مجله علمی پژوهشی «علوم و فناوری‌های پدافند غیرعامل»
سال چهارم، شماره۲، تابستان ۱۳۹۲؛ ص ۸۶- ۷۷

طراحی و شبیه‌سازی یک الگوریتم مسیریابی در شبکه‌های سیار اقتضایی مبتنی برشبکه‌های عصبی مصنوعی

محمود صالح اصفهانی [1*]، زهرا ستوده[2]، علی ناصری[3]

۱- استادیار، ۲- کارشناس ارشد و ۳- استادیار دانشگاه جامع امام حسین (ع)

(دریافت: ۱۳۹۱/۰۷/۰۹، پذیرش: ۱۳۹۲/۰۵/۲۷)

چکیده

شبکه‌های اقتضایی سیار یکی از گزینه‌های مناسب برای ایجاد ارتباط اضطراری در مناطق آسیب دیده از حوادث طبیعی و بحران‌های عمدی و نیز رزمایش است. این شبکه‌ها، از هیچ‌گونه مرکزیت کنترلی در مسیریابی بسته‌های خود استفاده نکرده و در موقعیت‌های خاص که وجود مرکزیت و یا پشتوانه سیمی از لحاظ فیزیکی غیرممکن و یا از لحاظ اقتصادی مقرون به صرفه نیست، کاربرد دارند. شبکه‌های اقتضایی سیار از تعدادی گره متحرک تشکیل شده است که مسیر ارتباطی بین مبدأ و مقصد در آن ممکن است به صورت لحظه‌ای تغییر نماید. متغیر بودن موقعیت نسبی گره‌های تشکیل دهنده، نیاز به الگوریتم مسیریابی چابکی دارد که بتواند تحرک گره‌ها را مدیریت نموده و بسته‌های انتقال یافته را به طرز صحیحی به مقصد برساند، به طوری که هیچ یک از دو طرف ارتباط از وجود تحرک در گره‌های شبکه مطلع نشوند. در مقاله حاضر ضمن بررسی الگوریتم‌های مسیریابی شبکه‌های اقتضایی سیار مبتنی بر شبکه‌های عصبی مصنوعی، یک الگوریتم جدید مبتنی بر کوتاه‌ترین مسیر در الگوریتم‌ها پیشنهاد و تشریح گردیده است. جهت سنجش معیارهای مورد نظر برای ارزیابی کارایی، الگوریتم پیشنهادی در محیط نرم‌افزار NS-2 و با استفاده از شبکه عصبی هاپفیلد شبیه‌سازی گردیده و فعالیت‌های گروهی گره‌ها به صورت تجمعی مورد تجزیه و تحلیل و ارزیابی قرار گرفت. یافته‌های حاصل از نتایج شبیه‌سازی، کارایی بهتر و بهبود در زمان یافتن کوتاه‌ترین مسیر حرکتی بسته‌ها را نشان می‌دهد.

کلید واژه‌ها: شبکه‌های اقتضایی سیار، سربار مسیریابی، تأخیر انتها به انتها، شبیه‌ساز شبکه NS-2.

Design and Simulation of a Routing Algorithm in Mobile Ad-Hoc Networks Based on Artificial Neural Networks

M. Saleh Esfahani[*], Z. Sotoudeh, A. Naseri
Imam Hossein University

(Received: 30/09/2012; Accepted: 18/08/2013)

Abstract

Wireless Ad Hoc networks are among different wireless networks which are extensively studied in the past few years. Ad Hoc networks consist of several mobile nodes with alternate locations. Hence, such a network, has to employ resilient routing algorithms to manage node movement and correct routing in the presence of broken links and outage of availability seamless to users of any two communicating nodes. In this paper different routing strategies based on neural network techniques are studied. Then a new algorithm based in shortest path method and Hopfield neural network is proposed. The proposed algorithm is then simulated using NS-2 network simulator. Numerical results show that the new algorithms can outperform traditional DSR and AODV and has shorter traffic overhead.

Keywords: Ad-hoc Networks, Routing Overhead, End-to-End Delay, NS-2 Network Simulator.

[*]Corresponding Author E-mail: msaleh@ihu.ac.ir

۱. مقدمه

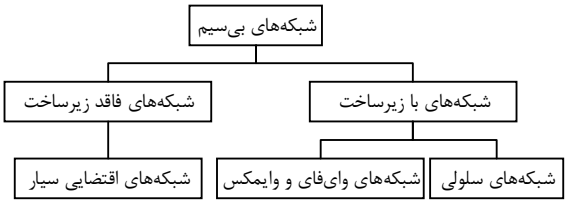
امروزه شبکه‌های بی‌سیم به دلیل ارائه کاربردهای وسیع و همچنین سرویس‌های مناسب، رشد چشمگیری داشته‌اند. علت این موضوع را می‌توان در دو عامل زیر جستجو کرد:

۱- پیشرفت‌هایی که در طراحی سخت‌افزارها رخ داده و سبب شده است که وسایل ارتباطی و محاسباتی بی‌سیم قابل حمل، با توان مصرفی پایین در دسترس همگان قرار گیرد.

۲- افزایش روز افزون زیرساختارهای[1] ارتباطی که دسترسی به شبکه‌های تلفنی و نیز داده را در همه مکان‌ها ممکن می‌سازند [۱]. این شبکه‌ها به سرعت در حال رشد هستند و سرویس‌های مبتنی بر آنها نیز، مرتباً بیشتر و بهتر می‌شود.

اصولاً سیستم‌های بی‌سیم را می‌توان به دو دسته تقسیم کرد [۲] (شکل ۱):

۱- شبکه‌های با زیرساخت[2]
۲- شبکه‌های بدون زیرساخت[3] اقتضایی یا شبکه‌های اقتضایی



شکل ۱. طبقه بندی شبکه‌های بی‌سیم [۲]

یک شبکه بدون زیرساخت یا شبکه اقتضایی تنها شامل گره‌های سیار است و هیچ ایستگاه ثابت مسیردهی در آن به‌کار گرفته نمی‌شود. هر گره سیار هم به عنوان یک میزبان و هم مانند یک مسیریاب عمل می‌کند. در این شبکه‌ها گره‌ها عموماً دارای تحرک هستند، به همین دلیل بافت[4] کاملاً پویایی در این شبکه‌ها وجود دارد. به دلیل وجود گره‌های سیار، امکان کشف مسیر برای ارسال اطلاعات، تنها با وجود الگوریتم‌های مسیریابی[5] پویای مناسب امکان‌پذیر است. در ضمن، این شبکه‌ها از نوع سیستم‌های چند پرشی[6] می‌باشند. به دلیل عدم وجود مدیر مرکزی و کنترل کننده در شبکه اقتضایی، گره‌ها به طور دلخواه می‌توانند وارد شبکه شده و یا خارج شوند.

یکی از موفق‌ترین فناوری‌های شبکه‌های بی‌سیم، استاندارد شبکه محلی IEEE، IEEE 802.11 می‌باشد [۳]. در این استاندارد دو روش برای پیکربندی شبکه‌های محلی بی‌سیم ارائه شده است. روش اول که بیشترین میزان کاربرد را دارد، روش زیرساخت‌دار است. در این روش یک شبکه از گره‌های ثابتی تحت عنوان نقاط دسترسی[7] وجود دارند که سایر گره‌ها برای ارتباط از آنها استفاده می‌کنند. روش دوم، روش بدون ساختار یا اقتضایی است. در این روش گره‌ها به طور مستقیم امکان ارتباط با یکدیگر را پیدا می‌کنند. نکته مهمی که در اینجا وجود دارد این است که پشتیبانی استاندارد IEEE 802.11 از شبکه‌ای اقتضایی بسیار ساده و ابتدایی و برای کاربردهای مهم، غیر کارآمد است. به عنوان مثال در این استاندارد تنها امکان یک پرش وجود دارد و اگر گره‌ها در دید مستقیم و در محدوده تشخیص یکدیگر نباشند، نمی‌توانند با یکدیگر ارتباط برقرار کنند [۳].

به علت طبیعت ناپایدار، کشف و نگهداری مسیر در شبکه‌های اقتضایی از اهمیت زیادی برخوردار است. دو عامل فقدان پهنای باند زیاد و توان باتری محدود در هر گره، سبب تلاش جهت دستیابی به شیوه مسیریابی مقرون به صرفه شده است. انتقال بسته‌ها، طی تعدادی عملیات پیوسته ذخیره‌سازی و ارسال مجدد[8]، توسط مجموعه‌ای از گره‌های واسطه و میانی صورت می‌گیرد. هدف از مسیریابی این است که یک بسته به صورت مطمئن از مبداء به مقصد با تأخیر کمینه انتقال یابد. یافتن مسیر بهینه نیازمند اطلاعاتی نظیر ساختار شبکه و تأخیر و ظرفیت اتصالات می‌باشد [۱].

در مقاله حاضر ضمن بررسی، تحلیل و شبیه‌سازی الگوریتم‌های مسیریابی شبکه‌های اقتضایی سیار مبتنی بر شبکه‌های عصبی مصنوعی، یک الگوریتم بهینه بر اساس محاسبه کوتاه‌ترین مسیر معرفی و نتایج حاصل ارائه می‌شود. در بخش ۲ و ۳ برخی تحقیقات مسیریابی مبتنی بر شبکه‌های عصبی معرفی می‌گردند. الگوریتم پیشنهادی در بخش ۴ معرفی شده و عملکرد آن در بخش ۵ تشریح می‌شود. در بخش ۶ نیز نتایج عددی و مقایسه الگوریتم پیشنهادی با الگوریتم‌های AODV و DSR انجام گردیده و جمع‌بندی مقاله در انتها آورده شده است.

۲. شبکه‌های سیار اقتضایی مبتنی بر شبکه‌های عصبی

یکی از اهداف الگوریتم‌های مسیریابی در شبکه‌های سیار اقتضایی، کاهش سربار ترافیکی در شبکه است. شبکه‌های اقتضایی به دلیل محدودیت ارسال امواج رادیویی مورد استفاده و همچنین عدم وجود زیرساخت دارای مسیرهای چند پرشی هستند. از آنجا که گره‌ها آزادانه و به دلخواه حرکت می‌کنند، مسیرهای موجود به طور مداوم شکسته می‌شوند. شکسته شدن مداوم مسیرها به نوبه خود باعث پویایی و تغییر مداوم بافت شبکه می‌شود. این ویژگی‌ها، نقش الگوریتم‌های مسیریابی را در شبکه‌های اقتضایی برجسته نموده است.

در این شبکه‌ها، الگوریتم‌های مسیریابی باید بتوانند ایجاد، نگهداری و بازسازی مسیرها را با سرعت قابل قبولی انجام دهند.

یکی دیگر از ویژگی‌هایی که باید در الگوریتم‌های مسیریابی در نظر

[1] Infrastructure
[2] Infrastructure Based Network
[3] Ad Hoc Network
[4] Topology
[5] Routing Protocols
[6] Multi-Hop
[7] Access Point
[8] Retransmission

گرفته شود، سربار کنترلی قابل قبول می‌باشد. بسیاری از روش‌های موجود در تحرک‌های بالا علاوه بر این که دچار اختلال عملکرد می‌گردند، سربار کنترلی زیادی نیز به شبکه تحمیل می‌کنند [۲].

۳. الگوریتم‌های مسیریابی مبتنی بر شبکه‌های عصبی

الگوریتم‌های مسیریابی مطالعه شده در شبکه‌های بی‌سیم اقتضایی مبتنی بر شبکه عصبی را می‌توان بر اساس رویکرد انتخاب مسیر یا ترافیک مسیرها به سه دسته کلی تقسیم نمود:

۱- مسیریابی مبتنی بر پیشگویی ترافیک در مسیر [۶-۴]
۲- مسیریابی مبتنی بر محاسبه کوتاه‌ترین مسیر [۷]
۳- مسیریابی مبتنی بر یافتن مسیریابی بهینه [۱۰-۸]

در یک تحقیق، توسعه الگوریتم مسیریابی جدید برای MANET که الگوریتم مسیریابی NEURon (NEURAL) نامیده شده گزارش گردیده است [۴]. در این تحقیق، مقایسه روش پیشنهادی با کارایی مسیریابی منبع متحرک (DSR[1]) و مسیریابی بردار- فاصله مبتنی بر تقاضای اقتضایی (AODV[2]) انجام گرفته است و الگوریتم مسیریابی چند مسیره مبتنی بر پیشگویی مسیر (MRATP[3]) در WMNها پیشنهاد شده که مشتمل بر ماژول‌های درخت به انضمام الگوریتم کشف مسیرهای چندگانه، مکانیسم‌های کشف ازدحام مبتنی بر روی شبکه عصبی موجک و الگوریتم تراز بار از طریق چند مسیر است.

این الگوریتم برای تضمین QOS انتها به انتها در WMN پیشنهاد شده است. نتایج حاصل از شبیه‌سازی در تحقیق نشان داد که این الگوریتم ویژگی‌هایی چون انعطاف‌پذیری، انطباق‌پذیری و قدرتمندی بهتری در صورت شکسته شدن مسیرهای موجود یا شبکه‌های پر ازدحام را دارا می‌باشد. گذشته از این، در مقایسه با الگوریتم‌های رایج، MRATP مقیاس‌پذیری بالاتر و بهبودی نرخ موفقیت بالاتر و تأخیر انتها به انتهای پایین‌تری داشته است. دیگر نتیجه حاصل از شبیه‌سازی معین کرد که الگوریتم پیشنهادی، درجه دقت و صحت پیشگویی بالاتری داشته است. این مدل می‌تواند ترافیک بلادرنگ از WMNها را پیشگویی کند. در نتیجه‌گیری تحقیق، عامل هوشمند سیار NEURAL، در مقایسه با DSR و AODV، نرخ تحویل بسته و تأخیر انتها به انتها که به طور قابل ملاحظه بهبود یافته را نشان داد.

در تحقیقت دیگری رویکرد سوئیچ چند مسیره مبتنی بر پیشگویی ترافیک مطابق با برخی مشخصات از شبکه‌های گسترده پیشنهاد شده است [۵]. در تحقیق یاد شده، علاوه بر مسیریابی چند جهته به انضمام روشی از برقراری مسیر، جستجوی قابل اطمینان و مسیرکارآمد و روش ارزیابی مصرف انرژی و کارایی مسیر، بر روی این چنین موارد وجود داشته که مشخص می‌کند نوعی از پیشگویی ترافیک مبتنی بر سوئیچ چند مسیره بین مسیر اولیه و مسیر پشتیبان امکان‌پذیر است. برای این منظور، مدل پیشگویی ترافیک مبتنی بر WNN شبکه عصبی موجک MWNN که از سه لایه: لایه ورودی، لایه پنهان و لایه خروجی ساخته شده دقت بسیار بالاتری برای هدایت پیشگویی دوره کوتاه مبتنی بر مشاهده داده را داشته است.

در یک تحقیق، روشی مبتنی بر شبکه‌های عصبی برای مدل سیستم‌های صف در شبکه مطرح گردیده است [۶]. شبکه عصبی توزیع شده بر روی گره‌ها و تربیت شده برای تخمین زمان انتقال، نرخ گم شدن و پارامتر مشخص کننده تغییرپذیری و جریان واگذاری صف، تشریح آماری معین و حداقل ترافیک وارد شونده به کار رفته است. برای هر سامانه، صف در شبکه هم پیوند شده با شبکه عصبی است. این شبکه عصبی برای پیشگویی عدد میانگین بسته‌های چشم به‌راه، نرخ گم شدن بسته و ضریب اختلاف زمان به یکدیگر مربوط بودن به بسته آموزش دیده است. در پژوهش یاد شده نتایج عددی نشان می‌دهد که استفاده از شبکه عصبی در مسیریابی مبتنی بر عصبی، توزیع بهتر بار در شبکه را در پی دارد. در واقع پیشگویی‌های حاصل از روش به کار رفته سبب نتایج قابل قبول از کنترل ترافیک، مسیریابی بهینه و همچنین اجتناب از تراکم در شبکه‌ها در ضمن حفظ QOS گردید.

آرایوو و همکارانش [۷] در مقاله خود، شبکه عصبی جدیدی برای حل مسئله کوتاه‌ترین مسیر جهت مسیریابی در شبکه‌ها را پیشنهاد داده‌اند. در راه حل پیشنهادی، معماری جدید، متغیرهای وابسته (DV) نامیده شده که توسعه معماری هاپفیلد بازگشتی یک لایه سنتی است. در مقایسه با نتایج قبلی، الگوریتم پیشنهادی به طور قابل ملاحظه‌ای کاهش، از تأخیر زمان برخوردهای بسته دارد. آزمون‌های متعدد نشان داد که روش پیشنهادی می‌تواند کارایی بهتر و پایداری بسیار خوبی را در دسترس قرار دهد.

در مقاله دیگر، با عنوان کاربرد هوش مصنوعی مسیریابی برای شبکه‌های بی‌سیم کاوشی درکارایی دو نمونه الگوریتم‌های مسیریابی مشهور، انتشار هدایت شده و مسیریابی آگاه از انرژی مسیریابی انجام شده است [۸]. این تحقیق نوآورانه کوششی برای ترغیب کاربرد روش‌های هوش مصنوعی درگره‌های حسگر بی‌سیم بوده است.

در تحقیقی یک شبکه عصبی حسگر- عملگر برای مسیریابی عملیات تخلیه مورد بررسی قرار گرفته است [۹]. شبکه‌های حسگر- عملگر نوعی شبکه بی‌سیم هستند که به طور نمونه برای پایش محیط پیرامونی، ساختمان (منزل/ دفترکار) و وسیله نقلیه خودکار، کنترل فرآیندهای تولید، مانیتورینگ قلب، و دیگر موارد به‌کار می‌روند. نتایج حاصل از این پژوهش نشان داد که استفاده از سامانه حسگر- عملگر شبکه بی‌سیم پیشنهادی با آموزش گره‌ها در یک شبکه عصبی برای یادگیری مشاهدات و به‌کار بردن دانش فرا گرفته برای کشف مسیر بهینه در عملیات تخلیه در موقعیت‌های بحرانی

[1] Dynamic Source Routing
[2] Ad Hoc On-demand Distance Vector
[3] Multiple Routing Algorithm Base on Traffic Prediction

کارآیی مناسبی داشته و می‌تواند سبب هدایت مردم به بیرون از ساختمان از طریق مسیرهای موجود تا زمان دور شدن از ناحیه پر خطر مورد استفاده قرار گیرد.

زارعی و همکاران [۱۰] یک الگوریتم جدید مسیریابی معکوس تقاضا محور برای شبکه‌های اقتضایی سیار مبتنی بر انتخاب بهتر مسیر با شبکه‌های عصبی بازگشتی پیشنهاد داده و آن را ارزیابی نموده‌اند. در الگوریتم پیشنهادی، در فازهای کشف مسیر، بهترین مسیر که دارای پایداری بالا بین مسیرهای دسترس‌پذیر است انتخاب می‌شود. ایده اصلی در استفاده از الگوریتم پیشنهاد شده، تغییر قابلیت زمان سازگاری مسیرهای شکست خورده است. همچنین با آگاهی‌هایی از سازگاری مسیرها، گره مبداء می‌تواند درزمانی که مسیرفعال شکست خورده، بهترین مسیر را در مجموعه‌هایی از مسیرهای دسترس‌پذیر انتخاب کند. روش به‌کار رفته، نسخه بهینه شده از الگوریتم‌های مسیریابی مبتنی بر بردار فاصله (AODV) اقتضایی و R-AODV که در تحقیق یاد شده MRAODV نامیده شده است، می‌باشد. نتایج شبیه‌سازی نشان داده است که هر دو الگوریتم AODV و R-AODV در سرعت کم اجرای خوب داشته اما در سرعت‌های بالا الگوریتم پیشنهادی آنان کارآیی بهتری از خود نشان می‌دهد. همچنین در ارزیابی مشخص ad MRAODV توانسته سربار کمتری نسبت به AODV و R-AODV داشته باشد. نتایج حاصل برای روش پیشنهادی نشان داد که این الگوریتم در مقایسه با دیگر نسخه از الگوریتمAODV برتر است.

۴. معرفی الگوریتم پیشنهادی

در این مقاله، روش جدیدی برای مسیریابی در شبکه‌های اقتضایی سیار ارائه می‌شود که ضمن استفاده از یک شبکه عصبی مصنوعی، بدون ایجاد سربار ترافیکی در شبکه، اقدام به یافتن مسیرهای مناسب بین گره‌های شبکه می‌کند. پس از مقایسه و ارزیابی انواع الگوریتم‌های ذکر شده در بخش ۳، الگوریتم پیشنهادی با نام ZSR[1] بر اساس دسته دوم، یعنی الگوریتم‌های مسیریابی مبتنی بر محاسبه کوتاه‌ترین مسیر پیشنهاد گردیده است.

۴-۱. شبکه عصبی انتخابی

جهت پیاده‌سازی الگوریتم ZSR از شبکه عصبی هاپفیلد به دلیل ویژگی‌های خاص و سنخیت با موضوع تحقیق، استفاده گردیده است. رویکرد هاپفیلد نشان دهنده دیدگاه فیزیکدان‌ها درباره فعالیت گروهی واحدهای پردازشی است که در آن نیازی به هماهنگی نیست. هر واحد به عنوان یک سیستم ابتدایی تعامل پیچیده‌ای با سایر واحدهای گروه دارد. شبکه هاپفیلد یک شبکه عصبی بازگشتی دارای اتصالات سیناپسی است که بستری برای یک تابع لیاپونوف فراهم می‌کند. با شروع از یک حالت پایه، سیستم دائماً تغییر می‌کند تا به یک حالت نهایی که یک کمینه محلی برای تابع لیاپونوف است، برسد. دو شکل متداول از مدل هاپفیلد وجود دارد:

۱- نرون‌های دودویی زمان گسسته:

$$V_j(t+1)=\begin{cases}1, & \sum_k T_{jk}V_k(t)+I_j>0\\ 0, & otherwise\end{cases} \quad (۱)$$

۲- نرون‌های نمره دار[2] یا زمان پیوسته:

$$\frac{dx_j}{dt}=-\frac{x_j}{\tau}+\sum T_{jk}\,g(x_k)+I_j \quad (۲)$$

شبکه‌های هاپفیلد دارای یک تابع لیاپونوف E (تابع انرژی) هستند که به کشف حالت‌های نهایی ممکن کمک می‌کنند. این تابع به صورت یکنوا نزولی و از پایین کراندار است. محاسبات در این سیستم می‌تواند با شروع از یک حالت و حرکت به سمت پایین بر روی منحنی E انجام شود [۱۱].

پدیده حافظه اشتراکی با ایده کنترل سازوکار به وسیله تابع لیاپونوف تطابق دارد. بهینه‌سازی جزء مسائل رایج محاسباتی هستند که به شکل‌های مختلفی ظاهر می‌شوند. بسیاری از مسئله‌های بهینه‌سازی می‌توانند به راحتی در یک شبکه هاپفیلد پیاده‌سازی شوند. برای اینکه مسئله به مجموعه‌ای از متغیرها تبدیل گردد به طوری که کمینه شدن تابع لیاپونوف مربوطه، بهینه‌سازی مورد نظر را نتیجه دهد [۱۱].

الگوریتم ZSR، بر اساس ترکیبی از یک شبکه عصبی مصنوعی و هوش تجمعی عمل می‌کند و با استفاده از بسته‌های داده و بدون نیاز به آگاهی آنها از یکدیگر مسیر مناسب بین گره‌های شبکه را می‌یابد. در این روش بسته‌ها با ایجاد تغییراتی در گره‌های شبکه هنگام عبور به مشخص شدن مسیر بهینه کمک می‌کنند.

۴-۲. الگوریتم مسیریابی با استفاده از بسته‌های داده

فرض کنید G(V,E) یک گراف همبند دارای n = |V| گره باشد. روش ساده مسیریابی برای پیدا کردن کوتاه‌ترین مسیر بین دو گره مبدأ (V_s) و مقصد (V_d) در گراف G مطرح می‌شود. طول مسیر بین دو گره بر اساس تعداد گره‌های مسیر مشخص می‌گردد. هر لبه $e(i,j)\in E$ عضو گراف که گره‌های V_i و V_j را به هم متصل می‌کند دارای یک متغیر $\varphi_{i,j}$ است که بیان کننده ارزش آن لبه است. این متغیر توسط بسته‌های داده که از لبه عبور می‌کنند تغییر داده می‌شود.

بسته‌ای که در گره V_i قرار دارد از متغیر $\varphi_{i,j}$ در گره $V_j \in N_i$ استفاده می‌کند تا احتمال انتخاب آن را به عنوان گره بعدی مسیر محاسبه کند. N_i مجموعه همسایه‌های گره V_i است:

[1] Zahra sotoodeh - Saleh Routing

[2] Graded Neuron

$$p_{i,j} = \begin{cases} \dfrac{\varphi_{i,j}}{\sum_{j\in N_i}\varphi_{i,j}} & \text{if } j \text{ is in } N_i \\ 0 & \text{if } j \text{ is not in } N_i \end{cases} \quad (۳)$$

که در آن، $p_{i,j}$ احتمال استفاده گره V_i از گره V_j است و در شرط زیر صدق می‌کند:

$$\sum_{j\in N_i} p_{i,j} = 1, \quad i \in [N_i] \quad (۴)$$

در حین مسیریابی، بسته‌ها ارزش لبه‌ها را افزایش می‌دهند. در ساده‌ترین حالت، هر بسته هنگام عبور ارزش لبه را به اندازه $\Delta\varphi$ افزایش می‌دهد. این کار در هنگام عبور بسته از لبه e(i,j) به صورت زیر انجام می‌شود:

$$\varphi_{i,j} := \varphi_{i,j} + \Delta_t \quad (۵)$$

ارزش لبه‌ها با گذشت زمان کاهش می‌یابد تا از تراکم زود هنگام ترافیک در یک لبه جلوگیری شود. در ساده‌ترین حالت این کاهش به صورت زیر اتفاق می‌افتد [۱۲]:

$$\varphi_{i,j} := (1-q)\varphi_{i,j} \quad q \in (0,1] \quad (۶)$$

۴-۳. الگوریتم مسیریابی

الگوریتم پیشنهادی ZSR از سه فاز کشف مسیر، نگهداری مسیر، و ترمیم مسیرهای شکسته شده تشکیل شده است:

فاز کشف مسیر: در این فاز، مسیرهای جدید کشف و ثبت می‌شوند. کشف مسیر جدید به یک بسته رفت و یک بسته برگشت نیاز دارد. بسته رفت، مسیر رسیدن از گره مقصد به مبدأ و بسته برگشت مسیر رسیدن از گره مبدأ به مقصد را مشخص می‌کند. بسته رفت، یک بسته کوچک دارای یک شناسه منحصر به فرد است. گره‌ها با استفاده از این شناسه و همچنین گره فرستنده بسته می‌توانند بسته‌های تکراری را شناسایی کنند.

یک بسته رفت، در ابتدا توسط گره مبدأ در شبکه منتشر می‌شود و توسط تمام گره‌های دریافت کننده به گره‌های بعدی انتقال می‌یابد. وقتی یک گره برای اولین بار بسته را دریافت می‌کند، یک ورودی جدید در جدول مسیریابی خود ایجاد می‌نماید. این ورودی دارای سه بخش است که به ترتیب عبارت است از: آدرس مقصد، گره بعدی و ارزش لینک. گره دریافت کننده بسته، اطلاعات مذکور را به این صورت از بسته استخراج می‌کند: آدرس مبدأ بسته به عنوان آدرس مقصد، آدرس گره قبلی به عنوان گره بعدی و تعداد گره‌های طی شده برای رسیدن بسته به اینجا به عنوان ارزش لینک در نظر گرفته می‌شود. سپس گره بسته را به همسایه‌های خود می‌فرستد. بسته‌های تکراری از روی شناسه تشخیص داده شده و توسط گره دریافت کننده از بین می‌روند. وقتی بسته رفت به گره مقصد می‌رسد، به شکل ویژه‌ای پردازش می‌شود. گره مقصد اطلاعات بسته را استخراج کرده و یک بسته برگشت ساخته و آن را به سمت گره مبدأ می‌فرستد. بسته برگشت وظیفه‌ای مشابه بسته رفت دارد: ایجاد مسیری به گره فرستنده‌اش. وقتی گره مبدأ بسته برگشت را دریافت می‌کند، مسیر تعیین شده است و می‌توان بسته‌های داده را ارسال کرد.

فاز نگهداری مسیر: در این فاز نیاز به ارسال بسته‌های جدیدی نیست. زمانی که بسته‌های رفت و برگشت مسیر را تعیین کردند، بسته‌های داده که در ادامه فرستاده می‌شوند برای نگهداری مسیر کافی هستند. وقتی گره V_i بسته‌ای به مقصد V_D را به همسایه خود V_j می‌فرستد، ارزش لینک را در ورودی (V_D, V_j, φ) در جدول مسیریابی خود به اندازه $\Delta\varphi$ افزایش می‌دهد. با این کار مسیر منتهی به مقصد توسط بسته‌های داده تقویت می‌شود. از سوی دیگر، V_j نیز ارزش لینک را در ورودی (V_S, V_i, φ) در جدول خود به اندازه $\Delta\varphi$ افزایش می‌دهد، یعنی مسیر منتهی به مبدأ نیز تقویت می‌شود. کاهش ارزش لینک‌ها در زمان بر اساس رابطه‌ای صورت می‌گیرد که ضرایب آن با استفاده از یک شبکه عصبی مصنوعی محاسبه می‌شوند و در بخشی جداگانه به آن پرداخته می‌شود.

روش مذکور ممکن است به ایجاد حلقه‌های ناخواسته منتهی شود. برای جلوگیری از این امر، یک روش ساده که در فاز کشف مسیر نیز وجود داشت، استفاده می‌شود. گره‌ها می‌توانند با استفاده از شناسه و آدرس فرستنده بسته‌های تکراری را شناسایی کنند. در صورتی که یک گره بسته تکراری دریافت کند، پرچم[1] خطای تکرار[2] را می‌افرازد و بسته را به گره قبلی باز می‌گرداند. گره قبلی لینک متصل به این گره را غیرفعال می‌کند تا بسته‌های داده دیگر از این مسیر فرستاده نشوند.

فاز رسیدگی به شکست مسیرها: این فاز مسئول ترمیم مسیرهای شکسته شده است که امری رایج در شبکه‌های اقتضایی سیار است، عمدتاً در اثر حرکت گره‌ها روی می‌دهد. مسیر از دست رفته با نرسیدن تأیید دریافت یک بسته مشخص می‌شود. زمانی که یک گره پیام خطای مسیر دریافت می‌کند، با تغییر ارزش لینک به صفر، آن را غیرفعال می‌کند. سپس گره در جدول مسیریابی خود به دنبال یک لینک جایگزین می‌گردد. اگر چنین لینکی پیدا شود بسته از طریق آن فرستاده می‌شود. در غیر این صورت گره، همسایه‌های خود را با خبر می‌کند و از آنها می‌خواهد تا بسته را منتقل کنند. گره‌های دیگر نیز مانند گره اول عمل می‌کنند و با جستجوی جدول مسیریابی خود سعی می‌کنند لینکی به سمت مقصد بیابند یا با اطلاع به همسایگانشان انتقال بسته را به آنها محول می‌کنند. در صورتی که این کار نتیجه ندهد و نهایتاً پیام خطا به گره مبدأ بسته برسد، فاز کشف مسیر برای گره‌های مبدأ و مقصد باید تکرار شود.

[1] Flag
[2] Duplicate Error

۴-۴. ویژگی‌های الگوریتم پیشنهادی

الگوریتم ZSR دارای چند ویژگی مهم است که کارآیی آن را تضمین می‌نماید:

- **عملیات توزیع شده:** در این الگوریتم هر گره جدول مسیریابی خودش را نگهداری می‌کند و خودش مسئول کنترل تغییرات ارزش لینک‌ها است.
- **عدم وجود حلقه:** از آنجایی که گره‌ها شناسه‌های بسته‌ها را به خاطر می‌سپارند، می‌توانند با تشخیص بسته‌های تکراری از ایجاد حلقه در مسیرها جلوگیری کنند.
- **عملیات تقاضا محور:** مسیرها در این الگوریتم براساس ارزش لینک‌ها در گره‌ها ساخته می‌شوند. با گذشت زمان و در صورت عدم عبور بسته‌ها از یک لینک، این ارزش کاهش می‌یابد و نهایتاً به صفر می‌رسد. کشف یک مسیر جدید تنها در صورت تقاضای فرستنده انجام می‌شود.
- **عملیات حالت خواب:** گره‌ها می‌توانند وقتی که ارزش لینک‌هایشان از حدی کمتر شد، به حالت خواب بروند. در این صورت گره‌های دیگر آنها را در نظر نخواهند گرفت.
- **محلی بودن:** جداول مسیریابی و سایر اطلاعات گره‌ها محلی بوده و نیازی به انتقال آنها بین گره‌ها نیست.
- **چند مسیری:** هرگره چند مسیر به هر مقصد خاص نگه می‌دارد. انتخاب مسیر به شرایط محیط و کیفیت لینک‌ها بستگی دارد.
- **حالت خواب:** وقتی یک گره در حالت خواب قرار دارد، تنها بسته‌هایی را که به مقصد خودش فرستاده شده‌اند، پردازش می‌کند تا در مصرف انرژی صرفه‌جویی کند.
- **سربار کم:** سربار الگوریتم پیشنهادی بر خلاف برخی الگوریتم‌های رایج با دلیل عدم تبادل جداول مسیریابی بین گره‌ها قابل توجه نیست. بسته‌های رفت و برگشت اطلاعات چندانی با خود حمل نمی‌کنند و تنها حاوی شناسه و گره‌های طی شده هستند. بیشتر کار نگهداری مسیر توسط خود بسته‌های داده انجام می‌شود و نیازی به ارسال اطلاعات مسیریابی اضافه نیست. الگوریتم پیشنهادی تنها به سرآیند بسته‌های داده نیاز دارد.

۴-۵. راه اندازی جدول مسیریابی

۱- به ازای هر گره k جداول مسیریابی بر اساس یک توزیع یکنواخت مقداردهی اولیه می‌شوند:

$$p_{ji} = \frac{1}{n_k}, \forall i \in N_k \qquad (۷)$$

در بازه‌های زمانی مشخص و به مدت نامحدود:

قدم اول: هر گره s یک بسته رفت $F_{s\to d}$ به سمت یک مقصد تصادفی ارسال می‌کند. زمانی که بسته به یک گره k که مقصد نیست می‌رسد قدم دوم را انجام می‌دهد. چنانچه گره k گره مقصد باشد، به قدم چهارم می‌رود.

قدم دوم: بسته $F_{s\to d}$ گره $S_{s\to d}(k)$ و زمان سپری شده از ارسال خود تا رسیدن به گره k را در پشته خود ذخیره می‌کند. این بسته سپس گره بعدی مسیر خود را به یکی از دو روش زیر انتخاب می‌کند:

الف. به صورت تصادفی بین i گره که همسایه‌های k هستند انتخاب می‌کند. احتمال انتخاب هر گره برابر P_{di} است که در جدول مسیریابی k آمده است. چنانچه گره انتخاب شده قبلاً ملاقات شده باشد:

ب. دوباره اقدام به انتخاب تصادفی گره بعد می‌کند اما این بار برای تمام گره‌های همسایه احتمال یکسانی در نظر گرفته می‌شود. اگر گره انتخابی قبلاً ملاقات شده بود:

قدم سوم: یک حلقه ایجاد شده است. بسته رفت، تمام اطلاعات گره‌های عضو حلقه را از پشته خود پاک می‌کند. مسیر بهینه نباید دارای حلقه باشد. اگر زمان سپری شده در حلقه کمتر از نصف کل زمان سفرش باشد، به قدم دوم باز می‌گردد و در غیر این صورت بسته نابود می‌شود تا از ایجاد حلقه بی‌نهایت جلوگیری شود.

قدم چهارم: بسته رفت، بسته دیگری به نام بسته برگشت $B_{s\to d}$ را تولید می‌کند، پشته خود را به آن منتقل کرده و خود نابود می‌شود. بسته برگشت از طریق همان مسیری که توسط بسته رفت استفاده شده بود به s باز می‌گردد.

مراحل زیر به ازای هر بسته برگشت انجام می‌شود: زمانی که بسته برگشت از یک گره f به گره k در همسایگی آن می‌رسد:

قدم پنجم: بسته برگشت، جدول مسیریابی گره k و لیست سفرها را برای تمام گره‌های 'k که بین k و d قرار دارند (شامل k و d) بر اساس داده موجود در $S_{s\to d}(k')$ به‌روزرسانی می‌کند. برای این کار، احتمال مربوط به مسیر استفاده شده را افزایش داده و ارزش سایر مسیرها را طبق معادله محاسبه شده در شبکه عصبی هاپفیلد کاهش می‌دهد [۱۲].

شکل شبه‌کد الگوریتم مذکور در ذیل آمده است:

```
BEGIN
{
Routing Tables Set-Up: For each node k the routing tables are
initialized with a uniform distribution:

DO always (in parallel)
{
    STEP 1: In regular time intervals, each node s launches a
    F_{s→d} forward packet to a randomly chosen destination d.
    /*when F_{s→d} reaches a node k, ( k ≠ d ), it performs step
    2*/
    DO (in parallel, for each F_{s→d} )
```

{

STEP 2: $F_{s\to d}$ pushes in its stack $S_{s\to d}(k)$ the node k identifier and the time between its launch from node s to its arrival at node k.

$F_{s\to d}$ selects the next node to visit in two possible ways:

(a) It draws between i nodes where each node i is a neighbor of k and has a probability P_{di} (in the routing table of k) to be selected.
IF the node selected in (a) was already visited

(b) It draws again the jumping node, but now with the same probability for all neighbor nodes
IF the selected node was already visited

STEP 3: A cycle is found. $F_{s\to d}$ pops all data of the cycle nodes from its stack. The optimal path must not have any cycles. $F_{s\to d}$ returns to step 2(a) if the time spent in the cycle is less than its half-trip time; else it dies, in order to avoid infinite loops.

END IF

END IF

} WHILE (jumping node ≠ d)

STEP 4: $F_{s\to d}$ generates another packet, called backward packet $B_{s\to d}$.

$F_{s\to d}$ transfers its stack Ss→ d to $B_{s\to d}$ and then dies.

/* $B_{s\to d}$, will return to s, following the same path used by $F_{s\to d}$ */

DO (in parallel, for each $B_{s\to d}$ packet)

{

/*When $B_{s\to d}$ arrives from a node f to a node k (k is a neighbor of f), it performs the step 5*/

STEP 5: $B_{s\to d}$ updates the k routing table and its list of trips, for entries of nodes k' between k and d (inclusive). According to the data carried in $S_{s\to d}(k')$, it increases probabilities associated to the path used and decreases other paths probabilities based on the equation acquired using a Hopfield network.

IF $k \neq s$

$B_{s\to d}$ will leave k and jump to a node given by $S_{s\to d}(k-1)$.

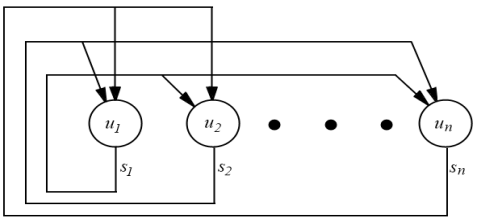
END IF

}WHILE ($k \neq s$)

}

}END

مدل مفهومی شبکه هاپفیلد استفاده شده که در آن تابع ریاضی لیاپونوف به عنوان تابع هزینه استفاده شده است، در شکل (۲) آمده است [۱۲]:



شکل ۲. مدل مفهومی شبکه هاپفیلد استفاده شده [۱۲]

و تابع انرژی کلاسیک این شبکه به صورت زیر است:

$$E = -\left(\frac{1}{2}\right)\left(\sum_i \sum_{i!=j} S_i S_j W_{ij}\right) - \sum_i S_i I_i \qquad (۸)$$

در این تابع انرژی با توجه به آنکه ماتریس وزن‌ها به صورت متقارن بوده ($W_{ji} = W_{ij}$) و نرون‌ها به صورت غیر همزمان به‌روزرسانی می‌شوند، تغییرات ناگهانی محلی که باعث تغییر موضعی مسیر می‌شوند وجود نخواهد داشت که خود باعث افزایش قابلیت اعتماد و اطمینان الگوریتم مسیریابی می‌شود. اما در این رابطه که ماتریس وزن‌ها بر اساس تغییرات بازپس آمده توسط بسته‌های بازگشتی و زمان، بر اساس رابطه آورده شده در قسمت بالا به‌روزرسانی می‌شوند، و S_i خروجی نرون iام و S_j خروجی نرون jام می‌باشد. I_i ورودی خارجی تابع و W_{ij} وزن بین نرون iام و نرون jام است. برای مثال، در صورتی که در جدول مسیریابی موجود در یک نود شبکه تعداد ۴ مسیر وجود داشته باشد، در این مدل پویا از شبکه هاپفیلد، چهار نرون وجود خواهد داشت.

۵. پیاده‌سازی الگوریتم پیشنهادی

الگوریتم ZSR در نرم‌افزار NS2 شبیه‌سازی گردیده و نتایج عددی از آن استخراج گردیده است. حالت عامل و جدول مسیریابی از مواردی هستند که نقش مهمی را در پیاده‌سازی ایفا کرده‌اند [۱۳].

در ابتدا الگوریتم مسیریابی اقتضایی سیار به عنوان یک مسئله یادگیری تقویتی، با شمارش حالت‌ها، عملیات، گذارها و تقویت‌های یک سیستم توصیف گردیده و سپس یک راهبرد یادگیری مطابق با محدودیت‌های شبکه اقتضایی سیار طراحی شده و مسیریابی بین جفت گره‌های داده شده انجام می‌شود. این گره‌ها با حروف S و D، به عنوان مبداء و مقصد، برچسب‌گذاری می‌شوند. یک تابع تقویتی[1] مبتنی بر هزینه یک عمل، بر حسب استفاده آن از منابع شبکه تعریف شده است که هدف آن تحویل هر بسته با حداقل هزینه ممکن است. یک بسته در داخل یک گره مبدا S قرار داده شده و فرآیند مسیریابی بسته آغاز گردیده است. این فرآیند وقتی به پایان می‌رسد که بسته در میانه راه سقوط کرده[2] یا توسط هرکدام از گره‌های موجود دریافت[3] شده باشد. به هرگره یک پارامتر زمان زندگی[4] نسبت داده می‌شود که با انتقال بسته کاهش می‌یابد. اگر مقدار این پارامتر برابر صفر باشد، بسته دور انداخته می‌شود.

۵-۱. مقایسه عملکرد الگوریتم پیشنهادی با سایر الگوریتم‌های رایج

برای انجام مقایسه عملکرد، با شبیه‌سازی ۳ سناریوی شبکه، الگوریتم پیشنهادی ZSR و الگوریتم‌هایAODV، DSR به ترتیب اجرا

[1] Reinforcement Function
[2] Drop
[3] Deliver
[4] Time to Live (TTL)

می‌شوند. مشخصات سناریوهای شبیه‌سازی شده به صورت جدول (۱) است.

۵-۲. شبیه‌سازی مدل حرکتی

برای شبیه‌سازی مدل حرکتی گره‌ها در شبکه، از نرم‌افزار BonnMotion استفاده شده است [۱۴]. مدل‌های متحرک مختلفی توسط این برنامه پشتیبانی می‌شوند که برخی از آنها عبارتند از:

- مدل Random Way Point
- مدل Random Walk
- مدل گاوس- مارکوف
- مدل Manhattan Grid

جدول ۱. مشخصات سناریوهای شبیه‌سازی شده

نوع پارامتر / متغیر	مقدار
ابعاد ناحیه شبیه‌سازی	۳۰۰ در ۱۵۰۰ متر
تعداد گره‌های موجود	۵۰
تعداد گره‌های در حرکت	۱۰
سرعت حرکت گره‌ها	صفر تا ۲۰ متر بر ثانیه
نوع ترافیک	CBR[1]
مدت زمان شبیه‌سازی	۱۵ دقیقه
اندازه بسته‌ها	۵۱۲ بیت (۶۴ بایت)
نرخ ارسال بسته‌ها	۴ بسته در ثانیه
پروتکل لایه تبادل داده	IEEE802.11
نوع صف در لایه تبادل داده	DropTail / PriQueue
ظرفیت صف‌ها	۵۰ بسته

چون سعی بر این است که مسیریابی پایدار به منظور کاهش سربار مسیریابی و نیز انتخاب مسیرهای پایدار در شبکه‌های سیار اقتضایی و نیز انتقال بر اساس پارامترهای حرکتی گره‌ها در شبکه صورت گیرد، یکی از روش‌ها برای انتخاب مسیرهای پایدار در یک شبکه سیار اقتضایی، استفاده از روش سیاست تقسیم‌بندی همسایه‌ها بر اساس فاصله برای اولویت‌بندی همسایه‌های یک گره است. بدین دلیل برای پیاده‌سازی این روش از مدل حرکتی Random Way point استفاده شده است. این مدل حرکتی شامل تعدادی زمان توقف قبل از انجام تغییرات در سرعت و جهت می‌باشد. هر گره سیار حرکت خود را با ماندن در یک مکان در یک دوره زمانی مشخص به نام زمان توقف (Pause Time) شروع می‌کند. هنگامی که این دوره زمانی تمام شود، گره سیار یک جهت تصادفی را در محیط شبیه‌سازی و نیز یک سرعت تصادفی را از بازه زمانی (minspeed, maxspeed) انتخاب می‌کند. سپس گره به سمت مقصد جدید با سرعت انتخاب شده حرکت می‌کند. زمانی که گره به مقصد رسید دوباره برای یک مدت زمانی به اندازه زمان توقف، ساکن می‌ماند و سپس دوباره عملیات قبلی را تکرار می‌نماید. یکی از کارها برای ساده کردن این مدل این است که زمان توقف در این مدل صفر در نظر گرفته شود.

۵-۳. ایجاد ترافیک مصنوعی

برای ایجاد ترافیک مصنوعی در سطح شبکه از منابع ترافیکی CBR در NS-2 استفاده شده است. منابع CBR بسته‌هایی با اندازه مشخص را در فواصل زمانی مشخص و ثابت به گره‌های مشخصی ارسال می‌کنند. در شبیه‌سازی انجام شده برای مقایسه الگوریتم‌ها این ترافیک ثابت به صورت جدول (۲) تولید می‌شود.

جدول ۲. ترافیک ثابت

زمان (s)	گره مبدأ	گره مقصد	نرخ ارسال (bps)	تعداد بسته
۸/۹۱۱۴	۰	۰	۱۶,۳۸۴	۱,۰۰۰,۰۰۰
۶/۱۶۸۳	۱	۴	۳۵,۲۴۰	۱۰,۰۰۰
۵/۹۴۰۶	۲	۴	۸۳۰	۱,۰۰۰,۰۰۰
۷/۱۵۳۱	۳	۹	۸,۱۹۲	۵۰,۰۰۰
۴/۷۲۰۴	۴	۳۲	۴,۰۹۶	۱۰,۰۰۰
۸/۵۴۲۶	۵	۸	۱۶,۳۸۴	۵۰۰,۰۰۰
۲/۹۹۵۳	۶	۷	۱۶,۳۸۴	۱,۰۰۰,۰۰۰
۷/۹۶۶۲	۷	۲۴	۲۴,۵۷۶	۱,۰۰۰,۰۰۰
۹/۶۵۵۹	۸	۰	۳۲,۷۶۸	۵,۰۰۰
۶/۵۸۹۷	۹	۱۱	۱۶,۳۸۴	۲,۰۰۰,۰۰۰

۵-۴. معیارهای شبیه‌سازی

معیارهای زیر در هرگونه شبیه‌سازی مد نظر قرار گرفته‌اند [۱۵]:

- **بار عرضه شده:** مقدار داده‌ای که مجموعاً توسط تمام مشتری‌ها فرستاده می‌شود. این مقدار توسط رابطه زیر محاسبه می‌شود: اندازه بسته x بسته بر ثانیه x تعداد مشتری‌ها.
- **نسبت دریافت:** کسری از بسته‌های فرستاده شده توسط گره‌های مبداء، در مقایسه با بسته‌هایی که در گره‌های مقصد دریافت شده‌اند.
- **گذردهی**[2]: مقدار داده‌ای که در سرویس‌گیرنده دریافت شده است.
- **انتقالات هر بسته فرستاده شده**[3]: تعداد انتقالاتی که برای هر هر بسته فرستاده شده توسط مشتری تولید می‌شود، چه آن بسته دریافت شده باشد و چه دریافت نشده باشد.
- **تأخیر سراسری**[4]: برای بسته‌های دریافت شده، زمان میانگین بین زمانی که بسته در منبع انتقال ساخته می‌شود و زمانی که توسط مقصد دریافت می‌شود، درنظر گرفته می‌شود.

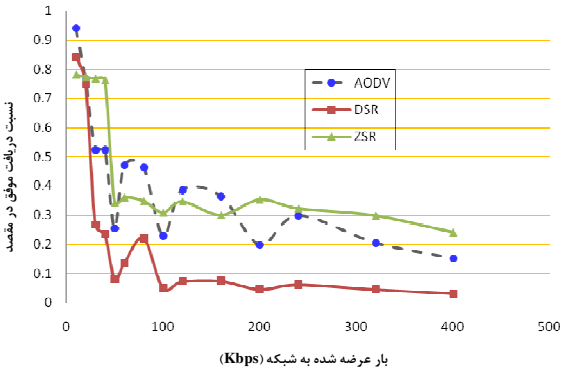
[1] Constant Bit Rate

[2] Throughput
[3] Transmissions Per Packet Sent
[4] End-to-End Latency

۶. نتایج و بحث

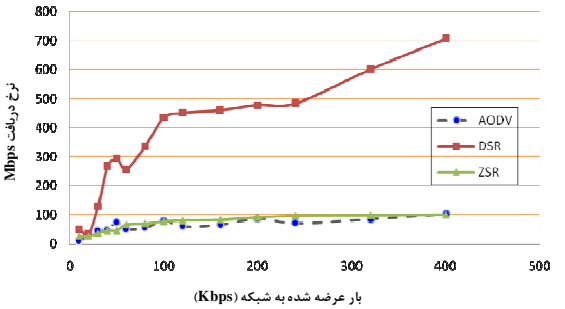
در الگوریتم‌های ADOV و DSR، معیار تعداد انتقالات، شامل بسته‌های مسیریابی است و تعداد بسته‌های مسیریابی جدا از بسته‌های داده شمارش نمی‌شوند.

در شکل (۳) کارایی الگوریتم پیشنهادی ZSR به همراه AODV و DSR برای بسته‌های ۶۴ بایتی و تعداد مشتری متغیر نشان داده شده است. در هر آزمایش، با افزایش تعداد مشتری‌ها، رقابت[1] برای دسترسی به کانال رادیویی نیز بیشتر می‌شود. با این حال کارآیی الگوریتم پیشنهادی نسبت به دو الگوریتم رایج دیگر بیشتر است.



شکل ۳. نمودار کارایی ۳ الگوریتم برای بسته‌های ۶۴ بایتی و تعداد مشتری متغیر

شکل (۴) نرخ دریافت و شکل (۵) نرخ ارسال بسته‌ها برای الگوریتم پیشنهادی با افزایش بار عرضه شده به شبکه را نشان می‌دهد. همان‌طور که رقابت در شبکه افزایش می‌یابد، هر بسته نیاز به تعداد بیشتری انتقال برای یک دریافت دارد. از آنجایی که الگوریتم پیشنهادی، بسته‌ها را پس از انتقال ناموفق دوباره منتقل می‌کند، این نرخ دریافت، افزایش یافته و انباشتگی در شبکه را افزایش می‌دهد.

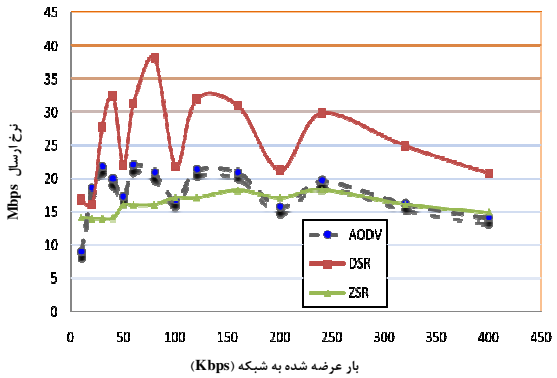


شکل ۴. نمودار نرخ دریافت بسته‌ها با افزایش بار عرضه شده به شبکه

مزیت روش پیشنهادی ZSR بر AODV و DSR با افزایش رقابت در شبکه مشخص می‌شود. با توجه به اینکه کارایی AODV و DSR به طور قابل توجهی با بار عرضه شده در شبکه افزایش می‌یابد، روش ZSR، سطح قابل قبول از کارایی را ارائه می‌دهد.

[1] Competition

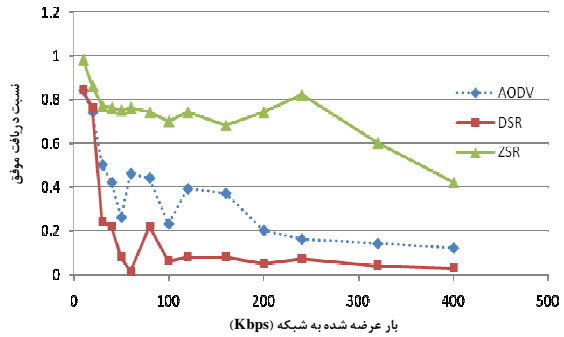
افزایش کارایی در AODV و DSR با افزایش بارگذاری، توسط این واقعیت که این الگوریتم‌ها یک انتقال ناموفق را به عنوان یک ارتباط منقطع تفسیر می‌کنند، قابل توجیه است. راه‌اندازی مکانیزم‌های به‌روز کردن مسیر به ارسال تعداد زیادی بسته در شبکه نیاز دارد که در انباشتگی شبکه سهیم هستند.



شکل ۵. نمودار نرخ ارسال بسته‌ها با افزایش بار عرضه شده به شبکه

کارایی روش پیشنهادی ZSR، با افزایش بار عرضه‌شده با مقدار بیشتر از ۱۵۰ Kbps به مرور کاهش می‌یابد. مطابق شکل (۶)، کاهش در کارایی ارسال، به ازای مقدار ۱۵۰Kbps با افزایش نرخ دریافت بسته همراه می‌شود. با گذشتن بار عرضه شده از مقدار ۱۵۰ Kbps تعداد ارسال‌های لازم برای دریافت موفق هر بسته نیز افزایش پیدا می‌کند.

مطابق شکل (۷)، ابتدا افزایش و سپس کاهش در کارایی ارسال، به ازای مقدار ۱۵۰Kbps با افزایش هزینه دریافت بسته همراه می‌شود. با گذشتن بار عرضه شده از مقدار ۱۵۰Kbps تعداد ارسال‌های لازم برای دریافت موفق هر بسته نیز افزایش پیدا می‌کند.

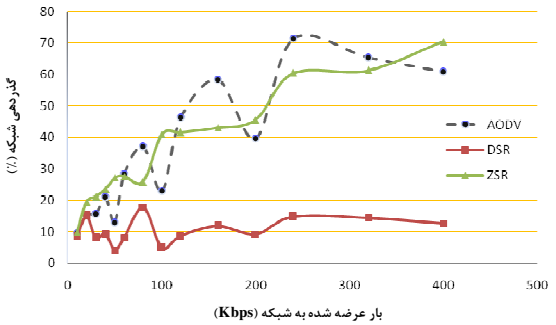


شکل ۶. نمودار نسبت تحویل نهایی ترافیک به ترافیک ورودی ارائه شده

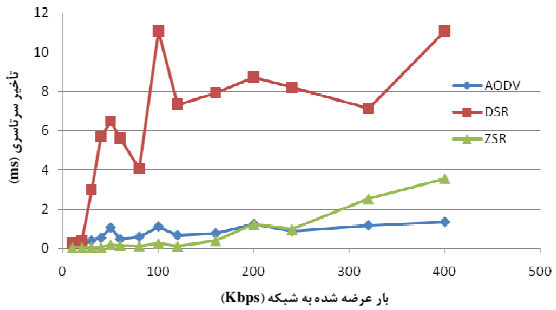
شکل (۸) نمایش می‌دهد که چگونه تأخیر سراسری با افزایش بار شبکه افزایش می‌یابد. کارایی روش پیشنهادی ارائه شده، با افزایش بار عرضه‌شده با مقدار بیشتر از ۱۵۰Kbps به مرور کاهش می‌یابد.

افزایش در تأخیر سراسری بدین صورت قابل توجیه است که در یک انتقال ناموفق، بسته بایستی دوباره فرستاده شود و باید به پشت

صف واسط برود. صف واسط ممکن است شامل تعداد زیادی بسته باشد، بنابراین یک بسته که دوباره فرستاده می‌شود می‌تواند تأخیر چشمگیری در ارسال قبل از دریافت نهایی داشته باشد.



شکل ۷. نمودار گذردهی نسبت به افزایش بار عرضه شده به شبکه با بسته‌های ۵۱۲ بیتی



شکل ۸. نمودار افزایش تأخیر سراسری با افزایش بار در شبکه

۷. نتیجه‌گیری

در این مقاله یک الگوریتم مسیریابی کوتاه‌ترین مسیر در شبکه‌های اقتضایی سیار مبتنی بر شبکه‌های عصبی مصنوعی پیشنهاد و روش جدیدی برای مسیریابی در شبکه‌های اقتضایی سیار ارائه شد که ضمن استفاده از یک شبکه عصبی مصنوعی، بدون ایجاد سربار ترافیکی در شبکه، اقدام به یافتن مسیرهای مناسب بین گره‌های شبکه می‌نماید. الگوریتم پیشنهادی، بر اساس ترکیبی از یک شبکه عصبی مصنوعی و هوش تجمعی عمل می‌کند و با استفاده از بسته‌های داده و بدون نیاز به آگاهی آنها از یکدیگر مسیر مناسب بین گره‌های شبکه را می‌یابد. روش پیشنهادی با استفاده از شبیه‌سازی ترافیک با روش‌های شناخته شده قبلی DSR و AODV در یک محیط متشکل از ۵۰ گره پیاده‌سازی شد و نتایج مقایسه گردید. نتایج حاصل از شبیه‌سازی و نمودارهای به دست آمده بر اساس معیارهای مورد نظر یعنی گذردهی و کارایی، نشان دهنده کارایی بهتر و نیز بهبودی در سرعتِ یافتن الگوریتم مسیریابی کوتاه‌ترین مسیر حرکتی را نشان می‌دهد.

۸. مراجع

[1] Moghim, N. “Study and Comparison of Different Routing Algorithms in Ad Hoc Networks and Improving AODV”; Postgraduate Seminar in Elect. Eng., Dept. of Elect. Eng., Isfahan Univ. of Tech., 2003 (In Persian).

[2] Azarmi, M. “Improving Routing Algorithms Based on Node Location in Ad Hoc Networks”; Master Thesis, Dept. of Computer Eng., Amir Kabir Univ. of Tech., 2006 (In Persian).

[3] IEEE Standard 802.11 “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”; August 1999.

[4] Vicente, E.; Mujica, V. E.; Sisalem, D.; Popescu-Zeletin, R. “NEURAL: A Self-Organizing Routing Algorithm for Ad Hoc Networks”; Proc. of Third Int. Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, pp. 259-266, April 2005.

[5] LI., Z.; Wang, R. “A Multipath Routing Algorithm Based on Traffic Prediction in Wireless Mesh Networks”; Communications and Network 2009, 1, 82-90.

[6] Feng, G.; Douligers, C. “A Neural Network Method for Minimum Delay Routing in Packet-Switched Networks”; Computer Communications 2001, 24, 933–941.

[7] Araújo, F.; Ribeiro, B.; Rodrigues, L. “A Neural Network for Shortest Path Computation”; Technical Reports, Campo Grande, 1700 Lisboa, Portugal, April 2000.

[8] Barbancho, J.; León, C.; Molina, F. J.; Barbancho, A. “Using Artificial Intelligence in Routing Schemes for Wireless Networks”; Computer Communications 2007, 30, 2802–2811.

[9] Jankowska, A.; Schut, M.; Schut, F. “A Wireless Actuator-Sensor Neural Network for Evacuation Routing”; Proc. of IEEE SENSORCOMM, Athens, Greece, 2009.

[10] Zarei, M.; Faez, K.; Alipour, H., Zarei, B. “A New on Demand Protocol Based on Recurrent Neural Network in Mobile Ad Hoc Networks”; 16th Telecommunications Forum TELFOR 2008, Serbia, Belgrade, Nov. 2008.

[11] Honfield, J. J. “Neural Networks and physical Systems with Emergent Collectivc Computational Abilities”; In Proc. National Academy Sci., USA, 1982, 79, 2554-2558.

[12] Feng, G. “A Neural Network Method for Minimum Delay Routing in Packet-Switched Networks”; Computer Communications 2001, 24, 933-944.

[13] Ros, F. J.; Ruiz, P. M. “Implementing a New Manet Unicast Routing Protocol in NS2”; Tech. Report, Dept. of Inf. and Communications Eng., Univ. of Murcia, Spain, 2004.

[14] Aschenbruck, N.; Ernst, R.; Gerhards-Padilla, E.; Schwamborn, M. “BonnMotion-A Mobility Scenario Generation and Analysis Tool”; In Proc. of SIMUTools, Torremolinos, Spain, 2010.

[15] Baran, B.; Sosa, R. “AntNet Routing Algorithm for Data Networks Based on Mobile Agents”; Intelligencia Artificial, Revista Beroamericana de Intelligencia Artificial, 12, 75-84, 2001.