

بهینه سازی ارسال فراگیر در شبکه های Ad Hoc و حسگر بی سیم به کمک داده کاوی مکانی

مهرداد الماسی^۱؛ محمد داوورپناه جزی^۲؛ محمد علی منتظری^۳

چکیده

در این مقاله، به کمک الگوریتم (DBSCAN (Density Based Spatial Clustering of Applications with Noise) راهی برای یافتن کوتاه ترین مسیر مشترک، در میان اعضای شبکه ای با عضوهای ناآگاه از موقعیت مکانی، ارائه شده است. یکی از دشواری های اصلی در شبکه های حسگر و Ad Hoc بی سیم، رساندن پیامی به تمام اعضا می باشد. با کاهش تعداد اعضای شرکت کننده در فرآیند ارسال، کارایی ارسال یک پیام به تمام اعضا افزایش خواهد یافت. در صورتی که اعضای شبکه آگاه از موقعیت مکانی باشند، تعیین مجموعه ارسال کننده چندان دشوار نیست، ولی آگاهی از موقعیت مکانی همیشه میسر نمی باشد. برای وضعیت عدم آگاهی از موقعیت مکانی، از الگوریتم پیشنهادی در این مقاله جهت بدست آوردن مجموعه بهینه استفاده کردیم. سپس نتایج حاصل از به کار گیری این الگوریتم را با دیگر الگوریتم ها، از طریق مقیاسی به نام بازده [۸] مقایسه نمودیم. برای مقایسه ابتدا بازده الگوریتم پیشنهادی از لحاظ تئوری تحلیل شد، سپس نتیجه با دیگر الگوریتم ها مقایسه گردید. الگوریتم را به کمک (JSIM (Java Simulator) شبیه سازی نمودیم. نتایج حاصل از بخش شبیه سازی، با نتایج حاصل از بخش تئوری تطبیق داشت.

کلمات کلیدی

DBSCAN، کوتاه ترین مسیر مشترک، شبکه Ad Hoc بی سیم، شبکه حسگر بی سیم، ارسال فراگیر.

Optimal broadcast in Wireless Ad Hoc and sensor networks with help of spatial data mining

Mehrdad Almasi; Mohammad Davarpanah Jazi; Mohammad Ali Montazeri

ABSTRACT

In this article, a way to find the shortest common path among the members of a network has been offered with help of DBSCAN (Density Based Spatial Clustering of Applications with Noise) algorithm. Notice, the members are unaware from their location.

Convey a message to all members is one of the main difficulties in the wireless Ad Hoc and sensor networks. By reducing number of participant members in sending operation the efficiency of sending a message to all members will be increased. In the case that the members of the network aware of their location creation sender collection isn't so difficult, but awareness of the location isn't always possible. We used our algorithm for covering lack of knowledge about the location and obtain an optimal sender collection. Then we compared our results with the other algorithms with help of efficiency metric [8]. For comparison, first the efficiency of our algorithm was analyzed theoretically then result was compared with other algorithms. We simulated our algorithm with JSIM (Java Simulator). The results of simulation were admitted with the results of analysis part.

KEYWORDS

DBSCAN, Shortest common path, Wireless Ad Hoc network, Wireless sensor network, Broadcast.

۱. دانشجوی کارشناسی ارشد هوش مصنوعی - دانشگاه صنعتی اصفهان - دانشکده مهندسی کامپیوتر - m.almasi@ec.iut.ac.ir

۲. استادیار کامپیوتر - دانشگاه صنعتی اصفهان - دانشکده مهندسی کامپیوتر - mdjazi@cc.iut.ac.ir

۳. استادیار کامپیوتر - دانشگاه صنعتی اصفهان - دانشکده مهندسی کامپیوتر - montazeri@cc.iut.ac.ir

۱. مقدمه

شبکه ی Ad Hoc بی سیم شبکه ای غیر متمرکز می باشد که دارای شالوده ای از پیش تعیین شده نیست و همه ی نودهای موجود در آن باید در مسیر یابی ارسال پیام به دیگر نودها شرکت کنند. شبکه های حسگر بی سیم حالت خاصی از شبکه های Ad Hoc می باشد. این شبکه ها مجموعه ای از ابزارهای کوچکی به نام نود هستند و امکان حس کردن کامل یک محیط را دارا می باشند. تا کنون کاربردهای متنوعی از شبکه های حسگر بی سیم ارائه و پیاده سازی شده است. این کاربردها از کنترل خانه [۱] تا رهگیری حیوانات و حتی کاربردهای نظامی [۲] را شامل می شود. ارسال فراگیر (broadcast) فرآیندی است که در آن پیامی که توسط یک نود ارسال می شود باید به دست همه ی اعضای شبکه برسد. یکی از چالش ها در شبکه های Ad Hoc و حسگر بی سیم، رساندن سریع پیام به همه ی اعضا در شبکه است. بعلاوه انتقال تنها یک بایت توسط یک نود، هزینه ای بیش از اجرای عملیات محاسباتی بر روی همان بایت و توسط CPU ی همان نود، خواهد داشت. پس باید تا حد امکان ارسال بیهوده انجام نداد. با توجه به آنچه گفته شد، کاهش مکالمات در شبکه های Ad Hoc و حسگرهای بی سیم مطلوب می باشد. به طور کلی الگوریتم های ارسال فراگیر به چهار دسته کلی [۴] زیر تقسیم می شوند:

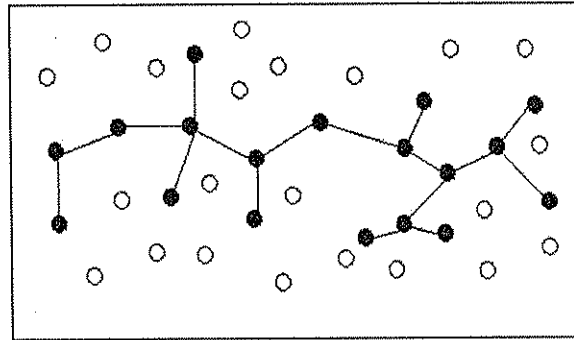
- Simple flooding [۵]: هر نود موجود در شبکه، هر پیام دریافتی را دقیقا یک بار ارسال مجدد (rebroadcast) می کند.
- Probability based [۶]: در روش های این دسته، به هر نود برای ارسال مجدد، احتمالی تخصیص داده می شود. نودها از این احتمال ها برای تعیین نحوه ی انجام عمل ارسال مجدد، استفاده می کنند.
- Area based [۶]: در روش های این دسته، فاصله ی مشترک ارسال، در میان نودها در نظر گرفته می شود و نودی عمل ارسال مجدد را انجام می دهد که ناحیه پوشش مناسبی داشته باشد.
- Neighborhood based: وضعیت همسایه ها توسط روش هایی که در این دسته قرار می گیرند، نگهداری می شود. نودها از این اطلاعات برای تعیین نحوه ارسال مجدد استفاده خواهند کرد.

در الگوریتم های دسته ی Simple flooding، هر گاه پیامی توسط نودی دریافت شود، اگر نود قبلا آن را دریافت نکرده باشد، عمل ارسال مجدد انجام خواهد داد. در غیر این صورت پیام را دور خواهد انداخت. الگوریتم های دسته ی Probability based برای کاهش ارسال ها در الگوریتم های در دسته ی Simple flooding مطرح شدند. در روش های این دسته به هر نود، احتمالی نسبت داده می شود. این احتمال معمولا بر اساس تراکم نودها در توپولوژی است. اگر میزان این احتمال بیش از مقداری معین باشد عمل ارسال مجدد توسط نود صورت می گیرد و در غیر این صورت عملی صورت نخواهد گرفت. مشکلی که در این روش وجود دارد، احتمال عدم رسیدن پیامی به همه ی نودهای شبکه می باشد. در الگوریتم های Area based هر نود نیارمند آگاهی از موقعیت مکانی است تا بتواند به نحو تقریبا دقیقی نودهایی که در برد رادپویی اش قرار می گیرند را محاسبه کند. برای این امر نیاز به پشتیبانی الگوریتم توسط سیستم موقعیت یاب جهانی (GPS) است. بعلاوه در این نوع الگوریتم ها پیام ها دارای (random assessment delay) RAD می باشند. نودی که پیامی را دریافت می کند، در صورت منقضی بودن RAD پیام را دور خواهد انداخت. در الگوریتم های Neighborhood based مجموعه ای به نام Gate way تعریف می شود. نودهای این مجموعه وظیفه انتقال پیام به سراسر شبکه را بر عهده خواهند داشت. هر چه مجموعه Gate way بهتر برگزیده شود، کارایی بهتری برای شبکه حاصل خواهد شد. ویژگی نودهای مجموعه ی Gate way داشتن همه ی نودهای شبکه در خود مجموعه یا در یک گام فاصله از مجموعه می باشد (تولید مجموعه ی Gate way بر همین اساس است). روش های ارسال فراگیر موجود، دارای معایب [۷] زیر می باشند:

- همه ی روش ها به جز روش هایی که در دسته Neighborhood based قرار می گیرند، ارسال های مجدد زیادی انجام می دهند که بخش زیادی از آنها زاید می باشد.
 - روش های که در دسته ی Area based قرار می گیرند از RAD استفاده می کنند در صورتی که تراکم نودها در توپولوژی بالا باشد. این روش زیان بار خواهد بود زیرا پیام های بیهوده زیادی در شبکه تولید و ارسال می شود که هیچ سودی ندارند (RAD پیام منقضی شده است).
 - روش هایی که در دسته ی Neighborhood based قرار می گیرند، اگر امکان آگاهی از موقعیت مکانی را نداشته باشند، در تصمیم گیری برای ارسال یا عدم ارسال یک پیام دچار دشواری هستند. این دشواری با افزایش تراکم نودها بیشتر خواهد شد.
- با توجه به آن چه گفته شد، در این مقاله بدنبال ارائه روشی هستیم که بتواند مشکلاتی که در بالا ذکر شد را حداقل کند. به این منظور به دنبال یافتن راهی برای ارسال سریع پیام به تمامی اعضا، بدون داشتن آگاهی از موقعیت مکانی و به کار گیری RAD هستیم. روشی که در این مقاله جهت ارسال فراگیر ارائه شده، در دسته ی الگوریتم های Neighborhood based و مبتنی بر الگوریتم DBSCAN می باشد. در روش پیشنهادی برای تامین داده های مورد نیاز الگوریتم DBSCAN از یک پروتکل تبادل پیام که در بخش معرفی روش به آن پرداخته شده، استفاده می شود. پیش از معرفی روش نیاز است تا منظورمان از کوتاهترین مسیر مشترک را بیان کنیم (شکل (۱)). مسیری همبند در میان اعضای شبکه که تمام اعضای شبکه یا در آن عضو باشند یا حداقل در برد یکی از اعضای این مجموعه قرار داشته باشند، کوتاهترین مسیر مشترک می باشد (تعریفی مشابه با تعریف مجموعه Gate way در الگوریتم های Neighborhood based).
- برای مقایسه الگوریتم پیشنهادی این مقاله با دیگر الگوریتم ها، از مقیاس بازده [۸] استفاده نمودیم و با کمک این مقیاس الگوریتم خود را با دو الگوریتم دیگر که در [۸] و [۹] ارائه شده، مقایسه کردیم. الگوریتم ارائه شده در [۹] یا به بیان دیگر الگوریتم Dominant pruning یکی از انواع الگوریتم های موجود در دسته ی Simple flooding محسوب می شود که از یک رهیافت حریصانه به منظور کاستن از تعداد نودهای ارسال کننده استفاده می کند. در این الگوریتم هنگامی که نود v پیامی را از نود u دریافت می کند، نود v مجموعه ای از نودها را برای ارسال مجدد پیام تعیین می کند به نحوی که بتوانند $N(N(v))$ را پوشش دهند. الگوریتم ارائه شده در [۸] در دسته الگوریتم های Neighborhood based قرار می گیرد. در این الگوریتم برای ارسال مجدد یک پیام از نودهایی که Periphery نامیده

می شوند، استفاده می شود. این نودها مسئول انتشار پیام در شبکه می باشند. نود **Periphery** نودی است که بیشترین فاصله را از نود ارسال کننده، داراست. به بیان دیگر نود p نود **Periphery** برای u محسوب می شود اگر رابطه ی زیر $[\lambda]$ برقرار باشد (در رابطه ی زیر منظور از توابع $N(x)$ ، $N(p)$ و $N(u)$ به ترتیب نودهای همسایه های با نود x ، p و u می باشد).

$$\#(N(p) \cap N(u)) \leq \#(N(x) \cap N(u)) \quad \forall x \in N(p)$$



شکل (۱) نمونه ای از تولید کوتاهترین مسیر مشترک در شبکه

در روش پیشنهادی این مقاله، برای تشکیل کوتاهترین مسیر مشترک از ارتباط مبتنی بر چگالی استفاده شده است. الگوریتم **DBSCAN**، ارائه شده توسط **Ester et al** در سال ۱۹۹۶ یکی از نماینده های اصلی الگوریتم های مبتنی بر چگالی و ارتباط می باشد. دو پارامتر ورودی ϵ و $MinPts$ برای تعریف این الگوریتم به کار گرفته می شوند که به ترتیب برابر با فاصله همسایگی و حداقل تعداد همسایه ها می باشند. از این دو پارامتر برای تعریف موارد زیر در الگوریتم **DBSCAN** استفاده می شود:

- **ε - Neighborhood** برای نقطه ی x ، به صورت رابطه ی $N_\epsilon(x) = \{y \in X \mid d(x,y) \leq \epsilon\}$ تعریف می شود. این رابطه معرفی کننده همسایه های نقطه ی x می باشد. منظور از $d(x,y)$ در این رابطه فاصله نقطه ی x از نقطه ی y است و X مجموعه نقاط موجود در محیط را نشان می دهد.
- یک **core object** نقطه ای با بیش از $MinPts$ همسایگی می باشد، به بیان دیگر یک **core object** بیش از $MinPts$ نقطه، همسایه دارد.
- **density - reachable** بودن نقطه ی y نسبت به **core object** x ، به معنی وجود ترتیبی متناهی از **core object** ها بین x و y می باشد. ارتباط در میان **core object** ها باید به نحوی باشد که در آن هر کدام از **core object** ها متعلق به ϵ - Neighborhood قبلی شان باشند.
- یک **density connectivity** بین دو نقطه ی x و y برقرار خواهد بود، اگر این دو نقطه از یک **core object** خاص، **density - reachable** باشند. با توجه به آنچه گفته شد، ویژگی **density connectivity** بین نقاط، یک رابطه ی متقارن است و مجموعه ی نقاط در دسترس **core object** ها کلاسترها را تشکیل خواهند داد. بعلاوه در این الگوریتم به ناطقی که به هیچ **core object** ای متصل نمی باشند **outlier** گفته می شود (این نقاط توسط هیچ کلاستری پوشش داده نمی شوند) و نقاط غیر **core object** درون یک کلاستر نیز **boundary** نامیده می شوند. پس الگوریتم **DBSCAN**، الگوریتمی مبتنی بر شمار فراوانی و همسایگی است. در الگوریتم پیشنهادی این مقاله دو پارامتر ورودی الگوریتم **DBSCAN** به ترتیب برابر با برد موج رادیویی نودها و بیشترین تعداد همسایه، در میان نودهای مجاور هم، در نظر گرفته شده است (در بخش معرفی روش به طور کامل به آن پرداخته شده است). بعلاوه نودهایی که در روش پیشنهادی به عنوان **outlier** تشخیص داده می شوند به کمک مرحله ی ترمیمی که به الگوریتم **DBSCAN** افزوده ایم به **core object** ای از کلاسترها متصل می شوند و به این طریق پوشش کامل شبکه تضمین می شود.

در پایان و پیش از تشکیل نهایی کوتاهترین مسیر مشترک، به کمک عملیات اصلاحی که مزیت اصلی در روش پیشنهادی مان محسوب می شود، تعداد نودهایی که در عمل ارسال فراگیر شرکت دارند، به حداقل می رسد. در نظر داشته باشید این عمل اصلاح به صورت محلی و مستقل، توسط هر نود صورت می گیرد (هیچ نودی اطلاعات شبکه را به صورت کامل در اختیار ندارد بلکه تنها دارای اطلاعات محلی است). در ادامه به تعریف مسئله، معرفی روش پیشنهادی، نحوه ی هرس کردن نودهای ارسال کننده در الگوریتم پیشنهادی و در انتها تحلیل و شبیه سازی روش پیشنهادی خواهیم پرداخت.

۲. تعریف مسئله

در کاربردهایی از شبکه های **Ad Hoc** یا حسگر بی سیم که نیاز است پیامی که توسط نودی ارسال می شود بدست همه ی نودهای شبکه برسد، حداقل کردن تعداد پیام های **forward** شده برای ارسال فراگیر، مسئله اصلی می باشد. پس باید تعداد نودهایی که عمل ارسال مجدد انجام می دهند، حداقل شود. از نمادگذاری های زیر برای تعریف این مسئله استفاده می کنیم.

- $N(u)$: مجموعه نودهای همسایه نود u .
- $S(u)$: مجموعه ی نودهای ارسال کننده و همسایه با نود u .

- $N_g(u) = \bigcup_{t \in S(u)} N(t)$ که این مجموعه شامل همسایه های نودهای ارسال کننده ی مجاور با نود u ، می باشد.
- نودی از همسایه های نود u ، عمل ارسال مجدد را انجام می دهد که دارای حداقل اشتراک با $N_g(u)$ باشد (رابطه ی ۱).

$$N(u) \cap N_g(u) \text{ be Minimized} \quad (1)$$

در واقع باید نودهایی در عمل ارسال مجدد شرکت کنند که حداقل همپوشانی را با همسایه های ارسال کننده شان دارا باشند.

۳. معرفی روش پیشنهادی

- پیش از معرفی جزئیات روش پیشنهادی، نیاز است مفاهیم زیر را معرفی کنیم.
- نود **header**: نودی از نودهای یک کلاستر می باشد که دارای بیشترین تعداد همسایه است.
- نود **overlap**: نودی که در بیش از یک کلاستر قرار دارد، نود **overlap** می باشد. به بیان دیگر نود **overlap** در محدوده ی ارسال بیش از یک نود **header** قرار دارد.
- نود **link**: ممکن است وضعیت ای پیش بیاید که یک نود **header** هیچ نود **header** یا **overlap** دیگری را در برد رادیویی خودش نداشته باشد. در این صورت به منظور اتصال این **header** به دیگر **header** ها، نودی در مجموعه همسایه های این **header** یافته می شود که در همسایگی اش دارای نودی متعلق به کلاستری دیگر باشد به چنین نودی و نودی از نودهای همسایه آن که عضو کلاستر همسایه می باشند، نودهای **link** می گوئیم. در واقع نودهای **link** در الگوریتم پیشنهادی، زیر مجموعه ای از نودهای **boundary** در الگوریتم DBSCAN محسوب می شود.
- در الگوریتم پیشنهادی فرض شده، هر نود دارای یک ID (شماره شناسایی) یکتا می باشد. توزیع نودها در شبکه به صورت تصادفی و یکنواخت بوده، بعلاوه ارتباط نودها از طریق امواج رادیویی صورت می گیرد (برد موج رادیویی نودها با یکدیگر برابر فرض شده است). الگوریتم پیشنهادی شامل دو مرحله می باشد، بخش اول شامل جمع آوری اطلاعات از شبکه به منظور یافتن نودهای ارسال کننده (همان **core object** های الگوریتم DBSCAN) می باشد. در این مرحله فرض کردیم شبکه برای بازه ی زمانی بسیار کوتاهی ثابت است، این بازه ی زمانی برابر با زمان لازم برای تبادل پیام ها جهت تشکیل کوتاهترین مسیر مشترک می باشد (در بخش ۱.۳ نحوه ی انجام تبادل پیام، به صورت مشروح بیان شده است). نودی به عنوان نود ارسال کننده، انتخاب می شود که ماگزیم تعداد همسایه ها را در میان نودهای مجاورش داشته باشد و از آنجایی که مجموعه ی نودهای ارسال کننده باید دارای ویژگی **density - reachable** در بین خودشان باشند، این ویژگی به نحوی که در قسمت ۱.۳ بیان شده، برقرار خواهد شد. در پایان از میان اعضای ارسال کننده، آنهایی که غیر ضروری هستند، حذف می شوند. بخش دوم الگوریتم، آغاز به کار گیری این مسیر برای ارسال فراگیر توسط اعضای شبکه، در بازه ی زمانی ای برابر با Δt می باشد. بعد از بازه ی زمانی Δt اعضای شبکه با بروز کردن اطلاعات همسایه هایشان کوتاهترین مسیر مشترک جدیدی را تولید خواهند نمود. به دلیل کاهش قابل ملاحظه تعداد نودهای ارسال کننده در شبکه های **Ad Hoc** یا حسگر بی سیم توسط الگوریتم پیشنهادی، هزینه ی مرحله ی جمع آوری اطلاعات از شبکه، قابل توجهی می باشد که کمک مسیر تولید شده، هر پیام آرسالی توسط هر نود در زمانی کوتاه به همه ی اعضای شبکه خواهد رسید.

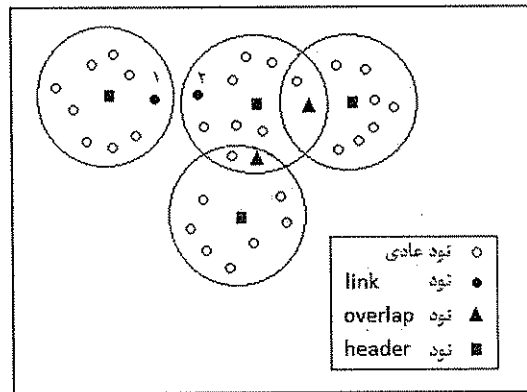
۱.۳. تشکیل کوتاهترین مسیر مشترک

- در بخش اول الگوریتم پیشنهادی، فعالیت های زیر صورت می گیرد:
- بررسی همسایگی: هر نود، وجود خود را از طریق یک ارسال محلی، به همسایگانش اعلام می دارد و همزمان تعداد همسایه های یک گام فاصله (**1-hop**) خود را با شمردن پیام های محلی دریافت شده از نودهای مجاورش محاسبه می کند. تعداد همسایه های هر نود، در واقع **MinPts** هر نود را نشان می دهد. هر نود **MinPts** خود را به منظور داشتن دید بهتر نودهای همسایه از خودش ارسال محلی می کند.
- کاندید شدن نودها به عنوان **header** توسط خود نودها: بر اساس دیدی که هر نود تا کنون بدست آورده است، اگر مقدار **MinPts** اش، بیش از مقدار **MinPts** نودهای همسایه اش باشد، خود را به عنوان یک **header** کاندید می کند و پیامی مبنی بر کاندید شدن خود به عنوان **header** برای نودهای همسایه اش ارسال می کند.
- افزودن کاندیدهایی به مجموعه کاندید شده به عنوان **header** توسط نودهای دیگر موجود در شبکه: اگر یک نود غیر کاندید به عنوان **header**، پیامی از نودهای همسایه اش مبنی بر کاندید بودن حداقل یکی از آنها به عنوان **header** دریافت نکرده باشد، به همسایه ای با بزرگترین **MinPts** پیامی مبنی بر این که خود را به عنوان کاندید برای **header** معرفی کند، ارسال می دارد. نودهایی که چنین پیامی را دریافت می کنند، پیامی مبنی بر کاندید بودنشان به عنوان **header** برای همسایه هایشان ارسال خواهند کرد.
- شناسایی نودهای کاندید نقش **overlap**: نودهایی که بیش از یک پیام مبنی بر **header** بودن از همسایه هایشان دریافت داشته اند، خود را به عنوان کاندید برای نقش **overlap** تشخیص می دهند و پیامی مبنی بر کاندید بودنشان به عنوان **overlap** برای همسایه هایشان ارسال می دارند.
- تبادل لیست همسایه ها: هر نود لیست نودهای همسایه اش را برای همسایه هایش ارسال می دارد. این لیست های همسایگی، توسط هر نود ذخیره می شود. این لیست ها توسط نودها، برای فرآیند هرس کردن به کار گرفته می شود.
- فرآیند هرس کردن: در این مرحله نودهای **header** و **overlap** غیر ضروری حذف می شوند فرآیند هرس کردن که در بخش ۲.۳ به صورت کامل معرفی خواهد شد، تعداد **header** ها و **overlap** ها در مناطق پر جمعیت را به خوبی کاهش می دهد. نودهایی که در پایان این مرحله همچنان به عنوان نود

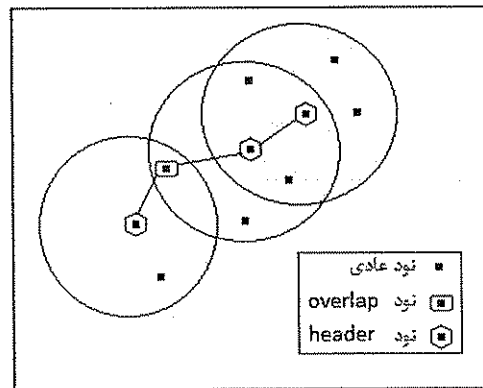
overlap یا **header** باقی می ماند، نودهای ارسال کننده در الگوریتم پیشنهادی که در واقع همان **core object** های الگوریتم DBSCAN هستند، را تشکیل خواهند داد و این نودها خود را از طریق ارسال پیامی مناسب به دیگران معرفی خواهند کرد.

انتخاب نودهای **link**: به منظور اجتناب از بخش بخش شدن شبکه و عدم پوشش صد در صدی، نیاز است نودهای **header** ای که در برد رادیویی خودشان دارای نود **header** یا **overlap** نمی باشند، از میان اعضای کلاسترشان نودهایی را طبق تعریف نود **link** در بخش ۳، انتخاب کنند و به مجموعه نودهای ارسال کننده، بیفزایند (شکل (۲)). اگر **header** ای در کلاسترش چند نود **link** پیدا کرد، طبق قراردادی (مثلا انتخاب کوچکترین ID در میان آن ها) یکی از آنها را انتخاب می کند و به آن نود این امر را اطلاع می دهد (نود **link** شماره ۱ در شکل (۲)) و آن نود **link**، دیگر نود **link** (نود **link** شماره ۲ در شکل (۲)) را از عضویت اش در کوتاهترین مسیر مشترک آگاه می کند.

با انجام فعالیت های بالا کوتاهترین مسیر مشترک برای ارسال پیام یافته می شود. این کوتاهترین مسیر مشترک شامل نودهای **header** و **overlap** ضروری و احتمالا تعدادی نود **link** می باشد، نودهای **overlap** و **link** اتصال میان نودهای **header** را برقرار می کنند. پس خاصیت **density-reachable** بودن بین **core object** ها، در الگوریتم پیشنهادی برقرار خواهد بود. علاوه چون در الگوریتم پیشنهادی هر نود حداقل یک نود به عنوان **header** در همسایگی اش داراست و اتصال **header** ها نیز به کمک نودهای **overlap** و **link** برقرار می باشد، پوشش صد در صدی توسط الگوریتم پیشنهادی قطعی است. شکل (۳) یک نمونه از اجرای الگوریتم پیشنهادی می باشد. در این شکل یک شبکه شامل ۱۰ نود، نشان داده شده است که نودها در آن به صورت تصادفی در محیط جای گرفته اند. با اجرای الگوریتم پیشنهادی بر روی شکل (۳) کوتاهترین مسیر مشترک برای این شبکه، شامل سه نود **header** و یک نود **overlap** خواهد بود. در شکل (۳) هرگاه نودی بخواهد پیامی را ارسال کند، یک ارسال محلی انجام می دهد. سپس اعضایی از شبکه که عضو کوتاهترین مسیر مشترک می باشند، عمل ارسال مجدد را انجام خواهند داد تا پیام بدست همه برسد (کوتاهترین مسیر مشترک بر روی شکل (۳) به صورت پاره خط هایی به هم پیوسته نشان داده شده است). همانطور که در بخش تحلیل خواهید دید، با این که دید هر نود از محیط اطرافش محلی است ولی روش پیشنهادی کارایی بالایی برای ارسال فراگیر دارا خواهد بود.



شکل (۲) نودهای **link**



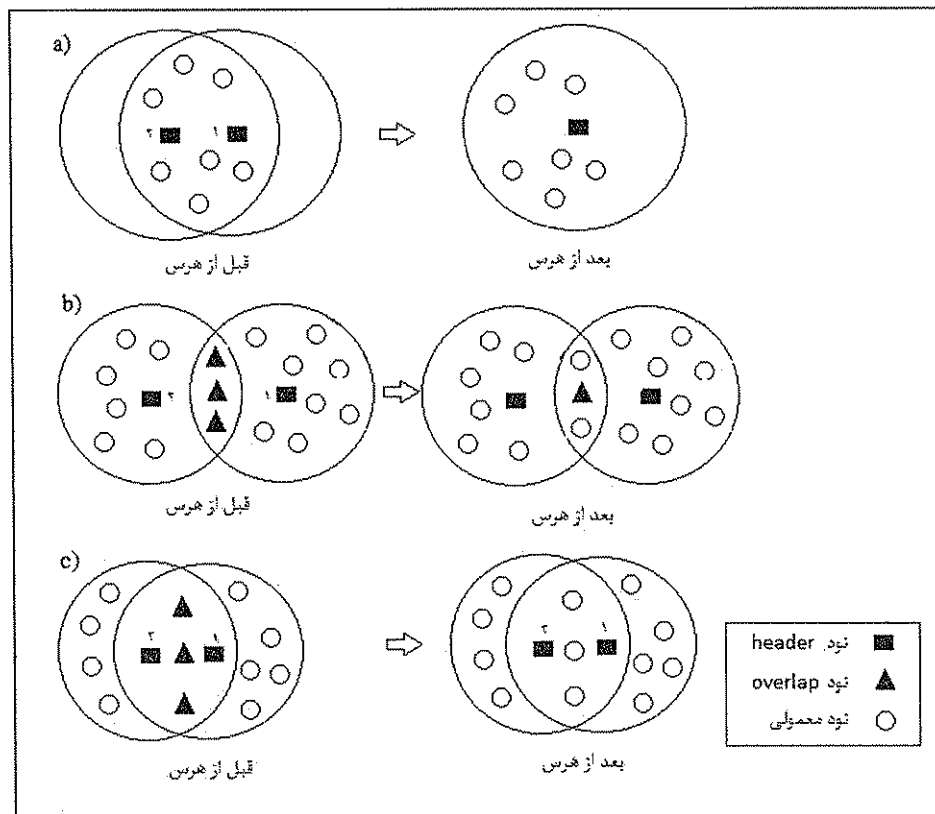
شکل (۳) یک نمونه از اجرای الگوریتم پیشنهادی بر روی یک شبکه ی ۱۰ نودی

۲.۲. فرآیند هرس کردن

همانطوری که در بخش های قبلی گفته شد، در الگوریتم پیشنهادی روی نتیجه حاصل از به کارگیری الگوریتم DBSCAN فرآیند هرس کردن اعمال می شود. این فرآیند هرس کردن عامل اصلی بالا بودن کارایی الگوریتم پیشنهادی برای ارسال فراگیر محسوب می شود و توسط هر نود به صورت مستقل انجام می گیرد. ویژگی اصلی فرآیند هرس کردن، توانایی حذف نودهای **header** و **overlap** دارای قابلیت یکسان، در مناطق پر جمعیت می باشد. توجه داشته باشید در فرآیند

هرس کردن، هدف، کاستن از تعداد نودهای شرکت کننده در عمل ارسال یک پیام به همه می باشد. هرچه تعداد نودهای شرکت کننده در تشکیل کوتاهترین مسیر مشترک کمتر باشد، تعداد ارسال های مجدد برای رسیدن پیامی به تمامی اعضا کاهش می یابد و سرعت ارسال و کارایی بیشتری در ارسال یک پیام حاصل می شود. در عمل هرس کردن، نودهایی با ویژگی های زیر از عمل ارسال مجدد حذف خواهند شد. به بیان دیگر نودهایی با ویژگی های زیر نود عادی محسوب می شوند و از کوتاهترین مسیر مشترک کنار گذاشته می شوند.

- اگر دو یا چند header دارای همسایه های یکسانی باشند تنها یکی از آنها باید باقی بماند و بقیه باید از کوتاهترین مسیر مشترک کنار گذاشته شوند. به شکل (۴) قسمت a نگاه کنید.
- اگر دو یا چند نود overlap متعلق به header های یکسانی باشند، تنها یکی از آنها باید باقی بماند و بقیه باید کنار گذاشته شوند. در شکل (۴) قسمت b سه نود overlap را می بینید که در میان header های ۱ و ۲ قرار گرفته اند ولی تنها به یکی از آنها نیاز است و باقی کنار گذاشته می شوند.
- نودهای overlap در میان header هایی که همدیگر را می بینند، باید کنار گذاشته شوند. این وضعیت در شکل (۴) قسمت c نشان داده شده است.



شکل (۴) فرآیند هرس کردن

۴. تحلیل

در این مقاله به منظور مقایسه الگوریتم پیشنهادی خود، با دیگر الگوریتم های ارسال فراگیر از مقیاسی به نام بازده [۸] استفاده نموده ایم. در این مقیاس تعداد انتقال های مورد نیاز برای پوشش یک شبکه توسط هر الگوریتمی با تعداد مورد نیاز در وضعیت ایده آل مقایسه می شود. قبل از محاسبه بازده ی الگوریتم پیشنهادی، به صورت خلاصه به معرفی مقیاس بازده [۸] می پردازیم.

$$\text{بازده} = \frac{T_{\text{ideal}}}{T_{\text{proposed}}} \quad (۲)$$

در رابطه ی ۲ منظور از T_{ideal} و T_{proposed} به ترتیب تعداد انتقال مورد نیاز برای ارسال فراگیر در وضعیت ایده آل و در الگوریتم پیشنهادی می باشد. توجه کنید در وضعیت ایده آل، هر نود هر پیام را دقیقاً یکبار دریافت می کند. بنابراین باید هیچ همپوشانی ای بین برد موج رادیویی نودهای ارسال کننده، وجود نداشته باشد.

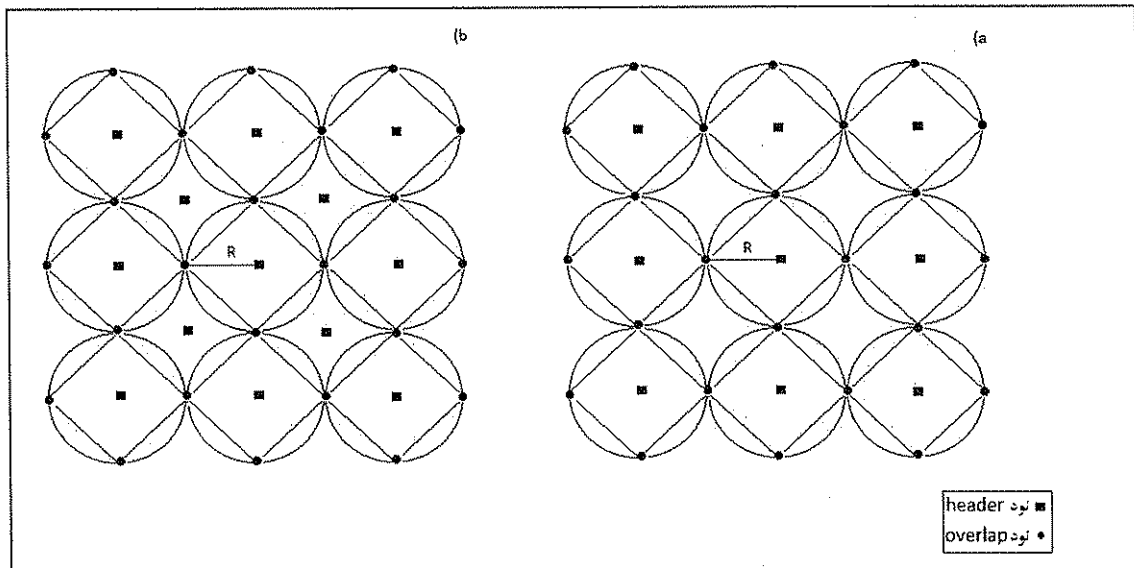
اگر N تعداد نودهای شرکت کننده در شبکه، R برد موج رادیویی و p چگالی نودها در شبکه باشد، ناحیه پوشانده شده توسط شبکه برابر با رابطه ی ۳ خواهد بود [۸].

$$\text{ناحیه پوشانده شده} = \frac{N}{p} \quad (۳)$$

با توجه به این که محدوده ی پوشش هر نود، دایره ای به مساحت πR^2 می باشد، تعداد انتقال ها در وضعیت ایده ال برابر با رابطه ی ۴ خواهد بود [۸].

$$T_{\text{ideal}} = \frac{N}{\pi R^2} \quad (۴)$$

از آنجایی که محاسبه تعداد انتقال های مورد نیاز در وضعیت کلی برای الگوریتم پیشنهادی دشوار می باشد، محدوده ی بازده الگوریتم پیشنهادی را به کمک در نظر گرفتن بهترین حالت و بدترین حالت بدست می آوریم. با توجه به ویژگی های الگوریتم پیشنهادی، شکل (۵) قسمت a بهترین حالت و شکل (۵) قسمت b بدترین حالت توپولوژی شبکه، برای الگوریتم پیشنهادی را نشان می دهد. در شکل (۵) فضا به کمک لوزی های مفرش شده است. نقاط مربع شکل نودهای header و نقاط دایره ای نودهای overlap می باشند.



شکل (۵) توپولوژی های بهترین وضعیت و بدترین وضعیت در الگوریتم پیشنهادی

اکنون می خواهیم تعداد انتقال های لازم برای ارسال فراگیر در الگوریتم پیشنهادی را محاسبه کنیم. از آنجایی که طبق رابطه ی ۳ مساحت پوشش داده شده توسط شبکه برابر با $\frac{N}{p}$ است و مساحت هر لوزی برابر با $2R^2$ می باشد، بنابراین تعداد لوزی ها در شبکه برابر با $\frac{N}{2pR^2}$ خواهد بود (تاثیر نودهای واقع در لبه های مرزی (انتهایی) توپولوژی شبکه را نادیده بگیرید). در شکل (۵) قسمت a دارای دو نوع لوزی هستیم. نیمی از لوزی ها دارای یک نود در مرکز خود هستند و نیمی دیگر نودی ندارند. بعلاوه نودهای واقع در رئوس لوزی های این توپولوژی با چهار لوزی اشتراک دارند. پس میانگین تعداد نودهای هر لوزی برابر با رابطه ی ۵ خواهد بود (در رابطه ی ۵ اعداد $\frac{1}{4}$ و $\frac{1}{2}$ به ترتیب، بدلیل وجود دو نوع لوزی و مشترک بودن هر نود overlap در میان ۴ لوزی درج شده است).

$$\frac{1}{2} \left(\left(1 + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} \right) + \left(0 + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} \right) \right) = \frac{3}{2} \quad (۵)$$

در شکل (۵) قسمت b، تنها یک نوع لوزی داریم که همه ی آنها دارای یک نود در مرکز خود هستند. بعلاوه نودهای واقع در رئوس لوزی ها در میان چهار لوزی مشترک می باشند. پس میانگین تعداد نودهای هر لوزی در شکل (۵) قسمت b برابر با رابطه ی ۶ خواهد بود.

$$\left(1 + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4}\right) = 2 \quad (6)$$

با توجه به رابطه های ۵ و ۶ تعداد نودهای انتقال دهنده پیام برای بهترین و بدترین حالت به ترتیب برابر با رابطه های ۷ و ۸ خواهد بود.

$$\text{برای بهترین وضعیت} = \left(\frac{N}{2\rho R^2}\right) \times \frac{3}{2} \quad (7)$$

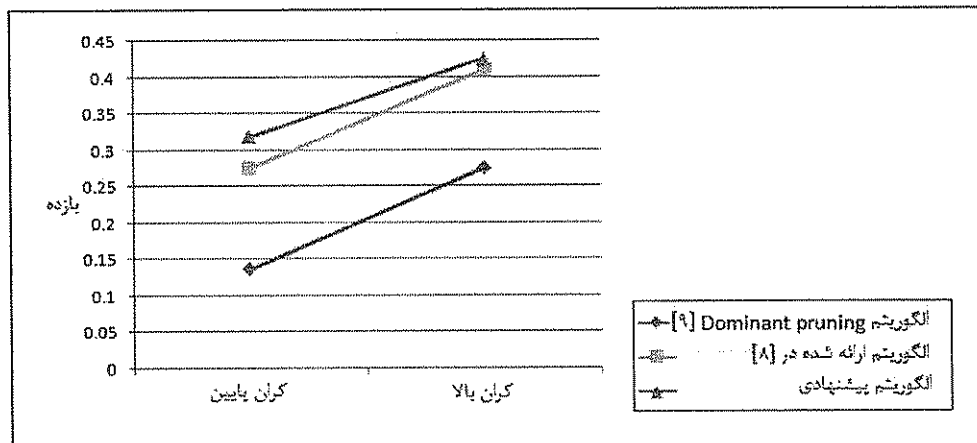
$$\text{برای بدترین وضعیت} = \left(\frac{N}{2\rho R^2}\right) \times 2 \quad (8)$$

پس با توجه به رابطه های ۲ و ۴ بازده الگوریتم پیشنهادی برابر بهترین حالت و بدترین حالت به ترتیب برابر با رابطه های ۰.۴۲۴۴ و ۰.۳۱۸۳ خواهد بود (روابط ۹ و ۱۰).

$$\text{بازده} = \frac{\left(\frac{N}{\pi\rho R^2}\right)}{\left(\frac{N}{2\rho R^2} \times \left(\frac{3}{2}\right)\right)} = 0.4244 \quad (9)$$

$$\text{بازده} = \frac{\left(\frac{N}{\pi\rho R^2}\right)}{\left(\frac{N}{2\rho R^2} \times (2)\right)} = 0.3183 \quad (10)$$

با توجه به آنچه محاسبه نمودیم، کران بازده پیش بینی شده برای روش پیشنهادی [۰.۳۱۸۳، ۰.۴۲۴۴] می باشد. به طریق مشابه در [۸] کران های بازده برای الگوریتم های ارائه شده در [۸] و [۹] به ترتیب برابر با [۰.۲۷۵، ۰.۴۱] و [۰.۱۳۶، ۰.۲۷۵] محاسبه شده است. همانطور که در شکل (۶) می بینید کران پایین و کران بالای بازده در الگوریتم پیشنهادی از کران های پایین و بالای الگوریتم ارائه شده در [۸] بهتر می باشد. بعلاوه کران پایین پیش بینی شده برای الگوریتم پیشنهادی حتی از کران بالای الگوریتم Dominant pruning [۹] نیز بهتر است (Dominant pruning نام الگوریتم ارائه شده در [۹] می باشد). در ادامه و در مرحله شبیه سازی، نتایج حاصل از شبیه سازی الگوریتم ها را مقایسه خواهیم کرد.

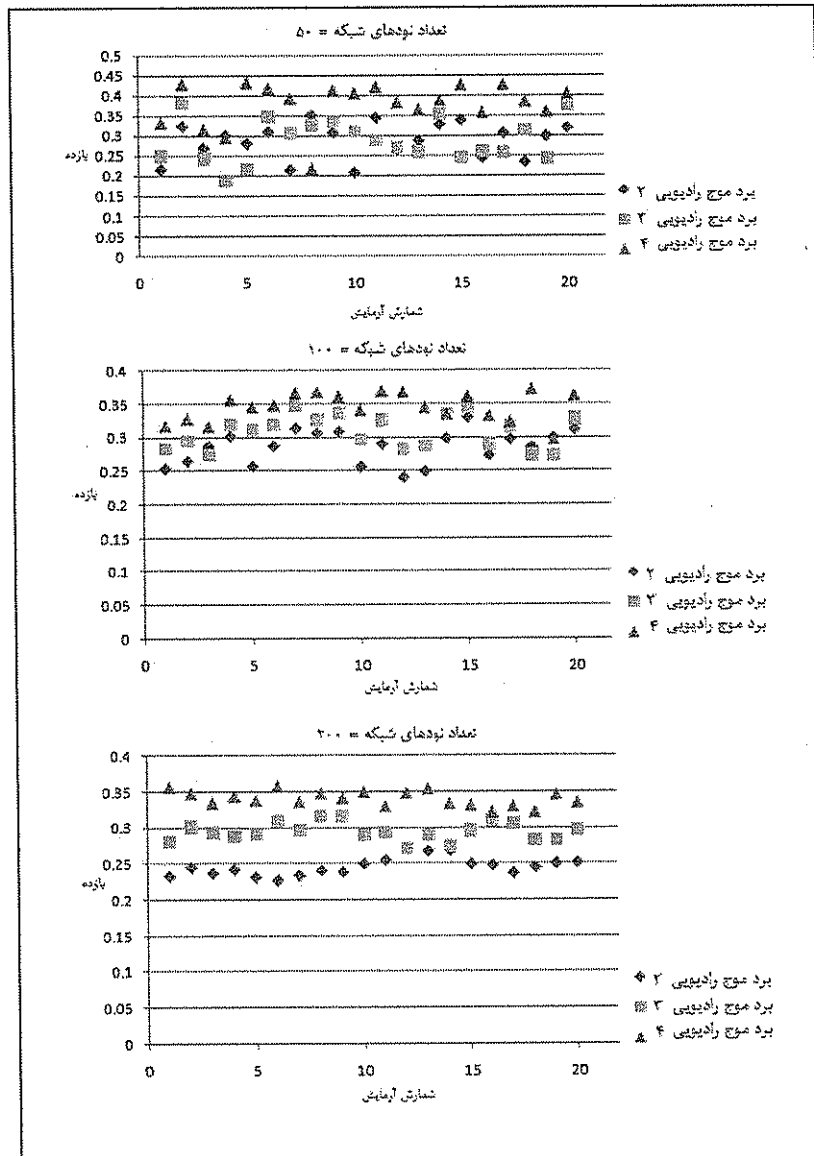


شکل (۶) مقایسه کران های بازده الگوریتم های پیشنهادی، الگوریتم ارائه شده در [۸] و الگوریتم Dominant pruning

۲.۳. شبیه سازی

الگوریتم پیشنهادی با استفاده از شبیه ساز JSIM (Java Simulator) شبیه سازی شده است. دلیل استفاده از JSIM امکان استفاده از ویژگی های زبان جاوا و داشتن قابلیت های ایجاد نودهای مستقل، امکان ارسال و دریافت پیام و بافر کردن پیام ها توسط نودها در این محیط شبیه سازی می باشد. برای شبیه سازی روش

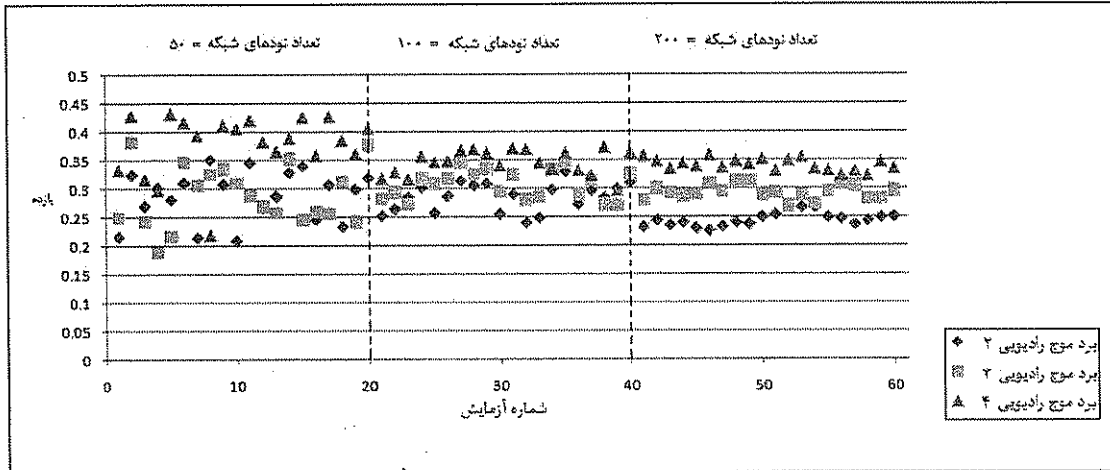
پیشنهادی یک فضای مربعی شکل ۱۰ در ۱۰ در نظر گرفته شد و نودها در این فضای مربعی شکل به صورت تصادفی و یکنواخت توزیع گردید. شبیه سازی روش پیشنهادی، بر روی شبکه هایی با ۵۰ تا ۲۰۰ نود و به ازای بردهایی برابر با ۲، ۳ و ۴ اجرا شده است. برای هر برد موج رادیویی، آزمایش را ۲۰ بار تکرار کرده ایم (شکل های (۷) و (۸)). همانطور که در شکل (۸) می بینید، با افزایش تعداد نودها میزان بازده بیشتر به کران های پیش بینی شده، نزدیک می شوند. روندی که در شکل (۸) دیده می شود، همگرایی الگوریتم را به سمت بازده ای در محدوده ۰.۲۵ تا ۰.۳۵ نشان می دهد، که در واقع تاییدی بر آنچه به صورت تئوری محاسبه شد، می باشد. هر نمودار موجود در شکل (۷) نتایج حاصل از ۶۰ بار آزمایش الگوریتم پیشنهادی بر روی شبکه ای با تعداد نودهایی معین می باشد. در واقع به ازای هر کدام از بردهای ۲، ۳ و ۴ بیست بار آزمایش تکرار شده است. شکل (۸) نتایج شکل (۷) را به صورت یکپارچه نشان می دهد.



شکل (۷) نتایج

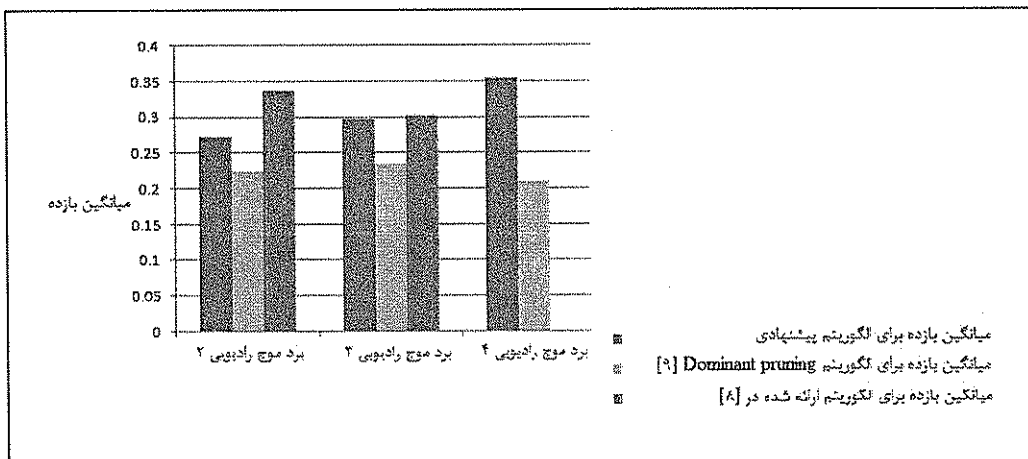
اکنون می خواهیم نتایج شبیه سازی الگوریتم پیشنهادی خود را با دو الگوریتم دیگر ارائه شده در [۸] و [۹] مقایسه کنیم. برای این امر، این دو روش را شبیه سازی نمودیم و آزمایش هایی مشابه با آنچه بر روی الگوریتم پیشنهادی انجام دادیم، بر روی آنها اجرا نمودیم. سپس با محاسبه میانگین بازده ی آزمایش های صورت گرفته روی آنها، نتایج حاصل را با نتایج حاصل از شبیه سازی الگوریتم پیشنهادی، مقایسه نمودیم. این مقایسه در شکل (۹) ارائه شده است (در واقع میانگین بازده ی

آزمایش‌ها در شکل (۸) به ازای هر یک از بردهای ۲، ۳ و ۴ محاسبه شده است و نتایج حاصل با میانگین متناظر در روش‌های ارائه شده در [۸] و [۹] مقایسه گردیده است.



شکل (۸) مجموعه نتایج شکل ۷ به صورت یکپارچه

شکل (۹) که مقایسه‌ی میانگین نتایج حاصل از شبیه‌سازی می‌باشد، نشان‌دهنده میانگین بازده همواره بهتر برای الگوریتم پیشنهادی نسبت به الگوریتم Dominant pruning [۹] به ازای تمامی بردهای ۲، ۳ و ۴ می‌باشد. اما بازده الگوریتم پیشنهادی و الگوریتم ارائه شده در [۸] برای بردهای ۲ و ۳ تقریباً برابر است. از آنجایی که بازده حاصل از به کارگیری الگوریتم ارائه شده در [۸] برای شبکه‌هایی که دارای نودهایی با برد ۴ می‌باشند ناچیز است، در شکل (۹) ستون مربوط به برد ۴ برای الگوریتم ارائه شده در [۸] خالی گذاشته شده است. البته در مقاله [۸] نیز، در قسمت نتایج شبیه‌سازی آن، برای شبکه‌هایی با ۱۰۰ و ۲۰۰ نود، به ازای برد ۴ نتیجه‌ای ارائه نشده است که دلیل این امر کارایی پایین این الگوریتم برای بردهای بزرگ می‌باشد. توجه داشته باشید، افزایش برد موج رادیویی نودها، باعث همپوشانی بیشتر در میان ناحیه پوشش داده شده توسط نودها می‌شود. روش پیشنهادی این مقاله بر روش ارائه شده در [۸] بدلیل توانایی فعالیت در برد های رادیویی بزرگ دارای مزیت می‌باشد. همانطور که در شکل (۹) می‌بینید، با افزایش برد نودها میانگین بازده الگوریتم ارائه شده در [۸] کاهش می‌یابد که نشان می‌دهد این الگوریتم در شبکه‌هایی با همپوشانی زیاد خوب عمل نمی‌کند. این نقطه‌ی مقابل الگوریتم پیشنهادی این مقاله می‌باشد. الگوریتم پیشنهادی این مقاله که مبتنی بر DBSCAN و هرس نودهای ارسال‌کننده در نواحی پرجمعیت می‌باشد برای بردهای بزرگ نیز دارای بازده مناسبی خواهد بود.

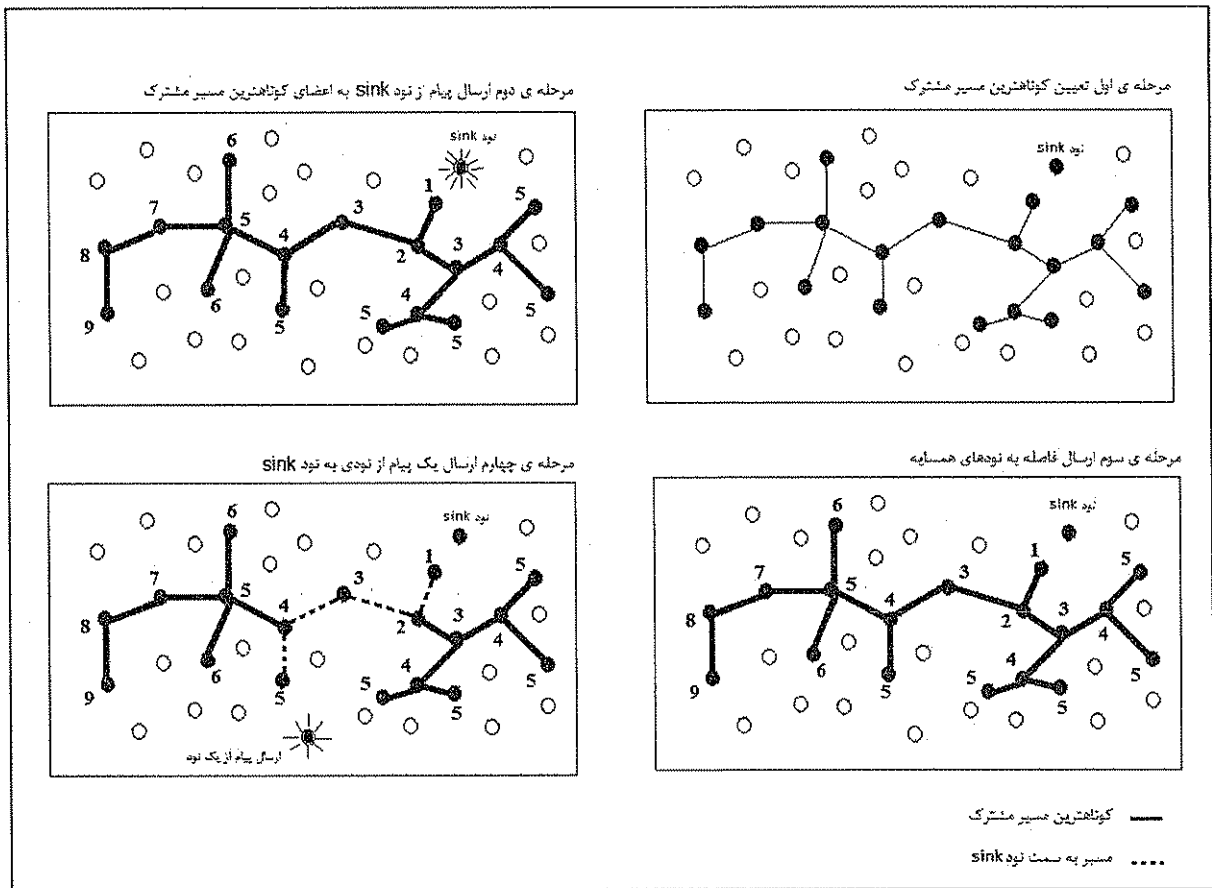


شکل (۹) مقایسه میانگین بازده الگوریتم‌ها

۵. شبکه‌های حسگر

شبکه‌های حسگر وضعیتی خاص از شبکه‌های Ad Hoc محسوب می‌شوند. در این شبکه‌ها به صورت دوره‌ای اطلاعاتی به یک نود خاص که sink نام دارد، ارسال می‌شود. نود sink، اطلاعات را از شبکه جمع‌آوری کرده و برای پردازش به سرورهای نهایی ارسال می‌کند. استفاده از روش‌های موجود در دسته‌ی

Simple flooding برای ارسال اطلاعات، می تواند کشنده باشد. چون حجم داده ی زیادی به شبکه تحمیل می شود که بخش بیشتر آن، زائد می باشد طبق تعریفی که در مقدمه ارائه شد، منظور از کوتاهترین مسیر مشترک در الگوریتم پیشنهادی، مسیری است که تمام نودها یا در آن عضو هستند یا در همسایگی یکی از اعضای عضو کوتاهترین مسیر مشترک، قرار دارند. از آنجایی که در شبکه های حسگر، پیام هر نود باید به دست نود sink برسد الگوریتم پیشنهادی برای شبکه های حسگر با کمی تفاوت اجرا می شود (شکل (۱۰)). وقتی یک کوتاهترین مسیر مشترک تشکیل شد، پیامی توسط نود sink ارسال می شود، این پیام حاوی یک counter (شمارنده) می باشد هر عضو کوتاهترین مسیر مشترک با دریافت این پیام یک واحد به آن اضافه می کند و آن را برای همسایه هایش ارسال می دارد. در پایان وقتی تمامی اعضای کوتاهترین مسیر مشترک پیام نود sink را دریافت کردند، هر نود از اعضای کوتاهترین مسیر مشترک می داند پیام نود sink را با چند ارسال دریافت نموده است یا به عبارتی فاصله اش از نود sink چقدر است. سپس هر نود عضو کوتاهترین مسیر مشترک این فاصله را برای همسایه هایش ارسال می دارد، نودهای عضو کوتاهترین مسیر مشترک، فاصله همسایه های عضوشان در کوتاهترین مسیر مشترک را نگهداری می کنند. هنگامی که پیامی از یکی از نودهای شبکه به دست یکی از اعضای کوتاهترین مسیر مشترک می رسد، آن عضو در میان همسایه های عضوش در کوتاهترین مسیر مشترک، نودی را که فاصله کمتری با نود sink دارد، انتخاب می کند، ID آن نود را به پیام اضافه کرده و پیام را ارسال می دارد. همسایه ای که ID خود را در پیام ارسالی می بیند، پیام دریافتی را به همین طریق ارسال مجدد می کند و این روند ادامه می یابد تا پیام به دست نود sink برسد با این کار تنها اعضای از نودهای کوتاهترین مسیر مشترک، پیام را ارسال کرده اند که نیاز به ارسال آنهاست. توجه داشته باشید در شبکه های حسگر رساندن پیام به نود sink اهمیت دارد نه رساندن پیام به تمامی اعضا. مزیت دیگری که در این روش وجود دارد امکان حرکت برای نود sink است نود sink می تواند در محدوده ی برد، اعضای عضو کوتاهترین مسیر مشترک حرکت کند و تمام پیام ها را دریافت کند.



شکل (۱۰) به کار گیری الگوریتم پیشنهادی این مقاله برای شبکه های حسگر بی سیم

۶- نتیجه

در این مقاله دسته های الگوریتم های ارسال فراگیر (broadcast) بررسی شد الگوریتمی مبتنی بر DBSCAN برای حداقل کردن تعداد ارسال مجدد، در ضمن حفظ خاصیت دریافت پیام توسط همه ی اعضا ارائه شد و به کمک آن کوتاهترین مسیر مشترک همبندی، جهت ارسال پیام در شبکه تولید گردید. بعلاوه برای

بالا بردن کارایی الگوریتم پیشنهادی، یک مرحله هرس کردن به الگوریتم افزودیم. برای مقایسه الگوریتم پیشنهادی با دیگر الگوریتم ها از معیار بازده [۸] استفاده نمودیم و به صورت تحلیلی محدوده ی بازده را برای الگوریتم پیشنهادی تعیین کردیم و آن را با کران های بازده برای الگوریتم های ارائه شده در [۸] و [۹] مقایسه نمودیم. سپس الگوریتم پیشنهادی را به کمک JSIM (Java Simulator) شبیه سازی کرده و نتایج حاصل از شبیه سازی را با نتایج حاصل از شبیه سازی الگوریتم های [۸] و [۹] مقایسه نمودیم، نتایج حاصل، آن چه در بخش تحلیل بدست آمد را تایید می کرد. در پایان نحوه به کار بردن الگوریتم پیشنهادی این مقاله برای شبکه های حسگر بی سیم که حالت خاصی از شبکه های Ad Hoc می باشند، ارائه شد.

۷ - تقدیر و تشکر

در پایان لازم می دانم از استادان گرامی، آقایان محمد داورپناه جزئی و محمد علی منتظری که در تهیه این مقاله مرا راهنمایی و کمک نمودند، تشکر کنم.

۸ - مراجع

- [۱] R. Szwedczyk; J. Polastre; A. Mainwaring; D. Culler; "Lessons from a sensor network expedition", in Proceedings of 1st European Workshop on Wireless Sensor Networks (EWSN '04), January 2004.
- [۲] E. Biagioni; K. Bridges; "The application of remote sensor technology to assist the recovery of rare and endangered species", Special issue on Distributed Sensor Networks for the International Journal of High Performance Computing Applications, Volume 16 N. 3, August 2002.
- [۳] P. Juang; H. Oki; Y. Wang; M. Martonosi; L. S. Peh; D. Rubenstein; "Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with zebraNet", in Tenth international conference on architectural support for programming languages and operating systems on (ASPLOS-X) ACM Press, 96-107, 2002.
- [۴] I. S. Committee; "Wireless LAN Medium ACCESS CONTROL (MAC) and Physical Layer Specifications", IEEE 802.11 Standard, New York, ISBN 1559379359, 1997.
- [۵] C. Ho; K. Obraczka; G. Tsudik; K. Viswanath; "Flooding for Reliable Multicast in Multi-hop Ad hoc Networks", International Workshop in Discrete Algorithms and Methods for Mobile Computing and Communication, 64-71, 1999.
- [۶] S. Ni; Y. Tseng; Y. Chen; J. Sheu; "The Broadcast Storm Problem in a Mobile Ad hoc Network", International Workshop on Mobile Computing and Networks, 151-162, 1999.
- [۷] B. Williams; T. Camp; "Comparison of Broadcasting Techniques for Mobile Ad hoc Networks", In Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC), 194-205, 2002.
- [۸] A. Samalam; S. Perreau; A. Dadej; "Optimal Broadcast in Ad-hoc and Sensor Network", Intelligent Sensors, Sensor and Networks and Information 3rd International Conference, ISSNIP 2007, 467 - 472, 2007.
- [۹] H. Lim; C. Kim; "Flooding in Wireless Ad Hoc Networks", Computer Comm. J., vol. 24, no. 3-4, 353-363, 2001.
- [۱۰] W. Lou; J. Wu; "On reducing broadcast redundancy in ad hoc wireless Network", IEEE trans. on mobile computing vol. 1 No 02, 111-122, 2002.
- [۱۱] J. Wu; F. Dai; "a generic distributed broadcast schema in ad hoc wireless Network", Vol. 53, No, 10, 1343-1354, 2004.
- [۱۲] Y. Li; S. Peng; W. Chu; "An efficient distributed broadcasting algorithm for wireless ad hoc networks", In: Proc. 6th Int. Conf. Par & Dist. Comp. (PDCat 05), Dalian China, 75-79, 2005.
- [۱۳] Y. Ganjali; N. McKeown; "Routing in highly dynamic topology", In: Proc. IEEE SECON 2005, Santa Clara Calif, 164-175, 2005.
- [۱۴] P. Kysanur; P. R. Choudhury; I. Gupta; "Smart Gossip: An adaptive gossip-based broadcasting services for sensor networks", In: Proc. IEEE Conf. On Mobile Ad hoc & Sensor Systems, Vancouver Canada, 91-100, 2006.