نشریه علمی پژوهشی مهندسی فناوری اطلاعات مکانی

سال پنجم، شماره دوم، تابستان ۱۳۹۶

Vol.5, No.2, Summer 2017

۱۴۱-۱۶۲

شاخص‌گذاری موقعیت جاری و گذشته اشیاء متحرک درمجموعه داده‌های بزرگ مقیاس

محمدرضا عباسی فرد[۱]، حسن نادری[۲*]

۱- دانشجوی دکتری دانشکده کامپیوتر، دانشگاه علم و صنعت ایران

۲- استادیار گروه نرم افزار دانشکده کامپیوتر، دانشگاه علم و صنعت ایران

تاریخ دریافت مقاله: ۱۳۹۵/۰۹/۰۵ تاریخ پذیرش مقاله: ۱۳۹۵/۱۲/۰۴

چکیده

با افزایش روز افزون سیستم‌های هوشمند حمل و نقل و سرویس‌های مبتنی برمکان که از داده‌های مکانی-زمانی بهره می‌برند، نیاز به تکنیک‌ها شاخص‌گذاری جدید برای این داده‌ها افزایش یافته است. این داده‌ها مربوط به زمان گذشته، حال یا آینده اشیاء متحرک هستند که شاخص‌های موجود عموماً امکان شاخص‌گذاری یکی از این سه زمان را دارند. داشتن شاخصی یکپارچه برای شاخص‌گذاری همزمان انواع داده‌ها و همچنین پاسخگویی به انواع پرس‌وجوها، یکی از چالش‌های این نوع داده‌ها می‌باشد که امکان کاهش فاصله زمانی بروزرسانی شاخص‌ها را داشته باشد. در این مقاله روشی یکپارچه با نام PCPI ارائه شده است که علاوه بر امکان ساخت و نگهداری شاخص‌های داده‌های مکانی-زمانی در زمان گذشته و حال، به‌صورت همزمان به ترتیب در دیسک و حافظه اصلی، توانایی پاسخ به انواع پرس‌وجوهای مکانی-زمانی را دارد. با استفاده از این تکنیک، از یک مجموعه منبع مشترک برای پردازش و ساخت شاخص‌ها در دو زمان مختلف استفاده شده و اختلاف فاز زمانی بین دو شاخص را کاهش داده است. در این روش از دو داده ساختار به‌صورت یکپارچه استفاده شده است. یک داده ساختار در حافظه اصلی برای شاخص‌گذاری و نگهداری موقعیت جاری اشیاء متحرک، و داده ساختار دوم برروی دیسک برای داده‌های خط‌سیر اشیاء متحرک که دارای حجم بالایی بوده و امکان نگهداری در حافظه اصلی را ندارند. علاوه بر این، در PCPI از روش‌های تطبیق برنقشه برای حذف اختلالات موجود در داده‌های دریافتی از اشیاء متحرک مانند اختلال حالت ساکن استفاده شده است که باعث افزایش دقت پرس‌وجوها می‌شود. همچنین تاثیر روش‌های کاهش داده بر افزایش سرعت شاخص‌گذاری، پردازش پرس‌وجو و کاهش اندازه شاخص‌ها در دیسک در مجموعه داده‌های بزرگ، در آزمایش‌ها موردبررسی قرار گرفت. نتایج و مقایسه‌های انجام شده در آزمایش‌های تجربی، نشان از کارایی بالای این ساختار شاخص‌گذاری دارد.

کلیدواژه‌ها: ماشین بردار پشتیبان، روش‌های دسته‌جمعی، فضای ویژگی، خود راه‌انداز تصادفی، تلفیق

*نویسنده مکاتبه کننده: تهران، کارگر شمالی، پردیس فنی دانشگاه تهران، دانشکده مهندسی نقشه‌برداری و فناوری اطلاعات مکانی.

تلفن: ۰۹۳۶۶۱۵۳۶۳۷

Email : jafarim@ut.ac.ir

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]

۱- مقدمه

در سال‌های اخیر پایگاه‌داده‌های مکانی-زمانی با تعداد زیادی از اشیاء متحرک مواجه هستند که تغییرات مکانی پیوسته آنها باعث به وجود آمدن حجم زیادی از داده‌های با ارزش شده است. با توجه به پیشرفت‌های زیادی که درابزارهای آگاه از محل، تکنولوژی‌های موقعیت‌یابی و همچنین ارتباطات بیسیم، به وجود آمده روزانه حجم انبوهی از داده‌های مکانی-زمانی اشیاء متحرک جمع‌آوری و ذخیره می‌شود. به دنبال رشد شدید این نوع داده‌ها، شاهد پیشرفت در سیستم‌های هوشمند حمل و نقل (ITS[1]) و سرویس‌های مبتنی بر مکان (LBS[2]) هستیم [۱، ۲ و ۳]. برای مدیریت این حجم از داده‌ها، استفاده از پایگاه‌داده‌های اشیاء متحرک (MODBs)[3] ضروری است. از مهمترین انواع داده‌های مکانی-زمانی می‌توان به داده‌های حرکتی و خط‌سیر[4] اشیاء متحرک که نمایانگر حرکت آن شیء می‌باشند، اشاره داشت که به‌صورت یک توالی از نقاط سه بعدی (x, y, t) ذخیره می‌شوند: دو بعد مختصات جغرافیایی و یک بعد زمان[۴، ۵، ۶ و ۷]. در کاربردهای مختلف حرکت اشیاء متحرک را می‌توان در سه گروه دسته‌بندی نمود: ۱. حرکت بدون محدودیت، ۲. حرکت محدود شده و ۳. حرکت در شبکه‌های جاده‌ای[۸]. در این مقاله حرکت اشیاء مانند حرکت تاکسی‌ها، اتوبوس‌ها، قطارها و حتی افراد محدود به شبکه‌های جاده‌ای است.

شاخص‌گذاری فراهم آوردن این امکان است که کاربران به‌صورت همروند و کارآمد بتوانند داده‌های مورد نیاز خود را از پایگاه‌داده‌های خیلی بزرگ مانند پایگاه داده‌های مکانی-زمانی، بازیابی نموده و زمان ارسال پرس‌وجوها را کاهش دهند. اهمیت تکنیک‌های شاخص‌گذاری از این جهت افزایش می‌یابد که سرعت رشد حجم داده‌های ذخیره شده به شدت بالا است، به‌طوری‌که بهبود نرخ انتقال داده‌ها بین دیسک و حافظه اصلی متناسب با آن رشد نکرده است [۹]. یکی از چالش‌های موجودکاهش فاصله زمانی آماده‌سازی و بروزرسانی داده در حافظه اصلی و دیسک است.

ساختارهای شاخص‌گذاری موجود، برای شاخص‌گذاری انواع داده‌های مکانی-زمانی اشیاء متحرک مانند داده‌های خط‌سیر اشیاء متحرک (که مربوط به زمان گذشته است)، موقعیت جاری و یا موقعیت آینده (پیش‌بینی شده) اشیاء متحرک استفاده می‌شوند [۱۰]. بیشتر این شاخص‌ها فقط برای شاخص‌گذاری داده‌ها در یکی از سه زمان گذشته، حال یا آینده مورد استفاده قرار می‌گیرند. ساختارهایی که برای شاخص‌گذاری دو زمان مختلف به‌صورت همزمان استفاده می‌شوند، معمولاً برروی یکی از این زمان‌ها تمرکز می‌کنند و یا تاکید برروی پرس‌وجوی خاصی دارند. نکته دیگری که باید به آن توجه داشت این است که برخی از این روش‌ها تاکید بر ساخت و نگهداری شاخص در حافظه اصلی دارند، در حالی‌که برای مجموعه داده‌های بزرگ، ساخت و نگهداری شاخص‌ها در حافظه اصلی مناسب نبوده و باید از روش‌های مبتنی بر دیسک استفاده نمود.

از نوآوری‌های روش ارائه شده در این مقاله ارائه شاخصی‌جدید برای شاخص‌گذاری و نگهداری داده‌های دو زمان گذشته و حال به‌صورت همزمان است که قابلیت پاسخگویی به انواع پرس‌وجوها را دارد. نوآوری دیگر این روش، برخلاف روش‌های موجود، استفاده از الگوریتم‌های تطبیق بر نقشه جهت کاهش خطای موجود در داده‌های ورودی است که باعث افزایش دقت پردازش‌ها و در نهایت پرس‌وجوها خواهد شد. همچنین در این روش با استفاده از روش‌های کاهش داده امکان شاخص‌گذاری و نگهداری مجموعه داده‌های بزرگتری را فراهم می‌آورد.

[1] Intelligent Transportation Systems
[2] Location Based Services
[3] Moving Object DataBases
[4] Trajectory

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

در این مقاله ساختار شاخص‌گذاری با نام PCPI[1] برای داده‌های مکانی-زمانی اشیاء متحرک پیشنهاد شده است. در این روش، داده‌های موقعیت جاری اشیاء متحرک در حافظه اصلی و با اصلاح و بهبود ساختار اولیه IMORS [۱۱] شاخص‌گذاری و نگهداری می‌شود. همچنین شاخص‌های داده‌های خط‌سیر برروی دیسک ایجاد و نگهداری می‌شود. در PCPI، از یک مجموعه منبع پردازشی، استفاده شده و با استفاده از یک سیستم یکپارچه، مشکلات ارتباط دو ساختار شاخص‌گذاری را کاهش داده است. همچنین حجم بالای داده‌ها و اختلالات زیاد موجود در این داده‌ها باعث کاهش کارایی و کارآمدی اجرای پرس‌وجوها شده است. تکنیک‌های رفع نویز، تطبیق برنقشه[۱۲، ۱۳ و ۱۴] و کاهش داده[۱۵، ۱۶ و ۱۷] به عنوان پیش‌پردازش شاخص‌گذاری در این مقاله پیشنهاد شده و نحوه اعمال و تاثیرگذاری آن بر شاخص‌گذاری و پردازش پرس‌وجو بررسی شده است. با استفاده از این ساختار امکان پاسخ به عنوان پرس‌وجوهای برنامه‌ها و سیستم‌های کاربردی فرآهم شده است. در ادامه ابتدا خلاصه‌ای از کارهای انجام شده و مرتبط بررسی می‌شوند. سپس در بخش سوم ساختار شاخص‌گذاری PCPI مطرح و همچنین نحوه انجام پیش‌پردازش بررسی می‌گردد. در بخش چهارم نتایج حاصل از آزمایش‌های انجام شده مورد ارزیابی و بحث قرار می‌گیرد. در نهایت در بخش پنجم جمع‌بندی از روش پیشنهادی و نتایج حاصله ارائه می‌شود.

۲- کارهای مرتبط

در پایگاه داده‌های اشیاء متحرک، ایده‌آل این است که ساختارهای شاخص گذاری کارآمدی استفاده شود که علاوه بر قابلیت پشتیبانی از مجموعه دادهای بزرگ که شامل تعداد بالای اشیاء متحرک و تعداد زیادی خط‌سیر است، موقعیت اشیاء متحرک در هر سه زمان گذشته، حال و آینده را به‌صورت بهینه پشتیبانی نمایند. در حال حاضر ساختارهای شاخص گذاری متعددی ارائه شده است، در دسته‌بندی ارائه شده در [۱۸ و ۱۹]، این ساختارها به چهار دسته تقسیم شده‌اند که پایه بیشتر این روش‌ها R-tree [۲۰] و خانواده آن می‌باشند: ۱. شاخص‌گذاری داده‌های زمان گذشته[2]، ۲. شاخص‌گذاری داده‌های زمان حال[3]، ۳. شاخص‌گذاری داده‌های آینده[4] و ۴. شاخص‌گذاری داده‌های در همه زمان‌ها (روش عمومی.)

برخی از ساختارهای شاخص‌گذاری مانند 3DR-tree[۲۱ و ۲۲]، TB-tree [۲۳]، STRtree [۲۳]، MV3R-tree[۲۱] و SETI[۷ و ۲۴] برای شاخص‌گذاری و نگهداری داده‌های زمان گذشته اشیاء متحرک مورد استفاده قرار می‌گیرند و بر پرس‌وجوهای مبتنی بر خط‌سیر تمرکز دارند. برای شاخص‌گذاری موقعیت جاری اشیاء متحرک نیز می‌توان از ساختارهایی مانند LUGrid[۲۵]، RUM-tree[۲۶] و IMORS[۱۱] استفاده نمود که با توجه به تغییرات زیاد موقعیت جاری اشیاء، چالش اصلی این ساختارها حجم بالای بروزرسانی در شاخص‌های ایجاد شده است. روش‌های شاخص‌گذاری نظیر SEB-Tree[۲۷]، SETI[۷ و ۲۴] و 2+3-RTree[۲۲] قابلیت شاخص‌گذاری داده‌ها و پاسخگویی به پرس‌وجوهای مربوط به زمان حال و گذشته را دارند و روش‌های شاخص‌گذاری مانند TPR-Tree[۲۰] و TPR*-Tree[۲۸] داده‌های زمان حال و آینده را شاخص‌گذاری و نگهداری می‌کند و به پرس‌وجوهای مرتبط پاسخ می‌دهند. برخی از روش‌ها برای شاخص‌گذاری پاسخ به پرس‌وجوهای هر سه زمان گذشته، حال و آینده مناسب هستند. ساختار شاخص‌گذاری PCFI+[۲۹] که با ترکیب

[1] Past and Current Position Indexing

[2] Indexing the past data
[3] Indexing the current data
[4] Indexing the future data

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

دو شاخص SETI[۷ و ۲۴] برروی دیسک و TPR*-Tree[۲۸] برروی حافظه اصلی، ایجاد شده است. روش PPFI که شامل 2D-R*-tree، تعدادی 1D-R*-tree و یک داده ساختار درهم‌سازی می‌باشد، علاوه بر شاخص‌گذاری داده‌ها و پاسخ به پرس‌وجوهای خط‌سیر اشیاء در شبکه‌های جاده‌ای، مکانیسم مناسبی جهت بروزرسانی‌های موقعیت جاری اشیاء متحرک و پاسخ به پرس‌وجوهای زمان حال ارائه داده و از پرس‌وجوهای پیش‌بینی خط‌سیر نیز پشتیبانی می‌کند[۸]. ساختار BB^x برای شاخص‌گذاری موقعیت اشیاء متحرک در هر لحظه با استفاده از یک تابع خطی از زمان استفاده می‌شود و برای ذخیره موقعیت اشیاء متحرک از مجموعه‌ای از B^+-Tree ها استفاده می‌کند[۱۰]. همچنین در منبع [۹] ساختار شاخص‌گذاری RPPF-tree ارائه شده که با استفاده از ساختار TPR-Tree شاخص‌گذاری موقعیت اشیاء متحرک رادر سه زمان گذشته و حال و آینده انجام می‌دهد.

۳- روش پیشنهادی

حجم بالای داده‌های مکانی-زمانی و همچنین تغییرات فراوان موقعیت اشیاء متحرک در طول زمان، از چالش‌های روش‌های شاخص‌گذاری می‌باشد. این تغییرات نیاز به حجم بالایی از بروزرسانی‌ها دارند. در نقطه مقابل، تغییرات در داده‌های خط‌سیر اشیاء متحرک، بسیار اندک است. برای شاخص‌گذاری و پاسخ به پرس‌وجوهای این نوع داده‌ها نیاز به منابع پردازشی مستقل ازهم می‌باشد. از چالش‌های دیگر در ترکیب دو ساختار شاخص‌گذاری زمان حال و گذشته، اختلاف سرعت دو ساختار می‌باشد. روش شاخص‌گذاری PCPI که در این مقاله پیشنهاد شده است، توانایی شاخص‌گذاری و نگهداری داده‌های مکانی-زمانی اشیاء متحرک را در زمان گذشته و حال، به‌صورت همزمان دارد. در روشPCPI، داده‌های موقعیت جاری در حافظه اصلی شاخص‌گذاری و نگهداری می‌شود ولی داده‌های خط‌سیر به دلیل حجم بالا، برروی دیسک شاخص‌گذاری و نگهداری می‌شود. ارتباط این دوبخش به کمک یک بافر به‌گونه‌ای برقرار می‌شود که وقتی موقعیت اشیاء متحرک تغییر می‌کند، امکان دسترسی سریع به داده‌های خط‌سیر آن شیء فراهم شود و بروزرسانی در خط‌سیر نیز در کمترین زمان انجام شود. در این روش با انجام پیش‌پردازش برروی داده‌ها، کیفیت داده‌ها افزایش یافته و در نتیجه دقت پاسخ پرس‌وجوها بالا می‌رود. در روش PCPI داده‌های خام اشیاء متحرک، به دو روش وارد سیستم می‌شوند. بخشی از این داده‌ها که موقعیت جاری اشیاء متحرک را نشان می‌دهند، به‌صورت آنلاین توسط دستگاه‌های آگاه از محل، جمع‌آوری و سپس با استفاده از روش PCPI شاخص‌گذاری می‌گردند. داده‌های خط‌سیر اشیاء متحرک که قبلاً توسط این دستگاه‌ها، جمع‌آوری شده و در مجموعه داده‌های مختلف نگهداری می‌شوند، به‌صورت آفلاین و دسته‌ای دریافت و سپس شاخص‌گذاری می‌گردند.

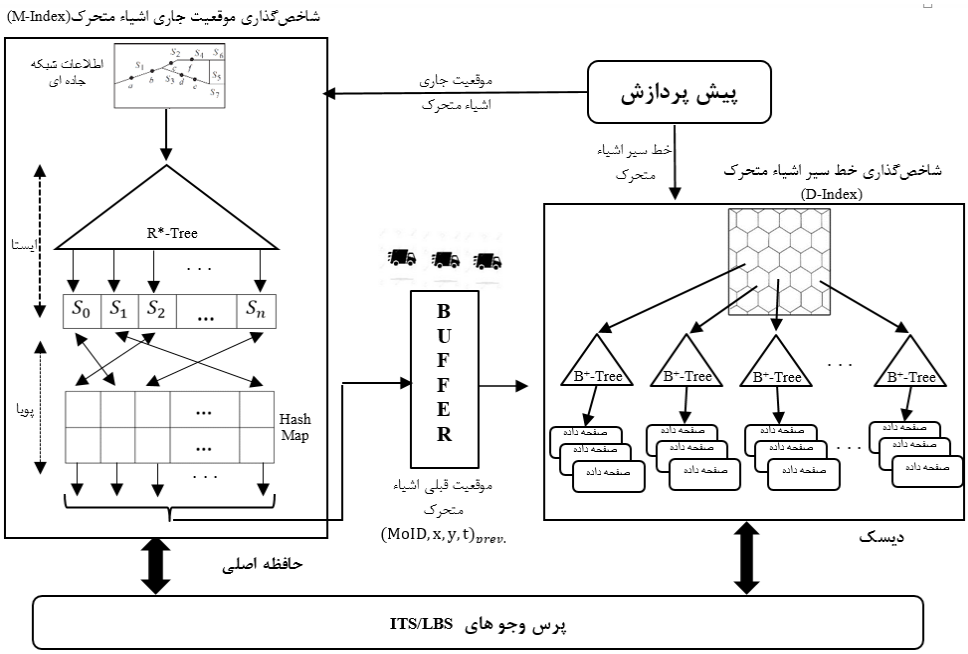
۳-۱- روش شاخص‌گذاری PCPI

همانگونه که در شکل (۱) مشاهده می‌شود، روش PCPI در دو بخش شاخص‌گذاری را انجام می‌دهد. در نتیجه می‌توان داده‌های دریافتی را به دودسته تقسیم کرد: ۱. موقعیت جاری اشیاء متحرک (زمان حال)، که شاخص این داده‌ها در حافظه اصلی ایجاد و نگهداری می‌شود و در شکل با عنوان M-Index مشخص شده است. ۲. داده‌های خط‌سیر اشیاء متحرک (زمان گذشته)، که شاخص این داده‌ها برروی دیسک ایجاد و نگهداری می‌شود و این بخش در شکل با عنوان D-Index مشخص شده است. این داده‌ها را می‌توان به‌صورت گروهی[1] از مجموعه داده‌های مختلف دریافت نمود و یا به‌صورت جریانی از نقاط، که پس از بروزرسانی موقعیت جاری اشیاء متحرک، موقعیت قبلی

[1] batch

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

آن را برای شاخص‌گذاری به‌صورت مرحله به مرحله دریافت کرد.



شکل ۱: ساختار شاخص‌گذاری PCPI

۳-۱-۱- شاخص‌گذاری موقعیت جاری اشیاء متحرک

یکی از ویژگی‌های اصلی شاخص‌گذاری موقعیت جاری اشیاء متحرک، بروزرسانی‌های تکراری و زیاد است که به‌عنوان یک چالش مهم در پایگاه داده‌های مکانی-زمانی و برنامه‌های کاربردی محسوب می‌شود. ساختار پیشنهادی با استفاده از جداسازی شاخص داده‌های ثابت و متغیر، به‌روزرسانی موقعیت جاری اشیاء را بهبود داده است. باید به این نکته توجه داشت که این ساختار برای شبکه‌های جاده‌ای مناسب است. این مرحله از PCPI دارای دو بخش ایستا و پویا است که شاخص‌گذاری هر دو بخش در حافظه اصلی انجام می‌شود. بخش ایستا بدون توجه به بروزرسانی‌ها بدون تغییر و ثابت بوده و بخش پویا به‌صورت مستقیم با سربار بروزرسانی در ارتباط است. در اینجا سعی در حداکثر کردن بخش ایستا وکمینه ساختن بخش پویا است که باعث کاهش تعداد بروزرسانی‌ها و در نتیجه افزایش کارایی خواهد شد. داده‌های دریافتی، در مرحله پیش‌پردازش به داده‌های منطبق بر شبکه تبدیل می‌شوند. همانطور که در بخش M-Index شکل (۱) مشخص است، اشیاء متحرک در این مرحله با کمک سگمنت‌های موجود در شبکه جاده‌ای[1] شاخص‌گذاری می‌شوند. سگمنت به قسمتی از جاده گفته می‌شود که با قسمت‌های دیگر جاده اشتراک نداشته باشد به عبارت دیگر قسمت‌هایی از جاده که می‌توان آنها را به‌صورت یک افراز تقسیم کرد. برای این منظور از دو ساختار R*-Tree و Hash Map استفاده شده است، در بخش ایستا، فقط یک R*-Tree برای سگمنت‌های شبکه جاده‌ای ساخته می‌شود. این سگمنت‌ها به‌کمک برگ‌های R*-Tree قابل دسترسی هستند. در این بخش R*-Tree مشابه دیگر R*-Tree ها استفاده می‌شود، با این تفاوت که به جای نقاط یا چند ضلعی‌ها، عناصر آن سگمنت‌های شبکه جاده‌ای هستند. در بخش پویا، داده‌ساختار Hash قرار دارد که شامل یک لیست از داده‌های اشیاء متحرکی است که در هر سگمنت

[1] Road Network

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

قرار دارند. کلید Hash استفاده شده، ID متناسب با آن می‌باشد. هر سگمنت به یکی از خانه‌های این Hash اشاره می‌کند که شامل شناسه اشیاء متحرک در آن سگمنت است و با این اشاره‌گر می‌توان اشیاء را از R*-Tree بازیابی نمود. علاوه برآن داده‌های دیگری مرتبط با اشیاء متحرک نظیر مختصات جغرافیایی، شناسه خط‌سیر، سرعت و دیگر ویژگی‌های مرتبط با هر شیء متحرک نیز در این ساختار ذخیره می‌شود. با استفاده از این داده ساختار یک ارتباط دو طرفه بین سگمنت‌ها و داده‌های اشیاء متحرک ساخته می‌شود. وقتی که موقعیت یک شیء متحرک تغییر می‌کند، داده‌های موقعیت و مختصات آن شیء در ساختار Hash بروزرسانی می‌شود و اگر سگمنت نیز تغییر کرده باشد، خط ارتباط بین شیء متحرک و سگمنت مربوطه (اشاره‌گر) نیز بروزرسانی می‌شود، ولی تغییری در بخش ایستا رخ نمی‌دهد. در واقع با جداسازی قسمت ایستا و پویا و محدود کردن بروزرسانی‌ها، تنها به قسمت پویا، باعث افزایش سرعت در بروزرسانی‌ها شده است. بعد از هر به‌روزرسانی موقعیت قبلی اشیاء متحرک برای اضافه شدن به خط‌سیر آنها و شاخص‌گذاری به بخش D-Index ارسال می‌شود. برای کاهش تعداد دسترسی‌ها به دیسک، اطلاعات اشیائی که موقعیت آنها تغییر کرده و بروزرسانی شده است و باید به شاخص خط‌سیر شیء که روی دیسک قرار دارد اضافه شود، به یک بافر با اندازه برابر با نصف اندازه بافر دیسک، منتقل می‌شود. این اطلاعات شامل $(M_{Id}, T_{Id}, x, y, t)$ می‌شود. بطوریکه M_{Id} شناسه شیء متحرک، T_{Id} شناسه خط‌سیر آن شیء، (x, y) مختصات شیء در شبکه جاده‌ای و t زمان تغییر موقعیت شیء می‌باشد. وقتی این بافر پر شد، اطلاعات آن به بخش D-Index منتقل و در آنجا شاخص‌گذاری می‌گردد که نحوه شاخص‌گذاری در بخش بعدی شرح داده شده است.

۳-۱-۲- شاخص‌گذاری خط‌سیر اشیاء متحرک

مجموعه داده‌هایی که حاوی خط‌سیر اشیاء متحرک است، به‌صورت گروهی دریافت و پس از انجام فرآیند پیش‌پردازش، شاخص‌گذاری می‌شود. علاوه برآن، اطلاعات مربوط به نقاط جاری اشیاء متحرک در بخش M-Index، پس از تغییر موقعیت شیء و بروزرسانی آن، باید به ادامه خط‌سیر اشیاء در بخش D-Index اضافه شود. برای این منظور و برای کاهش تعداد دسترسی‌ها به دیسک و افزایش سرعت شاخص‌گذاری، اطلاعات موقعیت مکانی-زمانی قبلی، به‌صورت جریانی از نقاط و مرحله به مرحله در بافر تعریف شده نگهداری و پس از پر شدن بافر به‌صورت گروهی در بخش D-Index شاخص خط‌سیر مربوطه را یافته و آن را بروزرسانی می‌کند. ساختارهای شاخص متعددی برای اطلاعات فضایی پیشنهاد شده است. یکی از روش‌های معمول، دسته‌بندی نمودن اشیا برحسب موقعیت مکانی آنها، تقسیم فضای داده و ذخیره‌سازی دسته‌های ایجادشده به‌عنوان صفحات در رسانه‌ی ذخیره‌سازی فیزیکی، می‌باشد[۳۰]. همانطور که در بخش D-Index شکل (۱) مشخص شده است، در روش PCPI برای شاخص‌گذاری خط‌سیرها از ساختار اصلاح شده SETI استفاده شده است[۳۱]. این ساختار برای شاخص‌گذاری خط‌سیرها در زمان گذشته مورد استفاده قرار می گیرد. به دلیل وجود تفاوت های اساسی بین ویژگی‌های ابعاد فضایی با بعد زمان، در این ساختار، شاخص گذاری ابعاد فضایی و زمانی از هم جدا می‌شود. با بررسی داده‌های خط‌سیر می‌توان دریافت که کران‌های ابعاد فضایی در طول زمان تغییرات بسیار کمی دارند. در حالی که بعد زمان به طور دائم در حال افزایش است. به دلیل اینکه ابعاد فضایی دارای تغییرات کمی هستند، این روش به‌صورت منطقی محدوده فضا را به سلول‌های شش‌ضلعی استاتیک و بدون همپوشانی تقسیم می‌کند. هر کدام از سلول ها تنها حاوی سگمنت‌هایی هستند که به‌طور کامل در داخل آن

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

سلول قرار می گیرند. اگر سگمنتی از خط‌سیر به‌صورت کامل درون سلول قرار نگیرد، در مرز بین دو سلول به دو زیر سگمنت تقسیم شده و هر کدام در سلول متناظر درج می‌شوند. هر سگمنت به‌عنوان یک چندتایی در یک صفحه داده‌ای بر روی دیسک ذخیره می‌شود. با این محدودیت که هر صفحه‌ی داده تنها شامل سگمنت‌هایی است که به همان سلول فضایی متعلق هستند. برای هر سلول یک شاخص زمانی پراکنده ساخته می شود. به این صورت که به‌ازای هر صفحه داده یک مقدار در شاخص ذخیره می‌شود (به جای آنکه به ازای هر سگمنت یک مقدار ذخیره شود). برای ساخت این شاخص، طول عمر هر صفحه داده محاسبه می‌شود. در این حالت برای هر صفحه ی داده، یک دوتایی به فرمت "حد پایین طول عمر صفحه ی داده و حد بالای صفحه ی داده" در درخت ذخیره می‌شود که برای این‌منظور در SETI از درخت *R برای استفاده می‌شود. بنابراین در اینجا داده‌هایی دوبعدی در درخت ذخیره می‌شوند. اما در روش PCPI به‌جای استفاده از درخت *R از درخت B^+ برای شاخص‌گذاری بر بعد زمان، استفاده‌شده است. در این درخت برای هر صفحه‌ی داده، حد پایین طول عمر آن صفحه، در درخت ذخیره می‌شود. این کار در جستجوی داده‌ها خللی را وارد نمی‌کند چراکه داده‌های درون درخت B^+، مرتب‌شده‌اند و عدم وجود حد بالای طول عمر صفحات، برای یافتن صفحات مرتبط با پرس‌وجو، مشکلی را ایجاد نمی‌نماید. علاوه بر آن، به‌جای شاخص‌گذاری بر داده‌های دوتایی، بر داده‌هایی تک‌مقداره شاخص‌گذاری انجام می‌شود و بنابراین با این کار از مشکلات چندتایی بودن مقادیر، پیش‌گیری می‌شود. همچنین زمان جستجو در یک درخت B^+ بسیار کوتاه‌تر از زمان جستجو در دیگر انواع درخت‌هاست. به دلیل مرتب بودن مقادیر در درخت B^+، طول عمر اکثر صفحات یافته شده در طی پرس و جوی محدوده‌ای[1]، در محدوده‌ی مکانی پرس و جو قرار دارد و در مرحله‌ی بعدی پردازش پرس‌وجو، نیاز به اعمال گزاره‌ی مکانی بر سگمنت‌های آن صفحات نخواهد بود. در روش‌های شاخص‌گذاری، با افزایش ابعاد، کارآمدی روش‌ها به شدت کاهش می‌یابد. از آنجایی که در PCPI از شاخص یک بعدی استفاده شده است، ساختار شاخص‌گذاری با مشکل چندبعدی بودن مواجه نخواهد شد. همچنین به‌جهت استفاده از شاخص‌های پراکنده سربار شاخص‌گذاری کمتر شده و کارآمدی درج ها بهبود می‌یابد. همچنین در روش شاخص‌گذاری SETI از یک جدول درهم سازی برای سرعت بخشیدن به عملیات درج استفاده شده است. این جدول درهم سازی، آخرین مکان تمام اشیاء متحرک را در حافظه اصلی نگهداری می‌کند که از آن برای افزودن سگمنت‌های جدید به انتهای خط سیرهای موجود استفاده می‌شود. در روش PCPI به جای این جدول درهم‌سازی، از داده‌ساختار Hash موجود در در بخش M-Index استفاده شده است. زیرا این ساختار حاوی موقعیت جاری اشیاء است که بعد از بروزرسانی به خط‌سیر هر شیء برروی دیسک اضافه می‌شود. با این روش علاوه بر دسترسی به آخرین مکان هرخط‌سیر، به سایر اطلاعات هر شیء دسترسی خواهیم داشت.

۳-۲- پیش پردازش

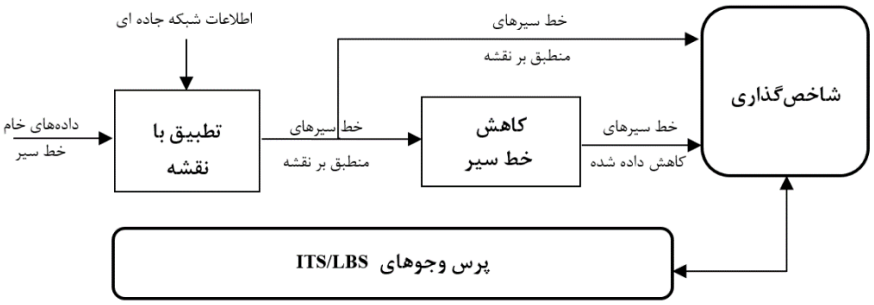
قبل از انجام شاخص‌گذاری، فرآیند پیش‌پردازشی بر روی داده‌ها انجام می‌شود که شکل (۲) شمای سطح بالا از فرآیند را نمایش می‌دهد. در این فرآیند از اطلاعات مربوط به شبکه جاده‌ای که به‌عنوان ورودی دریافت می‌شود، ساختار شبکه جاده‌ای تولید شده و سپس با در دست داشتن این ساختار، داده‌های خام خط‌سیر به عنوان ورودی این مرحله برای تحلیل

[1] Range query

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]

داده‌های مکانی-زمانی در محیط شبکه جاده‌ای، به داده‌های منطبق بر شبکه تبدیل می‌شوند. این داده‌ها، به دلیل اختلال حسگر در دستگاه‌های مکان‌یابی، سیگنال ضعیف ماهواره‌ای در فضاهای متراکم شهری و عوامل مختلف دیگر، هیچ‌گاه کاملاً دقیق نیستند و موقعیت اشیاء را با درصدی خطا گزارش می‌کنند که در برخی موارد این خطاها بسیار زیاد (در حد چند صد متر) است. این نوع خطاها که برای انجام پژوهش‌های گوناگون در حوزه پایگاه‌داده‌های اشیاء متحرک و تحلیل داده‌های مکانی-زمانی وسایل نقلیه مورد استفاده قرار می‌گیرند، زیاد هستند و تأثیر مستقیم بر عملکرد برنامه‌های کاربردی دارند. در مواردی که داده‌ها دارای خطای بسیار زیادی هستند، می‌توان با استفاده از فیلترهایی نظیر فیلتر کالمان یا فیلتر ذره‌ای، این خطاها را از بین برد. اما در بسیاری از موارد که خطای موردنظر نسبتاً کم می‌باشد، عمل تطبیق بر نقشه می‌تواند راهگشا باشد.



شکل ۲: فرآیند پیش‌پردازش

انجام عمل تطبیق بر نقشه در مواردی مانند اختلال حالت ساکن[1]، پل‌های روگذر و راه‌های موازی، مساله بسیار پیچیده ایست که تنها با اتکاء به روش‌های معمول هندسی نظیر تطبیق نقطه بر نزدیکترین خیابان قابل حل نیست. می‌توان با بهره‌گیری از ویژگی‌های توپولوژیک گراف شبکه جاده‌ای و روش‌های پیشرفته احتمالاتی نظیر مدل‌های مارکوف و منطق فازی، به دقت قابل قبولی در این زمینه دست یافت. روش‌های پیشرفته برای حل مسأله تطبیق نقشه، تلاش می‌کنند تا دنباله‌ای از یال‌های شبکه را بیابند که هم به منحنی ساخته شده از حرکت شیء متحرک نزدیک باشند و هم یک مسیر منطقی در شبکه جاده‌ای را شامل شوند. این روش‌ها همواره تلاش می‌کنند تا توازنی بین تطبیق نقطه GPS به نزدیکترین خیابان و تطبیق آن به خیابانی که با در نظر گرفتن نقاط تطبیق یافته قبلی، بامعنی‌تر است، برقرار سازند.

هدف روش‌ها کاهش داده، کاهش سربار ارتباطاتی و ذخیره‌سازی و افزایش سرعت پرس‌وجوهای مکانی-زمانی است. البته به شرط اینکه دقت داده‌های کاهش‌یافته در حد قابل‌قبولی باشد. در نتیجه در صورت نیاز به کاهش داده، قبل از مرحله شاخص‌گذاری، با بهره‌گیری از خوشه‌بندی خط‌سیرها اقدام به کاهش داده می‌شود. پس از انجام عمل خوشه‌بندی، مجموعه‌ای از دنباله‌های یال‌های شبکه جاده‌ای به‌عنوان نماینده هر خوشه به‌دست می‌آیند. با استفاده از اطلاعات استخراج‌شده از این نماینده‌ها، عمل کاهش داده بر روی خط‌سیرهای منطبق بر نقشه انجام می‌شود. برای این منظور، با در دست داشتن خوشه‌های کشف‌شده، ابتدا نیاز است که بخش‌هایی از هر خط‌سیر که منطبق بر بخش‌هایی از خوشه‌ها می‌باشند، شناسایی شوند. سپس با خواندن نقاط هر خط‌سیر، یک دنباله از یال‌های پیموده شده در طی خط‌سیر جاری ایجاد می‌شود. در ادامه زیردنباله‌های مختلف این دنباله در مجموعه

[1] stationary state noise

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

خوشه‌ها را با استفاده از داده ساختار ترای سه‌گانه جستجو[1] (TST) [۳۲]، جستجو می‌کنیم. ممکن است زیرخط‌سیرهایی از یک خط‌سیر بر قسمت‌هایی از یک خوشه با طول‌های متفاوت منطبق باشد. لذا با یافتن همه انطباق‌ها، در مرحله بعد انطباق‌هایی را انتخاب می‌کنیم که در آخر منجر به بهترین نتیجه از نظر حجم داده‌های نهایی شوند. درنهایت هم با در نظر گرفتن اطلاعات زمانی خط‌سیرها، داده‌های اولیه با اطلاعات خوشه‌ها جایگزین می‌شوند. در این مرحله با بهره‌گیری از درون‌یابی، اطلاعات زمانی خط‌سیرها تخمین زده‌شده و تنها این اطلاعات ذخیره می‌شوند. خط‌سیرهای تولیدشده دارای حجم پایین‌تر و درعین‌حال حاوی اطلاعات کافی موردنیاز برنامه‌های کاربردی در مورد حرکت اشیاء متحرک هستند. بنابراین پرس‌وجوهای لازم برای برنامه‌های کاربردی مختلف، می‌توانند به‌طور مستقیم بر روی داده‌های کاهش‌داده شده اعمال گردند. در این مقاله تاثیر کاهش داده برنتایج به‌دست آمده نیز بررسی شده است که در ادامه شرح داده شده است.

۴- ارزیابی روش پیشنهادی

در پیاده‌سازی و انجام آزمایش‌ها در روش پیشنهادی، سیستمی با پردازنده اینتل Core i5 3.30 GHz، 8096 MB حافظه اصلی و سیستم عامل ویندوز ۸ استفاده شده است. همچنین برای پیاده‌سازی الگوریتم‌ها از زبان برنامه‌نویسی جاوا استفاده شد. در این پژوهش ارزیابی‌ها و آزمایش‌ها بر روی داده‌های واقعی خط‌سیر Geolife انجام شده‌اند. این مجموعه داده شامل خط‌سیرهای ۱۸۲ کاربر در طول مدت ۵ سال از آوریل ۲۰۰۷ تا اوت ۲۰۱۲ است. این خط‌سیرها، به‌صورت دنباله‌ای از نقاط مکانی در محدوده شهر پکن ثبت شده‌اند که هر کدام دارای یک مهر زمانی می‌باشند. تعداد خط‌سیرهای این مجموعه ۱۸۶۷۰ خط‌سیر است که درمجموع دارای ۲۴,۸۷۶,۹۷۸ نقطه و مجموع طول ۱,۲۹۲,۹۵۱ کیلومتر و مدت‌زمان ۵۰,۱۷۶ ساعت است. از آنجا که تعدادی از این خط‌سیرها خارج از محدوده شبکه جاده‌ای پکن بودند، این خط‌سیرها از مجموعه داده حذف شدند. از طرف دیگر در بخش پیش پردازش، هر کدام از خط‌سیرها که دارای توقف‌های طولانی‌مدت بودند، مطابق با تعریف خط‌سیر، به‌عنوان چند خط‌سیر در نظر گرفته شده‌اند. لذا، در آزمایش‌های این پژوهش در مجموع ۳۶۵۶۳ خط‌سیر که شامل ۲۴.۸۱۹.۱۴۵ نقطه است، در این مجموعه داده درنظرگرفته شده است. همچنین در مرحله پیش‌پردازش با انجام عمل تطبیق بر نقشه بر روی داده‌های خط‌سیر خام موجود در مجموعه داده، به میزان قابل‌توجهی صحت و کیفیت این داده‌ها افزایش یافته و داده‌ها به یالی که در واقع روی آن در حال حرکت بوده‌اند، منطبق شده‌اند. لازم به یادآوری است که این روش برای داده‌های همه شهرها، ازجمله شهرهای ایران نیز قابل استفاده است. اما به‌دلیل دردسترس نبودن مجموعه داده مناسب در کشور که بتوان برای انجام آزمایش‌ها از آن بهره‌برد، این آزمایش برروی مجموعه داده Geolife انجام شد.

۴-۱- ارزیابی نتایج بخش کاهش داده

ازآنجایی‌که شاخص‌ها باید برروی دیسک ایجاد و نگهداری شوند، لذا کاهش حجم مجموعه داده و در نتیجه کاهش حجم شاخص‌ها می‌تواند کارایی روش PCPI را برای شاخص‌گذاری داده‌های زمان گذشته جهت استفاده در برنامه‌های کاربردی مختلف را افزایش دهد. باید به این نکته توجه داشت که برای زمان حال چون فقط یک نقطه داریم کاهش داده معنی ندارد. بیشتر آزمایش‌ها در این پژوهش برروی هر دو مجموعه داده اصلی و کاهش داده شده انجام شده است تا تاثیر کاهش داده‌ها برروی دقت و سرعت نتایج نیز بررسی گردد. حجم مجموعه داده اصلی ۹۷۶ مگابایت

[1] Ternary Search Trie

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]

است که می‌توان با انجام عمل کاهش داده و شاخص گذاری مجموعه حاصل، با یک شاخص ۳۷۰ مگابایتی این مجموعه داده را به‌صورت بهینه در حافظه اصلی نگه داشت. تعداد نقاط مجموعه داده کاهش داده شده برابر با ۲.۸۹۳.۵۴۷ نقطه می‌باشد.

از آنجا که برنامه‌های کاربردی، پرس‌وجوهای مختلفی از سوی کاربران را دریافت و پردازش می‌کنند، لذا برای ارزیابی تأثیر کاهش داده بر روی برنامه‌های کاربردی، چند پرس‌وجوی پرکاربرد در حوزه پایگاه داده‌های اشیاء متحرک برای بررسی انتخاب شدند. نتایج حاصل از اعمال این پرس‌وجوها بر روی داده‌های کاهش‌یافته می‌بایست مشابه نتایج حاصل از اعمال همان پرس‌وجوها بر روی داده‌های اصلی باشند. یعنی فاصله بین نتیجه حاصل از داده‌های کاهش‌یافته و نتیجه حاصل از داده‌های اصلی، کمتر از آستانه (ε) قابل قبول باشد (مقدار (ε) در این پژوهش ۱۰ متر می‌باشد). برای این ارزیابی، سه پرس‌وجوی *WhereAt*، *WhenAt*[۳۳] و پرس‌وجوی محدوده‌ای [۸, ۳۳, ۳۴] در نظر گرفته شدند. پرس‌وجوی *WhereAt* برای یافتن مکان خط‌سیر T در زمان t به‌کار می‌رود. منظور از مکان خط‌سیر T همان مکان شیء متحرکی است که خط‌سیر T توسط آن به ثبت رسیده است. در صورتی که این زمان، در مجموعه مهرهای زمانی خط‌سیر موجود باشد، مکان مربوط به آن مهر زمانی برگردانده می‌شود و در غیر این صورت، با یافتن دو مهر زمانی قبل و بعد از زمان t، مکان مربوط به زمان t از طریق درون‌یابی، محاسبه می‌شود. پرس‌وجوی *WhenAt* هم با روشی مشابه، زمانی که خط‌سیر T در مکان (x,y) بوده است را ارائه می‌دهد[۳۳]. این دو پرس‌وجو برای برای ارزیابی تاثیر کاهش داده برروی برنامه‌های کاربردی استفاده می‌شوند. به‌منظور بررسی تأثیر کاهش داده بر روی دو پرس‌وجوی *WhenAt* و *WhereAt*، ۱۰۰۰ نقطه جغرافیایی و ۱۰۰۰ نقطه زمانی، به‌صورت تصادفی در محدوده داده‌های مجموعه داده، استخراج و بر روی مجموعه داده‌ها اعمال شدند.

همچنین برای بررسی تأثیر روش PCPI بر روی زمان اجرای این پرس‌وجوها، زمان لازم برای پردازش آن‌ها، هم بر روی داده‌های اولیه و هم بر روی داده‌های کاهش‌یافته اندازه‌گیری شد. علاوه بر این، برای هر یک از ۱۰۰۰ پرس‌وجوی انجام‌شده، نسبت عملکرد پرس‌وجو[1] به‌صورت نسبت زمان پردازش بر روی مجموعه کاهش‌یافته به زمان پردازش بر روی مجموعه داده اولیه، محاسبه شد. میانگین زمان لازم برای پرس‌وجوی *WhenAt* و *WhereAt* در شکل(۳-الف) آمده است.

از آنجا که تعداد نقاط مجموعه داده کاهش داده‌شده، بسیار کمتر از داده‌های اولیه است، لذا زمان لازم برای پردازش این پرس‌وجوها، در دو حالت ذکرشده، تفاوت زیادی را نشان می‌دهد. البته مقایسه میانگین این زمان‌ها، نمی‌تواند معیار خوبی برای توصیف میزان بهبود سرعت انجام پرس‌وجوها باشد. ازاین‌رو نمودار میانگین *نسبت عملکرد پرس‌وجو* برای این پرس‌وجوها در شکل (۳-ب) آمده است. همان‌گونه که این شکل نشان می‌دهد، به‌طور میانگین پرس‌وجوی *whereAt* نزدیک به ۲ برابر سریع‌تر شده و پرس‌وجوی *WhenAt* نیز تقریباً ۳ برابر سریع‌تر شده است. یعنی زمان لازم برای پردازش پرس‌وجوی *WhereAt* در داده‌های کاهش داده‌شده، به‌طور میانگین ۵۴ درصد و برای پرس‌وجوی *WhenAt*، ۲۷ درصد زمان لازم برای پردازش این پرس‌وجوها بر روی داده‌های اصلی است. دلیل تفاوت در میزان سریع‌شدن این دو پرس‌وجو، تفاوت در انجام محاسباتی است که برای پاسخ‌گویی به این پرس‌وجوها مورد نیاز است.

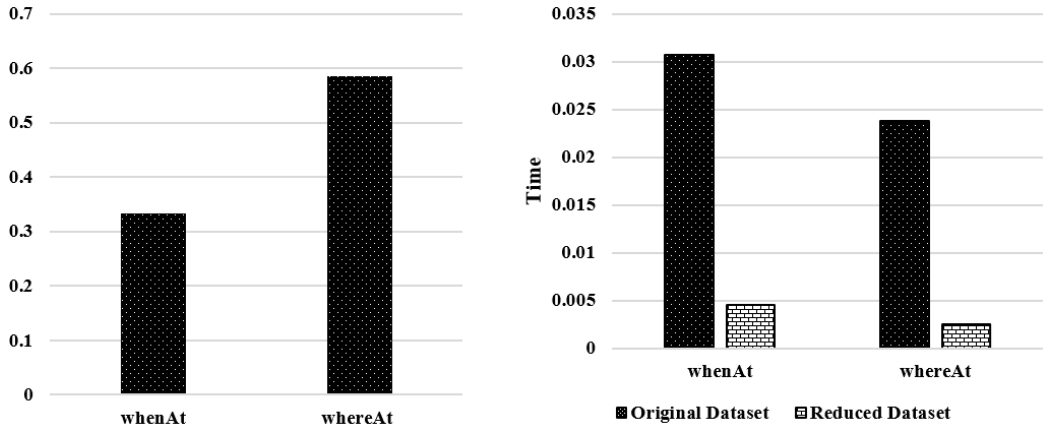
میزان افت دقت نتایج این پرس‌وجوها یعنی حداکثر خطای قابل‌تحمل توسط برنامه کاربردی برای روش کاهش داده نیز بررسی گردید، با توجه به قضیه‌های اثبات‌شده در [۳۳]، انتخاب این پارامتر، حداکثر خطای پرس‌وجوهایی که بر روی

[1] Query Performance Ratio

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

داده‌های کاهش داده‌شده اعمال می‌شوند را نیز ضمانت می‌کند. ازاین‌رو انتظار می‌رفت که خطای این پرس‌وجوها نیز همواره پایین‌تر از ε باشد.

برای محاسبه خطای پرس‌وجوی *WhereAt* از رابطه (۱) استفاده شده است.



شکل ۳: الف.میانگین زمان لازم برای پرس‌وجو WhenAt و WhereAt –ب. میانگین نسبت عملکرد پرس‌وجو

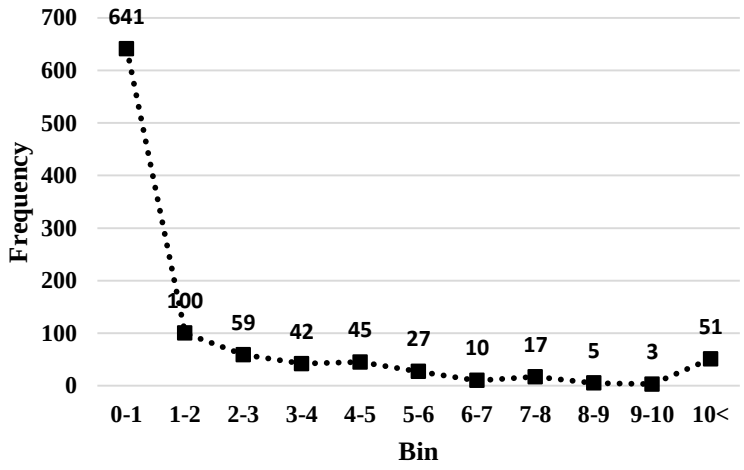
رابطه (۱)
$$distance = dist\big(WhereAt(T,t), WhereAt(T',t)\big) = dist\big((x,y),(x',y')\big)$$

تابع $dist()$ در این رابطه فاصله شبکه‌ای را نشان می‌دهد. این رابطه در واقع فاصله بین نقطه بازگردانده شده از پرس‌وجو بر روی خط‌سیر کاهش‌یافته و نقطه بازگردانده شده از همان پرس‌وجو روی خط‌سیر اولیه را محاسبه می‌کند. همین رابطه برای *WhenAt* نیز قابل فرض است. همانطور که در هیستوگرام شکل (۴) مشخص است، میانگین خطاهای ۱۰۰۰ پرس‌وجوی انجام‌شده کمتر از ε (۱۰ متر) می‌باشد (بیش از ۶۰ درصد این خطاها، در محدوده بین ۰ و ۱ می‌باشند) و فقط تعداد ۵۱ پرس‌وجو، یعنی حدود ۵ درصد پرس‌وجوها دارای خطای بالاتر از ε بوده است. علت این امر، وجود اختلال‌های بسیار زیاد این داده‌ها مانند اختلال حالت ساکن است که با توجه به مزایایی که دارد از این میزان می‌توان چشم پوشی کرد.

از نتایج به‌دست آمده از ارزیابی اعمال پرس‌وجوها برروی مجموعه داده کاهش داده‌شده، با در نظر گرفتن اختلال‌های متعدد در مجموعه داده واقعی، دارای دقت قابل‌قبولی بوده و از لحاظ زمان پردازش پرس‌وجوها بسیار مطلوب می‌باشد. لذا استفاده از مجموعه کاهش داده شده برای انجام پرس‌وجوها در برنامه‌های کاربردی قابل قبول بوده و در نتیجه در بیشتر آزمایش‌ها از این مجموعه داده نیز استفاده شده است.

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]



شکل ۴: میانگین خطاهای ۱۰۰۰ پرس‌وجوی انجام شده

۴-۲- ارزیابی شاخص‌گذاری موقعیت جاری اشیاء متحرک

تعداد بالای بروزرسانی‌ها از جمله چالش‌های شاخص‌گذاری موقعیت جاری اشیاء متحرک می‌باشد. علاوه بر این برای افزایش دقت نتایج پرس‌وجوها باید موقعیت اشیاء را با نرخ و دقت مناسب دریافت نمود. لذا روش کاهش داده در این بخش تاثیر منفی دارد و به همه نقاط مرتبط با اشیاء متحرک در پردازش‌ها نیاز است. اما انجام عمل تطبیق برنقشه برروی موقعیت اشیاء، باعث حذف برخی از خطاهای احتمالی و افزایش کیفیت داده‌ها شده است. برای انجام شاخص‌گذاری نقاط، ابتدا سگمنت‌های شبکه جاده‌ای با استفاده از ساختار *R*-Tree* شاخص‌گذاری می‌شوند. همانطور که قبلاً عنوان شد این ساختار ثابت بوده و تغییر نمی‌کند. اطلاعات مربوط به موقعیت جاری اشیاء متحرک نیز در ساختار *HashMap* درج و ذخیره می‌شود. همه بروزرسانی‌ها در این ساختار انجام می‌شود. اشاره‌گری دوطرفه بین این ساختار و *R*-Tree* وجود دارد که مشخص می‌کند شیء متحرک در کدام سگمنت قرار دارد. با تغییر موقعیت اشیاء فقط کافی است اطلاعات مختصات شیء متحرک در ساختار مورد نظر اصلاح شود و اگر سگمنت شیء متحرک نیز تغییر کرده بود، اشاره‌گر مربوطه نیز بروزرسانی گردد.

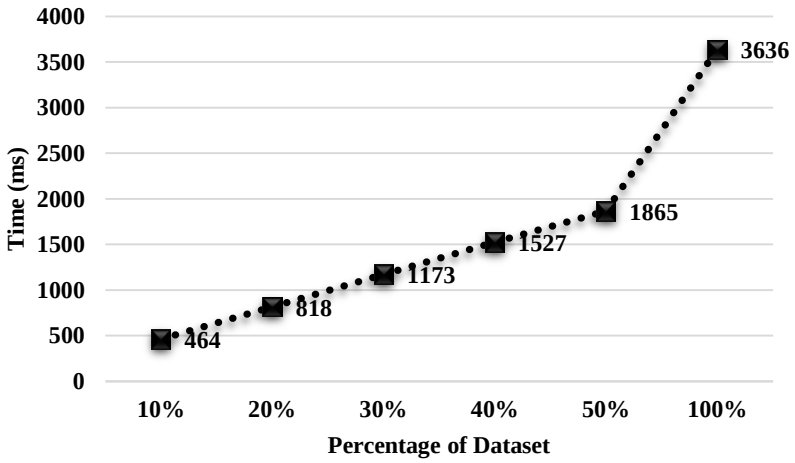
به‌منظور ارزیابی تاثیر حجم مجموعه داده برروی سرعت شاخص‌گذاری، آزمایش‌ها بر روی درصدهای مشخصی از این مجموعه داده، ۱۰، ۲۰، ۳۰، ۴۰، ۵۰ و ۱۰۰ درصد داده‌ها انجام شد. سپس بر روی هر مجموعه از شاخص‌های حاصل، پرس‌وجوهای مورد نظر اعمال گردید. کل زمان مورد نیاز برای بارگذاری کل مجموعه داده ۱۸۱۴۲ *ms* و همچنین کل زمان موردنیاز برای بارگذاری سگمنت‌های شبکه جاده‌ای برابر با ۲۳۴ *ms* است. زمان ساخت شاخص شبکه جاده‌ای نیز برابر با ۲۰۱۳ *ms* شد. زمان ساخت شاخص برای درصدهای مختلف از مجموعه داده نیز در شکل (۵) آمده است.

تعداد بروزرسانی‌های انجام شده برای کل مجموعه داده برابر با *24828783* می‌باشد. پس از ساخت شاخص‌ها برای ارزیابی بهتر این بخش در برنامه‌های کاربردی، دو پرس‌وجوی پرکاربرد محدوده‌ای [۸، ۳۳ و ۳۴] و *kNN* [۳۵، ۳۶ و ۳۷] در حوزه پایگاه داده‌های اشیاء متحرک برای بررسی انتخاب شدند. نتایج حاصل از اعمال این پرس‌وجوها بر روی داده‌ها در ادامه آماده است. در آزمایش‌ها هدف یافتن k شیء با کمترین فاصله تا نقطه پرس‌وجوی مورد نظر در یک لیست غیر مرتب است. برای ارزیابی هرچه بهتر نتایج بدست آمده در پرس‌وجوی *kNN* مقدار k به تدریج افزایش یافته و ۱،

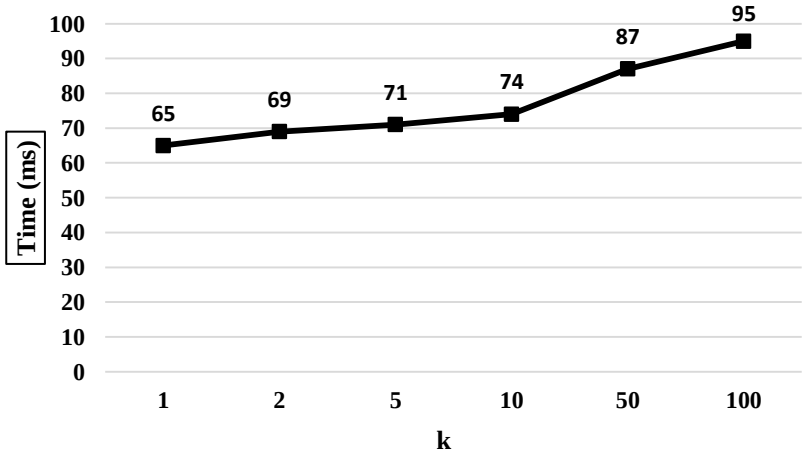
[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

۲، ۵، ۱۰، ۵۰، ۱۰۰،در نظر گرفته شد. در این پرس‌وجو از *MinHeap* استفاده شد که هزینه ساخت آن از مرتبه $\theta(n)$ می‌باشد. یافتن k شیء با کمترین فاصله در یک لیست غیرمرتب با استفاده از این روش حداکثر $\theta(n\ log\ n)$ زمان نیاز دارد. علاوه بر این خروجی ایجاد شده نیز مرتب شده است.

در نتایج حاصل که در شکل (۶) آمده است، وقتی‌که که مقدار k برابر ۱ یا ۱۰۰ باشد تفاوت چندانی در زمان اجرای پرس‌وجوی *kNN* ندارد، در واقع بیشترین زمان لازم در این روش برای ساخت *Heap* صرف می‌شود.



شکل ۵: زمان ساخت شاخص برای درصدهای مختلف مجموعه داده

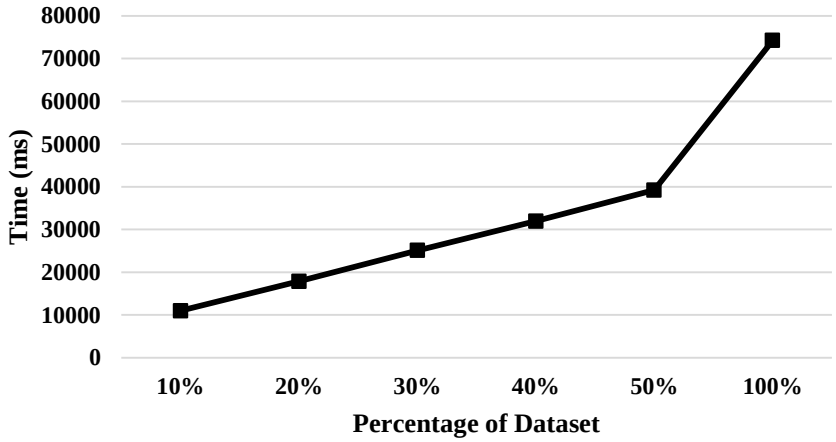


شکل ۶: زمان اجرای پرس‌وجوی kNN برای kهای مختلف

همچنین به‌منظور بررسی تأثیر روش پیشنهادی بر روی پرس‌وجوی محدوده‌ای، ۱۰۰۰ پرس‌وجو به‌صورت تصادفی در محدوده داده‌های مجموعه داده استخراج شده و آزمایش‌ها برروی این پرس‌وجوها انجام شده است. این آزمایش‌ها برروی درصدهای مختلفی از مجموعه داده تکرار شده و تاثیر این موضوع نیز مورد بررسی قرار گرفت. که نتایج حاصل در شکل (۷) آمده است.

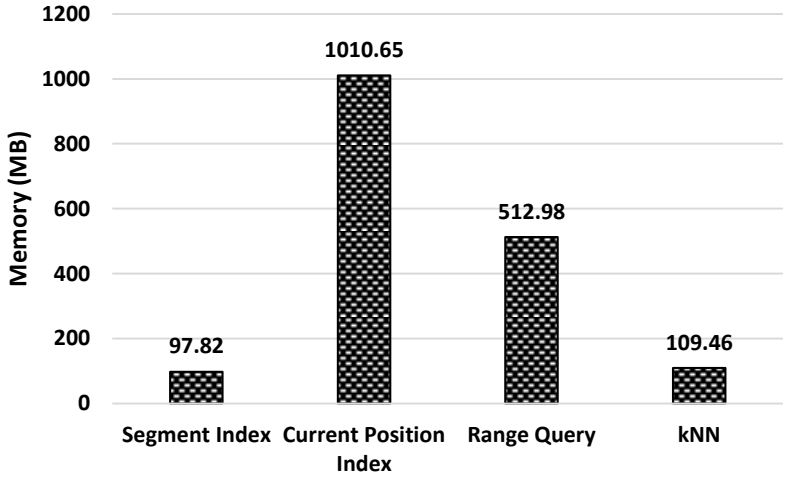
[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]



شکل ۷: زمان انجام پرس‌وجوی محدوده‌ای

ازآنجایی‌که این شاخص‌ها در حافظه اصلی ایجاد و نگهداری می‌شود علاوه بر زمان، میزان حافظه مصرفی نیز دارای اهمیت می‌باشد. لذا بعد از اجرای هر مرحله، از تکنیک *Garbage Collector* استفاده شده تا فضای استفاده شده به‌صورت دقیق محاسبه شود. شکل (۸) حداکثر حافظه مصرفی برای مرحله شاخص‌گذاری شبکه جاده‌ای، موقعیت جاری اشیاء متحرک و همچنین پرس‌وجوهای انجام شده را نمایش می‌دهد. همانطور که در این شکل مشخص شده است، بیشترین حافظه‌ای مصرفی برای شاخص‌گذاری موقعیت جاری اشیاء است که اطلاعات شاخص همه اشیاء و شاخص شبکه جاده‌ای در حافظه قرار می‌گیرند.



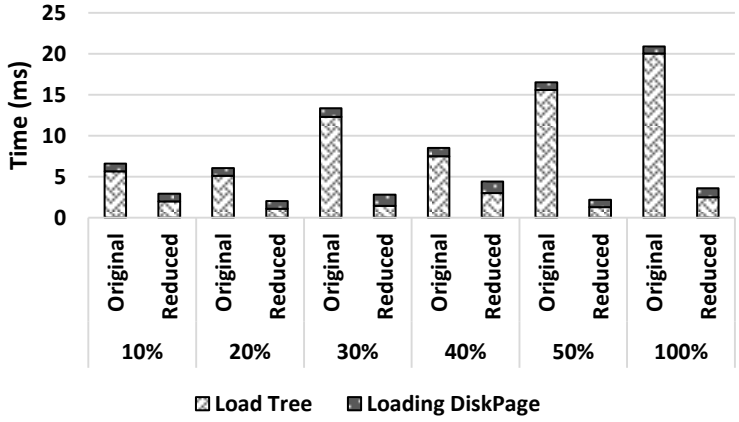
شکل ۸: میزان حافظه مصرفی در فرآیند شاخص‌گذاری و انجام پرس‌وجوها

۴-۳- ارزیابی شاخص‌گذاری خط‌سیر اشیاء متحرک

همان‌طور که در بخش ۳.۱.۲ اشاره شد، در روش PCPI برای شاخص‌گذاری داده‌های خط‌سیر اشیاء متحرک، از شاخص اصلاح شده SETI استفاده شد. از آنجایی‌که این شاخص برروی دیسک ساخته و نگهداری می‌شود برای بهره بیشتر، در قسمت شاخص‌های زمانی پراکنده از B^+-Tree به جای R*-Tree استفاده شده است. همچنین برای نگهداری آخرین مکان هر خط‌سیر در حافظه اصلی، بجای جدول درهم سازی مجزا،

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

از داده‌ساختار Hash موجود در قسمت M-Index استفاده شده است. با این روش اطلاعات علاوه بر افزایش سرعت، امکان نگهداری اطلاعات بیشتری نیز فراهم آمده است. آزمایش‌های انجام شده در این بخش برروی هر دومجموعه داده اصلی و کاهش داده شده، انجام شده است. در صورتی که از مجموعه داده کاهش داده‌شده برای پاسخگویی به پرس‌وجوها استفاده شود، می‌توان این مجموعه داده را با سرعت بسیار بیشتری شاخص‌گذاری کرد. از نظر حجمی که شاخص‌ها برروی دیسک اشغال می‌کنند در مجموعه داده اصلی GB 3.29 و در مجموعه داده کاهش داده شده برابر با MB 551 می‌باشد. این کل حجم شاخص‌های زمانی و صفحات داده‌ای است که برروی دیسک ذخیره شده است. البته بخش عمده این حجم را صفحات داده‌ای اشغال می‌کنند و شاخص‌های زمانی حجم بسیار اندکی دارند. این میزان کاهش حجم داده برروی دیسک، بسیار قابل توجه است و اگر دقت پاسخ‌ها به انواع پرس‌وجوها قابل قبول باشد، نشان دهنده این خواهد بود که روش کاهش داده برای شاخص‌گذاری داده‌ها در برنامه های کاربردی، مناسب و قابل استفاده است. در ادامه نشان خواهیم داد که این روش در پاسخ به انواع پرس‌وجوها نیز دارای دقت قابل قبولی است. از نظر میانگین زمانی برای پاسخ به پرس‌وجوها بیشترین میانگین زمان مربوط به بارگذاری شاخص‌های زمانی است و بارگذاری صفحات داده زمان کمتری را نیاز دارد. این موضوع برای هر دو مجموعه داده اصلی وکاهش داده شده، در شکل (۹) نمایش داده شده است. همچنین بین میانگین زمانی دو مجموعه داده اصلی و کاهش داده شده اختلاف قابل توجه وجود دارد که میانگین زمان مورد نیاز مجموعه داده کاهش داده شده در همه حالت‌ها کمتر است.



شکل ۹: میانگین زمان بارگذاری صفحات داده‌ای و شاخص‌های زمانی درصدهای مختلف مجموعه داده اصلی و کاهش داده شده

جدول (۱) میانگین زمان ساخت صفحات داده‌ای و همچنین میانگین زمان افزودن به صفحات داده‌ای برای شاخص‌گذاری داده‌ها در هر دو مجموعه داده اصلی و کاهش داده شده را نمایش می‌دهد. همانطور که در این جدول مشخص شده این زمان‌ها بسیار کوچک و در حد میکروثانیه و در برخی موارد حتی کوچکتر نیز می‌باشد.

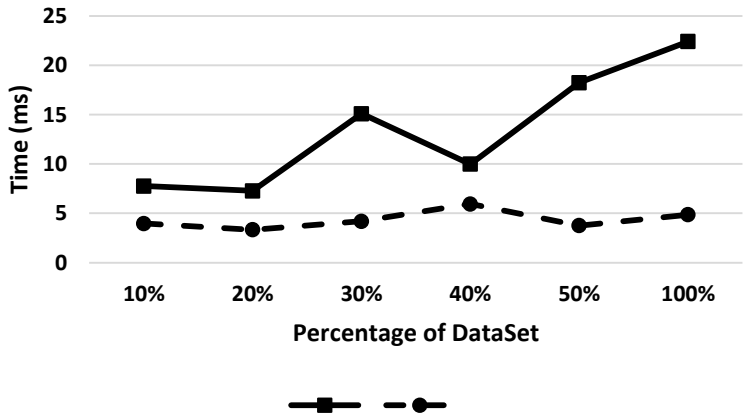
جدول ۱: میانگین زمان ساخت و افزودن به صفحات داده‌های

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]

درصد مجموعه داده	افزودن به صفحات داده ای		ایجاد صفحات داده ای	
	کاهش داده شده	اصلی	کاهش داده شده	اصلی
10%	0.000963391	0.000120351	0.000120351	0.043677696
20%	0.000271481	0.000235849	0.000235849	0.043200126
30%	0.001173021	0.000791766	0.000791766	0.060285753
40%	0.000729927	0.000407914	0.000407914	0.055106195
50%	0.000130582	0.000020351	<0.0001	0.059595634
100%	<0.0001	0.000035849	<0.0001	0.070595812

همچنین زمان افزودن سگمنت‌ها در درصدهای مختلف برای هر دو مجموعه داده اصلی و کاهش داده شده در شکل (۱۰) آمده است. لازم به ذکر است که این زمان برای افزودن همه سگمنت‌ها می باشد. این تعداد در کل مجموعه داده اصلی برابر با ۹۶۹۲۹۲۳ عملیات و در مجموعه داده کاهش داده شده برابر با ۱۸۸۱۵۷۹ عملیات افزودن سگمنت می‌باشد. همانطور که در شکل مشخص است در مجموعه داده اصلی با افزایش حجم مجموعه داده زمان نیز افزایش یافته ولی در مجموعه داده کاهش داده شده زمان تغییر چندانی ندارد.



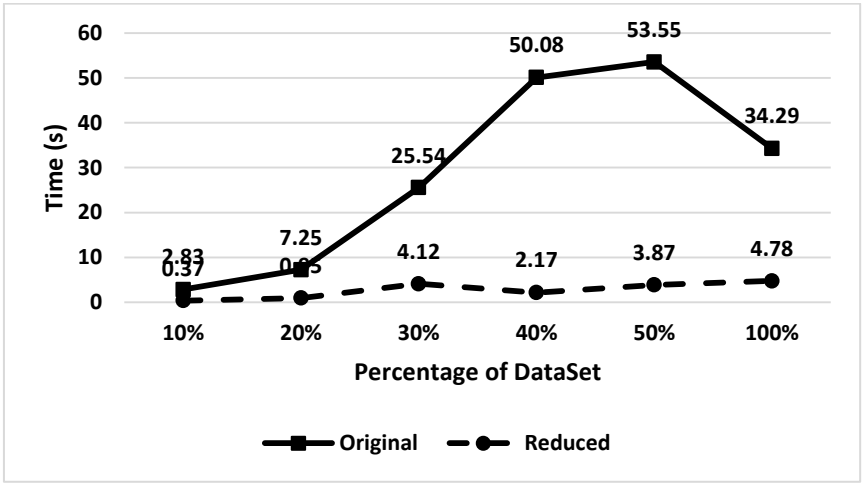
شکل ۱۰: میانگین زمان افزودن سگمنت‌ها به صفحات داده‌ای برای شاخص‌گذاری درصدهای مختلف مجموعه داده

برای بررسی و ارزیابی عملکرد روش شاخص‌گذاری PCPI، از چهار پرس‌وجوی محدوده‌ای، برش زمانی[1] [۳۴]، *kNN* و فاصله زمانی[2] [۲۹, ۳۴] استفاده شد. برای بررسی تاثیر هر یک از این پرس‌وجوها ۱۰۰۰ پرس‌وجو به‌صورت تصادفی در محدوده داده‌های مجموعه داده استخراج و آزمایش‌ها برروی آنها انجام شد. این آزمایش‌ها برای پرس‌وجوهای فوق به‌جز فاصله زمانی برروی درصدهای مختلفی از هر دو مجموعه داده اصلی و کاهش داده شده تکرار شده است که در ادامه به بررسی نتایج آن خواهیم پرداخت. شکل‌های (۱۱، ۱۲ و ۱۳) به ترتیب میانگین زمان پردازش پرس‌وجوهای پرس‌وجوی محدوده‌ای، برش زمانی و *kNN* در هردو مجموعه داده اصلی و کاهش داده شده و برای درصدهای مختلف می‌باشد. همانطور که در شکل‌ها مشخص است در دو پرس‌وجوی محدوده‌ای و *kNN* با افزایش حجم مجموعه

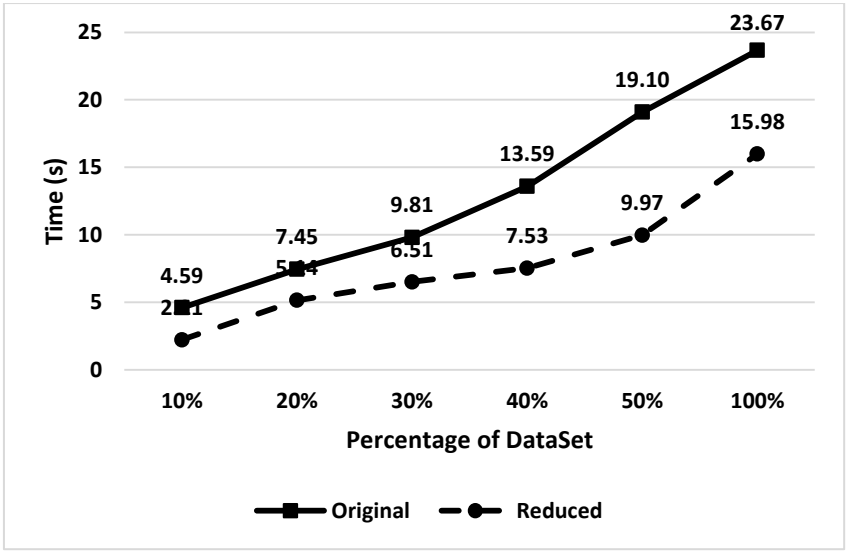
[1] Time-Slice query
[2] Time Interval Query

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

داده زمان پردازش در مجموعه داده اصلی نسبت به مجموعه داده کاهش داده شده افزایش قابل توجه خواهد داشت. اما در پرس‌وجوی برش زمانی به طور کلی زمان پردازش مجموعه داده اصلی بیشتر از مجموعه داده کاهش داده شده است. ولی با افزایش حجم مجموعه داده‌ها، نسبت افزایش زمان پردازش هر دو مجموعه داده تقریبا برابر است. همانطور که در همه این شکل‌ها مشخص شده است در مجموعه داده اصلی با افزایش حجم مجموعه داده زمان نسبت به مجموعه داده کاهش داده شده افزایش بیشتری داشته است.



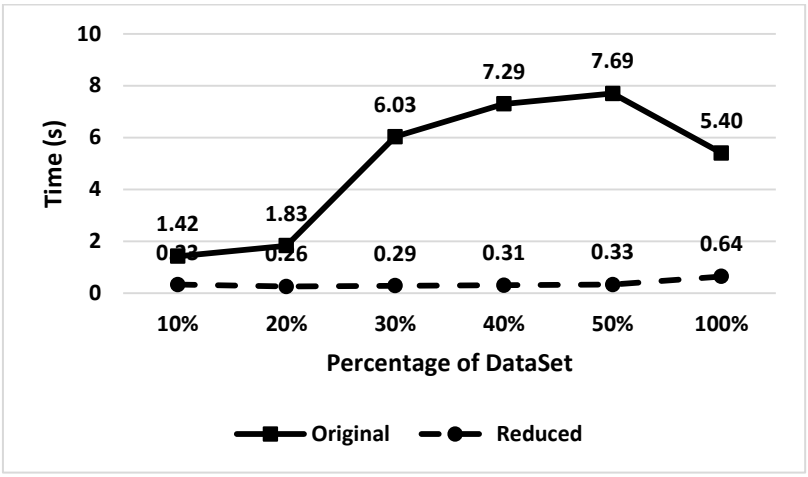
شکل ۱۱: میانگین زمان پردازش پرس‌وجوی محدوده‌ای



شکل ۱۲: میانگین زمان پردازش پرس‌وجوی برش زمانی

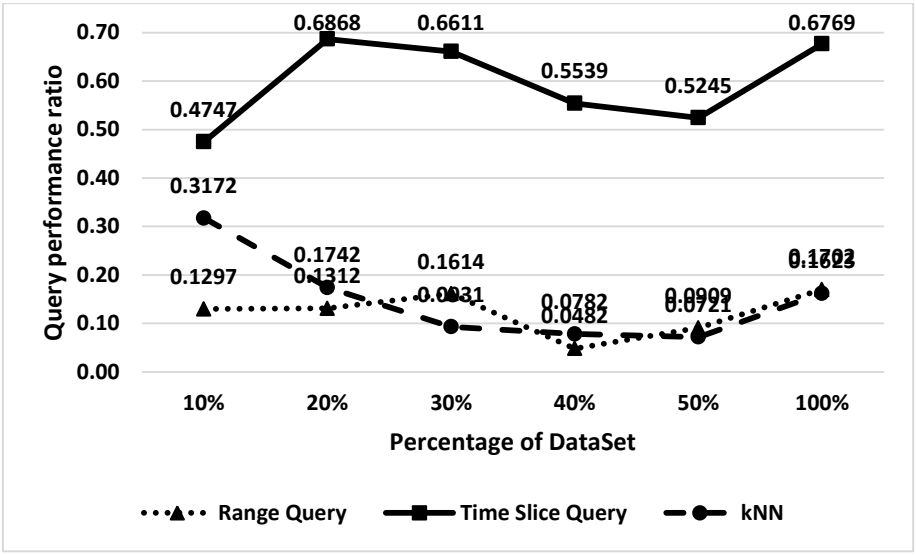
[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]



شکل ۱۳: میانگین زمان پردازش پرس وجوی kNN

برای بررسی بهتر این موضوع برای هر یک از ۱۰۰۰ پرس‌وجوی انجام‌شده در درصدهای مختلف مجموعه داده، نسبت عملکرد پرس‌وجو نیز محاسبه شد که در شکل (۱۴) نتایج آن برای هر سه پرس‌وجو ارائه شده است.

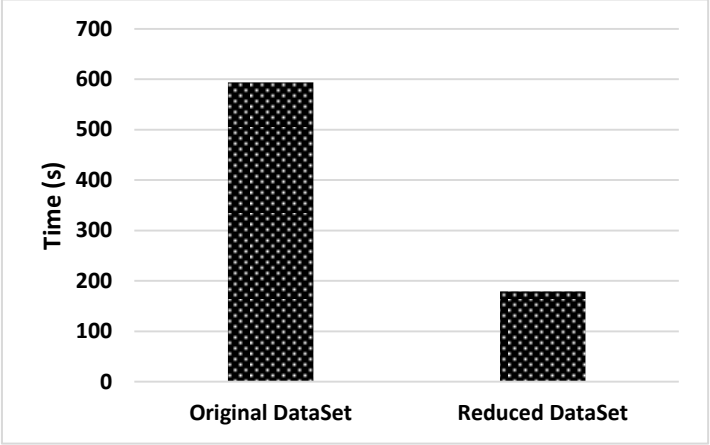


شکل ۱۴: نسبت عملکرد پرس‌وجوی برروی مجموعه داده اصلی و کاهش یافته شده

اما برای پرس‌وجوی فاصله زمانی برای دریافت پاسخ درست باید آزمایش‌ها برروی کل مجموعه داده اصلی و کاهش داده شده انجام می‌شد. در نتیجه میانگین زمان پردازش آن در کل هر دو مجموعه داده در شکل (۱۵) آمده است. همچنین دقت پاسخ این پرس وجو در مجموعه داده کاهش داده شده نسبت به مجموعه داده اصلی بیش از ۹۸ درصد بوده است.

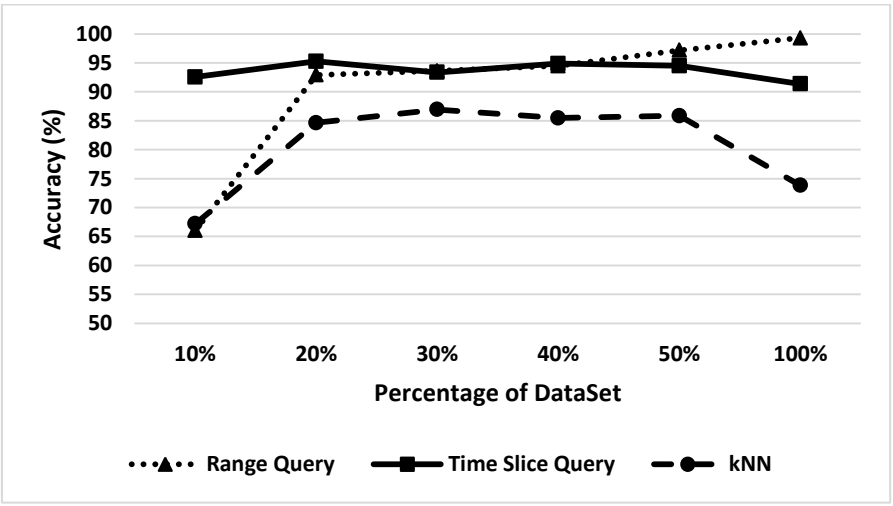
[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]



شکل ۱۵: میانگین زمان پردازش پرس وجوی فاصله زمانی

همچنین شکل (۱۶) نمودار میانگین دقت این پرس‌وجوها برای درصدهای مختلف را نشان می‌دهد. همانطور که در شکل مشخص می‌باشد دقت پرس‌وجوی برش زمانی در درصدهای مختلف بیش از ۹۰ درصد است و همچنین برای پرس‌وجوی محدوده‌ای به‌جز ۱۰ درصد در سایر موارد بیش از ۹۰ درصد می‌باشد و در واقع با افزایش حجم دقت نیز افزایش می‌یابد. فقط در پرس و جوی kNN در بیشتر موارد دقت بیش از ۸۰ درصد بود و در کل مجموعه داده نیز هنوز بیش از ۷۰ درصد بوده که با توجه به میزان کاهش انجام شده دقت قابل قبولی است.. دلیل این کاهش اندک نیز نقاط پرس وجویی می‌باشند که به‌صورت تصادفی انتخاب شده است و با انتخاب نقاط واقعی دقت افزایش خواهد یافت. همچنین دلیل دیگر کاهش دقت در هر سه پرس وجو خطاهایی هستند که حتی بعد از مرحله پیش پردازش هنوز در مجموعه داده وجود دارد. بطور کلی با توجه دقت‌های به‌دست آمده در آزمایش‌ها و همچنین میزان قابل توجه افزایش سرعت که با کاهش داده بدست می‌آید و کاهش فضای مورد نیاز برروی دیسک برای نگهداری شاخص‌ها، استفاده از روش‌های کاهش داده برای شاخص‌گذاری و در نهایت پاسخ به پرس‌وجوها در بیشتر کاربردها و سرویس‌های LBS/ITS مناسب بوده و پیشنهاد می‌گردد. فقط در کاربردهایی که دقت نزدیک به ۱۰۰ درصد مورد نیاز باشد روش‌های معمول کاهش داده مفید نخواهد بود.



شکل ۱۶: میانگین دقت پرس‌وجوی محدوده‌ای، برش زمانی و kNN برای درصدهای مختلف مجموعه داده

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]
[DOI: 10.29252/jgit.5.2.141]

۵- نتیجه گیری

با توجه به حجم بالای داده‌های مکانی-زمانی و تنوع بالای سیستم‌های هوشمند حمل و نقل و سرویس‌های مبتنی بر مکان، در این مقاله روشی جدید با نام PCPI برای شاخص‌گذاری داده‌های خط‌سیر اشیاء متحرک در دو زمان گذشته و حال به‌صورت همزمان ارائه شد. در این روش برای ساخت و نگهداری شاخص‌ها از منابع پردازشی یکپارچه استفاده می‌شود. در این روش از دو داده ساختار برای شاخص‌گذاری استفاده شده است. یک داده ساختار در حافظه اصلی وظیفه شاخص‌گذاری موقعیت جاری اشیاء متحرک را به عهده دارد. این داده‌ساختار، دارای دو بخش ایستا و پویا است که هدف آن حداکثر کردن بخش ایستا وکمینه ساختن بخش پویا برای بهینه سازی تعداد بروزرسانی‌ها است. برای داده‌های مربوط به زمان گذشته از ساختار شاخص‌گذاری SETI اصلاح شده برروی دیسک استفاده شد. در این داده ساختار بعد مکان و زمان جدا از هم در نظر گرفته شده است. فضای جغرافیایی موجود به کمک سلول‌های شش ضلعی تقسیم شده و داده‌های هر سلول در داخل صفحات داده‌ای مجزا به‌صورت سگمنت ذخیره شده است. همچنین برای هر سلول با استفاده از درخت+B یک شاخص‌زمانی پراکنده ایجادشده که به داده‌های آن سلول اشاره دارد. در روش PCPI از یک فرآیند پیش پردازشی برای افزایش کیفیت و دقت شاخص‌ها و پرس‌وجوها استفاده شده است. در این روش با استفاده از الگوریتم‌های تطبیق بر نقشه، اختلالات موجود در اطلاعات دریافتی از اشیاء متحرک، حذف می‌شوند. در این مقاله تاثیر الگوریتم‌های کاهش داده برای شاخص‌گذاری مجموعه داده‌های بزرگ مورد بررسی و ارزیابی قرار گرفت. با استفاده از این الگوریتم‌ها، علاوه بر کاهش قابل توجه حجم مجموعه داده‌ها برروی دیسک، پردازش و پاسخ‌گویی به پرس‌وجوها با سرعت بیشتر و دقت قابل بول انجام شد. کارایی بالای روش PCPI و همینطور تاثیر مثبت روش‌های کاهش داده در آزمایش‌های مختلف برروی دو مجموعه داده اصلی و کاهش داده شده، نشان داده شد.

مراجع

[1] C.-I. Cha, S.-W. Kim, J.-I. Won, J. Lee, and D.-H. Bae, "Efficient Indexing in Trajectory Databases.," International Journal of Database Theory and Application, vol. 1, pp. 21–28, 2008.

[2] C. Lv, Y. Xu, J. Song, and P. Lv, "A data frame based spatiotemporal indexing algorithm for moving objects," presented at the 12th World Congress on Intelligent Control and Automation (WCICA), Guilin, China, 2016.

[3] Y. Shi, J. Feng, Z. Ren, and W. Xie, "Hadoop-based Probabilistic Range Queries of Moving Objects on Road Network," 2016.

[4] Y. Zheng and X. Zhou. (2011). Computing with Spatial Trajectories [Online].

[5] C. Parent, S. Spaccapietra, C. Renso, G. Andrienko, N. Andrienko, V. Bogorny, et al., "Semantic trajectories modeling and analysis," ACM Computing Surveys (CSUR), vol. 45, pp. 1-32, 2013.

[6] C. Renso, S. Spaccapietra, and E. Zimányi, Mobility data: modeling management and understanding: Cambridge University Press, 2013.

[7] V. P. Chakka , A. C. Everspaugh, and J. M. Patel, "Indexing Large Trajectory Data SetsWith SETI," in CIDR 2003.

[8] Y. Fang , J. Cao, Y. Peng, N. Chen, and L. Liu, "Indexing the Past, Present and Future Positions of Moving Objects on Fixed Networks," presented at the International Conference on Computer Science and Software Engineering, Wuhan, Hubei, 2008.

[9] M. Pelanis, S. Šaltenis, and C. S. Jensen, "Indexing the past, present, and anticipated future positions of moving objects," ACM

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]

Transactions on Database Systems (TODS), vol. 31, pp. 255-298 March 2006.

[10] D. Lin, C. S. Jensen, B. C. Ooi, and S. Šaltenis, "Efficient indexing of the historical, present, and future positions of moving objects," in Proceedings of the 6th international conference on Mobile data management, ACM, New York, NY, USA, 2005, pp. 59-66.

[11] K.-S. Kim, S.-W. Kim, T.-W. Kim, and K.-J. Li, "Fast indexing and updating method for moving objects on road networks," in Proceedings of the Fourth international conference on Web information systems engineering workshops(WISEW'03), Rome, Italy, 2003, pp. 34-42.

[12] M. Hashemi and H. A. Karimi, "A weight-based map-matching algorithm for vehicle navigation in complex urban networks," Journal of Intelligent Transportation Systems, pp. 1-18, 17 March 2016.

[13] M. Bierlaire, J. Chen, and J. Newman, "A probabilistic map matching method for smartphone GPS data," Transportation Research Part C: Emerging Technologies, vol. 26, pp. 78-98, January 2013.

[14] P. Newson and J. Krumm, "Hidden Markov map matching through noise and sparseness," in 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, Washington, 2009, pp. 336-343.

[15] I. S. Popa, K. Zeitouni, V. Oria, and A. Kharrat, "Spatio-temporal compression of trajectories in road networks," GeoInformatica, vol. 19, pp. 117-145, January 2015.

[16] R. Song, W. Sun, B. Zheng, and Y. Zheng, "PRESS: A Novel Framework of Trajectory Compression in Road Networks," Proceedings of the VLDB Endowment, vol. 7, pp. 661-672, May 2014 2014.

[17] G. Kellaris, N. Pelekis, and Y. Theodoridis, "Map-matched trajectory compression," Journal of Systems and Software, vol. 86, pp. 1566-1579, June 2013.

[18] M. F. Mokbel, T. M. Ghanem, and W. G. Aref, "Spatio-Temporal Access Methods," IEEE Data Engineering Bulletin, vol. 26, pp. 40-49, 2003.

[19] L.-V. Nguyen-Dinh, W. G. Aref, and M. F. Mokbel, "Spatio-Temporal Access Methods: Part 2 (2003 - 2010)," IEEE Data Engineering Bulletin, vol. 33, pp. 46-55 2010.

[20] Y. Manolopoulos, A. Nanopoulos, A. N. Papadopoulos, and Y. Theodoridis. (2005, November 21). R-Trees: Theory and Applications (2006 ed.) [Online].

[21] Y. Tao and D. Papadias, "MV3R-Tree: A Spatio-Temporal Access Method for Timestamp and Interval Queries," in Proceedings of the 27th International Conference on Very Large Data Bases (VLDB '01), Roma, Italy, 2001, pp. 431-440.

[22] M. A. Nascimento, J. R. O. Silva, and Y. Theodoridis, "Evaluation of Access Structures for Discretely Moving Points," in Proceedings of the International Workshop on Spatio-Temporal Database Management (STDBM'99), 1999, pp. 171-188.

[23] D. Pfoser, C. S. Jensen, and Y. Theodoridis, "Novel Approaches to the Indexing of Moving Object Trajectories," in Proceedings of the 26th International Conference on Very Large Data Bases (VLDB '00), Cairo, Egypt, 2000, pp. 395-406.

[24] C.-I. Cha, S.-W. Kim, J.-I. Won, J. Lee, and D.-H. Bae, "Efficient Indexing in Trajectory Databases," International Journal of Database Theory and Application, vol. 1, pp. 21-28, 2008.

[25] X. Xiong, M. F. Mokbel, and W. G. Aref, "LUGrid: Update-tolerant Grid-based Indexing for Moving Objects," in the 7th International Conference on Mobile Data Management (MDM '06), Nara, Japan, 2006, p. 13.

[26] Y. N. Silva, X. Xiong, and W. G. Aref, "The RUM-tree: supporting frequent updates in R-trees using memos," The VLDB Journal

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]

— The International Journal on Very Large Data Bases, vol. 18, pp. 719-738 June 2009 2009.

[27] Z. Song and N. Roussopoulos, "SEB-tree: An Approach to Index Continuously Moving Objects," in MDM '03 Proceedings of the 4th International Conference on Mobile Data Management, Melbourne, Australia, 2003, pp. 340-344.

[28] Y. Tao, D. Papadias, and J. Sun, "The TPR*-Tree: An Optimized Spatio-temporal Access Method for Predictive Queries," in VLDB '03 Proceedings of the 29th international conference on Very large data bases, 2003, pp. 790-801.

[29] Z.-H. Liu, X.-L. Liu, J.-W. Ge, and H.-Y. Bae, "Indexing Large Moving Objects from Past to Future with PCFI+-Index," in Proceedings of the Eleventh International Conference on Management of Data(COMAD 2005), Goa, India, 2005, pp. 131–137.

[30] H. Ferhatosmanoğlu, D. Agrawal, Ö. Eğecioğlu, and A. El Abbadi, "Optimal Data-Space Partitioning of Spatial Data for Parallel I/O," Distributed and Parallel Databases vol. 17, pp. 75-101, January 2005.

[31] M. R. Abbasifard, H. Naderi, Z. Fallahnejad, and O. Isfahani Alamdari, "Approximate aggregate nearest neighbor search on moving objects trajectories," Journal of Central South University, vol. 22, pp. 4246–4253, 08 November 2015 2015.

[32] R. Sedgewick and K. Wayne, Algorithms, 4 ed.: Pearson Education, 2011.

[33] H. Cao, O. Wolfson, and G. Trajcevski, "Spatio-temporal data reduction with deterministic error bounds," The VLDB Journal—The International Journal on Very Large Data Bases, vol. 15, pp. 211-228 September 2006.

[34] Z. Song and N. Roussopoulos, "SEB-tree: An Approach to Index Continuously Moving Objects," presented at the Proceedings of the 4th International Conference on Mobile Data Management, 2003.

[35] M. R. Abbasifard, B. Ghahremani, and H. Naderi, "A Survey on Nearest Neighbor Search Methods," International Journal of Computer Applications, vol. 95, pp. 39-52, June 2014 2014.

[36] N. Bhatia and V. Ashev, "Survey of Nearest Neighbor Techniques," International Journal of Computer Science and Information Security, vol. 8, pp. 1-4, 2010.

[37] S. Dhanabal and S. Chandramathi, "A Review of various k-Nearest Neighbor Query Processing Techniques," Computer Applications, vol. 31, pp. 14-22, 2011.

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]

K. N. TOOSI UNIVERSITY OF TECHNOLOGY
FACULTY OF GEODESY AND GEOMATICS ENGINEERING

Journal of Geospatial Information Technology
Vol.5, No.2, Summer 2017

Indexing the past and current position of moving objects in large-scale dataset

Abbasifard, M.R.[1], Naderi, H.[2]*

1- Ph.D. Candidate of Software Engineering, Iran University of Science and Technology, Tehran, Iran
2- Assistant professor in School of Computer Engineering, Iran University of Science and Technology, Tehran, Iran

Abstract

By increasing intelligent transportation systems (ITS) and location based services (LBS) that take advantage of spatio-temporal data, these data have increased the necessity for new indexing techniques. Indexing methods index these data generally in the past, present or future. Creating an integrated index for indexing data and also answering to various queries which can reduce indices' updating time, is one of the challenges. The current study introduces an integrated method called "PCPI" (Past and Current Position Indexing) to index and store spatio-temporal data of the past and present in a simultaneous manner in the disk and main memory respectively that has ability to answer various spatio-temporal queries. PCPI uses a same resources for processing and creating indices in two different times. In this method, two data structure is used integratedly: the first data structure indexes and stores current position of moving objects in the main memory, and the second data structure on disk for trajectory data of moving objects that have high volume and cannot be stored in main memory. In addition, PCPI uses map matching methods to remove noises – e.g. stationary state noises- in the data received from the moving objects; this feature adds to accuracy and reliability of the query results. Effects of data reduction techniques on accelerating indexing and query processing and reducing disk space consumption (in massive datasets) were examined. Results of the comparisons made based on the experiments showed higher efficiency of the indexing structure.

Key words: Moving Objects Databases, Trajectory Indexing, Spatio-Temporal Data , Query Processing.

Correspondence Address. Moving Objects Database (MODB) Lab., School of Computer Engineering, Iran University of Science and Technology, Tehran 1684613114
Tel: +98-21-73225353
Email: naderi@iust.ac.ir

[Downloaded from jgit.kntu.ac.ir on 2026-11-12]

[DOI: 10.29252/jgit.5.2.141]