

یک الگوریتم هوشمند سیلاب سراسری در شبکه های موردنی

بی سیم مبتنی اتوماتاهاي يادگير توزيع شده

محمد رضا میبدی

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

دانشگاه صنعتی امیر کبیر

تهران ایران

meybodi@aut.ac.ir

جواد اکبری ترکستانی

دانشکده مهندسی کامپیوتر

دانشگاه آزاد اسلامی

اراک ایران

j-akbari@iau-arak.ac.ir

چکیده: در شبکه های موردنی بی سیم، با توجه به قابلیت تحرک گره ها، عملیات همه پخشی با فرکانس بالائی به منظور تعیین مسیر های ارسال مورد استفاده قرار می گیرد. بکارگیری مکانیزم سیلاب سراسری در انتشار پیام های همه پخشی، روش متداولی است که در شبکه های موردنی با توجه به محدودیت پهنای باند اتصال های بی سیم و انرژی گره ها و همچنین تغییرات مداوم توپولوژی، هزینه های بسیاری را به شبکه های مذکور تحمیل می نماید. یک مجموعه حاکم متصل، ستون فقراتی را در سطح شبکه ایجاد می نماید که ارسال پیام های همه پخشی از طریق آن، نرخ ارسال مجدد پیام در سطح شبکه، و در نتیجه پهنای باند و توان مصرفی را به حداقل ممکن می رساند. مسأله مجموعه حاکم متصل کمینه یک مسأله بفرنج است. در این مقاله، یک مکانیزم هوشمند سیلاب سراسری، موسوم به DLA-GF¹، مبتنی بر اتوماتاهاي يادگير توزيع شده، به منظور کاهش هزینه های ناشی از طوفان انتشار، در شبکه های موردنی بی سیم پیشنهاد می گردد. الگوریتم تقریبی پیشنهادی، با یافتن مجموعه حاکم متصل نزدیک به بهینه ای برای گراف شبکه، ستون فقراتی را در سطح شبکه ایجاد می نماید که هزینه های انتشار پیام را به حداقل می رساند. الگوریتم پیشنهادی بر حسب کاردينالیتی مجموعه حاکم متصل ایجاد شده، با بهترین الگوریتم های موجود مقایسه شده و نشان داده می شود که نتایج به مراتب بهتری را تولید می نماید.

کلمات کلیدی: سیلاب سراسری، مسأله طوفان انتشار، مجموعه حاکم متصل، اتوماتاهاي يادگير توزيع شده

۱. مقدمه

یک شبکه موردنی، شبکه بی سیم چند گامه ای است که فاقد هرگونه زیر ساخت شبکه ای ثابت بوده و بنا به نیاز، در یک پوشش موردنی ایجاد و سپس از بین می رود. شبکه های موردنی همچنین قادر به پشتیبانی از میزبان های متحرک نیز می باشند. محدودیت پهنای باند اتصال های بی سیم و تغییرات مداوم توپولوژی شبکه، مسیر یابی را در شبکه های موردنی به مسأله ای چالش انگیز مبدل ساخته است[2-3]. سیلاب سراسری روشی است که در اغلب پروتکل های مسیر یابی مبتنی بر توپولوژی موردن استفاده قرار می گیرد. مسأله طوفان انتشار، ناشی از ارسال غیر ضروری پیام ها توسط عده ای از گره های میانی، سیلاب سراسری را به روشی بسیار پر هزینه و ناکارامد مبدل ساخته است[25]. راه حل های مختلفی به منظور طراحی بک الگوی سیلاب سراسری کارامد پیشنهاد گردیده است، که ایجاد ستون فقرات شبکه از طریق تعیین مجموعه حاکم متصل کمینه از مؤثرترین روش های موجود به شمار می رود[7].

یک شبکه موردنی را می توان به کمک گراف ($G = (V, E)$) نشان داد که در آن، رئوس گراف معرف گره های میزبان بوده و هر یال از گراف، دو میزبان را که در محدوده ارسال یکدیگر قرار دارند متصل می نماید. مجموعه حاکم S ، زیر مجموعه ای از V می باشد مشروط به آنکه، هر گره در $S - V$ حداقل با یک گره از مجموعه S مجاور باشد. مجموعه حاکم متصل S ، یک مجموعه حاکم به صورت یک زیر گراف متصل از G می باشد. مجموعه حاکم متصل کمینه S ، مجموعه حاکم متصلی است که پایین ترین کاردينالیتی را داشته باشد. مجموعه حاکم کمینه و مجموعه حاکم متصل کمینه به ترتیب به عنوان روش هایی برای خوشه بندی[7-8, 9, 11-12, 23-26] و انتشار پیام [1, 3-4] در شبکه های موردنی بی سیم، مورد توجه قرار گرفته اند. نشان داده می شود که یافتن مجموعه حاکم کمینه و مجموعه حاکم متصل کمینه مسائلی

¹ Multi-hop

² Global Flooding

³ Dominating Set (DS)

⁴ Minimum Connected Dominating Set (MCDS)

بغنج هستند[5] و حتی برای یک گراف دیسک واحد نیز مساله مجموعه حاکم متصل کمینه مساله ای بفرنج است[6].

بهمن لحاظ، الگوریتم های تقریبی متمرکر و توزیع شده بسیاری برای حل آن پیشنهاد گردیده است. گوها و کاهلر[27] دو الگوریتم اکتشافی حریصانه را برای ایجاد مجموعه حاکم متصل پیشنهاد کردند، که در الگوریتم اول مجموعه حاکم متصل با یک گره شروع و سپس کامل می شود. اما در الگوریتم دوم ابتدا یک مجموعه حاکم با اتصال ضعیف ایجاد و سپس گره های میانی به منظور ایجاد مجموعه حاکم متصل انتخاب می شوند. بوتنکو و همکارانش[13] الگوریتم اکتشافی مبتنی بر هرس را پیشنهاد کردند. در این الگوریتم، ابتدا مجموعه رئوس گراف به عنوان مجموعه حاکم متصل در نظر گرفته می شود. سپس، چنانچه حذف گره ای موجب گستینگی گراف نشده و هر یک از همسایه های آن نیز حداقل با یک عضو دیگر از مجموعه حاکم مجاور باشد، گره مذکور حذف می گردد. لی و همکارانش[14] با اتصال گره های مجموعه مستقل بیشینه از طریق درخت اشتاینر با حداقل گره های اشتاینر روشی را برای تعیین مجموعه حاکم متصل پیشنهاد کردند. الذوبی، وان و فریدر[9,28] نیز دو الگوریتم توزیع شده مبتنی بر مجموعه مستقل بیشینه را برای تعیین مجموعه حاکم متصل پیشنهاد کردند. در این روش، ابتدا بکمک یک استراتژی برچسب زنی گره های مجموعه مستقل بیشینه تعیین و سپس از طریق درخت پوشای کمینه[10] متصل می گردند. و و دای [29] الگوریتمی توزیع شده مبتنی بر هرس کردن را پیشنهاد کردند که در آن، ابتدا هر راسی که دو همسایه غیر متصل داشته باشد، به مجموعه حاکم افزوده می شود. سپس با حذف گره هایی که تمامی همسایه های غیر متصل آنها از طریق اعضاء دیگر مجموعه پوشش داده می شوند، مجموعه حاکم متصل ایجاد می گردد.

در این مقاله، یک مکانیزم هوشمند سیلاب سراسری مبتنی بر اتوماتای یادگیر توزیع شده، به منظور کاهش هزینه های ناشی از طوفان انتشار، در شبکه های موردي بی سیم پیشنهادی با یافتن مجموعه حاکم متصل بسیار نزدیک به کمینه ای برای گراف شبکه، ستون فقراتی را در سطح شبکه ایجاد می نماید که هزینه های انتشار پیام را به حداقل می رساند. برای این منظور، ابتدا یک شبکه از اتوماتاهای یادگیر، همراه با گراف شبکه به گره های میزان تخصیص داده می شود. در هر مرحله، اتوماتاهای یادگیر با انتخاب تصادفی عمل های خود، یک مجموعه حاکم متصل ایجاد می نمایند. مجموعه انتخابی از طرف محیط تصادفی اتوماتاهای یادگیر ارزیابی می گردد. سرانجام، طی یک فرایند تکرار پذیر، اتوماتاهای یادگیر، در انتخاب عمل های خود به سیاست مشترکی همگرا می شوند که منجر به انتخاب بهینه مجموعه حاکم متصل برای گراف شبکه خواهد شد. نشان داده می شود که با انتخاب مناسب پارامتر یادگیری، الگوریتم همواره با احتمالی نزدیک به یک موفق به انتخاب مجموعه بهینه خواهد شد. کارائی روش پیشنهادی در مقایسه با سایر روش های موجود ارزیابی گردیده و نشان داده می شود که کارآینالیتی مجموعه حاکم ایجاد شده توسط الگوریتم پیشنهادی در مقایسه با بهترین الگوریتم های موجود، به مراتب کمتر است. در ادامه این مقاله و در بخش ۲ به ترتیب اتوماتاهای یادگیر و اتوماتاهای یادگیر توزیع شده به اختصار شرح داده می شود. در بخش ۳ نیز الگوریتم پیشنهادی به تفصیل مورد بررسی قرار داده می شود. در بخش ۴ نتایج آزمایش های شبیه سازی، و در بخش ۵ خلاصه و نتیجه گیری مقاله ارائه می شود.

۲. اتوماتاهای یادگیر

اتوماتای یادگیر[22-16] ماشینی است که می تواند تعدادی متناهی عمل را انجام دهد. هر عمل انتخاب شده توسط یک محیط احتمالی ارزیابی می شود و نتیجه ارزیابی در قالب سیگنالی مثبت یا منفی به اتوماتا داده می شود و اتوماتا از این پاسخ در انتخاب عمل بعدی تأثیر می گیرد. هدف نهایی این است که اتوماتا یاد بگیرد تا از بین اعمال خود، بهترین عمل را انتخاب کند. بهترین عمل، عملی است که احتمال دریافت پاداش از محیط را به حداکثر برساند.

اتوماتای یادگیر با ساختار متغیر را می توان توسط چهارتایی $\{\alpha, \beta, p, T\}$ نشان داد که $\alpha \equiv \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه اعمال اتوماتا، $\{\beta_1, \beta_2, \dots, \beta_r\} \equiv \{\beta_1, \beta_2, \dots, \beta_r\}$ مجموعه ورودی های اتوماتا، $p = \{p_1, \dots, p_r\}$ بردار احتمال انتخاب هریک از عمل ها و $T[\alpha(n), \beta(n), p(n+1)] = T[\alpha(n), \beta(n), p(n+1)]$ الگوریتم یادگیری می باشد. الگوریتم زیر یک نمونه از الگوریتم های یادگیری خطی است. فرض می کنیم عمل α_i در مرحله n انتخاب شود.

- پاسخ مطلوب از محیط

$$\begin{aligned} p_i(n+1) &= p_i(n) + a[1 - p_i(n)] \\ p_j(n+1) &= (1 - a)p_j(n) \quad \forall j \neq i \end{aligned} \tag{1}$$

- پاسخ نامطلوب از محیط

⁵ Unit Disk Graph(UDG)

$$\begin{aligned} p_i(n+1) &= (1-b)p_i(n) \\ p_j(n+1) &= (b/r-1)+(1-b)p_j(n) \quad \forall j \neq i \end{aligned} \quad (2)$$

در روابط (1) و (2)، a پارامتر پاداش و b پارامتر جریمه می‌باشند. با توجه به مقادیر a و b سه حالت را می‌توان در نظر گرفت: اگر a و b با هم برابر باشند، الگوریتم را L_{RP} ، هنگامی که b از a خیلی کوچکتر باشد، الگوریتم را L_{REP} و اگر b مساوی صفر باشد آن را L_{RI} می‌نامیم[22-16-20].

اتوماتاهای یادگیر توزیع شده یک شبکه از اتماتاهای یادگیر است که بمنظور حل یک مسأله خاص به صورت گروهی با یکدیگر همکاری می‌نمایند[22]. اتماتای یادگیر توزیع شده را می‌توان بوسیله یک گراف جهت دار، همانطور که در شکل ۳ دیده می‌شود، مدلسازی کرد بگونه‌ای که در آن مجموعه رئوس گراف متناظر است با مجموعه اتماتاهای یادگیر در سطح شبکه، و مجموعه یالهای خروجی برای هر راس متناظر است با مجموعه عمل‌های اتماتای متناظر با آن راس. اتماتای یادگیر توزیع شده را می‌توان توسط چند تایی V, E, T, V_0 نمایش داد که در آن $V = \{v_1, v_2, \dots, v_n\}$ معرف مجموعه رئوس گراف متناظر با مجموعه اتماتاهای یادگیر $E \subset A \times A$. $A = \{A_1, A_2, \dots, A_n\}$ معرف یالهای گراف و $E_i = \{e_{i1}, e_{i2}, \dots, e_{ij}, \dots, e_{ir_i}\}$ تعداد عمل‌های اتماتای A_i متناظر با مجموعه عمل‌های اتماتا A_i بگونه‌ای که یال e_{ij} متناظر است با عمل α_{ij} از اتماتای A_i . و $T = \{T^1, T^2, \dots, T^n\}$ معرف مجموعه الگوریتم‌های یادگیر و V_0 گره ریشه را در اتماتای یادگیر توزیع شده نشان می‌دهد.

عملکرد اتماتاهای یادگیر توزیع شده را می‌توان بین شکل تشرح کرد. در ابتدا اتماتای یادگیر ریشه با توجه به بردار احتمالات یکی از عمل‌های خود را انتخاب می‌نماید. انتخاب عمل مذکور موجب فعال سازی اتماتای یادگیری که در انتهای دیگر یال انتخابی قرار دارد، می‌شود. اتماتای فعال شده نیز به همین ترتیب عملی را انتخاب و اتماتای دیگری را فعال می‌نماید. این سلسله مراتب انتخاب عمل و فعال سازی تا زمانی که یک اتماتای برگ فعل گردد ادامه پیدا می‌کند. لازم به ذکر است که در هر تکرار همواره اتماتای ریشه فعل سازی می‌شود اما سایر اتماتاهای در یک تکرار به صورت احتمالی فعل می‌شوند. انتخاب مسیرهای مختلف میان اتماتای ریشه و اتماتاهای برگ آنقدر تکرار می‌گردد تا در نهایت یک مسیر با احتمالی نزدیک به ۱ انتخاب گردد. عمل انتخابی توسط اتماتای برگ به محیط اعمال می‌گردد و پاسخ محیط به تمامی اتماتاهای فعل شده در مسیر اتماتای ریشه تا اتماتای برگ برگشت داده می‌شود. سپس هر یک از اتماتاهای بر اساس پاسخ دریافتی از محیط، بكمک الگوریتم یادگیری، نسبت به بروز کردن بردار احتمالات عمل خود اقدام می‌نمایند[22].

۳. الگوریتم هوشمند سیالاب سراسری مبتنی بر اتماتاهای یادگیر توزیع شده

فرض می‌شود که گره‌های میزبان از طریق یک کanal بی سیم همه پخشی عمومی و با استفاده از آنتن‌های همه سویه، که همگی از محدوده ارسال رادیوئی یکسانی برخوردارند، با یکدیگر ارتباط برقرار می‌نمایند. به عبارت دیگر، در طراحی و تحلیل الگوریتم پیشنهادی، گراف معادل شبکه یک گراف دیسک واحد در نظر گرفته می‌شود. هر گره درون شبکه یک شماره شناسائی منحصر به فرد داشته و شماره شناسائی گره‌های مجاور خود را نیز می‌داند. هر دو گره دلخواه از شبکه تنها در صورتی که یک کanal ارتباطی مستقیم دو سویه میان آنها برقرار باشد، به یکدیگر متصل می‌شوند. بر این اساس، گراف معادل شبکه گرافی غیر جهت دار در نظر گرفته می‌شود. ابتدا به هر یک از گره‌های گراف یک اتماتای یادگیر اختصاص داده می‌شود. شبکه اتماتاهای یادگیر حاصل، هم‌ریخت با گراف شبکه، را می‌توان بكمک یک چندتائی به صورت A, α توصیف کرد، که در آن $A = \{A_1, A_2, \dots, A_m\}$ مجموعه اتماتاهای یادگیر، معادل با مجموعه گره‌های گراف و $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_m\}$ مجموعه عمل‌های اتماتاهای یادگیری باشد، بگونه‌ای که بازاء هر $\alpha_i = \{\alpha_{i1}, \alpha_{i2}, \dots, \alpha_{in_i}\}$ ، $\alpha_i \in \alpha$ معرف مجموعه عمل‌های اتماتای یادگیر A_i باشد که مجموعه تمامی گره‌ها به غیر از گره n_i را شامل می‌شود. به منظور بهینه سازی رفتار اتماتاهای یادگیر، در این مقاله، از اتماتاهایی با مجموعه عمل متغیر[18] استفاده شده است. وضعیت هر گره می‌تواند در دو حالت فعل و یا غیرفعال قرار گیرد، که در ابتدای هر مرحله تمامی گره‌ها در وضعیت غیر فعل قرار داده شده و به محض انتخاب فعل می‌شوند. سپس اتماتاهای یادگیر، طی یک فرایند تکرار پذیر، در انتخاب عمل‌های خود به سیاست مشترکی همگرا می‌شوند که منجر به انتخاب یک مجموعه حاکم متصل بسیار نزدیک به بهینه برای گراف شبکه خواهد شد. تشریح دقیق مراحل الگوریتم در ادامه آمده است.

گام ۱. گرده آغازین به صورت تصادفی و بکمک یک اتوماتی یادگیر مجزا (که اتوماتی آغازین خواند می شود.) انتخاب می گردد. شماره شناسائی گرده انتخابی به مجموعه گرده های حاکم اضافه و به همراه یک پیام فعال سازی برای گرده انتخابی ارسال می گردد.

گام ۲. با انتخاب تصادفی نخستین گرده، فعال سازی گرده ها آغاز می شود. در هر مرحله، گرده فعال با افزودن شماره شناسائی گرده های مجاور خود، لیست گرده های محکوم را بروز می نماید. بروز رسانی لیست عمل های مجاز نیز مطابق با آنچه در گام ۳ آمده صورت می گیرد. سپس، گرده فعال یکی از عمل های خود را انتخاب و شماره شناسائی گرده معادل با عمل انتخابی را به مجموعه حاکم اضافه می نماید. لیست گرده های محکوم و مجموعه حاکم به همراه یک پیام فعال سازی برای گرده انتخابی ارسال می گردد.

گام ۳. در این مرحله مجموعه عمل و بردار احتمالات تمامی گرده های فعال نشده، با حذف عمل معادل با گرده فعال شده در گام قبل، بروز می گردد.

گام ۴. عمل فعال سازی گرده ها و بروز رسانی لیست عمل ها (گام های ۲ و ۳) تا زمانی که کاردینالیتی مجموعه محکوم کمتر از تعداد گرده های گراف باشد، تکرار می گردد.

گام ۵. اگر مجموعه گرده های انتخاب شده به عنوان مجموعه حاکم، زیر گرافی متصل از گراف G باشد، الگوریتم به گام ۶ می رود. در غیر این صورت، الگوریتم تکرار فعلی را خاتمه می دهد.

گام ۶. آخرین گرده فعال مجموعه حاکم متصل را با کوچکترین مجموعه مشاهده شده مقایسه می نماید، چنانچه کاردینالیتی مجموعه انتخابی از کاردینالیتی کوچکترین مجموعه مشاهده شده کمتر باشد، مجموعه انتخابی جایگزین کوچکترین مجموعه حاکم می گردد و گرده مذکور با ارسال پیامی از گرده های حاکم می خواهد که عمل های انتخابی خود را بکمک الگوریتم یادگیری L_{R-1} پاداش دهند. در غیر این صورت، گرده مذکور با ارسال پیامی از گرده های حاکم می خواهد که عمل های انتخابی خود را جریمه نمایند.

گام ۷. تعیین گرده های حاکم و بروز رسانی بردار های انتخاب عمل گرده ها (گام های ۲ تا ۶) تا زمانی ادامه پیدا می کند که یا تعداد مجموعه های انتخابی از یک مقدار از پیش تعیین شده ای فراتر رود و یا حاصل ضرب احتمالات گرده های انتخاب شده از یک مقدار آستانه بیشتر شود. مجموعه حاکم متصل انتخاب شده در تکرار آخر و قبل از خاتمه الگوریتم به عنوان مجموعه کمینه در نظر گرفته می شود.

الگوریتم سیلاپ سراسری به صورت توزیع شده در گرده های شبکه اجراء می شود. در هر تکرار، روال سیلاپ سراسری طی گام های ۱ و ۷ با انتخاب نخستین گرده حاکم تا زمانی که شرط توقف الگوریتم برقرار شود، اجرا می شود. با اجرای گام ۱ انتخاب گرده های حاکم آغاز شده و با ارسال پیامی گرده انتخابی، فعال و ادامه اجراء به روال حاکم منتقل می گردد. گرده حاکم فعال با اجرای گامهای ۲ تا ۶، حاکم جدیدی را انتخاب و پیام فعال سازی را برای آن ارسال می نماید. این فرایند تا تکمیل مجموعه حاکم تکرار می گردد. پاسخ به صورت پیام جریمه و یا پاداش برای گرده های حاکم برگردانده می شود. با ارسال یک پیام مبنی بر خاتمه تکرار جاری ادامه اجراء در گام ۷ به روال سیلاپ سراسری منتقل می گردد.

۴. نتایج آزمایشات

در این بخش، به منظور ارزیابی عملکرد الگوریتم پیشنهادی، نتایج حاصل از شبیه سازی الگوریتم را با نتایج برخی از بهترین الگوریتم های موجود به لحاظ اندازه مجموعه حاکم (تعداد گرده های حاکم) مقایسه نموده ایم. از میان الگوریتم های مبتنی بر مجموعه مستقل بیشینه، نتایج الگوریتم حریصانه لی [14]، که با ایجاد مجموعه مستقل بیشینه گراف و سپس افزودن گرده های میانی به آن، قادر است کوچکترین مجموعه حاکم متصل را تولید نماید، و از میان الگوریتم های مبتنی بر هرس، نتایج الگوریتم بوتنکو [13] که حتی در اغلب موارد نتایج به مراتب بهتری را نسبت به الگوریتم لی تولید می نماید، و همچنین نتایج الگوریتم زی [15] که با پیچیدگی پیام نسبتا کمتری می تواند مجموعه های حاکم متصلی به کوچکی نتایج الگوریتم بوتنکو تولید نماید، برای مقایسه با نتایج بدست آمده توسط الگوریتم پیشنهادی انتخاب گردیده اند. در این آزمایشات، نرم افزار شبیه ساز، گراف های آزمایشی متصلی را از طریق توزیع تصادفی گرده های میزبان در یک ناحیه مربعی $m \times m$ تولید می نماید. فرض می شود که تمامی گرده های میزبان از محدوده ارسال رادیوئی یکسانی برخوردارند. گرده های میزبانی که در شعاع ارسال یکدیگر قرار دارند قادرند تا بطور مستقیم با یکدیگر ارتباط برقرار نمایند. تمامی ارتباطات نیز دو سویه در نظر

گرفته می شود. پس از توزیع تصادفی گره ها، بررسی می شود که لزوماً گراف تولید شده متصل باشد. مجموعه حاکم متصل بر روی گراف تولید شده ایجاد و کار دینالیتی آن تعیین می گردد. اندازه ناحیه مربعی، محدوده ارسال رادیوئی گره های میزبان و تعداد گره های میزبان درون شبکه، سه پارامتر اساسی هستند که اندازه مجموعه حاکم متصل تولید شده را تحت تاثیر قرار می دهند. در آزمایشات شبیه سازی به عمل آمده چگونگی تاثیر این سه پارامتر بر روی اندازه مجموعه مذکور، تولید شده بوسیله الگوریتم های مختلف مورد بررسی قرار داده می شود.

به منظور مطالعه اثرات تغییر پارامتر یادگیری بر روی اندازه مجموعه حاکم متصل ایجاد شده، ابتدا مقدار پارامتر یادگیری را از 0.04 تا 0.6 تغییر داده و سپس اندازه مجموعه حاکم ایجاد شده توسط الگوریتم پیشنهادی و همچنین تعداد تکرار های الگوریتم به ازاء هر یک از مقادیر فوق اندازه گیری می شود. آزمایشات برای هر مقدار از پارامتر یادگیری، 100 مرتبه تکرار شده و فرض می شود که محدوده ارسال رادیوئی هر گره میزبان 30×100 مربعی باشد. آزمایشات به ترتیب با تعداد گره های 50 ، 70 و 100 تکرار و نتایج بدست آمده در جدول ۱ آورده شده است. مشاهده می شود که در تمامی آزمایشات با افزایش مقدار پارامتر یادگیری اندازه مجموعه حاکم متصل ایجاد شده افزایش یافته و دقت الگوریتم در همکرانی به پاسخ بهینه کاهش می یابد، اما از سوی دیگر، با افزایش پارامتر یادگیری، تعداد تکرار های الگوریتم نیز به مقدار قابل توجهی کاهش می یابد. می توان نشان داد که با افزایش تعداد گره های میزبان، افزایشی کمی در اندازه مجموعه حاکم مشاهده می شود. برای مثال، با تعداد 70 گره و پارامتر یادگیری 0.05 ، اندازه مجموعه حاکم 1.98 و تعداد تکرار ها 258 است، در حالیکه بازه پارامتر یادگیری 0.55 ، اندازه مجموعه حاکم به 4.12 و تعداد تکرار ها به 6 می رسد.

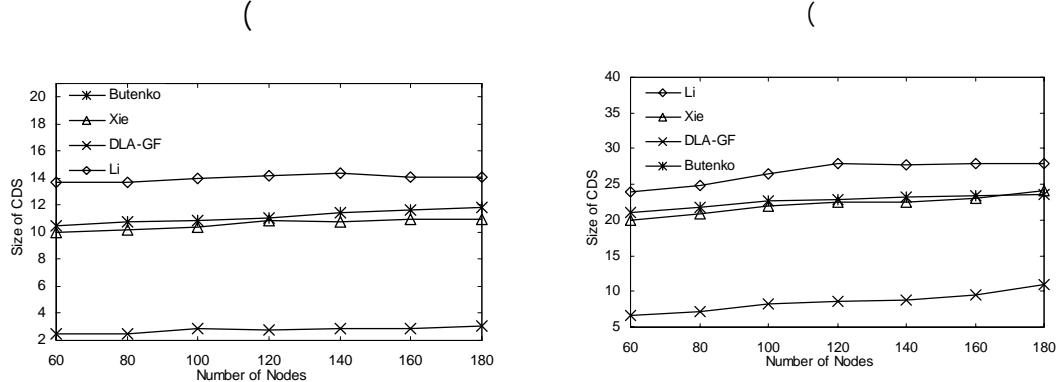
Learning Parameter	Number of Nodes = 50		Number of Nodes = 70		Number of Nodes = 100	
	Size of CDS	Number of Iterations	Size of CDS	Number of Iterations	Size of CDS	Number of Iterations
0.04	2.06	412	2.14	627	2.14	335
0.05	1.94	369	1.98	258	2.16	492
0.06	2.02	168	2.24	361	2.22	396
0.07	1.96	211	2.00	389	2.16	320
0.08	2.04	149	2.04	74	2.02	257
0.09	2.04	192	2.07	92	2.30	222
0.10	2.06	92	2.16	85	2.24	105
0.15	2.20	45	2.34	97	2.28	49
0.20	2.30	23	2.66	58	2.56	59
0.25	2.66	21	2.82	27	2.66	42
0.30	2.70	19	2.84	38	3.16	33
0.35	2.70	22	2.94	16	3.16	49
0.40	3.30	19	3.24	12	3.32	24
0.45	3.24	13	3.70	9	3.66	21
0.50	3.38	9	3.68	9	3.52	11
0.55	3.60	9	4.12	6	4.24	15
0.60	3.77	7	3.72	11	4.50	7

جدول ۱. اندازه متوسط مجموعه حاکم متصل و تعداد تکرار های الگوریتم، بازه تغییرات پارامتر یادگیری از 0.04 تا 0.6 ، با محدوده ارسال رادیوئی 30×100 ناحیه مربعی و تعداد گره های 50 ، 70 و 100 .

فرض می شود که محدوده ارسال رادیوئی هر گره میزبان 20×100 و اندازه ناحیه مربعی 100×100 باشد. تعداد گره های میزبان از 60 تا 180 بازه افزایش داده شده و اندازه مجموعه حاکم متصل تولید شده توسط هر یک از الگوریتم ها تعیین می گردد. آزمایشات برای هر یک از پیکره بندی های مذکور 100 مرتبه تکرار و میانگین نتایج حاصل از آنها برای هر یک از الگوریتم های در شکل ۱.الف نشان داده شده است. به منظور بررسی اثرات تغییر در شعاع ارسال بر روی اندازه مجموعه حاکم متصل، محدوده ارسال رادیوئی را به 30×100 افزایش داده و آزمایشات را به طور مشابه با قبل تکرار می نمائیم. نتایج این آزمایشات نیز در شکل ۱.ب نشان داده شده است. در پایان، به منظور ارزیابی اثرات اندازه ناحیه مربعی بر روی اندازه مجموعه حاکم متصل، اندازه ناحیه مذکور را به 120×120 افزایش داده و میانگین نتایج آزمایشات پایانی در اشکال ۱.الف و ۱.ب میزبان از 60 تا 180 و محدوده ارسال رادیوئی از 20×30 به 20×30 تکرار می نمائیم. نتایج آزمایشات پایانی در اشکال ۲.الف و ۲.ب نشان داده شده است. در شبیه سازی های الگوریتم پیشنهادی، مقدار پارامتر یادگیری برای تمامی تکرار ها 0.2 در نظر گرفته شده و الگوریتم زمانی خاتمه پیدا می کند که، حاصل ضرب احتمالات انتخاب گره ها به بیش از 90 درصد برسد.

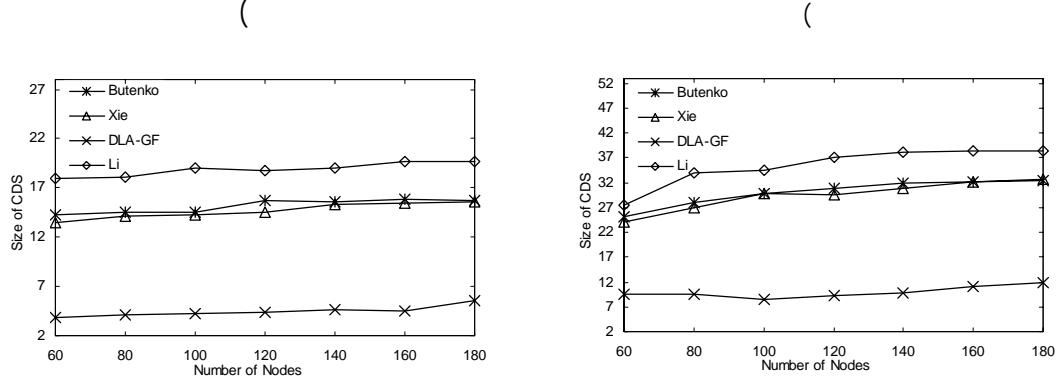
در تمامی نمودارهای اشکال ۱ و ۲ مشاهده می شود که میانگین اندازه مجموعه حاکم متصل برای تمامی الگوریتم ها با افزایش تعداد گره های میزبان در سطح شبکه، افزایش می یابد. با مقایسه نمودار های نشان داده شده در اشکال ۱.الف و ۱.الف با اشکال ۲.الف و ۲.ب می توان دریافت که با افزایش شعاع ارسال رادیوئی گره های میزبان، اندازه مجموعه حاکم متصل

کاهش خواهد یافت. دلیل این امر آنست که گره های حاکم با محدوده ارسال رادیوئی بیشتر، قادرند تا تعداد گره های میزبان بیشتری را پوشش دهند و بنا براین میزبان های توزیع شده در سطح شبکه با تعداد گره های حاکم کمتری پوشش داده می شوند.



شکل ۱. اندازه متوسط مجموعه حاکم متصل بازه تغییرات تعداد گره ها از ۶۰ تا ۱۸۰ ، در ناحیه مربعی 100×100 با محدوده ارسال رادیوئی (الف) ۲۰ و (ب)

از سوی دیگر، با مقایسه نمودار های نشان داده شده در اشکال ۱ و ۲ می توان مشاهده کرد که با افزایش ناحیه مربعی، اندازه مجموعه حاکم متصل نیز برای تمامی الگوریتم ها افزایش پیدا می کند، چرا که میزبان ها در ناحیه وسیع تری توزیع شده در حالیکه گره های میزبان تنها میزبان های محدود در شعاع ارسال خود را پوشش می دهند.



شکل ۲. اندازه متوسط مجموعه حاکم متصل بازه تغییرات تعداد گره ها از ۶۰ تا ۱۸۰ ، در ناحیه مربعی 120×120 با محدوده ارسال رادیوئی (الف) ۲۰ و (ب)

همانطور که در نمودارهای ارائه شده در اشکال ۱ و ۲ مشاهده می شود، الگوریتم لی همواره در تمامی موارد بدترین نتایج را تولید می نماید. نتایج تولید شده توسط الگوریتم بوتنکو بهبود قابل توجهی را نسبت به الگوریتم لی نشان می دهد، اما اندازه مجموعه حاکم تولید شده توسط آن در مقایسه با الگوریتم زی اندکی بزرگتر است. مشاهده می شود که الگوریتم پیشنهادی در مقایسه با الگوریتم های مذکور، قادر است تا مجموعه های حاکم متصل به مراتب کوچکتری را ایجاد نماید. برای مثال، در شکل ۱. الف بازه تعداد ۱۰۰ گره، اندازه متوسط مجموعه حاکم ایجاد شده توسط الگوریتم لی ۲۶,۴، الگوریتم بوتنکو ۲۲,۶، الگوریتم زی ۲۲ و الگوریتم پیشنهادی ۸,۲ می باشد، که نتیجه الگوریتم پیشنهادی تفاوت چشمگیری را نسبت به الگوریتم های دیگر نشان می دهد. مجموعه حاکم ایجاد شده توسط الگوریتم پیشنهادی ستون فقراتی را در شبکه ایجاد می نماید، که هزینه ارسال پیام های چند پخشی و همه پخشی را در مقایسه با سایر روش های موجود به کمترین مقدار ممکن می رساند.

۵. خلاصه و نتیجه گیری

در شبکه های موردنی با توجه به محدودیت های پهنای باند اتصال های بی سیم و انرژی گره ها و همچنین تغییرات مداوم تپیکلوزی شبکه، سیلان سراسری پیام های همه پخشی و طوفان انتشار ناشی از آن، به شدت موجب افت کارائی شبکه خواهد شد. یک مجموعه حاکم متصل، ستون فقراتی را در سطح شبکه ایجاد می نماید که ارسال پیام های همه پخشی از طریق آن، نرخ ارسال مجدد پیام در سطح شبکه، و در نتیجه پهنای باند و توان مصرفی را به حداقل ممکن می رساند. در این

مقاله، یک مکانیزم هوشمند سیلاب سراسری مبتنی بر اتوماتای یادگیر توزیع شده، به منظور کاهش هزینه های ناشی از طوفان انتشار، در شبکه های موردنی بی سیم پیشنهاد گردید. در الگوریتم پیشنهادی، طی یک فرایند تکرار پذیر، اتوماتاهای یادگیر، در انتخاب عمل های خود به سیاست مشترکی همگرا می شوند که منجر به انتخاب بهینه مجموعه حاکم متصل برای گراف شبکه خواهد شد. نشان داده می شود که با انتخاب مناسب پارامتر یادگیری، الگوریتم همواره با احتمالی نزدیک به یک موفق به انتخاب مجموعه بهینه خواهد شد. کارائی روش پیشنهادی در مقایسه با سایر روش های موجود و بكمک آزمایشات شبیه سازی ارزیابی گردید و نشان داده شد که کاربینالیتی مجموعه حاکم ایجاد شده توسط الگوریتم پیشنهادی در مقایسه با بهترین الگوریتم های موجود، به مراتب کمتر است.

مراجع

- Y. P. Chen, A. L. Liestman, "Maintaining Weakly-Connected Dominating Sets for Clustering Ad Hoc Networks," *Ad Hoc Networks*, Vol. 3, pp. 629–642, 2005.
- P. Gupta, P.R. Kumar, "The Capacity of Wireless Networks," *IEEE Transaction on Information Theory*, Vol. 46, No. 2, pp. 388–404, 2000.
- B. Han, W. Jia, "Clustering Wireless Ad Hoc Networks with Weakly Connected Dominating Set," *Journal of Parallel and Distributed Computing*, Vol. 67, pp. 727 – 737, 2007.
- R. Rajaraman, "Topology Control and Routing in Ad Hoc Networks: A Survey," *SIGACT News*, Vol. 33, No. 2, pp. 60–73, 2002.
- B. N. Clark, C. J. Colbourn, and D. S. Johnson, "Unit Disk Graphs," *Discrete Mathematics*, Vol. 86, pp. 165-177. 1990.
- M.V. Marathe, H. Breu, H.B. Hunt III, S.S. Ravi, D.J. Rosenkrantz, "Simple Heuristics for Unit Disk Graphs," *Networks*, Vol. 25, pp. 59–68, 1995.
- Y.Z. Chen, A.L. Liestman, "Approximating Minimum Size Weakly Connected Dominating Sets for Clustering Mobile Ad hoc Networks," *Proceedings of the Third ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'2002)*, pp. 157–164, 2002.
- Y.P. Chen, A.L. Liestman, "A Zonal Algorithm for Clustering Ad Hoc Networks," *International Journal of Foundations of Computer Science*, Vol. 14, No. 2, pp. 305–322, 2003.
- K.M. Alzoubi, P.-J. Wan, O. Frieder, "Maximal Independent Set, Weakly Connected Dominating Set, and Induced Spanners for Mobile Ad Hoc Networks", *International Journal of Foundations of Computer Science*, Vol. 14, No. 2, pp. 287-303, 2003.
- R.G. Gallager, P.A. Humblet and P.M. Spira, "A Distributed Algorithm for Minimum Weight Spanning Trees," *ACM Transaction on Programming Languages and Systems*, Vol. 5, pp. 66-77, 1983.
- J. Wu, B. Wu, and I. Stojmenovic, " Power-aware broadcasting and activity scheduling in ad hoc wireless networks using connected dominating sets," *Journal of Wireless Communications and Mobile Computing*, Vol. 3, pp. 425-438 , 2003.
- O. Dousse, F. Baccelli, and P. Thiran, "Impact of Interferences on Connectivity in Ad hoc Networks," *IEEE/ACM Transactions on Networking*, Vol. 13, No. 2, pp. 425-436, 2005.
- S. Butenko, X. Cheng, C. Oliveira, P. M. Pardalos, "A New Heuristic for the Minimum Connected Dominating Set Problem on Ad Hoc Wireless Networks," *In Recent Developments in Cooperative Control and Optimization*, Kluwer Academic Publishers, pp. 61–73, 2004.
- Y. Li, M. T. Thai, F. Wang, C. W. Yi, P. J. Wang, D. Z. Du, "On Greedy Construction of Connected Dominating Sets in Wireless Networks," *Special issue of Wireless Communications and Mobile Computing (WCMC)*, 2005.
- R. Xie, D. Qi, Y. Li, and J. Z. Wang, "A Novel Distributed MCDS Approximation Algorithm for Wireless Sensor Networks," *Journal of Wireless Communications and Mobile Computing*, 2007.
- K. S. Narendra and K. S. Thathachar, "Learning Automata: An Introduction", New York, *Printice-Hall*, 1989.
- M. A. L. Thathachat, P. S. Sastry, "A Hierarchical System of Learning Automata That Can Learn the Globally Optimal Path," *Information Science*, 1997, Vol.42, pp.743-766.
- M. A. L. Thathachar and B. R. Harita, "Learning Automata with Changing Number of Actions," *IEEE Transactions on Systems, Man, and Cybernetics*, 1987, Vol. SMG17, pp. 1095-1100.
- M. A. L. Thathachat, V.V.Phansalkar, "Convergence of Teams and Hierarchies of Learning Automata in Connectionist Systems," *IEEE Transactions on Systems, Man and Cybernetics*, 1995, Vol. 24, pp. 1459-1469.

20. S. Lakshmivarahan and M. A. L. Thathachar, "Bounds on the Convergence Probabilities of Learning Automata," *IEEE Transactions on Systems, Man, and Cybernetics*, 1976, Vol. SMC-6, pp. 756-763.
21. K. S. Narendra, and M. A. L. Thathachar, "On the Behavior of a Learning Automaton in a Changing Environment with Application to Telephone Traffic Routing," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-10, No. 5, pp. 262-269, 1980.
22. H. Beigy, M. R. Meybodi, "Utilizing Distributed Learning Automata to Solve Stochastic Shortest Path Problems," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol.14, pp. 591-615, 2006.
23. S. Basagni, D. Bruschi, I. Chlamtac, "A mobility-transparent deterministic broadcast mechanism for ad hoc network," *IEEE/ACM Transactions on Networking*, Vol. 7, No. 6, pp-799-807, 1999.
24. I. Chlamtac, S. Kutten, "Tree-based broadcasting in multihop radio networks," *IEEE Transactions on Computing*, Vol. 36, No. 10, pp. 1209-1223, 1987.
25. H. Lim, and C. Kim, "Flooding in wireless ad hoc networks," *Journal of Computer Communications*, Vol. 24, pp. 353-363, 2001.
26. X. Cheng, M. Ding, D. Hongwei, and X. Jia, " Virtual backbone construction in multihop ad hoc wireless networks," *Journal of Wireless Communications and Mobile Computing*, Vol. 6, pp. 183-190, 2006.
27. S. Guha and S. Khuller, "Approximation algorithms for Connected Dominating Sets," *Algorithmica*, Vol. 20, No. 4, pp. 374-387, 1998.
28. K.M. Alzoubi, P.J. Wan and O. Frieder, "Distributed Heuristics for Connected Dominating Sets in Wireless Ad Hoc Networks," *Journal of Communications and Networks*, Vol. 4, No. 1, pp. 22-29, 2002.
29. F. Dai, and J. Wu, "An Extended Localized Algorithm for Connected Dominating Set Formation in Ad Hoc Wireless Networks," *to appear in IEEE Transactions on Parallel and Distributed Systems*, 2004.