

پروتکل مسیریابی بهبود یافته HOPNET با استفاده از الگوریتم کلونی زنبور عسل

(Bee-HOPNET)

مصطفی حبیبی دهشیخی،¹ کارشناسی ارشد علوم کامپیوتر-هوش مصنوعی، دانشگاه باهنر کرمان، کرمان، ایران

مرجان کوچکی رفسنجانی²، دانشیار، دانشکده ریاضی و علوم کامپیوتر، دانشگاه باهنر کرمان، کرمان، ایران

چکیده

شبکه ویژه سیار (MANET) نوعی شبکه بی‌سیم می‌باشد که از مجموعه گره‌های سیار تشکیل شده است. در این نوع شبکه‌ها گره‌ها به صورت آزادانه حرکت می‌کند و توپولوژی شبکه دائماً در حال تغییر است. یکی از رویکردهای موفق این زمینه، استفاده از هوش مصنوعی در مسیریابی به منظور بهبود پروتکل‌های مسیریابی می‌باشد. در زمینه هوش مصنوعی، شاخه‌ای به نام "هوش جمعی" به وجود آمده که از رفتار کلونی جانوران در طبیعت الهام گرفته شده است. به دلیل آنکه خصوصیات گره‌های یک شبکه سیار به رفتار کلونی‌ها (مورچه‌ها، پرندگان، زنبورها، ماهی‌ها و...) شباهت دارند، کلونی‌ها توانسته‌اند کمک بزرگی در بهبود پروتکل‌های مسیریابی شبکه‌های ویژه سیار نمایند. در این مقاله نیز با استفاده از الگوریتم کلونی زنبور عسل سعی در بهبود پروتکل HOPNET شده است. این بهبود در دو قسمت از پروتکل اعمال شده است: ساختار عامل مسیریاب مورچه زنبور¹ و کشف مسیر درون ناحیه‌ای. از نتایج شبیه‌سازی مشاهده شد که این روش توانسته تأثیر قابل مشاهده‌ای در پارامترهای ارزیابی عملکرد شبکه از جمله تأخیر²، سربار مسیریابی³، نرخ تحویل بسته⁴ و میزان مصرف انرژی⁵ در گره داشته باشد.

کلید واژه‌ها:

پروتکل مسیریابی HOPNET⁶، کلونی زنبور عسل⁷، کلونی مورچه‌گان⁸، مسیریابی⁹، شبکه‌های ویژه سیار (MANET)¹⁰.

1. مقدمه

شبکه‌های ویژه سیار (MANET)¹¹ گروهی از گره‌های سیار هستند که بدون ساختار از پیش تعیین شده و یا مدیریت مرکزی¹² با هم در ارتباطند. این شبکه‌ها به صورت موقت شکل می‌گیرند و می‌توانند هر زمان در هر مکان بدون نیاز به ساختار قبلی تشکیل شوند و همین امر باعث استفاده گسترده از آن در تمام کاربردها شده است. شبکه‌های ویژه سیار پویایی¹³ و تحرک¹⁴ بالایی دارند به همین علت علاوه بر چالش‌ها و مسائل دیگر موجود در این زمینه، مسیریابی از اهمیت ویژه‌ای برخوردار است. توپولوژی متغیر¹⁵ و پویایی آن، ارائه‌ی یک پروتکل مؤثر و کارآمد را برای این‌گونه شبکه‌ها مشکل ساخته است [1]. از متداول‌ترین طبقه‌بندی این شبکه‌ها، سه دسته‌ی پیش‌کنشی¹⁶ (مبتنی بر جدول)، واکنشی¹⁷ (مبتنی بر تقاضا) و ترکیبی¹⁸ را می‌توان نام برد. این تحقیق بیشتر بر دسته‌ی سوم که از ترکیب دو دسته‌ی دیگر به وجود آمده متمرکز شده است. در بررسی‌های

¹AntBee routing agent

²Delay

³Routing overhead

⁴Packet delivery ratio

⁵Energy consumption

⁶Hybrid ant colony optimization routing algorithm (HOPNET)

⁷Bee Colony

⁸Ant colony

⁹Routing

¹⁰ Mobile Adhoc network (MANET)

¹¹ Mobile Ad Hoc Network (MANET)

¹² Centralized management

¹³ Dynamic

¹⁴ Mobility

¹⁵ Changing topology

¹⁶ Proactive

¹⁷ Reactive

¹⁸ Hybrid

انجام شده، الگوریتم‌های برگرفته شده از طبیعت¹ (هوش جمعی)² مانند الگوریتم بهینه‌سازی مورچه‌ها³ و زنبور عسل⁴ تکنیک‌های مؤثری برای پیشرفت و ارتقاء پروتکل‌های مسیریابی در شبکه‌های ویژه سیار را ارائه کرده‌اند [2].

2. پیشینه تحقیق

در این بخش به بیان الگوریتم‌های بهبود یافته با استفاده از رویکرد هوش جمعی پرداخته شده، که شامل الگوریتم‌های الهام گرفته از مورچه‌ها و کلونی زنبور عسل پرداخته شده است.

شوندروود⁵ و همکارانش [13] در الگوریتم⁶ ABC شمایی برای مسیریابی در شبکه‌های تلفن بوده را ارائه کرده‌اند. کاملاً پیش‌واکنشی بوده و با طبیعت شبکه‌های ویژه ی سیار، کاملاً مطابقت دارد. زیرا از ویژگی‌های آن می‌توان به بدون مرکز بودن، مقاومت در برابر از بین رفتن نودها، ایجاد تعادل در سربار شبکه و قابلیت انطباق با پویایی اشاره نمود. هر نود دارای جدولی است که ستون‌های آن مقصدهای در دسترس و سطرها همسایگان آن نود را تشکیل داده‌اند. مقادیر وارد شده در هر مدخل⁷ معادل مقدار فرومون⁸ آن نودها می‌باشد. عامل‌ها⁹ مورچه‌های کاشف هستند که به صورت تصادفی تولید شده و اطلاعات لازم از شبکه را جمع‌آوری می‌کنند.

الگوریتم ANTNET [12] در سال 1999 در مسیریابی شبکه‌ی مخابرات استفاده شده است و پیش‌کنشی بوده است. در این الگوریتم هر نود جدول مسیریابی دارد که با احتمالات استفاده رسیدن به یک مقصد از طریق یکی از همسایگان کامل شده است.

در هر فاز کشف مسیر، مورچه‌ها جداول مسیریابی را در صورت یافتن مسیر جدید به روزرسانی می‌کنند و احتمالات معادل آن را در جداول متناظر وارد می‌کنند. در این الگوریتم از دو عامل مورچه‌های جلو رونده و عقب رونده¹⁰ استفاده شده است.

ARA¹¹ از جمله الگوریتم‌های واکنشی بوده [14] که در سال 2002 ارائه و تا سال 2011 بهبود و ارتقا آن ادامه داشته است. همانند الگوریتم AODV مورچه‌ها را برحسب نیاز در شبکه پخش می‌کند و مقادیر اولیه اختصاص یافته به فرومون را در طول مسیریابی به روز رسانی می‌کند. همانند دیگر الگوریتم‌های واکنشی دارای سه فاز کشف مسیر، نگهداری مسیر و مدیریت خطا می‌باشد. در این الگوریتم نیز مورچه‌های جلو رونده و عقب رونده انجام هر فاز را بر عهده دارند.

در سال 2011 کومار و سینگ [15] الگوریتم¹² PERA را بهبود داده‌اند. در این الگوریتم جداول با اختصاص توزیع احتمال برای هر نود همسایه تکمیل می‌شود. از این توزیع احتمال برای انتخاب آن و تعیین احتمال رسیدن به مقصد استفاده می‌شود. البته به دلیل پخش مورچه‌ها در کل شبکه سربار زیادی را تولید می‌نماید. در واقع مسیر با احتمال بالاتر برای ارسال و دیگر مسیرها به عنوان جایگزین تعیین می‌شوند.

الگوریتم¹³ ADSR [16] از کلونی مورچه‌ها به منظور بهبود فاکتورهای QOS در الگوریتم DSR استفاده شده است. این الگوریتم شامل دو فاز کشف و نگهداری مسیر است و سعی در بهبود تاخیر، انرژی و برانگیختگی¹⁴ را دارد. در این روند اطلاعات در حافظه‌های پنهان مورچه‌ها ذخیره شده که مسیر را برای آن مشخص و در صورت یافتن مسیر جدید به روزرسانی می‌کند. عربشاهی¹⁵ و همکارانش الگوریتم¹⁶ ABC [3] را برای مسیریابی در شبکه‌های تلفن ارائه کرده‌اند. کاملاً پیش‌واکنشی بوده و با طبیعت شبکه‌های ویژه سیار، کاملاً مطابقت دارد. زیرا از ویژگی‌های آن می‌توان به بدون مرکز بودن، مقاومت در برابر از بین رفتن گره‌ها، ایجاد تعادل در سربار شبکه و قابلیت انطباق با پویایی اشاره نمود.

ونگ و همکارانش الگوریتم¹⁷ HOPNET [4] که الگوریتمی ترکیبی و براساس کلونی مورچه‌ها بنا شده را ارائه کردند. در این الگوریتم، هر مورچه از یک منطقه به منطقه دیگر می‌رود. این روند نسبت به دیگر الگوریتم‌های ارائه شده برای شبکه‌های عظیم مناسب‌تر می‌باشد. این الگوریتم شامل فاز اکتشاف مسیر محلی به صورت پیش‌کنشی در همسایه‌های یک گره و ایجاد ارتباط به

¹ Nature inspired algorithm

² Swarm intelligent

³ Ant Colony Optimization (ACO)

⁴ Bee

⁵ Schonderwoerd

⁶ Ant Based Control

⁷ Entry

⁸ Pheromone

⁹ Agents

¹⁰ FAnt(forward ant) Bant(backward ant)

¹¹ ANT colony Based Routing Algorithm

¹² Probabilistic Emergent Routing Algorithm

¹³ Ant Dynamic Source Routing

¹⁴ Jitter

¹⁵ Arabshahi

¹⁶ Ant Based Control (ABC)

¹⁷ Hybrid ant colony Optimization routing algorithm (HOPNET)

صورت واکنشی بین آنها می‌باشد. شبکه به مناطق مختلف تقسیم می‌شود که همسایه‌های محلی را نشان می‌دهد. در این تحقیق از این الگوریتم به عنوان الگوریتم پایه استفاده شده است و در بخش بعد به توضیح کامل آن پرداخته شده است. ود و همکارانش الگوریتم BEEAdHoc [5] را که از مسیریابی الهام گرفته شده از طبیعت برای شبکه‌های ویژه سیار است را براساس اصول به دنبال غذا گشتن¹ زنبورهای عسل کار می‌کند. به صورت عمده، از دو نوع عامل استفاده می‌کند: پیشاهنگ² برای اکتشاف مسیر و کاوشگر³ برای انتقال داده. در این پروتکل هر گره دارای یک کندو⁴ می‌باشد. هر کندو شامل سه قسمت ورودی⁵، طبقه‌ی بسته‌بندی⁶ و طبقه‌ی رقص⁷ می‌باشد. یک پیشاهنگ در صورت تمام نشدن زمان زندگی(TTL)⁸ و یا نرسیدن به مقصد دوباره در قسمت ورودی به کل شبکه پخش می‌شود. اطلاعات مربوط به پیشاهنگ و مبدأ آن در یک جدول نگهداری می‌شود.

ژانگ و همکارانش الگوریتم BeeHive [6] را براساس ارتباطات بین زنبورهای واقعی به وسیله‌ی زنبورهای عامل هوشمند طراحی و مدل شده است. این عامل‌ها قابلیت تصمیم‌گیری مسیر برای توپولوژی‌های عظیم و پیچیده را داراست. شبیه‌سازی‌ها و نتایج حاکی از آن است که زنبورهای عامل با تصرف طول باند کمتر، زمان پردازش کمتری نیز در مقایسه با عامل‌های موجود در الگوریتم‌های دیگر را به خود اختصاص می‌دهد.

چاد هاری و همکارانش الگوریتم BEE [7] را با تناظر شبکه‌های کامپیوتری با کلونی زنبور عسل ارائه کرده‌اند. در این الگوریتم زنبورها به دنبال منابع شهد غنی می‌گردند و نیاز به یافتن مسیرها با ظرفیت بالا به منظور انتقال منابع سودمند می‌باشد. در شبکه، بسته‌های کنترلی که به منظور یافتن مسیر بهینه ارسال می‌شود، معادل پیشاهنگ‌ها و بسته‌های داده متناظر کاوشگران است. مسیر یاب درون هر گره، از اطلاعات جمع‌آوری شده تمام گره‌های شبکه که توسط بسته‌های کنترلی جمع‌آوری می‌شود، استفاده می‌کند.

کاناس و همکارانش [8] الگوریتمی به نام AntOR ارائه کرده‌اند که تنظیمات مسیر را با رویکرد واکنشی و فاز نگهداری را با استفاده از متد پیش‌کنشی انجام پذیرفته است. نکته قابل توجه در این الگوریتم استفاده از لینک‌های گسسته و مسیرهای منفصل از گره می‌باشد که علاوه بر فرمون حقیقی و مجازی که در پروسه پخش و کشف مسیر استفاده شده است، فاکتور تعداد گام‌ها نیز به منظور یافتن بهترین مسیر در نظر گرفته شده است.

ردی و همکارانش [9] با استفاده از دیگر الگوریتم‌های بهینه‌سازی به بهبود پروتکل‌های مسیریابی پرداخته است. باسکاران و همکارانش با استفاده از ACOR در بهینه‌سازی الگوریتم AnthHocNet پروتکل بهبود یافته‌ای که از نظر بهروری شبکه و کارایی آن افزایش قابل توجهی حاصل شده است را ارائه کرده‌اند.

باسکار و همکارانش [10] با ارائه الگوریتم جدیدتر در زمینه مسیریابی به افزایش و بهبود فاکتور خاصی در شبکه توجه داشته است. استفاده از منطق فازی و بهرگیری از مسیریابی براساس الگوریتم زنبور عسل منجر به بهینه‌سازی QoS در شبکه‌های سیار ویژه شده است. در ادامه می‌توان به این نکته اشاره نمود که این الگوریتم با استفاده از منطق فازی توانسته با پویایی این‌گونه شبکه‌ها سازگاری یافته و با پوشش دادن فاکتورهای بیشتری در تصمیم‌گیری به منظور یافتن مسیر بهینه، تاثیر قابل توجهی در افزایش کارایی و ثبات مسیر و همچنین کاهش تاخیر را به همراه داشته باشد.

انوار و همکارانش [11] با استفاده از مسیریابی مورچه الگوریتم Aco MRA را ارائه کرده‌اند. این الگوریتم شامل فازهای ساخت و نگهداری مسیر است که اطلاعات هر گره به صورت تناوبی به روزرسانی می‌شود. این الگوریتم با ارسال پیام MRLVE برای تمامی مسیرهای موجود، از سلامت مسیر اطمینان حاصل می‌کند. استفاده از الگوریتم مورچه منجر به بهینه‌سازی فاکتور QoS در این پروتکل شده است. استفاده از شبکه Mesh نیز به هوشمند شدن مسیریابی و افزایش کارایی شبکه کمک بسزایی کرده است.

3. روش پیشنهادی (Bee-HOPNET)

الگوریتم کلونی مورچه برای اولین بار توسط دوریگو⁹ و همکارانش در سال 1991 به عنوان یک راه حل چند عامله¹⁰ برای مسائل مشکل بهینه سازی مثل فروشنده دوره گرد¹¹ با نام سیستم مورچگان، ارائه شد [44]. مورچه‌ها هنگام راه رفتن از خود،

¹ Foraging

² Scout

³ Forager

⁴ Hive

⁵ Entrance

⁶ Packing floor

⁷ Dance floor

⁸ Time To Live (TTL)

Dorigo⁹

Multi Agent¹⁰

TSP :Traveling Sales Person¹¹

ماده شیمیایی فرومون¹ بجای می‌گذارند البته این ماده به سرعت تبخیر می‌شود ولی در همین مدت کوتاه به عنوان رد مورچه بر سطح زمین باقی می‌ماند. یک رفتار پایه‌ای ساده در مورچه‌های وجود دارد: آنها هنگام انتخاب بین دو مسیر بصورت احتمالاتی² مسیری را انتخاب می‌کنند که فرومون بیشتری داشته باشد یا به عبارت دیگر مورچه‌های بیشتری قبلاً از آن عبور کرده باشند. الگوریتم زنبور، در سال 2005 توسط کاراباگا³ به منظور حل برخی مسائل بهینه‌سازی ابداع شد [42]. الگوریتم کلونی زنبورهای مصنوعی⁴ الهام گرفته از رفتار زنبورها در طبیعت می‌باشد. مشابه با کلونی زنبورهای طبیعی، این الگوریتم نیز از سه گروه زنبورهای کارگر، تماشاگر و پیش‌آهنگ تشکیل شده است. در ابتدا مجموعه‌ای از منابع غذایی به طور تصادفی انتخاب می‌شوند. زنبورهای کارگر به منابع مراجعه کرده و میزان شهد آن‌ها را محاسبه می‌کنند. سپس این زنبورها به کندو بازگشته و اطلاعات خود را با دیگر زنبورها (تماشاگران) به اشتراک می‌گذارند. در مرحله دوم بعد از تبادل اطلاعات، هر زنبور کارگر به سمت منبعی می‌رود که قبلاً دیده است و ممکن است براساس اطلاعات دیداری که از محیط می‌گیرد یک منبع جدید در همسایگی منبع قبلی برود و یا منبع جدیدی را انتخاب نماید. در مرحله سوم، تماشاگران با توجه به اطلاعاتی که از زنبورهای کارگر هنگام رقص دریافت کرده‌اند یک محدوده منبع غذایی را بر مبنای شهد آن ترجیح می‌دهند. بعد از رسیدن به محل ممکن است با توجه به اطلاعات دیداری، یک منبع جدید را که در همان اطراف قرار دارد انتخاب کنند. زمانی که یک منبع پایان پذیرد یا ترک شود یک منبع جدید که به‌طور تصادفی توسط پیش‌آهنگان دریافت شده است، جایگزین می‌شود. این چرخه تا برآورده شدن نیازها تکرار خواهد شد.

الگوریتم مسیریابی HOPNET

یک الگوریتم ترکیبی که برای کشف مسیر محلی بین همسایه‌های یک نود از رویکرد پیش‌کنشی و برای ارتباط میان همسایه‌ها به صورت واکنشی عمل می‌کند. این شبکه به مناطقی تقسیم می‌شود که هر منطقه شامل همسایه‌های محلی یک نود است. اندازه مناطق در ابتدا مشخص نیست اما طول شعاع آن‌ها به تعداد گام‌ها بستگی دارد. بنابراین یک منطقه مسیریابی شامل نودهای آن منطقه و تمام نودهای دیگر تا شعاع تعیین شده می‌باشد. یک نود می‌تواند در چندین ناحیه که دارای هم‌پوشانی هستند، قرار گیرد و این نواحی می‌توانند دارای سایزهای مختلفی باشند. نودها را می‌توان به دو دسته داخلی و مرزی دسته‌بندی نمود. نودهای مرزی از نود مرکزی دارای بیشترین فاصله می‌باشند و سایر نودها در فاصله کمتر از شعاع قرار دارند. برای تشکیل یک ناحیه، یک نود و نودهای مرزی نیاز به شناسایی همسایه‌های محلی خود دارند. به منظور شناسایی نودهای همسایه، فرآیند شناسایی با ارسال بسته‌های Hello و دریافت پاسخ‌های آن‌ها برای هر نود صورت می‌پذیرد.

جدول مسیریابی

هر نود دارای دو جدول مسیریابی می‌باشد: جدول مسیریابی درون ناحیه‌ای⁵ و جدول مسیریابی بین ناحیه‌ای⁶. جدول مسیریابی درون ناحیه‌ای: در یک ناحیه نود می‌تواند به صورت پیش‌کنشی اطلاعات لازم مسیریابی برای رسیدن به مقصد داخل منطقه‌ی خود را به دست آورد. در این متد به صورت دوره‌ای هر نود یک سری مورچه‌های جلورونده در منطقه خود می‌فرستد و هر گونه تغییرات در توپولوژی و دیگر فاکتورهای شبکه را به بقیه نودها اعلام می‌کند. هنگامی که یک مورچه جلورونده به مقصد برسد، یک مورچه عقبگرد به سمت مبدا در مسیر کشف شده، ارسال می‌شود. جدول مسیریابی بین ناحیه‌ای: در این جدول مسیرهای یک نود به خارج از منطقه آن ذخیره می‌شود. این جدول مسیریابی به صورت مبتنی بر تقاضا ساخته می‌شود و مسیرهای خارج از منطقه یک نود را نشان می‌دهد. به منظور کشف مسیرهای بین منطقه‌ای از نودهای خارجی (جانبی، مرزی) استفاده می‌شود.

کشف مسیر

در این الگوریتم دو نوع کشف مسیر وجود دارد: داخل ناحیه‌ای و بین ناحیه‌ای. می‌توان شبکه را به صورت گراف $G=(V,E)$ در نظر گرفت که در آن V که تعداد رئوس (نود) و E تعداد یال‌ها (لینک‌ها) می‌باشد. به منظور پیدا کردن مسیر بهینه و کوتاه‌تر از مبدا به مقصد از الگوریتم بهینه‌سازی کلونی مورچه استفاده می‌شود. میزان فرومون با گذشتن مورچه‌ها از آن مسیر به‌روزرسانی می‌شود. مورچه‌ها مقدار فرومون را هنگام رفتن به سمت مقصد و در راه بازگشت به مبدا تغییر می‌دهد.

کشف مسیر درون ناحیه‌ای

این عمل با استفاده از جدول مسیریابی درون ناحیه‌ای انجام می‌شود. یک نود برای کلیه همسایه‌هایش به‌صورت دوره‌ای، به منظور حفظ اطلاعات این جدول، مورچه جلورونده می‌فرستد. در این الگوریتم بین مورچه جلورونده داخل و خارجی تمایز قائل می‌شود.

¹Pheromone

²Statistical

³Karabaga

⁴ABC : Artificial Bee Colony

⁵Intra Zone (Intra RT)

⁶Inter Zone (Inter RT)

زمانی که نود مبدا می‌خواهد بسته‌ای را برای مقصد ارسال کند، اول به ستون جدول مسیریابی درون ناحیه‌ای مراجعه می‌کند تا تشخیص دهد که آیا نود مقصد در همان منطقه قرار دارد یا خیر. اگر مقصد را در جدول مسیریابی درون ناحیه‌ای پیدا کند، آنگاه فرآیند کشف مسیر به پایان می‌رسد. در نظر داشته باشید که اگر مقصد در همان ناحیه باشد، مسیری که در جدول مسیریابی درون ناحیه‌ای به صورت پیش‌کنشی نگهداری می‌شود. در این الگوریتم مورچه بهترین مسیر را برای رسیدن به مقصد انتخاب می‌کند. در صورتی که مسیر در جدول وجود نداشته باشد، مورچه نودی را انتخاب می‌کند که تا کنون ملاقات نشده باشد. مورچه همگی لینک‌های مجاور به نود ملاقات نشده را کشف می‌کند، این فرآیند قبل از انتخاب گام برای نود بعد صورت می‌پذیرد. اگر نود ملاقات نشده‌ای وجود نداشته باشد، مورچه برای رفتن به گام بعد براساس مقدار فرمون نود اقدام می‌کند. استراژی کشف تضمین می‌کند که هیچ لینکی و مسیر بهینه‌ای گم نشده باشد.

کشف مسیر بین ناحیه‌ای¹

از جدول مسیریابی بین ناحیه‌ای زمانی استفاده می‌شود که نود مقصد در ناحیه نباشد و مقدار متناظری برای آن در جدول مسیریابی درون ناحیه‌ای پیدا نشده باشد.

هنگامی که مبدا می‌خواهد بسته داده‌ای را به مقصد ارسال کند که در خارج از ناحیه قرار گرفته باشد، جدول مسیریابی بین ناحیه‌ای را چک می‌کند تا مسیر معتبر که به تازگی توسط مورچه قبلی کشف شده باشد را بیابد. اگر مسیر منقضی نشده باشد، بسته از آن مسیر ارسال می‌شود که اصطلاحاً به آن خود-مستقیم² گفته می‌شود. در غیر این صورت مورچه جلورونده خارجی برای یافتن مسیر ارسال می‌شود.

مورچه جلورونده خارجی در ابتدا به نودهای مرزی فرستاده می‌شود و در صورتی که جدول مسیریابی درون ناحیه‌ای به‌روزرسانی شده باشد، این فرآیند با سرعت و به آسانی انجام می‌شود. در منطقه‌ی مرزی نودها جداول مسیریابی بین ناحیه‌ای و مسیریابی درون ناحیه‌ای را چک می‌کنند تا مشخص شود که نود مقصد در داخل و یا خارج از ناحیه قرار گرفته است. اگر نود مقصد در ناحیه کنونی واقع نباشد و مسیر آن منقضی شده باشد، مورچه‌ها توسط نودهای مرزی آن ناحیه به منطقه بعد فرستاده می‌شوند. این فرآیند تا یافتن نود مقصد ادامه پیدا می‌کند.

نگهداری مسیر

یک مسیر در صورتی که نودهای تشکیل دهنده آن حرکت کنند و یا لینکی در طول آن شکسته شود، از اعتبار ساقط می‌شود. اگر مسیر آسیب دیده یک مسیر درون ناحیه‌ای باشد، پس از یک مدت زمان دوره‌ای با به‌روزرسانی جدول مسیریابی درون ناحیه‌ای ترمیم خواهد شد. اگر یک مسیر بین ناحیه‌ای غیر معتبر باشد نزدیک‌ترین نود به لینک شکسته شده یک رویه‌ی ترمیم محلی اجرا خواهد کرد که این ترمیم به منظور یافتن یک مسیر جایگزین به سمت نود مقصد می‌باشد. در هنگام اجرای فرآیند ترمیم، تمام بسته‌های داده در بافر نود ذخیره خواهند شد. اگر نود بتواند مسیر جایگزین را به مقصد پیدا کند، بسته‌های داده‌ی ذخیره شده در بافر خود را از طریق مسیر جدید پیدا شده به نود مقصد ارسال می‌کند؛ همچنین یک مورچه اطلاع دهنده را به مبدا ارسال خواهد کرد تا آن نود، از تغییر ایجاد شده در مسیر با خبر شود. تمامی نودها در مسیر که مورچه‌ای اطلاع دهنده آن‌ها را ملاقات می‌کنند، جدول مسیریابی بین ناحیه‌ای خود را به‌روزرسانی کرده و مسیرهای غیر معتبر را از آن حذف می‌کند. نود مبدا مسیر شکسته شده را با مقدار مسیر جدید که مورچه اطلاع دهنده اطلاعات مربوط به آن را حمل می‌کند جایگزین خواهد کرد. اگر مسیر جایگزین پیدا نشود، یک مورچه خطا به نود مبدا ارسال خواهد شد. پس از دریافت مورچه خطا اگر نود مبدا همچنان به مسیری برای ارسال بسته‌ها به نود مقصد نیاز داشته باشد، آنگاه یک مورچه جلورونده جدید را به منظور یافتن مسیر جدید به سمت مقصد ارسال خواهد کرد و فرآیند کشف مسیر همانطور که در قسمت قبل ذکر گردید، اجرا خواهد شد.[17]

روش پیشنهادی به بهبود پروتکل مسیریابی HOPNET با استفاده از مزایای الگوریتم‌های کلونی مورچه‌ها و زنبور عسل در مسیریابی درون ناحیه‌ای و پارامترهای مرتبط با آن پرداخته و عملکرد عامل مسیریاب را با الهام از رفتار هوشمند مورچه و زنبور عسل در طبیعت بهینه کرده است.

شبکه مشابه با پروتکل HOPNET ناحیه‌بندی شده که هر ناحیه شامل همسایه‌های یک گره است. اندازه نواحی با توجه به تعداد گام‌ها مشخص می‌شود. یک ناحیه مسیریابی شامل گره‌های آن ناحیه و تمام گره‌های دیگر تا شعاع تعیین شده می‌باشد که الزاماً شعاع نواحی با یکدیگر برابر است و یک گره می‌تواند در چندین ناحیه همپوشانی قرار گیرند. گره‌ها را می‌توان به دو دسته داخلی و مرزی دسته‌بندی نمود. گره‌های مرزی از گره مرکزی دارای بیشترین فاصله می‌باشند و سایر گره‌ها در فاصله کمتر از شعاع قرار دارند. برای تشکیل یک ناحیه، گره‌ها نیاز به شناسایی همسایه‌های خود دارند که این فرآیند با ارسال بسته‌های "سلام" و دریافت پاسخ‌های آن‌ها صورت می‌پذیرد. روش پیشنهادی یک پروتکل ترکیبی است که برای کشف مسیر بین همسایه‌های یک گره از رویکرد مسیریابی مورچه زنبور (در بخش بعدی به آن پرداخته خواهد شد) و برای ارتباط بین ناحیه‌ای به صورت واکنشی عمل می‌کند. در ادامه جزئیات مرتبط با روش پیشنهادی شامل ساختار عامل مورچه زنبور، جداول مسیریابی و ساختار جدید پیشنهادی و ارائه فاز کشف مسیر بهبود یافته بیان شده است.

ساختار عامل‌های مسیریاب

در روش پیشنهادی از دو نوع عامل مسیریاب استفاده شده: مورچه و عامل مورچه زنبور. از مورچه برای فرآیند مسیریابی بین ناحیه‌ای و از عامل مورچه زنبور به منظور مسیریابی درون ناحیه‌ای استفاده می‌شود. عامل مورچه دارای ساختار مشابه مورچه استفاده شده در پروتکل HOPNET می‌باشد اما ساختار جدید ارائه شده برای عامل مورچه زنبور به صورت زیر می‌باشد. ساختمان داده هر عامل مورچه زنبور از 7 فیلد تشکیل شده است:

- **مبدأ:** گره‌های مرکزی هر ناحیه که الگوریتم HOPNET آن نواحی را تشکیل داده، عامل را ارسال می‌کند.
- **مقصد:** آدرس گره مقصد را نگهداری می‌کند.
- **حالت¹:** پارامتر حالت در عامل مورچه زنبور موقعیت عامل را در آن لحظه تعریف می‌کند.
- **پوشش ناحیه‌ای²:** تعداد گام‌های بین گره مبدأ و دیگر گره‌های درون ناحیه را نشان می‌دهد.
- **گره‌های ملاقات شده³:** دنباله‌ی گره‌های ملاقات شده توسط عامل می‌باشد.
- **دنباله عددی:** دوتایی (مبدأ، دنباله عددی) شناسه یکتای است که به منظور جلوگیری از ارسال عامل‌های تکراری به یک مسیر مشابه استفاده می‌شود.
- **فاصله انتخاب شده⁴:** عدد شبه تصادفی است که بیشترین فاصله‌ی مجاز عامل از کندو را مشخص می‌کند. این فیلد معادل شعاع ناحیه می‌باشد.

ساختار جداول مسیریابی

هر گره دارای دو جدول مسیریابی می‌باشد: جدول مسیریابی درون ناحیه‌ای و جدول مسیریابی بین ناحیه‌ای.

- **جدول مسیریابی بین ناحیه‌ای:** ساختار این جدول مشابه نظیر آن در پروتکل HOPNET می‌باشد که در آن مسیرهای یک گره به خارج از ناحیه ذخیره می‌شود. این جدول مسیریابی به صورت مبتنی بر تقاضا بوده و به منظور کشف مسیرهای بین ناحیه‌ای از گره‌های مرزی استفاده می‌شود.
- **جدول مسیریابی درون ناحیه‌ای:** گره‌های مرکزی در هر ناحیه معادل یک کندو می‌باشد که ساختار جدول مسیریابی آنها شامل مقادیر زیر می‌باشد:
- **گام‌ها:** تعداد گام‌ها بین گره مبدأ و دیگر گره‌های ناحیه که معادل فیلد پوشش ناحیه‌ای در عامل می‌باشد.
- **زمان بازدید:** تعداد دفعاتی که اتصال توسط عامل بازدید شده است.
- **دنباله عددی:** معادل فیلد دنباله عددی در عامل می‌باشد.
- **میزان فرومون:** مجموع فرومون اتصال است که با توجه به تعداد دفعات عبور عامل از آن محاسبه می‌شود.

کشف مسیر در پارامترهای عملکردی پروتکل‌های مسیریابی تأثیر بسزایی دارد. در ادامه به توضیح جزئیات این فاز در روش پیشنهادی پرداخته شده است.

کشف مسیر

در روش پیشنهادی همانند پروتکل HOPNET دو نوع کشف مسیر وجود دارد: درون ناحیه‌ای و بین ناحیه‌ای. فاز کشف مسیر بین ناحیه‌ای به طور مشابه پروتکل HOPNET صورت می‌گیرد. اما کشف مسیر درون ناحیه‌ای با استفاده از الگوریتم مسیریابی مورچه زنبور بهبود داده شده و به طور مفصل به آن پرداخته شده است.

الگوریتم مسیریابی درون ناحیه‌ای مورچه زنبور⁵

در روش پیشنهادی، الگوریتم‌های مورچه و زنبور را به منظور ارائه یک الگوریتم جدید با هم ترکیب نموده ایم. شاخص اصلی کیفیت مسیر، فاصله آن در نظر گرفته شده است، به این منظور که کوتاه‌ترین مسیر دارای بیشترین کیفیت می‌باشد. این الگوریتم قابلیت ارتباط فرومون در مورچه‌ها و رقص در زنبورها را باهم ترکیب نموده است و از این مقادیر و قابلیت‌ها برای تبادل اطلاعات مسیر کشف شده استفاده می‌شود. الگوریتم مسیر مورچه زنبور با ارسال مجموعه‌ای از عامل‌ها شروع به کار می‌کند.

¹ State

² Distance covered

³ Visited node

⁴ Elected Distance

⁵ AntBee Routing Algorithm

عملکرد این الگوریتم در شکل 1 توضیح داده شده است. در ابتدا عامل مورچه زنبور از کلونی خارج شده و شروع به جستجو در شبکه می‌کند. این عامل می‌تواند در 5 حالت قرار بگیرد: کاوش¹، بازگشت²، انتظار³، بازخوانی⁴، دنباله‌رو⁵.

- **کاوش:** بیانگر حالت جستجوی عامل در شبکه می‌باشد. عامل مورچه زنبور در ابتدا گره‌های که از مکان کنونی قابل دسترس هستند را جستجو می‌کند. از مجموعه‌ی گره‌های قابل دسترس یکی از آن‌ها بر اساس رابطه وزندار انتخاب می‌شود. احتمال p از انتخاب گره بعدی این مجموعه یک تابع از مقدار فرومون اتصال بین گره‌ها می‌باشد که این رابطه یک به یک نمی‌باشد. هر عامل مورچه زنبور در هر تکرار فقط می‌تواند به سمت یکی از این گره‌ها حرکت کند و مقدار فرومون را در طول مسیر مربوط به آن برجای بگذارد. عامل در حالت کاوش تا زمانی که یکی از دو رویداد زیر رخ بدهد باقی می‌ماند:

- ❖ تمام گره‌ها در مسیر کنونی ملاقات شده باشند.
- ❖ پارامتر منطقه‌ای انتخاب شده برابر با منطقه‌ای پوششی قرار گیرد به آن منظور که عامل به بیشترین مقدار فاصله از مبدأ رسیده باشد.

در این موقعیت عامل مورچه زنبور به حالت بازگشت می‌رود.

- **بازگشت:** در این حالت عامل مورچه زنبور به کلونی در خلاف جهت طی شده از مسیر کشف شده باز می‌گردد. هنگامی که عامل در این مسیر به طرف کلونی حرکت می‌کند مقدار فرومون مربوط به آن را افزایش می‌دهد. برای هر تکرار عامل از یک گره باز می‌گردد. عامل‌های مورچه زنبور در حالت بازگشت باقی می‌مانند تا زمانی که به کلونی بازگردند. هنگامی که این عامل‌ها به کلونی رسیده‌اند، حالت آن‌ها از بازگشت به انتظار تغییر می‌کند.

- **انتظار:** در این حالت عامل مورچه زنبور به کلونی بازگشته است، که احتمال وقوع سه عمل⁶ وجود دارد:
- ❖ اگر عامل مورچه زنبور در آخرین کاوش خود مسیری بهتر از مسیر ذخیره شده در جدول مسیریابی کلونی پیدا کرده باشد،

حالت اول: - مورچه زنبور در شبکه مسافرت می‌کند. - انتخاب گره با احتمال P . - مقدار فرومون مسیر افزایش می‌یابد.

حالت دوم: - مورچه زنبور مسیر را از عامل بازخوان کپی می‌کند. - در گره‌های شبکه مسافرت می‌کند. - مقدار فرومون مسیر افزایش می‌یابد.

حالت سوم: - مورچه زنبور به عقب باز می‌گردد. - مقدار فرومون مسیر کاهش می‌یابد.

حالت چهارم (گره مقصد): - مورچه زنبور مسیر جدید را با قبلی مقایسه کرده و در صورت بهتر بودن آن را بروزرسانی می‌کند. - مورچه زنبور کلونی را برای یافتن عامل دیگر در حال رقص چک می‌کند. - در غیر این صورت به کاوش شبکه ادامه می‌دهد.

حالت پنجم: - مورچه زنبور با رقص خود بقیه عامل‌های در حال انتظار را بازخوانی می‌کند. [اطلاعات مسیر بهتر کشف شده در بردار رقص کپی می‌شود].

- ❖ از حالت انتظار به بازخوانی می‌رود. در این حالت عامل دیگری در حال رقص می‌باشد و به احتمال q به وسیله عامل در حال رقص بازخوانی شده است و به حالت دنباله‌رو می‌رود.

- ❖ عامل مورچه زنبور، جستجوی خود را با رفتن به حالت کاوش ادامه می‌دهد.

- **بازخوانی:** عامل مورچه زنبور مسیر کشف شده را با مسیر ذخیره شده در جدول کلونی مقایسه می‌کند، اگر مسیر کشف شده کوتاه‌تر از مسیر ذخیره شده در جدول مسیریابی کلونی باشد. عامل به منظور بازخوانی و ارسال عوامل بیشتر به آن مسیر رقص خود را شروع می‌کند. تنها عوامل مورچه زنبوری که در آن لحظه در کلونی حاضر هستند می‌توانند بازخوانی شود. در این حالت آن عامل می‌تواند به حالت کاوش برگردد.

- **دنباله‌رو:** در این حالت یک عامل مورچه زنبور که به وسیله‌ی عامل دیگر فراخوانی شده است، به مسیر کشف شده توسط عامل قبلی می‌رود. این عمل با کپی کردن لیست گره‌های ملاقات شده انجام می‌شود. عامل مورچه زنبور بازخوانی شده مسیر را ادامه داده و فرومون آن را در راه رسیدن به مقصد افزایش می‌دهد و می‌تواند به حالت بازگشت برود.

در بخش قبل به بیان کلیات الگوریتم پیشنهادی، ساختار جداول مسیریابی و بیان حالات مختلف در الگوریتم پرداخته شد. در ادامه به منظور تکمیل مطالب به بیان فلوچارت مربوط به عملکرد الگوریتم در شکل 2 پرداخته شده است.

نگهداری مسیر

¹ Exploring

² Returning

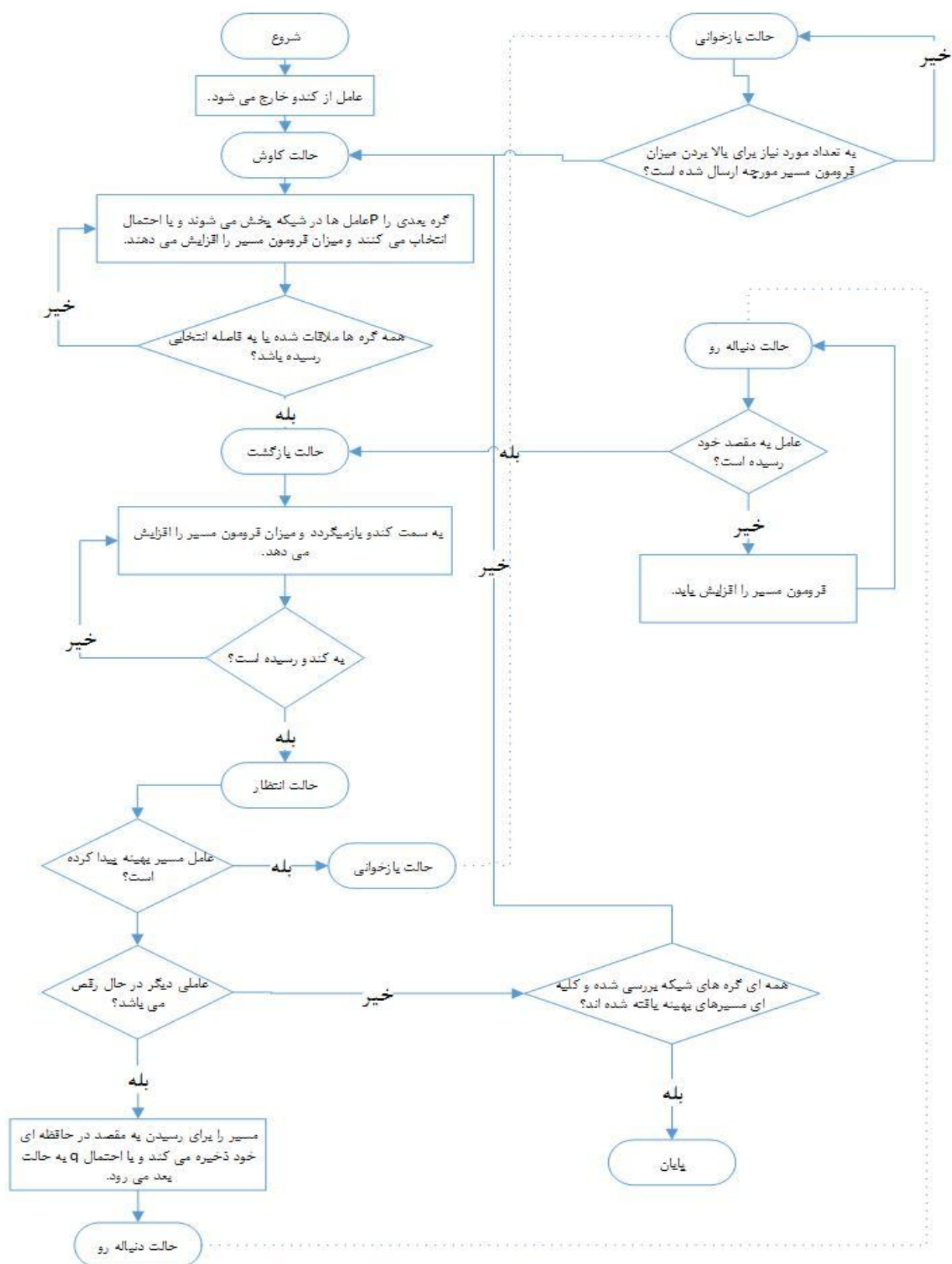
³ Waiting

⁴ Recruiting

⁵ Following

⁶ Action

در صورتی که یک مسیر شکسته شود و درون ناحیه‌ای باشد، روش پیشنهادی با ارسال عامل مورچه زنبور به منظور کشف مسیر جدید درون ناحیه‌ای کلیه مسیرهای شکسته شده تا آن شعاع را ترمیم خواهد شد و سربار شبکه نسبت به روش مشابه آن در پروتکل HOPNET کمتر خواهد بود. اگر این مسیر بین ناحیه‌ای غیر معتبر باشد مشابه عملکرد پروتکل HOPNET عمل شده و نزدیک‌ترین گره به اتصال شکسته شده یک روبه‌ی ترمیم محلی اجرا خواهد کرد. در هنگام اجرای فرآیند ترمیم، تمام بسته‌های داده در بافر گره ذخیره خواهند شد که اگر گره بتواند مسیر جایگزین را به مقصد پیدا کند، بسته‌های داده‌ی ذخیره شده را از طریق مسیر جدید به گره مقصد ارسال می‌کند و همچنین یک مورچه اطلاع دهنده را به مقصد پیدا کند، بسته‌های داده‌ی ذخیره شده را از طریق مسیر جدید به گره مقصد ارسال می‌کند و همچنین یک مورچه اطلاع دهنده را به مقصد پیدا کند. تمامی گره‌ها در مسیر که مورچه‌ای اطلاع دهنده آن‌ها را ملاقات می‌کنند، جدول مسیریابی بین ناحیه‌ای خود را بروزرسانی می‌کنند. اگر مسیر جایگزین پیدا نشود، یک مورچه خطا به گره مبدأ ارسال خواهد شد. پس از دریافت مورچه خطا اگر گره مبدأ همچنان به مسیری برای ارسال بسته‌ها به گره مقصد نیاز داشته باشد، فرآیند کشف مسیر همانطور که در قسمت قبل ذکر گردید، اجرا خواهد شد.



شکل 1: فلوچارت کشف مسیر درون ناحیه‌ای روش پیشنهادی

4. شبیه‌سازی و نتایج آن

کلیه شبیه‌سازی‌های انجام شده در این فصل با استفاده از نرم افزار NS-2 بوده که آن یک شبیه‌ساز رویداد گسسته است. این شبیه‌ساز از دو زبان برنامه‌نویسی C++ و TCL بهره می‌برد. و بر روی سیستمی با مشخصات سخت افزاری Intel Core I5

Mainboard: Intel Z97 3400 و RAM: 4GB، CPU: M560 2.67GHz
 شده در مقایسه الگوریتم‌های مسیریابی به شرح زیر می‌باشد:

- **نرخ تحویل بسته¹:** نرخ موفقیت بسته‌های داده‌ای تحویل داده شده به تعداد کل بسته‌های ارسال شده از مبدأ به مقصد می‌باشد.
- **سربر²:** میزان سربر با استفاده از نرخ مقدار اطلاعات لازم به منظور انتقال (حمل) ترافیک کنترلی به مقدار کل ترافیک ارسال شده می‌باشد.
- **تاخیر انتها به انتها³:** اختلاف زمان بین دریافت بسته در لایه‌ی کاربردی گره مقصد با زمان ارسال آن بسته از لایه‌ی کاربردی گره مبدأ را بیان می‌کند.
- **Delay:** میانگین اختلاف زمان بین-ورود⁴ بین بسته‌های داده‌ای می‌باشد.
- **تعداد بسته‌های از دست رفته⁵:** تعداد بسته‌های اطلاعاتی که به صورت موفق به مقصد نرسیده‌اند.
- **توان عملیاتی⁶:** مقدار این پارامتر به ظرفیت کانون ارتباطی بستگی دارد. این پارامتر نشان دهنده‌ی حداکثر مقدار ممکن تحویل پیام در کانال می‌باشد. معمولاً با مقیاس bit/sec انداز مگیری می‌شود.
- **مصرف انرژی⁷:** مقدار کل انرژی مصرف شده که شامل انرژی مصرفی توسط بسته‌های کنترلی و ارسال یک کیلوبایت اطلاعات به مقصد می‌باشد.

الگوریتم پیشنهادی و HOPNET را در شبیه ساز NS-2 پیاده‌سازی شده است. شرایط شبیه‌سازی و سناریو انتخاب شده در جدول 2 موجود است.

جدول 2: سناریوهای شبیه‌سازی روش پیشنهادی

پارامترها	سناریوها			
	سناریوی اول	سناریوی دوم	سناریوی سوم	سناریوی چهارم
نوع صف	Droptail	Droptail	Droptail	Droptail
اندازه شبکه	100×100	100×100	100×100	100×100
آنتن	همه جهته ⁸	همه جهته	همه جهته	همه جهته
زمان شبیه سازی	1000	1000	1000	1000
تعداد گره‌ها	25	50	100	200
موقعیت گره‌ها	تصادفی	تصادفی	تصادفی	تصادفی
انرژی اولیه	1 ژول	1 ژول	1 ژول	1 ژول
محدوده انتقال گره‌ها	50 متر	50 متر	50 متر	50 متر
طول صف	50	50	50	50

¹ Packet Delivery Ratio

² Overhead

³ End-to-End Delay

⁴ Inter- Arrival

⁵ Number of Packet Dropped

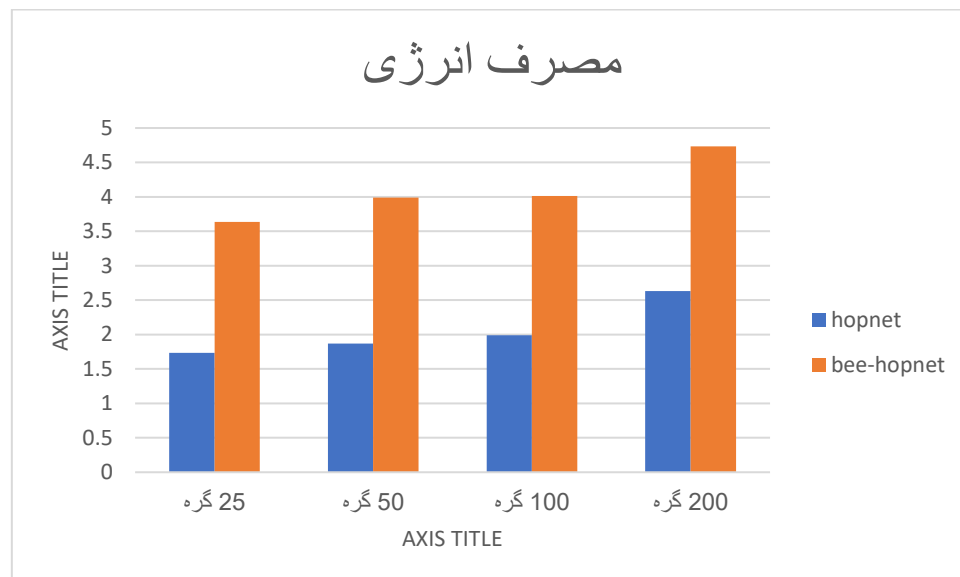
⁶ Throughput

⁷ Energy

⁸ Omni Directional

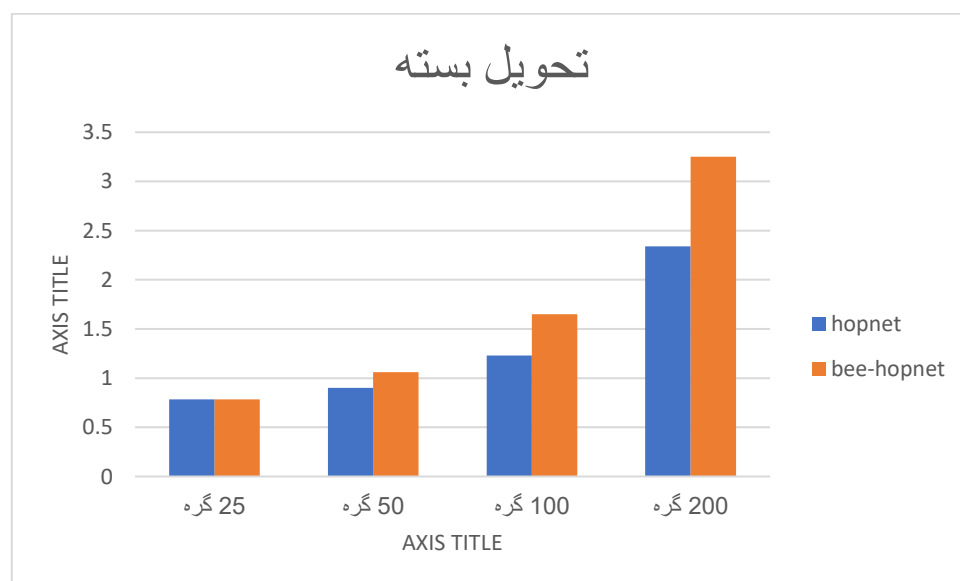
همانطور که در جدول مشاهده می‌شود، چهار سناریوی مختلف را برای شبیه‌سازی در نظر گرفته شده است. تمام سناریوها به مدت 1000 ثانیه اجرا شدند و پهنای باند 1 مگابایت در نظر گرفته شده است. هر یک از سناریوها هشت بار اجرا شده و سپس از خروجی اطلاعات لازم را استخراج شده است. سپس اطلاعات به دست آمده را یک به یک با هم مقایسه نموده که در بخش‌های بعدی ذکر شده است.

در این بخش میزان مصرف انرژی گره‌ها بررسی شده است. محاسبه مرگ آخرین گره، به معنای مصرف انرژی کامل تمامی گره‌ها است. در این حالت فقط می‌توان مدت زمان اتمام انرژی گره‌ها را محاسبه نمود. برای مقایسه این پارامتر، نیاز به اجرای مجدد شبیه‌سازی است اما در این شبیه‌سازی‌ها انرژی اولیه گره‌ها برابر با پنج ژول می‌باشد.



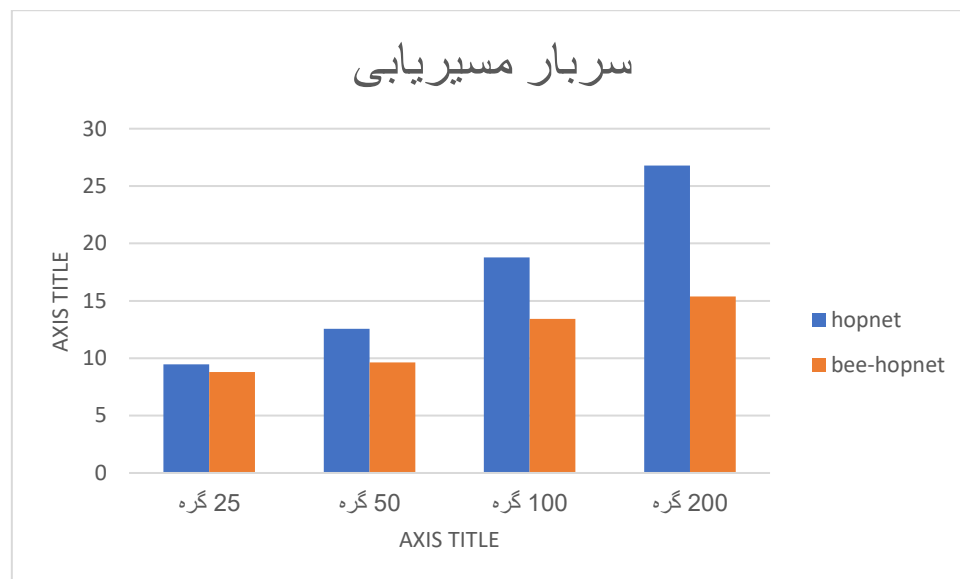
نمودار 1: مصرف انرژی

نرخ تحویل تعداد بسته‌هایی است که با موفقیت در مقصد دریافت شده‌اند. برای محاسبه تعداد گره‌های دریافتی در مقصد نسبت تعداد بسته‌های دریافتی در مقصد بر تعداد بسته‌های ارسالی محاسبه شده است. در این بخش فقط تعداد بسته‌های ارسالی و دریافتی داده در لایه کاربرد محاسبه شده، چرا که در این لایه بسته‌های دیگری هم ارسال شده است.



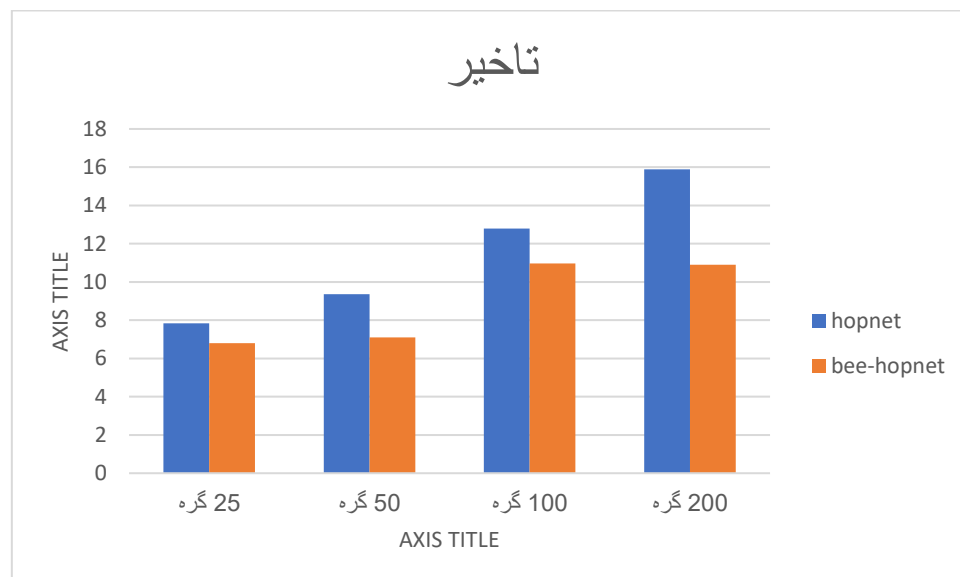
نمودار 2: نرخ تحویل بسته به ایستگاه پایه

در این بخش سربار مسیریابی محاسبه شده است. این سربار برابر با نسبت تعداد بسته‌های ارسالی برای ارسال یک بسته اطلاعاتی است. این پارامتر تنها برای لایه کاربرد محاسبه شده است (نسبت میزان حجم بسته‌های داده را بر میزان حجم اطلاعات ارسالی برای تشکیل خوشه‌ها و تحمل پذیری خط).



نمودار 3: نرخ سربار مسیریابی

از آنجایی که در این پروتکل‌ها مسیرها از قبل مشخص است لذا نرخ تأخیر در این پروتکل‌ها معنایی را که در پروتکل‌های مسیریابی مرسوم دارد، نمی‌دهد در عمل نرخ تأخیر لایه پایین‌تر محاسبه شده است. در این بخش منظور از نرخ تأخیر میزان زمان لازم برای تشکیل خوشه و ارسال اولین بیت بسته داده است.



نمودار 4: نرخ تأخیر

5. نتیجه‌گیری

در این پژوهش به بهبود پروتکل مسیریابی HOPNET با استفاده از الگوریتم کلونی زنبور عسل پرداخته شده است. این بهبود در دو بخشاز پروتکل اعمال گردید: کشف مسیر درون ناحیه‌ای و ساختار عامل مسیریاب مورچه زنبور. در این روش با الهام از عملکرد زنبور عسل در طبیعت، فاز کشف مسیر بهبود یافته و اعمال این تغییر منجر به بهبود روند نگهداری مسیر نیز شده است. استفاده از روش پیشنهادی در عامل مسیریاب، سربار مسیریابی را کاهش داده است. این امر باعث بهبود نرخ تأخیر آنها به انتها

و انرژی مصرفی گره‌ها شده است. نتایج حاصل از شبیه‌سازی نشان داده است که با افزایش تعداد گره‌ها نیز مقادیر پارامترهای مورد بررسی از نظیر خود در پروتکل‌های بحث شده دیگر بهتر بوده است. اما از مشکلات همراه با این بهبود می‌توان به افزایش حجم اطلاعات به طور نسبی در شبکه اشاره نمود. این پارامتر با بالا رفتن تعداد گره‌ها و متعاقباً تعداد عامل‌های لازم برای کشف، نگهداری و ترمیم مسیر افزایش می‌یابد. به طور خلاصه می‌توان نتیجه گرفت که استفاده از الگوریتم زنبور عسل در بهبود مسیریابی و به صورت ترکیب با الگوریتم کلونی مورچه‌ها منجر به بهبود برخی پارامترهای ارزیابی عملکرد شبکه شده اما به دلیل بالا رفتن فضای لازم برای ذخیره‌سازی مقادیر موجود در عامل‌های مسیریابی، حجم اطلاعات جاری در شبکه را افزایش می‌دهد. البته باید به این نکته اشاره نمود که استفاده از این روش پیشنهادی در ترمیم و نگهداری مسیر نیز موثر بوده و این فرایندها را سرعت می‌بخشد. با توجه به نتایج ذکر شده از شبیه‌سازی، پارامترهای مورد بررسی به صورت چشمگیری بهبود یافته‌اند به طوریکه نرخ تأخیر 65%، سربار شبکه 60%، انرژی مصرف شده 52% کاهش و نرخ تحویل بسته 40% افزایش داشته است. در نهایت می‌توان گفت که استفاده از الگوریتم‌های هوش جمعی می‌تواند در بهبود پارامترهای ارزیابی مسیریابی در شبکه تأثیر قابل توجهی داشته باشد. با توجه به کارایی الگوریتم‌های هوش جمعی، می‌توان از ترکیب آنها در بهبود تمام جنبه‌های یک شبکه استفاده نمود. به عنوان مثال بهینه‌سازی مصرف انرژی در شبکه‌های حسگر اهمیت بسزایی دارد. از اینرو می‌توان این دسته الگوریتم‌ها را در افزایش طول عمر این شبکه‌ها به کار برد و تأثیر آن را مطالعه نمود.

مراجع

- [1] J. Macker and S. Corson, Mobile Ad Hoc Networks (MANET), IEEE, 1997.
- [2] P. Dempsey and A. Schuster, Swarm intelligence for network routing optimization, Journal of Telecommunications and Information technology, vol.12, pp. 24-28, 2005.
- [3] P. Arabshahi, A. Gray, I. Kassabalidis, A. Das, S. Narayanan, M. El-Sharkawi and R.J. Marks, Adaptive routing in wireless communication networks using swarm intelligence, Proceeding of the 9th International Communications Satellite Systems Conference, Toulouse, France, pp. 219-230, 17-20 April 2001.
- [4] J. Wang, E. Osagie, P. Thulasiraman and R.K. Thulasiram, HOPNET: A hybrid ant colony optimization routing algorithm for mobile ad hoc network, Ad Hoc Networks, vol.7, no. 4, pp. 690-705, 2009.
- [5] H.F. Wedde, M. Farooq, T. Pannenbaecker, B. Vogel, C. Mueller, J. Meth and R. Jeruschkat, BeeAdhoc: An energy efficient routing algorithm for mobile ad hoc networks inspired by bee behavior, Proceeding of the Genetic and Evolutionary Computation, vol. 1, pp. 153-160, 2005.
- [6] H.F. Wedde, M. Farooq and Y. Zhang, BeeHive: An efficient fault-tolerant routing algorithm inspired by honey bee behavior, Ant Colony Optimization and Swarm Intelligence, Springer, 2004.
- [7] D. Chaudhary, Bee-inspired routing protocols for mobile ad hoc network (MANET), Journal of Emerging Technologies in Web Intelligence, vol. 2, no. 2, pp. 86-88, 2010.
- [8] Rupérez Cañas D, Sandoval Orozco AL, García Villalba LJ, Kim TH. A Family of ACO Routing Protocols for Mobile Ad Hoc Networks. Sensors. 2017 May 22;17(5):1179.
- [9] Jayalalitha B, Reddy PC. Design of an ANT based routing algorithm for mobile ad-hoc networks. In Communication and Electronics Systems (ICCES), International Conference on 2016 Oct 21 (pp. 1-3). IEEE.
- [10] Baskaran A, Chaudhari N, Caytiles RD, Iyengar NC. Fuzzy Optimized and Bee inspired Routing Protocol for Improved QoS in Mobile Ad Hoc Networks. International Journal of Control and Automation. 2016;9(8):391-402.

- [11] Anwar N, Deng H. Ant Colony Optimization based multicast routing algorithm for mobile ad hoc networks. In *Advances in Wireless and Optical Communications (RTUWO)*, 2015 Nov 5 (pp. 62-67). IEEE.
- [12] Di Caro G, Dorigo M. AntNet: Distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research*. 1998;9:317-65.
- [13] Kaur A, Singh A. A Review on Ant Based Routing Protocols for Manet. *International Journal*. 2015 Mar;5(3).
- [14] Jagtap, Anjali, Ashok Shinde, Smita Kadam, Dipak Raut, and Parag Hirulkar. "IMPLEMENTATION OF OPTIMIZED ANT BASED ROUTING ALGORITHM FOR MANET." (2017).
- [15] Agnihotri S, Ramkumar KR. A Survey and Comparative Analysis of the Various Routing Protocols of Internet of Things. *International Journal of Pervasive Computing and Communications*. 2017 Aug 21(just-accepted)
- [16] Kaur A, Dhaka VS, Singh G. Voyaging Through Various Ant Based Routing Algorithms For MANETs. *International Journal of Advanced Technology in Engineering & Science*. 2015 Sep;3(1):468-80.
- [17] Wang J, Osagie E, Thulasiraman P, Thulasiram RK. HOPNET: A hybrid ant colony optimization routing algorithm for mobile ad hoc network. *Ad Hoc Networks*. 2009 Jun 30;7(4):690-705.
- [18] Dorigo, M. and M. Birattari, Ant colony optimization, in *Encyclopedia of machine learning*. 2010, Springer. p. 36-39
- [19] Karaboga, D., et al., A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review*, 2014. 42(1): p. 21-57