Multidisciplinary Journal of Research in Engineering and Technology
ISSN: 2348-6953
Volume 13 Issue 2s (2026) | 384-395

Co-rider: A Privacy-Aware Real-Time Multi-User Location Tracking System with Adaptive Update Optimization

Yash Yogesh Wase[1], Dipannita Mondal[2], Sairaj Jadhav[3], Akash Suradkar[4], Gaurav Ghorpade[5]

[1,2,3,4,5]Department of Artificial Intelligence & Data Science, Dr. D. Y. Patil College of Engineering & Innovation, Talegaon, India.
[1]yashwase13@gmail.com, [2]hod_aids@dypatilef.com, [3]Sairajjadhav433@gmail.com, [4]akashsuradkar586@gmail.com, [5]gauravghorpade642005@gmail.com

Peer Review Information

Type: *Article*
Received: *18 April 2026*
Revised: *06 May 2026*
Accepted: *15 June 2026*
Published: *11 July 2026*

Abstract

Group mobility scenarios, such as bike trains, touring rides, and convoyed travel, often encounter synchronization interruptions due to traffic, divergent pathways, asymmetric riding times and speeds, and distributed situational awareness. Existing location navigation and live location sharing tools are designed for focused individual use, and do not provide scalable, real-time shared coordination services for dynamic groups on the move. This paper presents Co-Rider, a web-first, real-time convoy coordination service designed to enable multiple riders to exist within a single shared navigation environment.

The system introduces Adaptive Update Optimization (AUO), a flexible synchronization procedure that adapts GPS point transmission (or “poll rate") to current convoy conditions based on convoy spread, rider pace, and pairing divergence. In contrast with services sharing locations at relatively fixed poll rates, this design expedites network communication while staying true to convoy steering context, to ultimately optimize on battery duration.

The platform incorporates real-time socket-based convoy coordinate sharing, shared node navigation, convoy-optimized rider tracking, message broadcasting, and session-specific rider operations into a comprehensive mobility coordination architecture. The lightweight, web first implementation uses Next.js, Node.js, Socket.IO, Leaflet.js, and OpenStreetMap. This work examines how flexible synchronization and convoy-centric navigation paradigms can enhance group mobility coordination in aWeb application, yet be fast and scalable enough for free-tier implementation. It lays the foundation for future mobility coordination research.

Keywords: Real-Time Location Tracking; Adaptive Update Optimization; Convoy Coordination; GPS Synchronization; Group Mobility; Socket-Based Communication

How to Cite This Article

Wase, Y. Y., Mondal, D., Jadhav, S., Suradkar, A., & Ghorpade, G. (2026). Co-rider: A privacy-aware real-time multi-user location tracking system with adaptive update optimization. *Multidisciplinary Journal of Research in Engineering and Technology,* 13(2s), 384–395.

Introduction

The advancement and proliferation of mobile navigation technologies in recent years has revolutionized individual mobility by improving geographic information handling. Navigation applications such as Google Maps or dedicated GPS navigators provide accurate route finding, traffic analysis and reporting, location sharing and other individual identification services that have enhanced personal GPS usage. Nevertheless, these navigation systems have been optimized to allow individual access and have not been combined to allow multi-user convoy navigation.

In real world convoy applications such as side by side bike touring groups, highway road trips, long-distance travel convoys, or synchronized mobility teams, online convoy grouping necessitates continuous multi-subscriber synchronization. Due to traffic, multiple vehicle locations and navigation latency resulting in divisions in convoy distance, or by mismatched routing decision, convoy groups are split up and will need to re-sync afterwards. Still, existing systems are not providing any multi-user convoy aware continuous synchronization or process-aligned situation awareness.

Standard location sharing systems use brute-force, non-adaptive GPS polling at fixed-intervals, where subscriber sends location messages continuously at scheduled times no matter what is the current state of diversion. Such systems waste a lot of network bandwidth and radio as well as consumes additional battery due to simple time-driven polling and high level of redundancy in message transmission. Additionally, such approaches do not scale well under a multi-user, multi-route convoy.

This paper presents Co-Rider, a convoy travel coordination framework to provide convoys with a shared navigation environment where multi-subscriber convoy in the same shared space operating in the same timeframe with consistent segments of time moving toward shared destination. One of the main contributions of this work is designing an adaptive update algorithm (AUO) that calculates dynamic GPS scheduling process that optimizes the system scalability as well as maintains a continuous, real time synchronization of different tracking sub-tasks, by estimating the movement speed, a variety of convoy suspension factors and target divergence outside the convoy.

The following sections will describe the design and implementation details of the system architecture, the methodology, the proposed synchronization dynamics, the implementations and observed experiments as well as directions for future system scalability.

Problem Statement

Group travel environments involve sustaining real-time convoy coordination among large numbers of interdependent, constantly moving parties. Navigational tools available today cannot solve these convoy mobility issues because they focus on single rider preferences or individual wayfinding.

Here are some predictable problems that occur in group-riding scenarios:

- Riders diverge due to traffic conditions or speed differential.
- There is no holistic ensemble-based convoy discernment.

There are inefficiencies in re-grouping:

- Periodic knowledge of position causes excessive network calls.
- Battery life diminishes due to GPS feedback.
- Scalability limitations in multi-user situations with high update frequency.
- No collaborative environment exists for navigation.

Most location-sharing systems are not convoy-aware; they are passive data observers. They are not equipped with:

- Adaptive concurrent updates,
- Collaborative ride management,
- Bidirectional communication,
- Convoy-sensitive routing,
- Real-time situational awareness,
- Public message broadcasting.

This persisting situation necessitates a specialized, synchronization-oriented mobility solution for group convoy operation.

Objectives

The general goal of such a proposed system is to design a scalable and efficient real-time convoy coordination system that will allow multiple uniquely and dynamically moving riders to continuously maintain synchronized group mobility with an intelligent mobile multimedia navigation platform. The system should include continuous situation awareness, adaptively synchronized status updates, as well as a unified mirroring interaction within the concurrent VMI platform.

The specific goals of proposed work are summarized as follows:

1. Design a real-time synchronized mobility environment, which has the capability to provide multiple operators a simultaneously riding session and continuously monitor the convoy status and members' coordination.

2. Implement an Adaptive Update Optimization (AUO) algorithm, which dynamically adjusts GPS synchronization update rate, that depends on instantaneous riders' velocity, inter-riders' distance, and Convoy divergence in a convoy setting, to reduce redundant communication and battery drain;

3. Develop a mobile prioritized, web-oriented infrastructure with a highly scalable, real-time delivery framework, and with the latest real-time streaming protocols supported, which is additionally capable to effectively support real-time, low latency synchronization, broadcasting and alert propagation, and mobile first interaction.

Literature Survey

Current navigation and location sharing solutions have enabled massive advances in individual route guidance and geospatial accessibility in the last decade. services like Google Maps, Life360 and other real time tracking platforms provide navigation guidance, route updating and limited real-time location sharing to individuals. Existing information-sharing platforms are better suited for individual mobility than amp; cluster synchronized group mobility. Various studies on real-time communication and distributed systems detail the need for low-latency synchronization facilities suitable for amp; multi-user contexts.

WebSocket-based architectures have proven to be efficient solution for such multi-user applications by facilitating bi-directional communication between client-server over HTTP. Other benefits include quicker communication latency, low-bandwidth overhead and better synchronization performance than conventional client-server request-response protocols.

Research on GPS-tracing solutions also shows that fixed-interval synchronization demonstrates redundant transmission and response lag in stationary en-vironmental circumstances. As a result, continuous periodic location dispatching leads to over-utilization of bandwidth, energy, propagation time and system resources for amp; multi-user operations.

The field of enterprise-level fleet control and logistics tracking platforms offers centralized monitoring tools, though these systems consume much customer resources and infrastructure to be feasible. They are not well-suited for lightweight amp; consumer-focused synchronized group mobility systems. Additionally, decentralized and adaptive synchronization schemes are yet to be explored.

The Co-Rider system presents a hybrid solution by offering:

- Direct real-time socket-based control,
- Adaptive gps update streamlining,
- Amp; convoy-responsive amp; rider group merging,
- Group navigation coordination in shared space,
- Nowledge-mobile-first interaction modes,
- Lightweight free-tier publicly available implementation.

By merging adaptive synchronization with convoy-aware rider group merging, this work attempts to address the shortcomings of existing navigation solutions while fulfilling the demand for amp; scalable amp; synchronized convoy mobility system.

Proposed System

The presented solution Co-Rider is a real-time convoy coordination system targeted towards applications of synchronized group mobility, e.g. cross-country cycling, touring convoys or other expedition traveling groups. The conventional approach of individual operation separate navigation solutions for each convoy participant is replaced by the shared navigation environment with all members participating in the same synchronized convoy session. All convoy members constantly share movement updates, receive instant convoy state updates, rider locations and map progress. The architecture incorporates elements of real-time convoy messaging, adaptive real-time synchronize with incremental updates, shared navigation and rider monitoring in a single control framework and alert broadcasting hub.

Major modules of the proposed solution and its mobile-first architecture are as follows:

- Authentication
- Ride Session control and create

- Real-time synchronize engine
- Adaptive update optimization module
- Shared navigation & convoy progress browsing
- Rider monitoring and location update engine
- Alert messaging broadcast and dispatch center
- Ride progress summary & analytics engine

The system mainly addresses convoy cohesion and rider separation minimization through constant localized synchronized awareness.

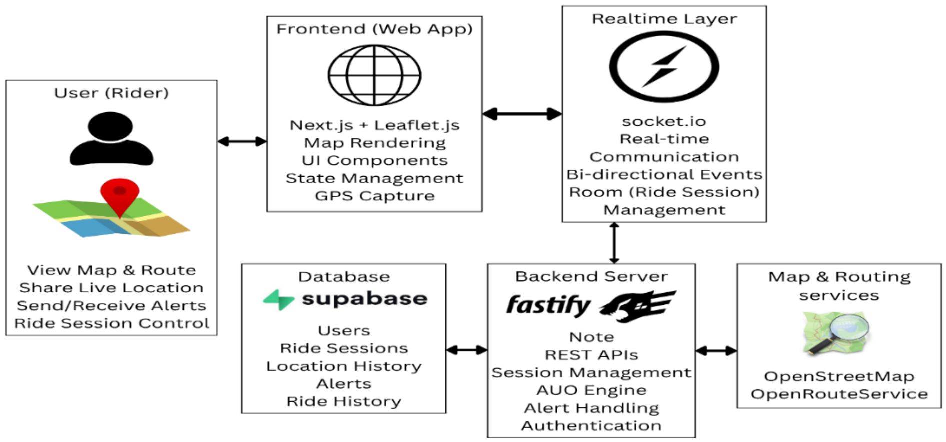

Fig. 1. *System Tech Stack*

System Architecture

The architecture of Co-Rider operates within a layered client-server architecture designed to achieve realtime synchronization over a lightweight, mobile-hardware first design. The design supports low-latency communication, scalable, session-based coordination, and multi-user synchronization while remaining lightweight enough to operate on the internet-equipped infrastructure.

The architecture is composed of four high-level layers:

- Frontend Layer
- Backend Layer
- Realtime Communication Layer
- Database & Mapping Layer

Frontend Layer

The frontend of the Co-Rider service is implemented in Next.js, using TypeScript, Tailwind CSS, Zustand, and Leaflet.js. The frontend performs:

- Responsive Presentation Of The Interaction
- Gps Coordinate Retrieval
- Route Visualization
- Rider Rendering
- Map Interaction
- Socket Interaction
- State Management

The service has a mobile-first design approach in recognition of the point-to-point drive situation where the the actual interaction only occurs over a smartphone. Presented interfaces are overly broad to be usable over a tablet or desktop, although not optimized for such situations.

Backend Layer

The backend of the service is implemented through a Node.js application with a Fastify plugin. The backend performs:

- Authentication Management

- Session Creation
- Ride Management
- Api Management
- Realtime Synchronization Management
- Alert Generation and management

The backend is also the primary controlling layer over the Adaptive Update Optimization logic.

Realtime Communication Layer

The realtime connection between riders and between a rider and the server is implemented via Socket.IO. In a typical Co-Rider usage scenario, each run session represents an independent broadcast room from which rider movement, alerts, and session states are broadcast across.

Database & Mapping Layer

Data are stored both temporarily in a Supabase Postgres database and persistently in the database through a variety of database tables and relationships. These data models persist:

- User Data
- Session Data
- Ride History
- Alert Data
- Chat Data
- User Preference Data

Map rendering, rendering visualizations, and navigation direction visualization are implemented through Leaflet.js, and OpenStreetMap.

The architecture explicitly does not store time-sensitive repeated GPS streams for the entire duration of a ride in order to minimize transit database overhead.

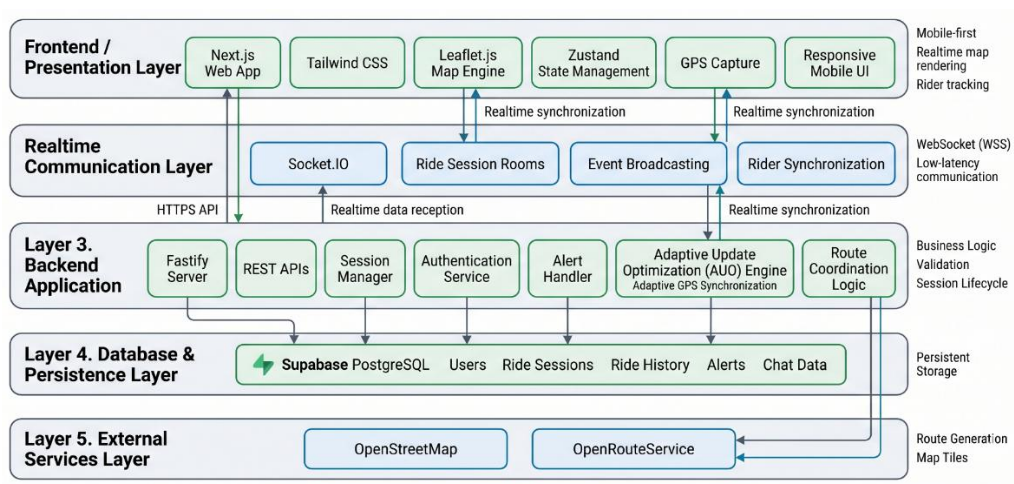

***Fig. 2.** Layered Architecture for corider's workflow*

Adaptive Update Optimization (AUO)

Adaptive Update Optimization (AUO) is the leading technical contribution of this system. It automates the synchronization process by modulating GPS update frequency based on current range, speed, and other dynamics.

Conventional location-sharing approaches use standard fixed-interval synchronization, which transmits GPS coordinates at regular time frames without regard to convoy progress or rider position stability. This method introduces high communication overhead and battery-drain inefficiencies.

The proposed AUO mechanism enables more efficient synchronization rate determination by considering the following parameters:

- Rider Speed,
- Rider Relative Range,

- Convoy Spread,
- Rider Differential Speeds.

Operation

Convoy To Maintain Stability

Convoy is stable when riders are:

sprintf('roughly %d meters apart',range),

co-directional,

traveling at similar speeds,

leads to an increased synchronization time interval.

Convoy is not stable when riders are:

sprintf('moving away from each other'),

going off course,

abruptly accelerating or decelerating,

which results in an update transmission rate increase.

This adaptive communication property maximizes efficiency while maintaining convoys with competing demands on satellite bandwidth, and phone applications.

Advantages

AuO develops a reduction in:

- Apparent GPS Reading Frequency,
- Capacity Loading,
- Data Bandwidth consumption,
- function usage,
- mobile battery drain,
- CONVIF platform scalability,
- free-tier deployment, and
- convoy management awareness.

AuO is the computational heart of the Co-Rider synchronization engine, and forms a significant implementation differentiator for proposed convoy.

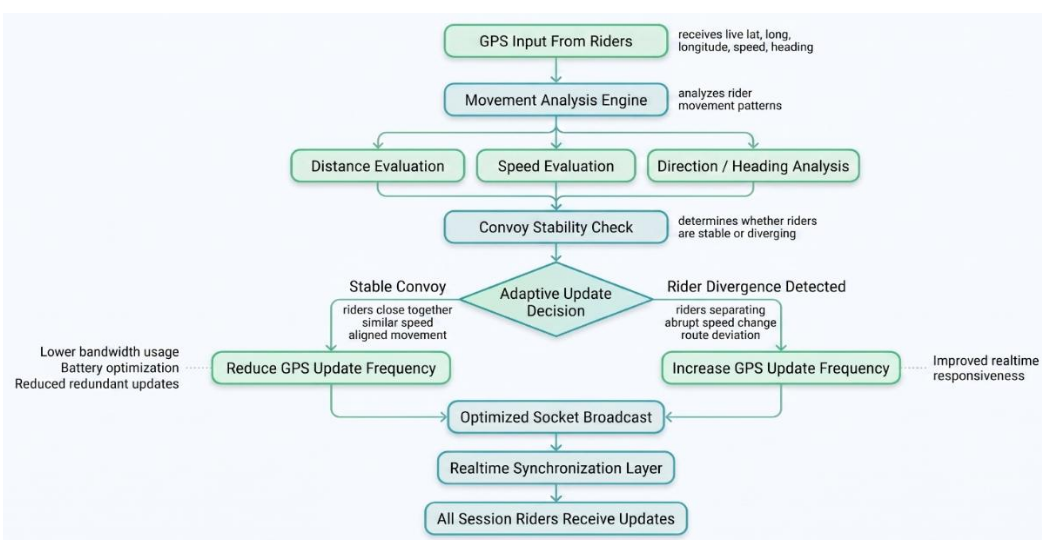

Fig. 3. *Adaptive Update Optimization (AUO) Workflow*

Parameter	Fixed Interval	AUO
Battery Usage	High	Lower
Redundant Updates	High	Reduced
Scalability	Limited	Improved
Network Load	Higher	Optimized

Implementation Methodology

Design process of Co-Rider included a modular, mobile-first development methodology with decentralization, responsiveness, scalability and real time synchronization efficiency.

The development process was modularized to several stages for system clean integration and maintainability. Then the development process followed the path:

Stage 1: UI & UX Design
Focusing on designing a map-oriented, mobile-suitable UI system, which involved the following:

- Responsive Layouts
- Floating controls
- Bottom Navigation
- Draggable panels
- Light overlay
- High contrast visual hierachy

The key goal was finishing a functioning environment to be used outside during riding.

Stage 2: Front End Development
Front End system was set up and using re-usable React component as basic elements, with states managed by zustand, on which the separate parts have encapsulated designed, such as:

- Authentication State
- Ride State
- Map State
- Socket State
- Overlay State

In which the map element has used the Leaflet.js for realtime visualizing the riders.

Stage 3: Backend Development
Back End was designed by using Node.js with Fastify, creating a REST site for:

- Authentication
- User profile
- Session
- Ride History
- Alert

Moreover using socket.io to realize parallel communication.

Stage 4: Realtime Synchronization
Socket.io rooms can be used to isolate parallel environment for each ride, Rider can join a room automatically after joining the corresponding ride.

Stage 5: AUO Integrate
Designing adaptive realtime synchronization, dynamically extended the GPS update rate according to the detected rider situation.

Stage 6: Testing & Optimization
Prototype testing focused on the stability of realtime synchronization, tracking reliability, mobile adaptability, robustness of socket connection, realization efficiency and UI performance.

Outputs and Results

The Co-Rider prototype, as a functional implementation of the realtime convoy coordination platform, coherently accomplishes the ability of stepwise maintaining synchronized group mobility in a fluid travelling circumstance.

The prototype system implementation underwent evaluation methodologies based on the criteria of:

- Realtime Synchronization,
- Rider Tracking Stability,
- Communication Latency,
- Mobile-First Usability,
- Auo Synchronization Adaptation,
- User Interface Response.

The prototype implementation authenticates realtime communication from Socket.IO based messaging streams maintaining continuous convoy synchronization via a realtime realtime platform.

The AUO mechanism observed to independently scale synchronization behavior, rapidly fielding redundant GPS position update broadcasts in appropriately stable convoy subgroups while rapidly increasing convergence responsiveness during rider separation events.

The platform prototype toolset fulfilled the requirements of:

- Realtime rider spatial awareness,
- Shared destination convergence,
- Time-based messaging,
- A message broadcast protocol,
- Responsive map interface,
- Mobile-first interactive system usability.

The frontend implementation maintained stable responsiveness for the mobile viewport variations under concurrent realtime operational interaction across active ride sessions.

The system implementation affirms the likelihood of developing a minimal synchronized mobility platform with the support of contemporary web technologies and free deployable hosting infrastructure.

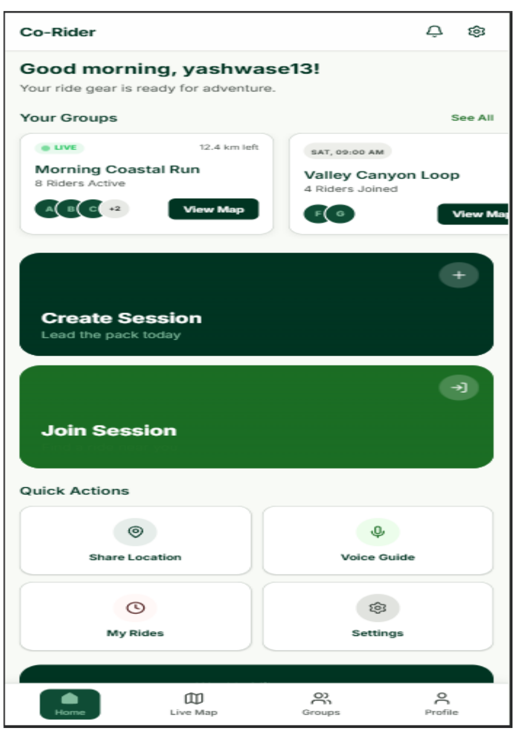

Fig. 4. Co-Rider dashboard interface displaying ride sessions, group management, and convoy coordination controls.

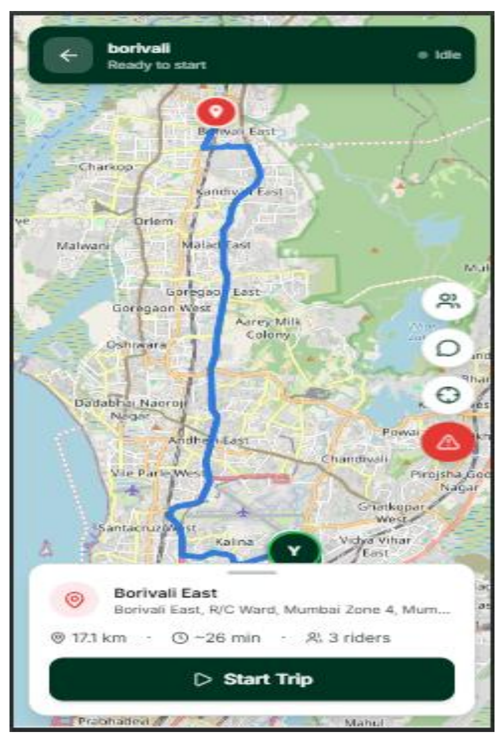

Fig. 5. direction from every user to destination with all the ETA details before starting the trip

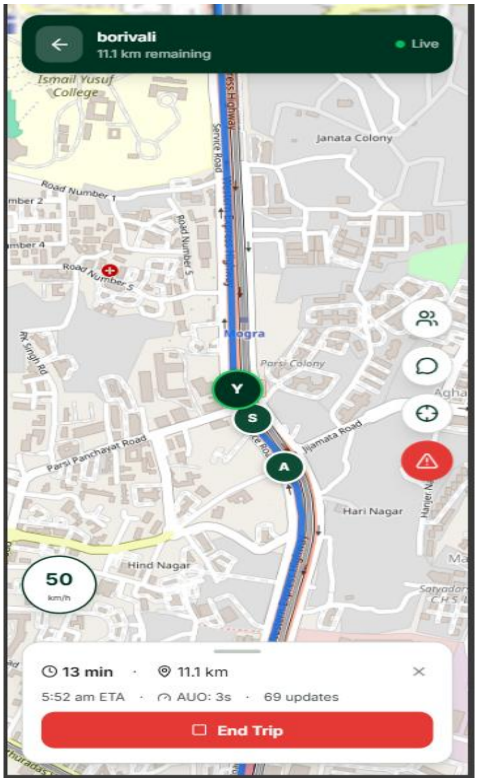

Fig. 6. *Real-time location sharing and updates within a synchronized ride sessions.*

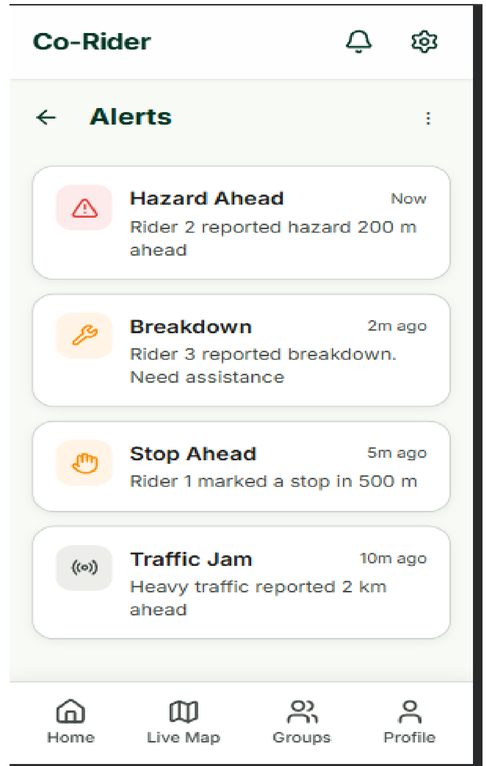

Fig. 7. *Real-time alert broadcasting system integrated within synchronized ride sessions.*

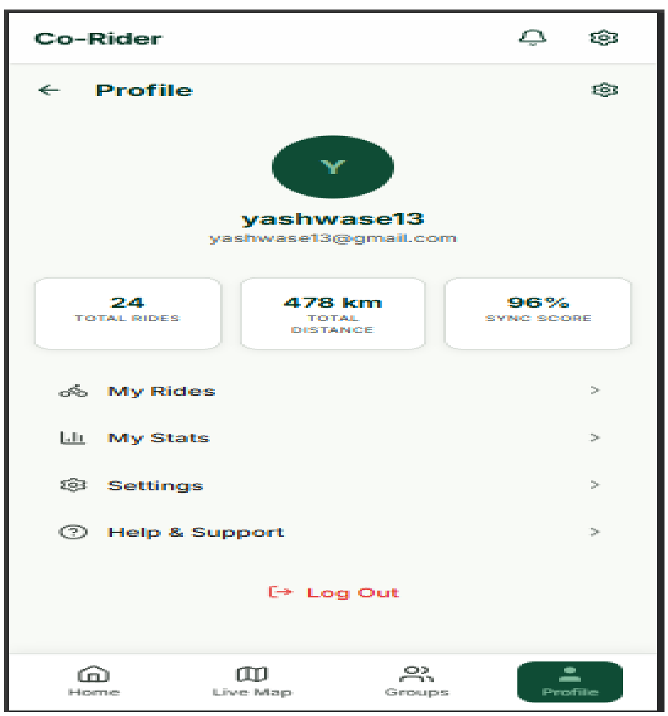

Fig. 8. User profile and ride analytics interface displaying synchronized mobility statistics and ride information.

Advantages of the Proposed System

The Co-Rider platform offers a number of operational as well as technical benefits relative to conventional navigation and real-time location sharing approaches; these include:

Technical benefits
Flexible synchronization

The automatically controlled AUO method adjusts GPS transmission timing to convoy driving dynamics to improve efficiency and reduce redundant update overhead.

Real-time interaction

A socket.io based system allows rapid, bi-directional ridersync within shared ride session groups.

Lightweight operation

An open-source stack was applied to produce a scalable deployment; Next.js, socket.io, leaflet.js, OpenStreetMap, Supabase.

Scalable session framework

Userspace sessions focused inward as discrete dyads or small groups on separate server broadcast buses allow scalable modular multi-group operation.

User benefits

Enhanced convoy awareness

Live profile positioning and other ride observer / communicator tools enable constant oversight of convoy status on the road.

Rapid assembly

Temporary ridersync and live profiles on demand at the destination more easily bring separated riders together.

Shared destination environment

A common destination interface enhances optimal shared-way mobility effort.

Session-wide user alerts

Instant broadcast of riding hazards, planned stops and breakdowns to all session members.

Mobile first access

Lean interactive web interface with expandability, no app installation needed.

Limitations

While the current MVP system proves effective as a demonstration for the system capabilities, there are some limitations on the system design that can affect adaptability. One limitation is the browser leve background GPS restriction, as browsers on mobile devices optimizes battery management by minimizing background synchronization frequency, which may effect location broadcast precision for long screen-inactive times. The minimum reliance on socket.io realtime connection requires the synchronization to be in the same network environment, as experiencing packet loss or delay until socket reestablished when facing weak network environment. Other delays include:

- Dependence on device GPS precision
- Free-tier scale constraints
- Lack of offline routing capabilities
- Browser throttling strategy
- Lack of extensived high volume testing experiences

These limitations restrict wide scale ride groups current deployment in prototype form. However, this platform proves adaptable to allow use across small and medium sized ride groups in a prototype level deployment environments for the use of easily integrated adaptation.

Future Scope

The work by the proposed Co-Rider platform offers a broad infrastructures upon which future work can be based, for development of other synchronized group mobility systems and intelligent convoy coordination platforms.

In the future, Co-Rider system can adapt to include additional features and intelligent capabilities, including:

AI-powered Convoy Intelligence

ML, predictive analytics, and data mining techniques may be used to analyze convoy strategies and movement trends, to proactively detect and correct issues such as:

- Rider exclusion,
- Convoy instability,
- Delivery disruptions due to traffic,
- Synchronization delays in cooperation.

ML-based suggestion engines could power optimized group regroup and convoy stabilization.

Offline Budget Synchronization

The system can be evolve to support inherent offline technique, allowing for reliable operation in low bandwidth environments through:

- Local Synchronys Buffers,
- Sinc Delay Window,
- Oblivious State Handling,
- Partially Peer-to-peer, backup assisted operation.

Native Mobile Infrastructure

In the future application you may develop:

- Responding Android/ios applications,
- Background optimization,
- Tighter GPS integration,
- Improved notification services.

Native support will facilitate device normalization and stable mobility tracking.

Advanced Ride Analytics

Analysis and visualization deliverables on future iteration may include:

- Convoy Unity Metrics,
- Round-Trip Optimization Calculations,
- Heatmaps,
- Time-Distance Travel Features,
- Efficiency & Sustainability Coaching.
- Enhanced Analytical tools measure operational success and sustainability improvement.

Wearable Market Tilt

Additional developer focus may be directed toward the extensive development of wearable ecosystems such as:

- Smart Watches,
- Smart Helmet Rovers,
- 2nd-Screen Bluetooth Systems,
- Hud-Enabled Gps Systems.

This reduces reliance on the phone and enables reader access in difficult environmental conditions.

Horizontal Decentralized Infrastructure

In the future your system would begin to leverage, for scalability and stability:

- Redis Powered Horizontal Socket Infrastructure,
- Bringing Fast Horizontal Scaling To The Backend,
- Having A Distributed Synchronization Protocol,
- Multiple Local/Region Expanding Duplication,
- Very Large Convoy Support.
- Convoy scalable worldwide.

This work establishes a solid aerodynamic foundation upon which future intelligent group margin systems can be built.

Conclusion

This paper has proposed Co-Rider, a realtime convoy coordination platform to enhance synchronized groupmobility through adaptive realtime inter-rider communication and convoy-aware session synchronization strategy. The platform overcame the key limitations of existing free navigation and location sharing applications, by presenting:

- Shared navigation environment,
- Ealtime rider synchronization,
- Adaptive gps optimization,
- Centralized alert broadcasting service,

- Session-based convoy coordination.

The implementation of adaptive update optimization successfully eliminated redundant user-device synchronization overhead without sacrificing realtime responsiveness and convoy-awareness, as result, the system achieved bandwidth savings, energy savings and also realized scalable user-device synchronization. The web-first approach successfully utilized lightweight open-source technologies to develop realtime convoy coordination platform at no additional cost for infrastructure investment. The field experiment proved that realtime socket-based synchronization, map-centric user-focused interaction, adaptive communication optimization and session-oriented convoy management could be combined together in a common platform for free infrastructure while supported at a reasonable scale.

With the promising result shown in off-the-shelf deployment, this study contributed to the research body of dispersed synchronized mobility systems, as a platform to branch further into the intelligent convoy management approaches. Furthermore, the demonstrated convoy capable navigation environment could significantly surpass the traditional individual-centric navigation experience.

References

1. Manali Gopinath Pednekar, Sanjay Bodke, Rushikesh Shelake, Ashwin Shedge, "A Real Time Tracking System using Group Coordination," *International Journal of Engineering Research & Technology (IJERT)*, Vol. 14, Issue 02, 2025. Available: IJERT Research Paper (IJERT)
2. Idriss Moumen, Najat Rafalia, Jaafar Abouchabaka, Marouane Aoufi, "Real-time GPS Tracking System for IoT-Enabled Connected Vehicles," *E3S Web of Conferences*, 2023. DOI: 10.1051/e3sconf/202341201095 | Available: E3S Conference Paper PDF (E3S Conferences)
3. Nazneen Pendhari, Azaz Ahmad, Belal Ansari, Farhan Shaikh, Mohammad Saad Shaikh, "SYNCFLEET: Real-Time Group Tracking and Context-Aware Anomaly Detection for Synchronous Mobility in Travel Scenarios," *International Journal of Computer Technology*, 2025. DOI: 10.21917/ijct.2025.0549 | Available: SYNCFLEET Research Paper PDF (ICTACT Journals)
4. "REAL-TIME TRACKING FOR EFFICIENT PUBLIC TRANSPORT MANAGEMENT," *International Journal of Novel Research and Development (IJNRD)*, 2025. Available: IJNRD Paper PDF (IJNRD)
5. "Developing a Cross-Platform Vehicle Tracking System using Node.js and WebSocket," *ISR Journals*. Available: ISR Journal Paper (ISR Journals)
6. Narayan Ghatul, Shyam More, Dipali Adhav, S. S. Gadekar, "Real Time Vehicle Tracking System," *International Journal of Progressive Research in Engineering Management and Science (IJPREMS)*, Vol. 04, Issue 07, 2024. DOI: 10.58257/IJPREMS35557 | Available: IJPREMS Research Paper PDF (IJPREMS)
7. Zheng Dong, Yan Lu, Guangmo Tong, Yuanchao Shu, Shuai Wang, Weisong Shi, "WatchDog: Real-time Vehicle Tracking on Geo-distributed Edge Nodes," 2020. Available: Arxiv Paper - WatchDog (arXiv)
8. Wenjie Luo, Bin Yang, Raquel Urtasun, "Fast and Furious: Real Time End-to-End 3D Detection, Tracking and Motion Forecasting with a Single Convolutional Net," 2020. Available: Arxiv Paper - Fast and Furious (arXiv)
9. Nicolai Wojke, Alex Bewley, Dietrich Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," 2017. Available: Arxiv Paper - Deep SORT (arXiv)
10. "A Comprehensive Study of Web-Based Real-Time Vehicle Tracking and Analytics System," 2026. Available: ResearchGate Paper (ResearchGate)