته‌های درخواست ارتباط (GREG PACKET) به کار می‌روند.

الگوریتم CAMP برای شبکه‌های ADHOC که تحرک فراوان در آنها وجود دارد طراحی شده است. در این الگوریتم ثابت می‌شود کوتاه‌ترین مسیر بین مبدا و مقصد روی عضو یک گروه، روی مش گروه قرار دارد. به خاطر اینکه هر مسیریاب عضو یک مش دارای چندین مسیر تا مسیریاب مش دیگر است، قطع شدن یک مسیر بین آنها باعث قطع ارتباط نمی‌شود و نیاز به بازسازی ساختار مسیریابی نیز ندارد.

همه گره‌ها در الگوریتم CAMP دارای یک سری اطلاعات شامل جداول مسیریابی و عضویت هستند. بعلاوه همه گره‌ها یک سری اطلاعات پنهان شامل اطلاعات بسته‌های قبلی ارسال شده و درخواست‌های عضویت رد شده می‌باشند. الگوریتم CAMP گره‌ها را به دو دسته، دوطرفه و تک طرفه تقسیم‌بندی می‌کند.

این الگوریتم شامل ساختن مش‌ها و نگهداری آنهاست. هر گره‌ای که می‌خواهد با یک مش چندپخشی ارتباط برقرار کند، ابتدا در جدول خود نگاه می‌کند تا ببیند که آیا این مش در همسایگی او قرار دارد یا خیر. اگر چنین بود آن گره عضویت خود را با یک پیام به‌روزکننده اعلام می‌کند. در غیر اینصورت گره یک پیام درخواست ارتباط را به یکی از هسته‌های گروه‌های چندپخشی می‌فرستد.

جداول مسیریابی بر اساس الگوریتم‌های مسیریابی تک‌پخشی تک‌پخشی ساخته می‌شود، هنگامی که نیاز باشد که گروه‌های چند پخشی حذف و یا اضافه شوند، الگوریتم **CAMP** این تغییرات را در جداول ایجاد می‌کند. در الگوریتم **CAMP** برای یک مسیر یاب دو روش مختلف برای برقراری ارتباط را به آن می‌فرستد. در عوض اگر آن مسیر یاب به هسته مش وصل باشد یک پیام درخواست ارتباط ساده یک‌جهته^{۱۵۰} به آن هسته می‌فرستد. بعلاوه بعد از ارسال یا گرفتن داده، هر عضو مش ممکن است تقاضایی مبنی بر پیدا کردن مسیر یاب‌هایی در کوتاه‌ترین فاصله از آن، توسط پیام درخواست ارتباط **PUSH**، ارسال کند.

در الگوریتم **CAMP** قطع شدن ارتباط یک مشکل اساسی و بحرانی نیست. با قطع شدن یک ارتباط، کوتاه‌ترین مسیر به منبع قطع می‌شود ولی مسیریابی ممکن است اصلاً تأثیری نپذیرد، زیرا مسیر دیگری روی مش وجود دارد. به خاطر سادگی ایجاد ارتباط یک مسیر یاب با یک گروه قطع ارتباط کامل نیز اثر کمی روی الگوریتم دارد. در الگوریتم **CAMP** سعی می‌شود برای ارتباط با یک مش تکیه بر روی یک هسته نباشد. همچنین اگر دو مش از نظر فیزیکی به هم نزدیک باشند می‌توانند درهم ادغام شوند.

الگوریتم **CAMP** از قابلیت ارتباط قوی‌تری نسبت به درختها برخوردار است. همچنین این الگوریتم مشکل ایجاد حلقه در ارسال داده را ندارد. الگوریتم **CAMP** تضمین می‌کند که در یک زمان محدود هر گیرنده و هر فرستنده دارای یک مسیر کوتاه‌ترین، روی مش هستند. همچنین این الگوریتم از ایجاد سیل داده و حجم زیاد داده کنترلی جلوگیری می‌کند. یک مسیر یاب می‌تواند از تمام همسایگان خود در مش داده دریافت کند و به خاطر وجود چند مسیر اضافه برای هر مسیر یاب نسبت به یک مش تغییرات توپولوژیکی در شبکه، نیازی به بازسازی مسیریابی ایجاد نمی‌کند. **CAMP** فرستنده‌های داده را دنبال نمی‌کند به همین خاطر چک کردن عضویت فرستنده‌ها ممکن نیست. الگوریتم‌های مسیریابی مبتنی بر الگوریتم بلمان - فورد نمی‌توانند با **CAMP** به کار روند. همچنین برای کار با الگوریتم‌های مسیریابی بر اساس نیاز، الگوریتم **CAMP** باید گسترش یابد. شبیه‌سازی‌ها نشان می‌دهد گم شدن بسته در این الگوریتم به مقدار کمی از الگوریتم‌های مشابه مانند **ODMRP** بیشتر است [24].

۳-۱۴-۲-۴ پروتکل **ODMRP**^{۱۵۱}

ODMRP یک پروتکل براساس مش است. این الگوریتم راهکاری را براساس نیاز برای ساختن مسیرهای دینامیکی و دستیابی به عضویت گروه چندپخشی به کار می‌برد. **ODMRP** برای شبکه‌های بیسیم **ADHOC**

¹⁵⁰. **SIMPLEX**

¹⁵¹. on-demand Multicast Routing Protocol

که دارای گره‌های متحرک و پهنای باند کم و تغییرات سریع در توپولوژی شبکه نیاز به مصرف توان کم هستند، کاربرد دارد [۱].

با استفاده از مفهوم مشابه به جای درخت مشکلاتی که در درختهای چندپخشی ایجاد می‌شود در این الگوریتم وجود ندارد از خصوصیات ODMRP کمتر بودن سرباره و راحت‌تر بودن ارتباط در شبکه‌های بی‌سیم است. مشابه پرتکل‌های مسیریابی براساس نیاز، در این الگوریتم یک فاز جستجو و یک فاز باز پاسخ این الگوریتم را تشکیل می‌دهند. هنگامی که یک منبع چندپخشی داده‌ای برای ارسال داشته باشد ولی مسیر مورد نیاز آن را نداشته باشد، سیلی از بسته‌های عضویت را در شبکه ایجاد می‌کند و می‌فرستد.

این بسته Join-Query نامیده می‌شود و به طور متناوب از شبکه همه‌پخشی می‌شود و اطلاعات عضویت و مسیرها را به روز می‌کند. هنگامی که یک گره پیام Join-Query غیر تکراری دریافت کند ID گره بالایی خود را در جدول مسیریابی ذخیره می‌کند و بسته را دوباره همه‌پخشی می‌کند. هنگامی که بسته Join-Query به گیرنده چندپخشی می‌رسد، این گیرنده یک پیام Replay-Join تولید می‌کند و آنرا به همسایگان خود همه-پخشی می‌کند. هنگامی که هر گره این بسته را دریافت می‌کند، ID آنرا با داده‌های ذخیره شده قبلی مقایسه می‌کند و در صورت یکی بودن داده‌ها، متوجه می‌شود که به سر راه منبع واقع است، بنابراین یک پیام Join Replay می‌سازد و FG- Flag^{۱۵۲} را فعال می‌کند و آنرا همه‌پخش می‌کند.

این کار تا زمانی ادامه پیدا می‌کند که این پیام به گره منبع در کوتاه‌ترین مسیر برسد. در این عمل حاشیه اطمینان ارسال پیام Join-Replay نقش مهمی بازی می‌کند. بنابراین در بعضی از استانداردها در صورت نرسیدن پیام Join-Replay به منبع، منبع دوباره درخواست خود را ارسال می‌کند یا در بعضی از الگوریتم‌های دیگر پیام Join-Replay علاوه بر گره بالایی به گره‌های دیگر نیز فرستاده می‌شود. بعد از ساخت و برقراری یک ارتباط منبع داده‌های خود را توسط مسیرهای انتخاب شده به گیرنده‌ها می‌فرستد.

هنگامی که یک گره بسته داده را دریافت می‌کند در صورتی که نشانه FG- Flag آن فعال شده باشد که نشانه این است که گره بر روی مسیر واقع است، آنرا به گره‌های بعدی می‌فرستد.

در ODMRP هیچگونه پیام خاصی برای ترک یک گره نیاز نیست. هنگامی که یک منبع داده بخواهد یک گروه را ترک کند، کافی است ارسال پیام Join-Query را متوقف کند. و اگر یک گیرنده بخواهد گروه را ترک کند کافی است از باز پاسخ پیام Join – Reply صرفنظر کند. گره‌هایی که با فعال شدن G-Flag عضویت یک مسیر هستند بعد از دوره زمانی خاصی کمه داده‌ای توسط آنها دریافت نشد، Flag خود را در غیر فعال می‌کنند.

¹⁵². Forwarding Grope Flag

این مقادیر دوره زمانی، تأثیر زیادی در کارایی ODMRP دارند، انتخاب این مقادیر به محیط شبکه که شامل نوع ترافیک، بارترافیک، شیوه حرکت، سرعت حرکت، ظرفیت کانال و غیره می‌باشد. بستگی دارد [25].

گره‌هایی که ODMRP در آنها اجرا می‌شود دارای جداول زیر هستند:

۱- جدول مسیریابی، که این جدول براساس نیاز مسیریابی ساخته می‌شود و براساس پیام‌های Join Query و Join Replay به روز می‌شود.

۲- جدول اعضاء گروه ارسال^{۱۵۳}، که این جدول مسیریابی که گره عضو آن است را مشخص می‌کند و ID گروه چندبخشی متناظر با آن و زمان دیدن آخرین پیام مربوط به آن گروه را در خود ذخیره می‌کند.

۳- حافظ پنهان پیام، که این حافظ برای تشخیص دریافت تکراری پیام‌ها به کار می‌رود.

در این الگوریتم چون پهنای باند یک پارامتر مهم است و ایجاد سیل بسته‌های جستجو برای بروزرسانی مسیریابی، معمولاً مشکلاتی را پدید می‌آورد، بنابراین دوره تناوب بروز رسانی یک پارامتر مهم است که براساس حرکت و محل گره‌ها در شبکه تعیین می‌شود.

شایان ذکر است که انتخاب مسیر در ODMRP بدین نحو است که یک گیرنده یک مسیر را به گونه‌ای انتخاب می‌کند که دارای کمترین مقدار تأخیر باشد (اولین Join Query که دریافت می‌کند).

[25]

جدول مربوط به نام‌های مخفف و کامل پروتکل‌ها

نام پروتکل / الگوریتم	مخفف پروتکل	# /
AMRIS Ad-hoc Multicast Routing protocol utilizing Increasing id numbers protocol	AMRIS	۱
Multicast Ad-hoc On-Demand Vector protocol	MAODV	۲
Ad-hoc Multicast Routing protocol	AMRoute	۳
Lightweight Adaptive Multicast algorithm	LAM	۴
On-Demand Multicast Routing Protocol	ODMRP	۵

¹⁵³. Forwarding Grope Flag

٦	CAMP	Core Assisted Mesh Protocol
٧	LGT	Location Guided Tree algorithm
٨	DDM	Differential Destination Multicast algorithm
٩	HQMRP	Hierarchical QoS Multicast Routing Protocol
١٠	SOM	Self-Organizing Map algorithm
١١	LGF	Location-based Geocasting and Forwarding algorithm
١٢	SPBM	Scalable Position-Based Multicast algorithm
١٣	STAMP	Shared Tree Ad-hoc Multicast Protocol
١٤	ACMP	Adaptive Core-based Multicast Routing Protocol
١٥	CQMP	Mesh-based Multicast Routing Protocol with Consolidated Query Packets
١٦	E-ODMRP	Enhanced On-Demand Multicast Routing Protocol
١٧	BODS	Bandwidth Optimized and Delay Sensitive algorithm
١٨	EHMRP	Efficient Hybrid Multicast Routing Protocol
١٩	MWIA	Minimum Weight Incremental Arborescence algorithm
٢٠	PCHMR	Power-Controlled Hybrid Multicast Routing protocol
٢١	RAMP	Reliable Adaptive Multicast Protocol
٢٢	RORP	Reliable On-demand Routing Protocol
٢٣	ROMANT	Robust Multicasting in Ad-hoc Network using Tree
٢٤	OPHMR	Optimized Polymorphic Hybrid Multicast Routing protocol
٢٥	MAMR	Mobile Agents aided Multicast Routing protocol
٢٦	MPGC	Multicast Power Greedy Clustering protocol
٢٧	P=REMiT	Probability for Refining Energy-efficiency of Multicast Tree
٢٨	PMRP	Distributed Minimum Energy Multicast
٢٩	CCMRP	CodeCast Multicast Routing Protocol
٣٠	ONCRM	On Network Coding and Routing in dynamic wireless Multicast network
٣١	DQMRP	Distributed QoS Multicast Routing Protocol
٣٢	HVDB	logical Hypercube-based Virtual Dynamic Backbone

جدول مربوط به مقایسه پروتکل ها

#	نام/مخفف پروتکل ها	E *	F	H	H Y	L	M	P	Q	R	S	T	های معیار مسیریابی چندپخشی	دوطبقه کلی پروتکل ها
														AI AD
1	AMRIS	N	Y	N	N	N	N	Y	N	N	Y	Y	حداقل پرش	N N
2	MAODV	N	Y	N	N	N	N	N	N	Y	Y	Y	حداقل پرش	N N
3	AMRoute	N	Y	N	N	N	N	Y	N	N	Y	Y	حداقل پرش	N N
4	LAM	N	Y	N	N	N	N	N	N	Y	N	Y	وابسته به اتصال	N N
5	ODMRP	N	Y	N	N	N	Y	Y	N	N	N	N	حداقل پرش	N N
6	CAMP	N	Y	N	N	N	Y	Y	N	N	N	N	وابسته به اتصال	N N
7	LGT	N	Y	N	N	N	N	N	N	Y	N	Y	بیشترین ارسال به جلو	N N
8	DDM	N	Y	N	Y	N	N	Y	N	N	N	N	حداقل پرش	N N
9	HQMRP	N	N	Y	N	N	N	Y	N	N	Y	N	تمرکز بر تغییر توپولوژی و واکنش های مربوط به آن	N Y
10	SOM	N	N	Y	N	N	N	N	N	Y	Y	N	وابسته به اتصال	N Y
11	LGF	N	Y	N	N	Y	Y	N	N	Y	N	N	بیشترین ارسال به جلو	N N
12	SPBM	N	Y	N	N	Y	N	N	N	Y	N	Y	بیشترین ارسال به جلو	N Y
13	STAMP	N	Y	N	N	N	N	N	N	Y	Y	T	بیشترین ارسال به جلو/کمترین پرش	N Y
14	ACMP	N	Y	N	N	N	N	N	N	Y	Y	Y	بیشترین ارسال به جلو/کمترین پرش	N N
15	CQMP	N	Y	N	N	N	Y	N	N	Y	Y	Y	حداقل پرش	N N

16	E-ODMRP	N	Y	N	N	N	Y	N	N	Y	N	Y	حداقل پرش	N	N
17	BODS	N	Y	N	N	N	Y	N	N	Y	Y	Y	حداقل پرش	N	N
18	EHRP	N	Y	N	Y	N	N	Y	N	N	Y	Y	بیشترین ارسال به جلو/کمترین پرش	N	N
19	MWIA	Y	Y	N	N	N	N	Y	N	Y	Y	N	کمترین قدرت انتقال کلی	N	Y
20	PCHMR	Y	Y	N	N	Y	N	Y	N	N	N	Y	کمترین قدرت انتقال	N	Y
21	RAMP	N	Y	N	N	N	Y	Y	N	N	N	N	کمترین پرش	N	Y
22	RORP	N	Y	N	N	N	N	Y	N	N	Y	Y	بیشترین ارسال به جلو/کمترین پرش	N	Y
23	ROMANT	N	Y	N	N	N	N	N	N	Y	Y	Y	بیشترین ارسال به جلو/کمترین پرش	N	Y
24	OPHMR	N	Y	N	Y	N	N	N	N	Y	N	N	بیشترین گام	N	Y
25	MAMR	N	Y	N	N	N	N	N	N	Y	N	Y	کمترین پرش بر کانال های قویتر	N	Y
26	MPGC	Y	Y	N	N	N	N	Y	N	N	Y	Y	کمترین تاخیر در انتقال	N	Y
27	P-REMiT	Y	Y	N	N	N	Y	Y	N	N	Y	Y	کمترین قدرت انتقال	N	Y
28	PMRP	Y	Y	N	N	N	Y	Y	N	N	Y	N	کمترین قدرت انتقال کلی	N	Y
29	CCMRP	N	Y	N	N	N	N	Y	N	N	Y	N	کمترین پرش بر کانال های قویتر	N	Y
30	ONCRM	N	Y	N	N	N	N	Y	N	N	Y	N	کمترین پرش بر کانال های قویتر	N	Y
31	DQMRP	N	Y	N	N	N	N	Y	Y	N	Y	Y	بیشترین ارسال رو به جلو	N	Y

32	HVDB	N	N	Y	Y	N	N	Y	Y	N	Y	Y	بیشترین ارسال رو	N	Y
													به جلو		

*در جدول زیر خصوصیات پروتکل ها به همراه حرف اول آنها آمده است.

	charectristic	مشخصه
E	Energy aware	وابسته به انرژی
F	Flat	مسطح
H	Hierarchical	سلسله ای
HY	Hybrid	ترکیبی
L	Location aware	وابسته به موقعیت
M	Mesh	مش
P	Proactive	پیش فعال
Q	QoS aware	وابسته به کیفیت
R	Reactive	فعال
S	Stabilit	پایداری
T	Tree	درختی
AI	Application Independence	مستقل از برنامه
AD	Application Dependence	وابسته به برنامه

جدول مربوط به مزایا و معایب پروتکل ها

مزایا	معایب	نام پروتکل ها
بدون حلقه بودن در تمام لحظات، در دسترس بودن مسیرها برای کلیه گره های مقصد، حداقل تعداد مسیرهای زائد	پشتیبانی نکردن از از مسیریابی چند مسیری و ارسال چند بخشی، به هدر رفتن پهنای باند	1-DSDV
نداشتن حلقه مسیریابی	ایجاد مقدار زیادی سرباره حافظه در هر گره	2-WRP
بهبود انتشار اطلاعات در الگوریتم حالت لینک	استفاده زیاد از پهنای باند	3-GSR
باعث کاهش سرباره پیغام شده، مقیاس خوبی برای شبکه های بزرگ	-	4-FSR
جدایی مدیریت تحرک از سطح فیزیکی	دارای سرباره اضافه ای برای آرایش و نگهداری خوشه ها	5- HSR
به حداقل رساندن حجم ترافیک کنترل		6-OLSR
سرباره کمتر	هزینه ی بالای ناشی از نگهداری خوشه ها	7-CGSR
- کاهش قابل توجهی در پهنای باند مصرفی، - جلوگیری از به هنگام رسانی متناوب در الگوریتم حالت لینک		8-STAR
کاهش سرباره، ترافیک کنترلی و بروز رسانی کمتر	استفاده کردن از ERPF	9-TBRPF
مصرف پهنای باند کمتری کاهش سرباره مسیریابی		10-DREAM
کاهش سرباره پهنای باند شبکه، حفظ انرژی باتری، کاهش بروزکننده های مسیریابی در شبکه	برای شبکه های بزرگ مناسب نیست.	11-DSR

12-AODV	تاخیر در برقراری مسیرها- عدم پشتیبانی از کشف و نگهداری چند مسیره بین هر جفت مبدا و مقصد.	اطمینان از عدم ایجاد مسیرهای حلقوی، قابل استفاده در هر سه نوع ارتباطات (تک پخشی، چند پخشی و انتشاری) بهره وری در استفاده از پهنای باند.
13-TORA	ایجاد مسیرهای ناصحیح موقتی شبه LMR	- سازگار - بدون حلقه - قابلیت چند مسیریابی - بهینه نمودن مسیر - کاهش پیام های کنترلی به مجموعه همسایه ها
14-LAR	گره باید مجهز به GPS باش	کوتاهترین مسیر به مقصد
15-ABR	ممکن به ایجاد کوتاهترین مسیر نشود	نداشتن حلقه عدم وجود قفل شدن بی نهایت و بسته های اضافی
16-SSA	- ممکن است کوتاهترین مسیر به مقصد انتخابی نشود - عدم تعمیر لینک های تخریب شده - تاخیر زیاد	انتخاب مسیر بر اساس قدرت سیگنال
17-LMR	ممکن است مسیرهای ناصحیح موقتی ایجاد کند	جلوگیری از تاخیر زیاد و سرباره
18-CBRP	وجود حلقه های مسیر موقتی	سرباره کنترلی ارسالی بسیار کم
19-RDMAR	نداشتن قابلیت مقیاس پذیری در شبکه های بزرگ	عدم نیاز به تکنولوژی موقعیت مانند GPS

20-ARA	نداشتن قابلیت مقیاس پذیری در صورت زیاد شدن تعداد گره ها و جریان ها	میزان سرباره به ازای بسته کنترلی کمینه است
21-FORP	مشکلات مقیاس پذیری در شبکه های بزرگ	تخریب مسیرها را کمینه میکند
22-ROAM	اجازه نداشتن گره به حالت استراحت برای کنترل توان در شبکه هایی که تغییرات زیاد	بهبود پهنای باند و جدول مسیریابی
23-ZRP	خالص عمل کردن در منطقه مسیریابی کنترل	کاهش هزینه مسیریابی BORDER CAST
24-ZHLS	برای محدوده جغرافیایی شبکه دینامیکی کاربرد ندارد	دارای سرباره کمترین در مقایسه با پروتکل های واکنش دار، وفق پذیری در برابر تغییرات پیکربندی
25-DDR	در شبکه های با ترافیک بالا به کاهش توان عملیاتی می انجامد	در محدوده جغرافیایی شبکه دینامیکی کاربرد دارد.
26-AMROUTE	کاهش بازدهی، تاخیر زیاد	پایداری و کم شدن سیگنالینگ، نیاز نبودن باز پاسخ به سیگنالینگ های پروتکل. IP چند پخشی ندارد.
27-MAODV	افزایش سرباری، کاهش نسبت رسیده داده	نبود حلقه، مصرف بهینه از پهنای باند
28-CAMP	ممکن نبودن چک کردن عضویت فرستنده ها، گم شدن بسته در این الگوریتم بیشتر است.	نداشتن حلقه، جلوگیری از ایجاد سیل داده و حجم زیادی کنترلی
29-ODMRP	هنگامی که تعداد زیادی فرستنده چند پخشی بسته های جستجوی خود را در شبکه به صورت سیل رها کنند شبکه دارای مشکل می شود	سادگی-کم بودن سرباره کانال و حافظه، استفاده از کوتاهترین مسیرهای بروز شده، ساخت مسیرهای مطمئن بر اساس گره های ارسال