مجله محاسبات نرم
SOFT COMPUTING JOURNAL
تارنمای مجله: scj.kashanu.ac.ir
دانشگاه کاشان
University of Kashan

موقعیت‌یابی چندگامه مبتنی بر الگوریتم ازدحام ذرات برای شبکه‌های حسگر بی‌سیم✧

سعید دوست‌علی[۱]، دانشجوی دکتری، محمد خلیلی درمنی[۲*]، استادیار
[۱] دانشکده مهندسی کامپیوتر، گروه مهندسی کامپیوتر، دانشگاه کاشان، کاشان، ایران.
[۲] دانشکده مهندسی کامپیوتر، دانشگاه آزاد اسلامی، واحد خمین، خمین، ایران.

اطلاعات مقاله

تاریخچه مقاله:
دریافت ۱۹ خرداد ماه ۱۳۹۹
پذیرش ۰۳ شهریور ماه ۱۳۹۹

کلمات کلیدی:
موقعیت‌یابی چندگامی
شبکه‌های حسگر بی‌سیم
مثلث‌سازی
الگوریتم بهینه‌سازی ازدحام ذرات
متوسط خطای موقعیت‌یابی

چکیده

یک شبکهٔ حسگر از تعداد زیادی گره حسگر تشکیل شده که اطلاعات محیط جغرافیایی بزرگی را که در آن پخش شده‌اند جمع‌آوری می‌کنند. به‌دلیل اهمیت تعیین محل وقوع یک رخداد، موقعیت‌یابی یکی از موضوعات کلیدی و مهم در حوزهٔ شبکه‌های حسگر بی‌سیم محسوب می‌شود. از سوی دیگر، استفاده از موقعیت‌یاب جهانی GPS برای پیدا کردن موقعیت حسگرها به‌دلیل برخی محدودیت‌های موجود در گره‌های حسگر مانند قیمت و اندازهٔ فیزیکی مناسب نیست. در برخی کاربردها، تعدادی گره راهنما که از موقعیت خود اطلاع دارند به گره‌های حسگر اطلاعاتی می‌دهند تا بتوانند موقعیت خویش را تعیین کنند. به‌دلیل وجود خطا، دقت پایین روش‌های فاصله‌یابی و از همه مهم‌تر فاصلهٔ چندگامهٔ گره‌های حسگر از گره‌های راهنما، ممکن است موقعیت به‌دست‌آمده دقت مناسبی نداشته باشد. در این مقاله، یک الگوریتم ازدحام ذرات توزیع‌شده برای تعیین موقعیت گره‌های حسگر ارائه شده است. در الگوریتم پیشنهادی، از میانگین طول گام و تعداد گام فاصله بین گره حسگر تا گره‌های راهنما برای تعیین موقعیت کمک گرفته شده است. نتایج شبیه‌سازی و همچنین مقایسهٔ خطای موقعیت‌یابی در الگوریتم پیشنهادی و الگوریتم‌های رایج، حاکی از کارایی مناسب الگوریتم ارائه‌شده در فراهم کردن موقعیت دقیق مکانی است.

© ۱۳۹۹ – مجله محاسبات نرم، کلیه حقوق محفوظ است.

۱. مقدمه

شبکه‌های حسگر بی‌سیم از تعداد زیادی گره حسگر ارزان‌قیمت تشکیل شده است که گره‌ها منابع انرژی، پردازشی و ارتباطی ضعیفی دارند. گره‌های حسگر قابلیت حس کردن و تأثیرگذاری بر محیط را دارند و کاربردهای متنوعی برای این شبکه‌ها مطرح می‌شود. معمولاً از این شبکه‌ها در یک فضای جغرافیایی بزرگ استفاده می‌شود تا رویدادهای محیطی را تشخیص دهند. اولین سؤالی که دربارهٔ داده‌های به‌دست‌آمده از حسگرها مطرح می‌شود، این است که رویدادهای گزارش‌شده در چه مکانی رخ داده‌اند. برای پاسخ به این سؤال لازم است که حسگر اطلاعات لازم را در مورد موقعیت خود داشته باشد. بنابراین، موقعیت‌یابی یکی از موضوعات کلیدی و مهم در حوزهٔ تحقیقاتی شبکه‌های حسگر بی‌سیم محسوب می‌شود [۱].

در شبکه‌ای که از هزاران گره تشکیل شده است، نمی‌توان موقعیت هر گره را به‌طور دقیق از قبل تعیین کرد [۲]. از سوی

✧ نوع مقاله: پژوهشی
* نویسنده مسئول
پست‌های الکترونیک: doostali.s@gmail.com (دوست‌علی)
md.khalili@gmail.com (خلیلی درمنی)

نحوه ارجاع به مقاله: دوست‌علی، سعید، خلیلی درمنی، محمد، «موقعیت‌یابی چندگامه مبتنی بر الگوریتم ازدحام ذرات برای شبکه‌های حسگر بی‌سیم»، مجله محاسبات نرم، جلد ۸، شماره ۱، ص۵۸ـ۶۹، بهار و تابستان ۱۳۹۸.

دیگر، استفاده از موقعیت‌یاب جهانی (GPS)[1] برای شبکهٔ حسگر بی‌سیم مناسب نیست [۳]. دلایل این موضوع عبارت‌اند از: ۱. موقعیت‌یاب جهانی در محیط‌های بسته و مناطق با پوشش گیاهی فراوان، دقت و صحت مناسبی ندارد؛ ۲. استفاده از موقعیت‌یاب جهانی باعث افزایش انرژی مصرفی گره‌ها می‌شود که به‌نوبهٔ خود سبب کاهش طول عمر شبکه می‌گردد؛ ۳. ابعاد موقعیت‌یاب جهانی و آنتن آن، از نظر فیزیکی، اندازهٔ حسگرها را افزایش می‌دهد؛ ۴. با توجه به اینکه گره‌های حسگر باید در تعداد بالا و ارزان به کار گرفته شوند، استفاده از موقعیت از دستگاه‌های موقعیت‌یاب جهانی به‌دلیل قیمت بسیار بالای آن‌ها، مقرون‌به‌صرفه نیست. بنابراین یک الگوریتم موقعیت‌یابی برای تعیین موقعیت گره‌های حسگر در یک شبکهٔ حسگر بی‌سیم مورد نیاز است که با توجه به محدودیت‌های مربوط به اندازه، انرژی اولیه و هزینهٔ گره‌های حسگر، اطلاعات موقعیتی هر گره را ارائه دهد. بر این مبنا بسیاری از برنامه‌های کاربردی در سال‌های اخیر روی این مسئله متمرکز بوده‌اند.

در برخی از کاربردها تعدادی گره راهنما در محیط مورد نظر پخش می‌شود که موقعیت مشخص‌شده‌ای دارند. سپس، گره‌های حسگر مجاور این گره‌های راهنما، با کمک تخمین فاصلهٔ خود از گره‌های راهنما، موقعیت خویش را به دست می‌آورند. گره‌های غیرمجاور نیز با تخمین فاصلهٔ خود با گره‌هایی که موقعیت خود را به دست آورده‌اند، موقعیت خود را تخمین می‌زنند. واضح است که هر چقدر تعداد گام فاصلهٔ گره حسگر از گره‌های راهنما افزایش یابد، دقت و صحت موقعیت‌یابی کاهش پیدا می‌کند. حساسیت بالا به توپولوژی شبکه، انتشار و انباشته شدن خطا در شبکه از مشکلات رایج در روش‌های موقعیت‌یابی چندگامی است [۴]. در تحقیقات اخیر، از روش‌های بهینه‌سازی و فراابتکاری برای بهبود تخمین موقعیت گره انجام شده است [۴ـ۷]. اما با توجه به متمرکز بودن این روش‌ها، همهٔ اطلاعات شبکه از جمله فاصلهٔ بین تمام گره‌ها از یکدیگر باید به گره مرکزی گزارش شده و سپس الگوریتم موقعیت‌یابی روی این داده‌ها اجرا شود. واضح است که با افزایش تعداد گره‌ها و همچنین پویایی شبکه، پیچیدگی این روش‌ها افزایش یافته و تطبیق‌پذیری آن‌ها کاهش خواهد یافت.

در این مقاله، یک الگوریتم موقعیت‌یابی توزیع‌شده بر مبنای بهینه‌سازی ازدحام ذرات (PSO)[2] برای موقعیت‌یابی گره‌های حسگر ارائه شده است. بار محاسباتی پایین، تعداد کم پارامترهای مورد نیاز الگوریتم و در نهایت پیچیدگی کم پیاده‌سازی، دلایل انتخاب الگوریتم ازدحام ذرات برای این منظور است. علاوه بر این الگوریتم ازدحام ذرات در مقایسه با سایر روش‌های فراابتکاری کمتر تحت‌تأثیر جمعیت اولیه قرار گرفته و دارای سرعت همگرایی بیشتری است، که این امر به نوبهٔ خود سبب کاهش انرژی مصرفی در گره‌های حسگر می‌شود [۸ و ۹]. در الگوریتم پیشنهادی، از میانگین طول گام و تعداد گام فاصله بین گره حسگر تا گره‌های راهنما برای تعیین موقعیت تخمینی کمک گرفته شده است. سپس از الگوریتم بهینه‌سازی ازدحام ذرات برای تعیین موقعیت دقیق گره حسگر بهره برده می‌شود. الگوریتم پیشنهادی توسط گره‌های حسگر اجرا شده و لازم نیست که اطلاعات به گره مرکزی منتقل شوند. بنابراین، روش پیشنهادی در شرایط پویایی شبکهٔ حسگر بی‌سیم با تعداد گره زیاد کارایی مناسبی خواهد داشت. نتایج شبیه‌سازی نشان می‌دهد که استفاده از الگوریتم پیشنهادی در شبکهٔ حسگر بی‌سیم، میانگین خطای موقعیت‌یابی را کاهش می‌دهد و موقعیت‌های به‌دست‌آمده دارای دقت و صحت مناسبی خواهند بود. از سوی دیگر، استفادهٔ حداقل از گره‌های راهنما و سخت‌افزار اضافه مانند موقعیت‌یاب جهانی، موجب کاهش هزینه‌ها می‌شود. همچنین، الگوریتم پیشنهادی نسبت به الگوریتم‌های مشابه موقعیت‌یابی دارای بار محاسباتی کمتری است.

این مقاله در پنج بخش سازمان‌دهی شده است. بخش دوم به مرور پیشینهٔ موقعیت‌یابی و الگوریتم‌های موقعیت‌یابی در شبکه‌های بی‌سیم می‌پردازد. با توجه به اشکالات الگوریتم‌های موقعیت‌یابی موجود یک الگوریتم موقعیت‌یابی جدید در

1. Global Positioning System

2. Particle Swarm Optimization

شبکه‌های حسگر بی‌سیم با مقیاس بزرگ با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات در بخش سوم ارائه شده است. در بخش چهارم، کارایی روش پیشنهادی در مقایسه با سایر روش‌ها ارزیابی شده و در نهایت در بخش پنجم، نتیجه‌گیری و کارهای آتی بیان شده است.

۲. مرور کارهای گذشته

پیدا کردن موقعیت گره حسگری که در مجاورت آن رویدادی رخ داده است، برای خیلی از کاربردهای شبکهٔ حسگر بی‌سیم لازم بوده و یک چالش عمدهٔ این شبکه‌ها محسوب می‌شود؛ زیرا بدون تعیین موقعیت حسگر، اطلاعات گزارش‌شده مفید نیستند [۱۰]. در روش‌های تعیین موقعیت فرض می‌شود تعدادی از گره‌های حسگر از طریق پیکربندی دستی یا با استفاده از موقعیت‌یاب جهانی از موقعیت خود اطلاع دارند. به این گره‌ها، گره‌های راهنما گفته می‌شود. گره‌های راهنما در واقع گره‌هایی از شبکه هستند که مختصات خود را در دستگاه مختصات مطلق می‌دانند. سایر گره‌های حسگر با کمک تخمین فاصله از گره‌های راهنما موقعیت خود را به دست می‌آورند. برای تخمین فاصله از الگوریتم‌های موقعیت‌یابی مختلف استفاده می‌شود که به دو دسته تقسیم می‌شود: الگوریتم‌های موقعیت‌یابی تک‌گامی[1] و چندگامی[2] [۱۲ و ۱۱]. در الگوریتم‌های موقعیت‌یابی تک‌گامی، گرهی با مختصات نامعلوم می‌تواند به‌طور مستقیم با گره‌های راهنما ارتباط برقرار کرده و در واقع بین گره راهنما و گره مذکور تنها یک گام فاصله وجود دارد. اما در الگوریتم موقعیت‌یابی چندگامی، از آنجایی که گره‌ها نمی‌توانند با گره‌های راهنما به‌طور مستقیم در ارتباط باشند، با استفاده از گره‌های میانی و همکاری بین گره‌ها موقعیت گره‌های حسگر به دست می‌آید. در ادامهٔ این بخش، ابتدا روش‌های تعیین فاصلهٔ گره‌ها از یکدیگر بررسی شده، سپس روش‌های موقعیت‌یابی تک‌گامی و در انتها روش‌های موقعیت‌یابی چندگامی مورد بررسی قرار خواهند گرفت.

۲.۱. تعیین فاصلهٔ گره‌ها از یکدیگر

در روش‌های موقعیت‌یابی، لازم است که تخمینی از فاصلهٔ گره تا تعدادی از گره‌های راهنما فراهم شود. روش‌های متعددی برای تعیین فاصله وجود دارد که در ادامهٔ این روش‌ها توصیف شده است.

روش شاخص قدرت سیگنال[3] (RSSI): با فرض اینکه توان سیگنال ارسالی P_{tx}، مدل افت مسیر و ضریب افت مسیر α باشند، گیرنده می‌تواند از توان سیگنال دریافتی، P_{rcvd}، برای محاسبهٔ فاصله، d، بین خودش و فرستنده استفاده کند. در عمل، در این روش فاصله با فرمول $d = \sqrt[\alpha]{\frac{cP_{tx}}{P_{rcvd}}}$ محاسبه می‌شود [۱۳]. از آنجا که این روش نیازی به سخت‌افزار اضافی ندارد و همچنین فاصله به‌راحتی و بدون نیاز به هماهنگی و ارسال سیگنال‌های اضافی به دست می‌آید، روش مناسبی به نظر می‌رسد. البته اشکال این روش این است که مقادیر RSSI ثابت نیستند، حتی اگر گیرنده و فرستنده ثابت باشند و حرکت نکنند مقدار توان دریافتی دائم در حال تغییر است. البته می‌توان با استفاده از ارسال مکرر احتمال خطا را کاهش داد. ازاین‌رو هنگام استفاده از RSSI به‌عنوان روشی برای تعیین فاصله، می‌بایست طراحان درصدی از خطا را بپذیرند [۱۳].

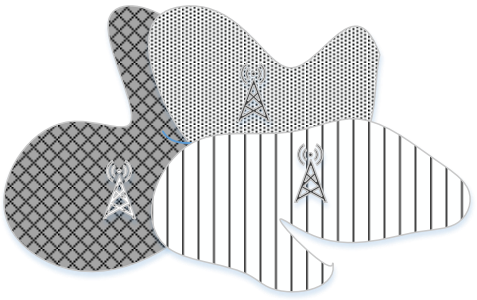
روش زمان ورود[4] (ToA): زمان ورود، روشی است که در آن از رابطهٔ بین فاصله و زمان ارسال سیگنال، در صورت دانستن سرعت انتشار می‌توان استفاده کرد. با فرض آنکه فرستنده و گیرنده زمان شروع ارسال را می‌دانند، با استفاده از زمانی که طول می‌کشد سیگنال از لحظه شروع به گیرنده برسد مانند t و سرعت انتشار سیگنال مانند v می‌توان فاصلهٔ بین دو گره را به‌صورت $d = v.t$ محاسبه کرد. از آنجایی که نیاز است بین فرستنده و گیرنده همزمان‌سازی دقیقی صورت گیرد، در صورت همزمان نبودن فرستنده و گیرنده خطای اندازه‌گیری رخ خواهد داد. از اشکالات دیگر این روش آن است که

1. Single Hop
2. Multi Hop
3. Received Signal Strength Indicator
4. Time of Arrival

سرعت انتشار بسته در محیط‌های مختلف، متفاوت است، لـذا شرایط محیطی بر اندازه‌گیری‌ها تأثیرگذار خواهد بود [۱۴].

روش اختلاف زمان ورود[1] (TDoA): برای غلبه بر مشکل همزمان‌سازی، روش اختلاف زمان ورود یا TDoA معرفی شد. در این روش از دو سـیگنال بـا سـرعت‌هـای متفـاوت بـرای اندازه‌گیری فاصله استفاده می‌شود. اشکال این روش استفاده از دو فرستنده و گیرنده است، ولی از روش‌هایی که تاکنون برای تخمین فاصله بیان شد، دقیق‌تـر عمـل مـی‌کنـد. ایـن روش و گونه‌های دیگر این روش که در مقالات متعددی مورد استفاده قرار گرفته‌اند، توانسته‌اند با دقت ۲ سانتی‌متر در موقعیـت‌یـابی موفق باشند. دقت شود که روش TDoA در مقابل روش RSSI از دقت بسیار بیشتری برخوردار است، درحـالی‌کـه تجهیـزات سخت‌افزاری و پیچیدگی‌های آن به‌مراتب بیشتر است [۱۰].

اندازه‌گیری زاویهٔ ورود[2] (AoA): روش دیگر اندازه‌گیری فاصله، استفاده از اندازه‌گیری زاویه است. این زاویه مـی‌توانـد زاویهٔ بین خط واصل یک گره به گره راهنمـا نسـبت بـه یـک مرجع (به‌عنوان مثل صفر درجهٔ شمالی) باشـد. در روش‌هـای قدیمی زاویه را با استفاده از آنتن‌های جهت‌دار مـدور هماننـد ایستگاه‌های رادار اندازه‌گیری می‌کردند. روش دیگر، استفاده از چندین آنتن نصب‌شده با فواصل معین بر روی حسگر است که در این حالت از اختلاف زمانی سیگنال دریافت‌شده توسط این آنتن‌ها جهت سیگنال ارسال‌شده محاسبه می‌شود. هرچه اندازه و فواصل این آنتن‌ها به هم نزدیک‌تر باشد، جهت اندازه‌گیری‌شده دقیق‌تر خواهد بود. از آنجایی که این راه‌حل بسیار پـرحجم و گران‌تر از روش‌های یادشده است، از آن کمتر استفاده می‌شود [۱۵]. دقت داشته باشید که در یک محیط به‌علت وجود عواملی مانند ساختمان‌ها و دیگر موانع موجود، پخش سیگنال در عمل به‌صورت دایره‌ای نیست [۴] و همان طـور کـه در شـکل (۱) نشان داده شده، پوشش سیگنال به‌صورت ابری، دور گره راهنما است.

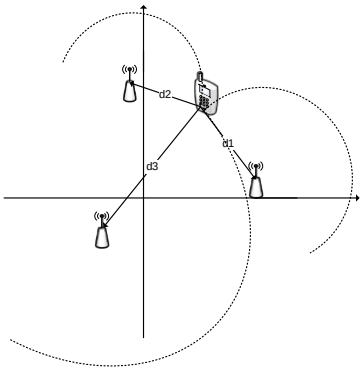


شکل (۱): پوشش سیگنال رادیویی اطراف گره راهنما [۱۶]

۲.۲. موقعیت‌یابی در محیط‌های تک‌گامی

برای تشخیص موقعیت به روش مثلث‌سازی[3] با داشتن فاصلهٔ گره از چنـد گـره راهنمـا مـی‌تـوان مکـان گـره را بـه دسـت آورد. مثلث‌سازی به دو روش موقعیت را به دست می‌آورد: مثلث‌سـازی از طریق محاسبهٔ فاصله[4] و مثلث‌سازی از طریق محاسبهٔ زاویه[5].

مثلث‌سازی از طریق محاسبهٔ فاصله: در مثلـث‌سـازی از طریق فاصلهٔ موقعیت گره با داشتن فاصلهٔ آن تا گره راهنمـا و استفاده از روابط ریاضی مثلث محاسبه می‌شود. در موقعیت‌یابی دوبعدی، حداقل به سه گره راهنما و در موقعیت‌یابی سه‌بعدی به چهار گره راهنما نیاز است. هرچه تعداد این گره راهنما بیشتر شود، موقعیت‌یابی دقیق‌تـر انجـام مـی‌شـود. الگـوریتم مثلـث دوبعدی به سه مقدار متفاوت از سه نقطه که در یک راستا قرار نگرفته باشند و تشکیل یک مثلث بدهند، نیاز دارد و موقعیـت گره حسگر در تقاطع سه دایره قرار داده می‌شـود. پیـدا کـردن موقعیت گره به این روش در شکل (۲) نشان داده شده است.



شکل (۲): موقعیت‌یابی با روش مثلث‌سازی از طریق فاصله در فضای دوبعدی [۱۶]

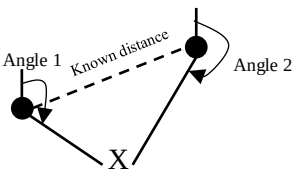
1. Time Difference of Arrival
2. Angle of Arrival
3. Triangulation
4. Lateration
5. Angulation

اگر سه مرجع دارای مختصات (x_1,y_1)، (x_2,y_2)، (x_3,y_3) باشند و گرهی با مختصات (x_i,y_i) وجود داشته باشد، با استفاده از رابطهٔ ریاضی (۱) فاصلهٔ موجودیت مشخص می‌شود.

$$\begin{aligned}(x_1-x_i)^2+(y_1-y_i)^2&=d_1\\(x_2-x_i)^2+(y_2-y_i)^2&=d_2\\(x_3-x_i)^2+(y_3-y_i)^2&=d_3\end{aligned} \quad (۱)$$

که در آن، d_1، d_2 و d_3 فاصلهٔ گره‌های ۱، ۲ و ۳ از گره ناشناخته هستند. البته به‌علت دقت پایین تخمین فاصله از گره‌های راهنما و همچنین موقعیت نادقیق گره‌های راهنما (یعنی (x_1,y_1)، (x_2,y_2) و (x_3,y_3))، ممکن است دستگاه فوق جواب نداشته باشد. به همین دلیل است که گره‌های حسگر از بیشتر از سه گره راهنما برای تخمین موقعیت خود استفاده می‌کنند. البته استفاده از بیش از سه گره راهنما نیز با چالش‌های اساسی روبه‌رو خواهد بود و ممکن است یک جواب یکتا برای معادلات وجود نداشته باشد. دقت کنید زمانی که گره‌های شبکه از گره‌های راهنما چند گام فاصله داشته باشند، پیچیدگی این موضوع بیشتر می‌شود. در بخش ۳.۲ راهکارهای موجود برای رفع مشکلات مذکور و تخمین موقعیت گره به‌صورت چندگامه توصیف شده است.

مثلث‌سازی از طریق زاویه: در مثلث‌سازی از طریق زاویهٔ موقعیت با محاسبهٔ زاویهٔ مراجع نسبت به موجودیت و استفاده از روابط ریاضی مثلث به دست می‌آید. در این روش، برای موقعیت‌یابی دوبعدی به حداقل سه مرجع یا دو مرجع و یک فاصله از پیش مشخص (مانند شکل ۳) نیاز است. سیستم ناوبری هوایی یک مثال از انجام موقعیت‌یابی بر پایهٔ زاویهٔ است. در موقعیت‌یابی سه‌بعدی به چهار مرجع و یا سه مرجع و یک فاصلهٔ از پیش مشخص نیاز است، همچنین در موقعیت‌یابی سه‌بعدی مراجع باید در زوایای غیرهم‌سطح قرار داشته باشند.



شکل (۳): موقعیت‌یابی با روش مثلث‌سازی از طریق زاویه در فضای دوبعدی [۱۶]

۳.۲. موقعیت‌یابی در محیط‌های چندگامی

از آنجا که در شبکه‌های حسگر بی‌سیم، تمامی گره‌ها نمی‌توانند با حداقل سه گره راهنما در ارتباط باشند، روش‌هایی نیاز است تا بتوان موقعیت گره‌های حسگر را به‌صورت چندگامی محاسبه کرد. در ادامه، تحقیقات اخیر در این زمینه مورد بررسی قرار می‌گیرد.

در [۱۷] الگوریتم موقعیت‌یابی گام به گام، بردار فاصله ارائه شده است. این الگوریتم یک استاندارد برای موقعیت‌یابی توزیع‌شده است؛ زیرا این الگوریتم بسیار قوی و ساده است و همچنین نیاز به سخت‌افزار اضافه ندارد. این الگوریتم شامل ۳ مرحله است: انتشار بسته‌های راهنما، محاسبهٔ فاصلهٔ تخمینی هر گره (اعم از گره‌های راهنما و گره‌های ناشناخته) تا گره‌های راهنما، تخمین حداقل مربع خطا و محاسبهٔ مختصات گره‌های ناشناخته. الگوریتم گام به گام بردار فاصله در مقیاس بزرگ (LRDV-Hop) یکی از روش‌های بهبودیافتهٔ الگوریتم موقعیت‌یابی گام به گام بردار فاصله است. در این الگوریتم فرض می‌شود که گره‌های راهنما محدودهٔ فرستنده‌های رادیویی بلند و قابل تنظیم دارند. الگوریتم LRDV-Hop با کاهش ارتباطات در منطقهٔ تحت پوشش، مصرف انرژی را کاهش می‌دهد. از آنجایی که حداکثر منطقه تحت پوشش گره راهنما مشخص شده است، نیاز به ارسال پخشی گسترده در کل شبکه وجود نخواهد داشت [۱۸]. الگوریتم CDV-Hop برای تخمین مکان حسگرهای ناشناخته از بهینه‌سازی نامحدود به‌جای مثلث‌سازی که در الگوریتم گام به گام بردار فاصله مورد استفاده قرار می‌گیرد، بهره می‌برد. در این روش محدودیت‌ها در واقع اساس ارتباط حسگر ناشناخته و فاصلهٔ بین حسگر و گره راهنماست [۱۹]. در [۲۰ـ۲۳] گسترش‌های دیگری از الگوریتم موقعیت‌یابی گام به گام بردار فاصله ارائه شده است. چن و همکاران در [۲۰] برای تعیین فاصله بین گره‌های ناشناخته و گره‌های راهنما از متوسط اندازهٔ گام به‌جای اندازهٔ گام شخصی هر گره بهره برده‌اند. در ادامه، آن‌ها روشی بهبودیافته برای DV-Hop ارائه کردند که در آن فاصله، گام با تعیین خطای موقعیت‌یابی بهتر تخمین زده می‌شود

[۲۱]. در [۲۲] نویسندگان روش جدیدی برای شبکه‌های حسگر توزیع‌شدهٔ متقارن ارائه دادند که از طریق پیشرفت در اندازهٔ گام، موقعیت‌یابی را بهبود می‌بخشد. بهبود اندازهٔ گام در نتیجهٔ استفاده از خطای افتراقی دامنه[1] مانند ضریب وزن حاصل می‌شود. لی در [۲۳] علاوه بر اصلاح دقیق خطای دامنهٔ گره‌های ناشناخته در اطراف یک گره راهنما، یک روش بهبودیافته برای موقعیت‌یابی را با استفاده از ارزیابی محلی اندازهٔ گام ایجاد کرد.

پیشنهاد الگوریتم ترکیبی HDV-Hop نه‌تنها باعث دستیابی به دقت موقعیت‌یابی بالا بلکه به حداقل رساندن پخش سیل‌آسای پیام‌ها و مصرف انرژی در گره‌های حسگر شد [۲۴]. همچنین گره‌های راهنمای لازم برای موقعیت‌یابی دقیق در الگوریتم HDV-Hop به حداقل رسیده و مکان آن‌ها به محیط شبکه منتقل شده و نیاز به پخش آن‌ها در شبکه نیست. در HDV-Hop، مثلث‌سازی در ایستگاه پایه انجام خواهد شد که می‌تواند محاسبات NLLS[2] و همچنین دریافت و حفظ مطالب در مورد تمام فواصل موجود تا گره‌های راهنما را بهبود بخشد. در این روش با استفاده از تمام فاصله‌های محاسبه‌شده از تمام گره‌های راهنما به یک حسگر گزارش‌دهنده، یک الگوریتم مثلث‌سازی برای پیدا کردن موقعیت این حسگر استفاده می‌شود.

آمورفوس[3] یک الگوریتم بدون محدوده بر اساس روش چندگامی، برای ایجاد یک سیستم منطقی دقیق و هماهنگ برای شبکه‌های حسگر بی‌سیم به‌صورت تصادفی است. در آمورفوس همانند الگوریتم گام به گام بردار فاصلهٔ هر گره تعداد گام تا گره راهنما را به‌صورت پخشی در شبکه می‌یابد. برخلاف الگوریتم گام به گام بردار فاصله که در آن گره راهنما میانگین طول گام را محاسبه می‌کند، آمورفوس یک رویکرد متفاوت برای برآورد متوسط به‌وسیلهٔ یک تک‌گام ایجاد می‌کند. در این روش، فرض می‌شود که تعدادی از گره‌های همسایه از قبل شناخته شده‌اند؛ یعنی مکان آن‌ها مشخص است. در این الگوریتم متوسط طول گام را به‌وسیلهٔ فرمول کلینوک-سیلوستر[4] محاسبه می‌شود. در روش آمورفوس، به‌دلیل استفاده از اندازهٔ همسایه‌های محلی در برآورد فاصلهٔ گام، با افزایش تراکم گره‌ها، کارایی بهبود می‌یابد، درحالی‌که با پراکنده شدن گره‌ها، عملکرد روش به‌شدت تحت‌تأثیر قرار گرفته و کاهش می‌یابد [۲۵].

علاوه بر روش‌های بهینه‌سازی که اشاره شد، در سال‌های اخیر تعدادی از الگوریتم‌های فراابتکاری نیز برای بهبود موقعیت‌یابی مورد استفاده قرار گرفته‌اند که در ادامه به بیان آن‌ها می‌پردازیم.

در [۲۶] نویسندگان الگوریتم GADV-Hop را بر اساس الگوریتم ژنتیک برای یافتن مکان گره‌های ناشناخته پیشنهاد دادند. آن‌ها برای بهبود نرخ همگرایی الگوریتم پیشنهادی جمعیت اولیه را با استفاده از منطقهٔ امکان‌پذیر برای گره، محدود کردند. علاوه بر این، آن‌ها از ادغام ریاضی به‌عنوان عملگر ادغام ژنتیک بهره بردند. در [۵] نویسندگان از ترکیب الگوریتم‌های ژنتیک و ازدحام ذرات برای بهبود تخمین مکان گره‌های حسگر استفاده کرده‌اند. یکی دیگر از الگوریتم‌های فراابتکاری که برای موقعیت‌یابی گره‌های حسگر مورد استفاده قرار گرفته، الگوریتم تکامل تفاضلی[5] است. در [۶] نویسندگان دربارهٔ به‌کارگیری این الگوریتم برای به حداقل رساندن خطای تخمین فاصله بحث کرده‌اند. در ضمن آن‌ها به‌جای استفاده از شمارش گام برای تخمین فاصله ناشناخته، از قدرت سیگنال رادیویی استفاده کرده‌اند. در ادامه، آن‌ها با ترکیب الگوریتم تکامل تفاضلی و الگوریتم کرم شب‌تاب[6] سعی کردند دقت موقعیت‌یابی را بهبود دهند [۷]. در [۴] الگوریتم DECHDV-Hop به‌عنوان ترکیبی از الگوریتم گام به گام بردار فاصله و تکامل تفاضلی برای بهبود موقعیت‌یابی در شبکه‌های حسگر بی‌سیم بدون استفاده از تجهیزات سخت‌افزاری اضافه ارائه شده است. در واقع این روش به‌منظور کاهش هرچه بیشتر خطای موقعیت‌یابی، از الگوریتم تکامل تفاضلی برای تعیین

1 .Range's differential error
2. Non-linear least squares
3. Amorphous
4. Kleinvock-silvester
5. Differential evolution
6. Firefly algorithm

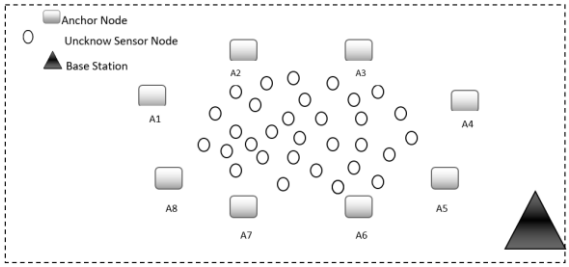
مکان گره‌های ناشناخته با فرموله کردن فرایند تخمین مکان به‌وسیلهٔ یک مسئلهٔ بهینه‌سازی استفاده کرده است. اما دقت داشته باشید که در این روش‌ها، فاصلهٔ بین تمام گره‌ها از یکدیگر به گره مرکزی گزارش شده و الگوریتم موقعیت‌یابی به‌صورت متمرکز اجرا می‌گردد. واضح است که با افزایش تعداد گره‌ها و همچنین پویایی شبکه، پیچیدگی روش‌های متمرکز افزایش می‌یابد. همچنین، الگوریتم‌های متمرکز وفق‌پذیری[1] پایینی داشته و در تغییرات و پویایی فیزیکی که در شبکهٔ حسگر بی‌سیم مطرح است (مانند اضافه شدن گره‌های جدید، نقص تعدادی از گره‌ها و یا تغییر مکان گره‌ها)، کارایی پایینی دارند. لذا برای غلبه بر مشکلات مطرح‌شده، در این مقاله یک روش توزیع‌شده بر اساس الگوریتم ازدحام ذرات برای یافتن مکان گره‌های ناشناخته در شبکهٔ حسگر بی‌سیم ارائه شده است.

۳. روش پژوهش

در این بخش، یک الگوریتم موقعیت‌یابی چندگامی برای شبکه‌های حسگر بی‌سیم با مقیاس بزرگ ارائه می‌شود. الگوریتم پیشنهادی از اصول اساسی الگوریتم بهینه‌سازی ازدحام ذرات بهره‌برداری کرده و اطلاعات مربوط به موقعیت گره‌های حسگر را بر اساس فاصلهٔ گره تا گره‌های راهنما به دست می‌آورد. بدین منظور الگوریتمی در دو بخش پیشنهاد شده است. در بخش اول، حسگر به محاسبهٔ فاصلهٔ مکانی خود تا گره‌های راهنما می‌پردازد و در بخش دوم با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات، موقعیت مکانی گره حسگر ناشناخته محاسبه می‌شود.

۱.۳. فرض‌های مسئله

در این مقاله فرض می‌شود که گره‌های مورد مطالعه در شبکه‌های حسگر بی‌سیم ثابت و در فضایی بزرگ در شبکه پخش شده‌اند. گره‌های موجود در شبکهٔ حسگر به دو دسته تقسیم می‌شوند. تعداد محدودی از گره‌ها مجهز به موقعیت‌یاب جهانی هستند که از مختصات دقیق موقعیت خود اطلاع دارند که به آن‌ها گره‌های راهنما گفته می‌شود. با توجه به محدودیت‌های موجود در گره‌های حسگر (از نظر اقتصادی، صرفه‌جویی در انرژی حسگر و استفاده کمتر از سخت‌افزارهای اضافه)، هرچه تعداد گره‌های راهنما کمتر باشد، مفیدتر خواهد بود. دستهٔ دیگری از گره‌ها که در شبکه وجود دارند، در سطح شبکه پخش شده و به مأموریت‌های خاصی که به آن‌ها محول شده مشغول هستند (مانند پایش دما، فشار، رطوبت و...). این گره‌ها موقعیت خود را با استفاده از گره‌های راهنما تخمین می‌زنند. به این گره‌ها در شبکه، گره‌های حسگر ناشناخته گفته می‌شود. در شکل (۴) نمایی از شبکهٔ مورد تحقیق نشان داده شده است. گره‌هایی که با نام A نمایش داده شده‌اند گره‌های راهنما و دایره‌ها هم حسگرهای ناشناخته هستند.

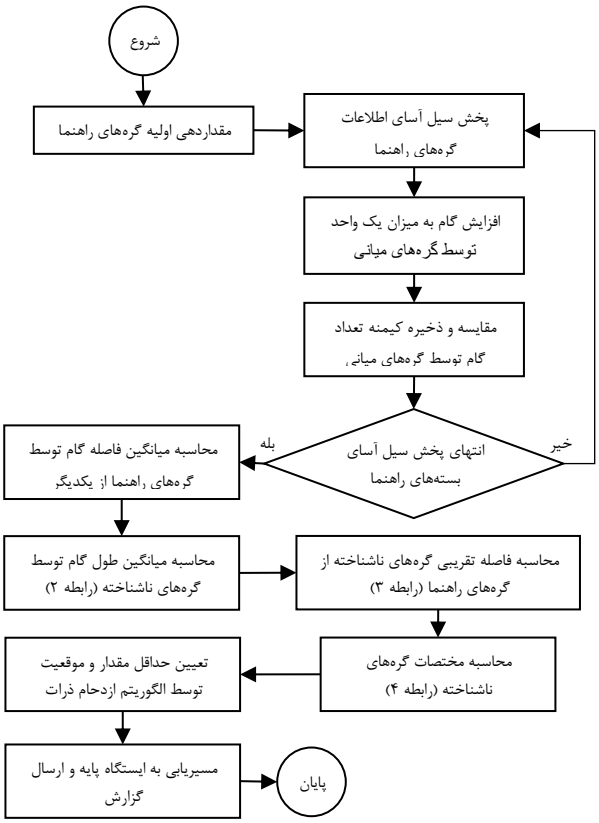


شکل (۴): یک شبکهٔ حسگر شامل گره‌های راهنما و حسگرهای ناشناخته

۲.۳. شرح الگوریتم موقعیت‌یابی پیشنهادی

ایدهٔ اصلی این الگوریتم بر این اساس است که هر گره اطلاعات مکانی خود را با گره‌های همسایه مبادله کرده و فاصلهٔ گره‌های ناشناخته با گره راهنما توسط ضرب متوسط فاصلهٔ گام هر گره در تعداد گام‌ها میان دو گره به دست آمده و در انتها الگوریتم بهینه‌سازی ازدحام ذرات به‌منظور برآورد موقعیت تخمینی دقیق با خطای کم مورد استفاده قرار می‌گیرد. این امر سبب بهبود موقعیت‌یابی در شبکهٔ حسگر بی‌سیم خواهد شد. در این الگوریتم، گره‌های راهنما در محیط شبکه قرار گرفته و گره‌های ناشناخته در سطح شبکه توزیع شده‌اند. همچنین گره‌های راهنما با یکدیگر در ارتباط بوده و در نهایت به ایستگاه پایه متصل هستند. شکل (۵) الگوریتم پیشنهادی را که دارای ۶ مرحله است نشان می‌دهد. در ادامه به جزئیات مراحل این الگوریتم پرداخته می‌شود.

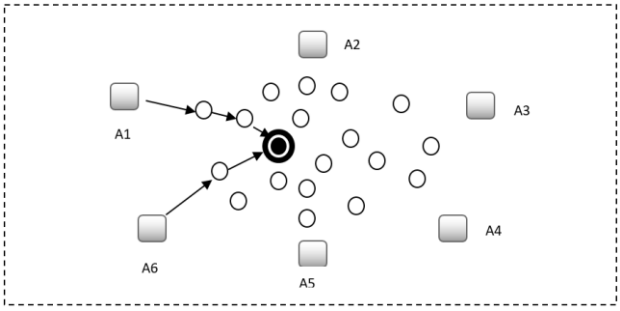
1. Adaptability



شکل (۵): نمودار جریان الگوریتم پیشنهادی

مرحلهٔ ۱. مقداردهی اولیهٔ گره‌های راهنما: با استفاده از الگوریتم موقعیت‌یابی در شبکهٔ حسگر با مقیاس بزرگ تعداد محدودی از گره‌های راهنما در محیط شبکه مستقر می‌شوند و با استفاده از موقعیت‌یاب جهانی مختصات خود را تعیین می‌کنند و مقداردهی اولیه می‌شوند. در این الگوریتم از متغیر A_i استفاده شده که i شماره سریال هر گره راهنماست.

مرحلهٔ ۲. شمارش گام از گره راهنما تا گره‌های ناشناخته: در این الگوریتم هر گره راهنما مقدار اولیهٔ تعداد گام خود را صفر قرار می‌دهد. سپس مختصات خود و همچنین تعداد گام خود را به‌صورت سیل‌آسا در شبکه پخش می‌کند. هر گره میانی تعداد گام را یک واحد افزایش داده و سپس دوباره اقدام به پخش بسته می‌کند. گره‌های راهنما از تعداد گام‌ها به‌منظور برآورد میانگین فاصلهٔ گام به دیگر گره‌های راهنما استفاده می‌کنند. هر گره حسگر میانی حداقل مقدار گام مربوط به گره‌های راهنما را نگهداری کرده و مقدار گام‌های دیگر از گره‌های راهنمای دیگر را نادیده می‌گیرد. حسگر ناشناخته آدرس همسایه‌ای را که حداقل مقدار را از او دریافت کرده است حفظ می‌کند. حسگر از این حداقل مقدار به نام کمینهٔ سراسری[1] بهره می‌برد. حسگر، همسایه‌ای را که مقدار کمینه را از آن تحویل گرفته، همسایهٔ کمینهٔ سراسری[2] و گره راهنمایی را که این مقدار مربوط به اوست، والد راهنما[3] می‌نامد. شکل (۶) یک مثال از الگوریتم پیشنهادی در فاز شمارش گام را نشان می‌دهد. در این شکل، گره مشخص‌شده با گره‌های راهنما دو گام فاصله دارد.



شکل (۶): فاصلهٔ دو گامی گره ناشناخته با گره‌های راهنما

مرحله ۳. محاسبهٔ میانگین طول گام فاصلهٔ واقعی میان گره ناشناخته تا گره راهنما: در این مرحله، هر گره راهنما با استفاده از فرمول زیر، میانگین طول گام خود را محاسبه می‌کند [۲۴]

$$\overline{HL_i} = \frac{\sum_{j=1}^{m} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{j=1}^{m} HC_{i,j}} \qquad (۲)$$

در این فرمول، (x_i, y_i) مختصات گره راهنما i ام، (x_j, y_j) مختصات گره راهنما j ام، m هم تعداد کل گره‌های راهنما و $HC_{i,j}$ تعداد گام‌های بین گره‌های i و j است.

پس از محاسبهٔ میانگین طول گام هر گره راهنما در شبکه، این مقدار به‌صورت سیل‌آسا انتشار و در اختیار تمام گره‌های حسگر قرار می‌گیرد. گره‌های ناشناخته، اولین میانگین طول گامی را که دریافت کردند نگهداری می‌کنند و آن را در اختیار دیگر گره‌های همسایه نیز قرار می‌دهند.

هر گره حسگر ناشناخته پس از دریافت میانگین طول گام،

1. Global minimum
2. Global minimum neighbor
3. Parent anchor

فاصلهٔ تقریبی خود تا گره‌های راهنما را توسط ضرب کردن متوسط فاصلهٔ گام در تعداد گام‌ها تا گره راهنما طبق فرمول زیر محاسبه می‌کند [۲۴]:

$$d_j = \overline{HL_j} \times HC_{i,j} \quad (۳)$$

که d_j فاصلهٔ گره ناشناخته i از گره راهنمای j است.

مرحلهٔ ۴. محاسبهٔ مختصات گره‌های ناشناخته: در این مرحله، گره‌های ناشناخته فاصلهٔ خود تا گره راهنما را می‌دانند و با استفاده از حداقل سه فاصله تا گره‌های راهنما و همچنین استفاده از الگوریتم بهینه‌سازی ازدحام ذرات موقعیت دقیق گره ناشناخته محاسبه می‌شود. برای محاسبهٔ موقعیت یک گره ناشناخته از حل مدل بهینه‌سازی بدون محدودیت زیر استفاده می‌شود [۲۴]:

$$\min_{x_u,y_u} \sum_{i=1}^{m} ((x_i - x_u)^2 + (y_i - y_u)^2 - {d_i}^2)^2 \quad (۴)$$

که (x_u, y_u) مختصات گره ناشناخته و d_i فاصلهٔ گره ناشناخته تا گره راهنما و m تعداد کل گره‌های راهنماست.

مرحلهٔ ۵. تعیین حداقل مقدار و موقعیت با استفاده از الگوریتم بهینه‌سازی ازدحام ذرات: برای تعیین موقعیت دقیق یک گره ناشناخته از الگوریتم بهینه‌سازی ازدحام ذرات استفاده می‌کنیم (الگوریتم ۱). برای این منظور، به‌جای مختصات هر گره ناشناخته که قصد تعیین موقعیت خود را دارد، یک ذره در نظر می‌گیریم. بدیهی است هرچه تعداد این ذرات در شبکه بیشتر باشد، جواب بهتری به دست خواهد آمد.

الگوریتم (۱): تعیین موقعیت دقیق گره‌های ناشناخته با استفاده از الگوریتم ازدحام ذرات

```
for each particle p in P
   X_p ← random 2D position;
   V_p ← random velocity;
end for
do
   for each particle p in P
      F_p ← calculate fitness of p according to eq. 5;
      if F_p < F_pbest then
         X_pbest ← X_p;
      end if
   end for
   X_gbest ← best X_p;
   for each particle p in P
      Update the velocity and position of p according to eq. 6;
   end for
while max_iteration or min_error is not attained
```

گام ۱. ذرات در نظر گرفته‌شده در شبکه را مقداردهی اولیه می‌کنیم و به هر ذره یک مختصات تصادفی داده می‌شود و با اجرای چندباره الگوریتم این ذرات به‌روزرسانی می‌شوند.

گام ۲. برازندگی ذرهٔ p که دارای موقعیت (x_p, y_p) است با استفاده از فرمول زیر محاسبه می‌شود:

$$F_p = \sum_{i=1}^{m} ((x_i - x_p)^2 + (y_i - y_p)^2 - {d_i}^2)^2 \quad (۵)$$

گام ۳. برازندگی ذره p (یعنی F_p) با بهترین برازندگی که ذره قبلاً در آن قرار گرفته است (F_{pbest}) مقایسه می‌شود؛ اگر موقعیت فعلی بهتر از موقعیتی بود که قبلاً آن را تجربه کرده است ($F_p < F_{pbest}$)، موقعیت فعلی به‌عنوان بهترین موقعیت محلی در نظر گرفته خواهد شد.

گام ۴. در الگوریتم بهینه‌سازی ازدحام ذرات موقعیت بهینهٔ محلی هر ذره (F_{pbest}) با موقعیت بهینه‌ای که از جمع ذرات به دست آمده یا به‌عبارتی بهترین موقعیت جمعی ذرات F_{gbest} مقایسه می‌شود؛ اگر موقعیت فعلی ذره از موقعیت بهینه جمعی بهتر بود ($F_{pbest} < F_{gbest}$)، موقعیت فعلی به‌عنوان موقعیت بهینهٔ جمعی در نظر گرفته خواهد شد و جایگزین موقعیت قبلی خواهد شد.

گام ۵. با استفاده از فرمول زیر سرعت و موقعیت هر ذره در هر مرحله از اجرای الگوریتم به‌روز می‌شود و با استفاده از فرمول زیر، موقعیت هر ذره در هر مرحله از الگوریتم محاسبه می‌شود.

$$\begin{aligned} V_i[t+1] &= wV_i[t] + c_1R_1 + c_2R_2 \\ R_1 &= \text{Rand}(X_{ipbest}[t] - X_i[t]) \\ R_2 &= \text{Rand}(X_{igbest}[t] - X_i[t]) \\ X_i[t+1] &= X_i[t] + V_i[t+1] \end{aligned} \quad (۶)$$

که در آن، w و c_1 و c_2 مقادیر ثابت و R_1 و R_2 مقادیر تصادفی، X_{ipbest} ارزش مکانی بهینهٔ محلی (یعنی خاطرهٔ شخصی هر ذره)، X_{igbest} ارزش مکانی بهینهٔ جمعی ذرات (یا همان خاطرهٔ جمعی ذرات) و $X_i[t]$ موقعیت فعلی ذرهٔ i است. در معادلهٔ بالا $X_{ipbest}[t] - X_i[t]$ فاصلهٔ موقعیت فعلی ذره با موقعیت بهینهٔ محلی در شبکه و $X_{igbest}[t] - X_i[t]$ فاصلهٔ موقعیت فعلی ذره با موقعیت بهینهٔ جمعی ذرات است. دقت داشته باشید که سرعت V به‌وسیلهٔ حداکثر سرعت (V^{max}) محدود می‌شود.

گام ۶. در ادامهٔ الگوریتم، به شرایط توقف الگوریتم در

شبکه می‌پردازیم. اگر موقعیت ذره بهترین موقعیت جمعی ذرات را برآورده کرد (یعنی بهترین موقعیت فعلی ذره از بهترین موقعیت بهینهٔ جمعی ذرات بهتر بود و دیگر هیچ اطلاعات بهینه‌تری از ذرات دیگر دریافت نشد) و یا الگوریتم به تعداد مورد نظر تکرار شد، الگوریتم متوقف می‌شود و در غیر این صورت به گام ۲ باز خواهد گشت.

مرحلهٔ ۶. مسیریابی به ایستگاه پایه و ارسال گزارش: یک حسگر ناشناخته داده‌های به‌دست‌آمده را به‌همراه موقعیت تخمین زده‌شده به ایستگاه پایه گزارش می‌دهد. در اولین گزارش حسگر یک بستهٔ جدید را که دارای دو بخش است، می‌سازد: بخش اول گزارش پیام و بخش دوم مختصات و آدرس گره حسگر.

۴. ارزیابی روش پیشنهادی

در این مقاله، روشی برای تعیین موقعیت گره‌های حسگر ناشناخته با استفاده از الگوریتم ازدحام ذرات توصیف شد. در این بخش به بررسی کارایی الگوریتم پیشنهادی و مقایسهٔ آن با کارهای گذشته پرداخته خواهد شد. برای شبیه‌سازی، شبکه‌ای متشکل از n گره حسگر در یک محیط دوبعدی (به شکل مربع به ضلع l) در نظر گرفته شده است. در این شبکه فرض شده است که گره‌های حسگر به صورت ثابت در محیط پراکنده شده‌اند. گره‌های راهنما در محیط شبکه مستقر و دارای مختصات دقیق هستند و گره‌های حسگر ناشناخته در یک منطقهٔ مسطح با مختصات تصادفی قرار گرفته‌اند. برای ارزیابی کارایی الگوریتم موقعیت‌یابی پیشنهادی (که در شبیه‌سازی‌ها KhaDost نامیده شده است)، آن را با الگوریتم‌های CDV-Hop [۱۹]، آمورفوس [۲۵] و DV-Hop بهبودیافته [۵] مقایسه کرده‌ایم. واضح است روش‌های توزیع‌شده در مقابل روش‌های متمرکز در شرایط پویایی شبکه (مانند وقوع خطا و خرابی گره‌ها، جابه‌جایی گره‌ها و تغییر تخمین فاصله) عملکرد بهتری خواهند داشت. به‌خصوص در شبکه‌های بزرگ‌مقیاس با تعداد گره زیاد الگوریتم‌های توزیع‌شده کارایی مناسب‌تری ارائه می‌کنند.

الگوریتم پیشنهادی و الگوریتم‌های مورد مقایسه با استفاده از زبان برنامه‌نویسی پایتون نسخه ۲,۷ بر روی یک کامپیوتر با پردازنده Intel i5 2.53GHz و ۴ گیگابایت حافظه شبیه‌سازی و ارزیابی شده‌اند. در ضمن برای پیاده‌سازی الگوریتم ازدحام ذرات از بسته PSOPy استفاده شده است. توجه کنید که برای ایجاد یک مقایسهٔ مناسب تمام پارامترهای شبیه‌سازی مشابه مراجع اصلی روش‌های ذکرشده، در نظر گرفته شده‌اند. به‌منظور بررسی تأثیر عوامل مختلف، تعداد گره‌های حسگر بین ۱۰۰ تا ۲۵۰۰ گره، تعداد گره راهنما بین ۱۰ تا ۲۵۰ گره و مساحت شبکه بین $۱۰^۶$ تا ۲۰×۱۰^۶ مترمربع در نظر گرفته شده است.

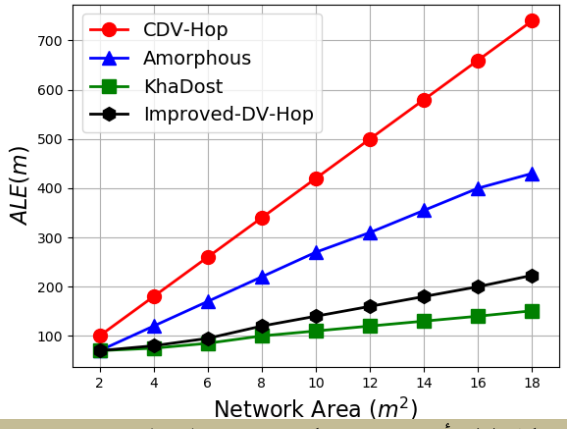
برای ارزیابی و تحلیل روش پیشنهادی از معیار متوسط خطای موقعیت‌یابی[1] (ALE) (برحسب متر) استفاده شده است که با استفاده از رابطهٔ زیر محاسبه می‌شود [۲۴].

$$ALE = \frac{\sum_{i=0}^{n} dis(P_i^{est}, P_i^{act})}{n} \quad (۷)$$

که P_i^{est} برآورد موقعیت حسگر i با استفاده از الگوریتم موقعیت‌یابی پیشنهادی و P_i^{act} موقعیت واقعی حسگر i است. $dis(P_i^{est}, P_i^{act})$ فاصلهٔ اقلیدسی بین موقعیت واقعی حسگر i و موقعیت برآورد شده و n تعداد حسگرها در شبکه است.

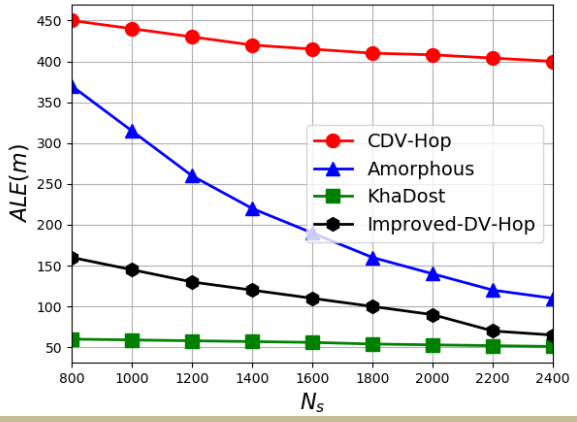
شکل (۷) متوسط خطای اندازه‌گیری بر مبنای مساحت شبکه را نشان می‌دهد. با افزایش مساحت شبکه، میانگین خطای موقعیت‌یابی افزایش می‌یابد. این افزایش به‌دلیل افزایش خطای اندازه‌گیری فاصله بین گره‌هاست. از سوی دیگر، الگوریتم پیشنهادی نسبت به سه الگوریتم دیگر، عملکرد بهتری داشته و با افزایش مساحت شبکه، افزایش خطای کمتری داشته است. در نتیجه از نظر موقعیت‌یابی، الگوریتم پیشنهادی نسبت به الگوریتم دیگر در شبکه‌های با مقیاس بزرگ، مکان دقیق‌تری را تخمین می‌زند. نتایج همچنین نشان می‌دهد که الگوریتم DV-Hop بهبودیافته زمانی که سطح شبکه کوچک است، بهتر از CDV-Hop و آمورفوس عمل می‌کند.

1. Average Localization Error



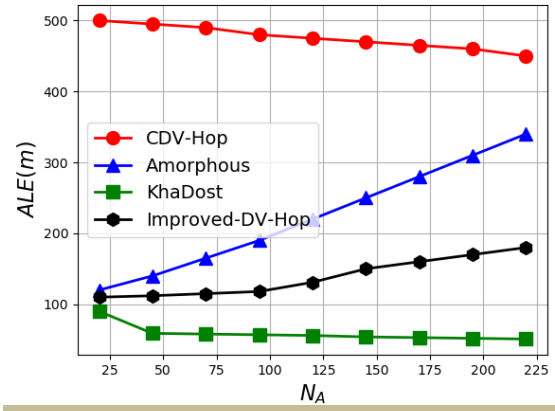
شکل (۷): تأثیر مساحت شبکه روی متوسط خطای موقعیت‌یابی

در شکل (۸)، چهار الگوریتم CDV-Hop، آمورفوس، DV-Hop بهبودیافته و الگوریتم پیشنهادی بر مبنای تأثیر تعداد گره حسگر در میانگین خطای موقعیت‌یابی مورد ارزیابی و بررسی قرار گرفته‌اند. در این شبیه‌سازی، گره‌های راهنما در شبکه‌ای با مساحت ۷ کیلومتر مربع به تعداد ۱۵۰ گره و به‌صورت ثابت در نظر گرفته شده‌اند. تعداد گره‌های حسگر از ۷۰۰ تا ۲۵۰۰ گره افزایش یافته و در هر مرحله از اجرای الگوریتم میانگین خطای موقعیت‌یابی اندازه‌گیری شده است. نتایج این شکل نشان می‌دهد که متوسط خطای موقعیت‌یابی الگوریتم پیشنهادی با افزایش تعداد گره‌های حسگر نه‌تنها افزایش نیافته، بلکه الگوریتم عملکرد بهتری را نیز به دست آورده و خطای آن از ۷۰ به کمتر از ۵۰ متر رسیده است.



شکل (۸): متوسط خطای موقعیت‌یابی بر مبنای تعداد گره حسگر

در شکل (۹) با در نظر گرفتن تعداد ۱۵۰۰ گره حسگر در مساحت ۷ کیلومتر مربع، تعداد گره‌های راهنما از ۱۰ تا ۲۵۰ گره افزایش یافته و نتایج مربوط به میانگین خطای موقعیت‌یابی برای الگوریتم‌های مورد مقایسه گزارش شده است. واضح است که میانگین خطای موقعیت‌یابی الگوریتم پیشنهادی با افزایش تعداد گره‌های راهنما تا ۱۰۰ گره، بهبود یافته است، درحالی‌که با توجه به نتایج می‌توان بیان کرد که این الگوریتم با افزایش تعداد گره راهنما چندان دچار تغییر در موقعیت‌یابی نمی‌شود. این امر در حالی رخ می‌دهد که میانگین خطای موقعیت‌یابی در الگوریتم آمورفوس و DV-Hop بهبود یافته با افزایش تعداد گره‌های راهنما افزایش یافته است.



شکل (۹): متوسط خطای موقعیت‌یابی بر مبنای تعداد گره راهنما

۵. نتیجه‌گیری

پیدا کردن موقعیت حسگری که در مجاورت آن رویدادی رخ داده، بخش ذاتی و جدایی‌ناپذیر از هر شبکهٔ حسگر بی‌سیم بوده و نشان‌دهندهٔ یک چالش عمده است. ولی استفاده از موقعیت‌یاب جهانی در گره‌های حسگر، موجب بزرگ‌تر شدن حسگر و همچنین بالا رفتن بار مالی در گره حسگر می‌شود. به همین دلیل الگوریتم‌های موقعیت‌یابی مورد توجه قرار گرفته‌اند. در این مقاله با استفاده از الگوریتم ازدحام ذرات، روشی برای موقعیت‌یابی گره‌های ناشناخته شبکه ارائه شده است. نتایج شبیه‌سازی نشان می‌دهند که الگوریتم پیشنهادی می‌تواند از حداقل گره‌های راهنما نتایج مطلوبی به دست آورد، که این امر به‌نوبهٔ خود باعث کاهش هزینه و کاهش استفاده از سخت‌افزارهای اضافه در گره‌های حسگر می‌شود. علاوه بر این، استفاده از الگوریتم موقعیت‌یابی پیشنهادی در مقیاس بزرگ بدون توجه به مساحت شبکه سبب بهبود

موقعیت‌یابی نسبت به دیگر الگوریتم‌های موقعیت‌یابی در شبکه می‌شود. به هر حال، اجرای الگوریتم ازدحام ذرات در گره‌های حسگر ارزان قیمت و کم‌حافظه و همچنین رد و بدل کردن اطلاعات با گره‌های همسایه در شبکه‌های حسگر بی‌سیم از مشکلات روش پیشنهادی است. محققان به‌عنوان کار آتی می‌توانند از ماشین یادگیر برای تخمین مناسب‌تر موقعیت گره‌های حسگر استفاده کنند.

مراجع

[1] Han G., Xu H., Duong T.Q., Jiang J. and Hara T., "Localization algorithms of wireless sensor networks: a survey", *Telecommunication Systems*, vol. 52, no. 4, pp. 2419-2436, 2013.

[2] Karl H. and Willig A., Protocols and architectures for wireless sensor networks. John Wiley & Sons, 2007.

[3] Wang L. and Xu Q., "GPS-free localization algorithm for wireless sensor networks," Sensors, vol. 10, no. 6, pp. 5899-5926, 2010.

[4] Cui L., Xu C., Li G., Ming Z., Feng Y. and Lu N., "A high accurate localization algorithm with DV-Hop and differential evolution for wireless sensor network", Applied Soft Computing, vol. 68, pp. 39-52, 2018.

[5] Mehrabi M., Taheri H. and Taghdiri P., "An improved DV-Hop localization algorithm based on evolutionary algorithms", Telecommun System, vol. 64, pp. 1-9, 2017.

[6] Harikrishnan R., Kumar V.J.S. and Ponmalar P.S., "Differential evolution approach for localization in wireless sensor networks", IEEE International Conference on Computational Intelligence and Computing Research, Coimbatore, pp. 1-4, 2014.

[7] Harikrishnan R., Kumar V.J.S. and Ponmalar P.S., "A Comparative Analysis of Intelligent Algorithms for Localization in Wireless Sensor Networks", Wireless Pers. Commun., vol. 87, no. 3, pp. 1057-1069, 2016.

[8] Sahoo B.M., Amgoth T. and Pandey H.M., "Particle swarm optimization based energy efficient clustering and sink mobility in heterogeneous wireless sensor network", Ad Hoc Networks, vol. 106, pp. 102237, 2020.

[9] Zhang Y., "Coverage Optimization and Simulation of Wireless Sensor Networks Based on Particle Swarm Optimization", International Journal of Wireless Information Networks, vol. 27, no. 2, pp. 307-316, 2020.

[10] Savvides A., Han C.C. and Strivastava M.B., "Dynamic fine-grained localization in ad-hoc networks of sensors", in Proceedings of the 7th annual international conference on Mobile computing and networking, ACM, pp. 166-179, 2001.

[11] Lee J., Chung W. and Kim E., "A new range-free localization method using quadratic programming", Computer Communications, vol. 34, no. 8, pp. 998-1010, 2011.

[12] Iniewski K., Optical, acoustic, magnetic, and mechanical sensor technologies. CRC Press, 2012.

[13] Wessels A., Wang X., Laur R., and Lang W., "Dynamic indoor localization using multilateration with RSSI in wireless sensor networks for transport logistics", Procedia Engineering, vol. 5, pp. 220-223, 2010.

[14] Huang J., Xue Y. and Yang L., "An efficient closed-form solution for joint synchronization and localization using TOA", Future Generation Computer Systems, vol. 29, no. 3, pp. 776-781, 2013.

[15] Nasipuri A. and Li K., "A directionality based location discovery scheme for wireless sensor networks", in Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, ACM, pp. 105-111, 2002.

[16] Karl H. and Willig A., Protocols and Architectures for Wireless Sensor Networks. Wiley, ISBN 978-0-470-09510-2, pp. 1-497, 2005.

[17] Niculescu D. and Nath B., "DV based positioning in ad hoc networks," Telecommunication Systems, vol. 22, no. 1-4, pp. 267-280, 2003.

[18] Sun Z., Yu R. and Mei S., "A power-aware and range-free localization algorithm for sensor networks", in Communications, 2006. APCC'06. Asia-Pacific Conference on, IEEE, pp. 1-5, 2006.

[19] Ji W.-W. and Liu Z., "An improvement of DV-Hop algorithm in wireless sensor networks", in Wireless Communications, Networking and Mobile Computing, 2006. WiCOM 2006. International Conference on, IEEE, pp. 1-4, 2006.

[20] Chen H., Sezaki K., Deng P. and So H.C., "An improved DV-Hop localization algorithm for wireless sensor networks", in Proceedings of the 3rd IEEE Conference on Industrial Electronics and Applications, Singapore, pp. 1557–61, 2008.

[21] Chen H., Sezaki K., Deng P. and So H.C. "An improved DV-Hop localization algorithm with reduced node location error for WSNs", IEICE Tran. Fund .Electron. Commun. Comput. Sci., vol. 91, no. 8, pp. 2232–36, 2008.

[22] Hou S., Zhou X. and Liu X., "A novel DV-Hop localization algorithm for asymmetry distributed WSNs", in Proceedings of the 3rd IEEE Conference on Computer Science and Information Technology, Chengdu, China, pp. 243–8, 2010.

[23] Li Y., "Improved DV-Hop localization algorithm based on local estimating and dynamic correction in location for WSNs", Int. J. Digit. Content Technol. Appl., vol. 5, no. 8, pp. 196–202, 2011.

[24] Safa H., "A novel localization algorithm for large scale wireless sensor networks", *Computer Communications*, vol. 45, pp. 32-46, 2014.

[25] Nagpal R., Shrobe H. and Bachrach J., "Organizing a global coordinate system from local information on an ad hoc sensor network", in Information Processing in Sensor networks, Springer, pp. 333-348, 2003.

[26] Peng B. and Lei L., "An improved localization algorithm based on genetic algorithm in wireless sensor networks", Cogn. Neurodynamics, vol. 9, no. 2, pp. 249–256, 2015.