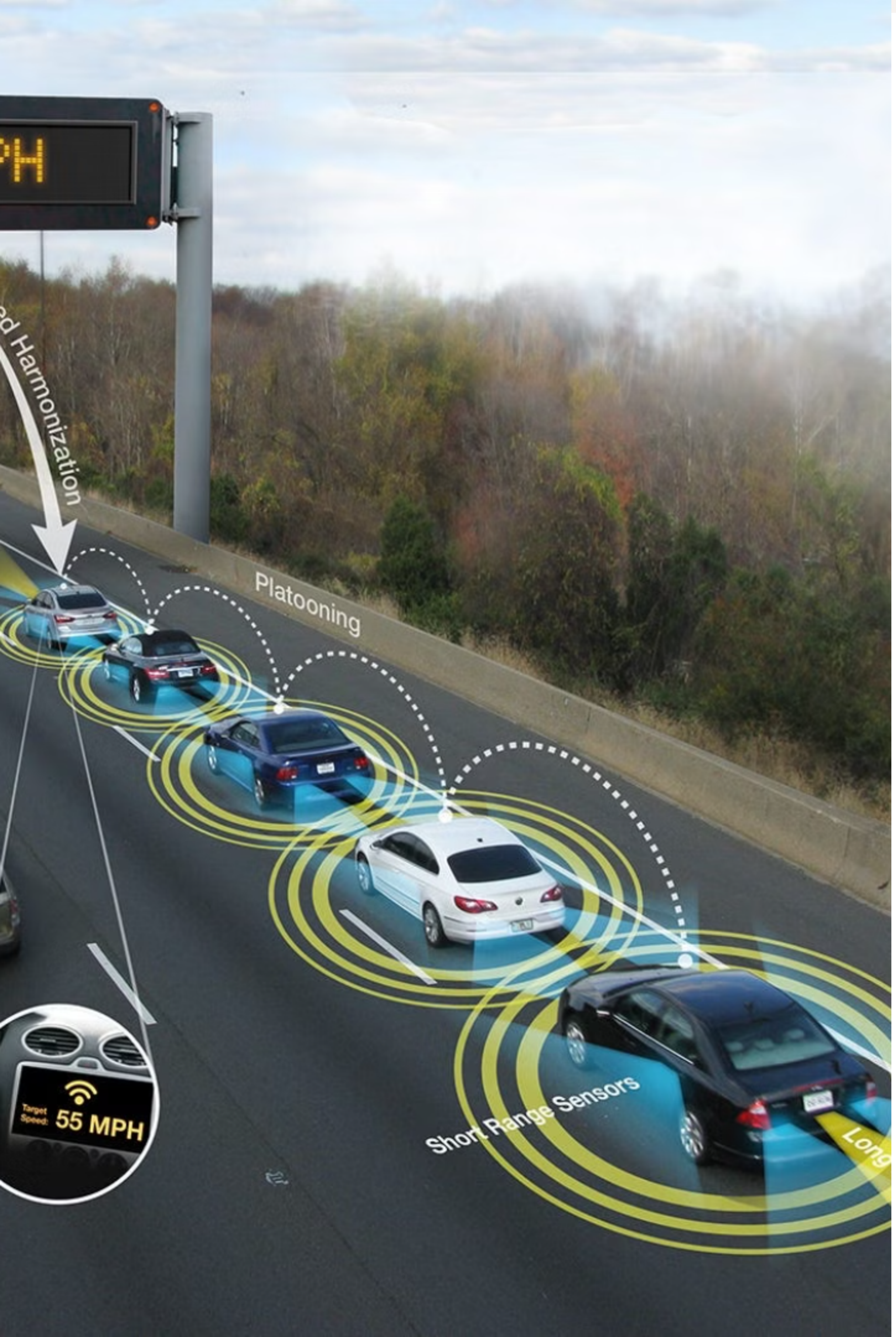


Delay-Optimized V2V Computation Offloading for Autonomous Vehicles

The rapid advancement of autonomous vehicle technology demands efficient communication networks to support resource-intensive and latency-sensitive applications. This project focuses on optimizing computation offloading in a Vehicle-to-Vehicle (V2V) communication environment. By dynamically utilizing idle computational resources of neighboring vehicles, we aim to minimize task execution delays and enhance the performance of self-driving vehicles in urban environments.

The Need for Efficient Communication

Traditional Approach

Traditional computation offloading to centralized cloud systems introduces significant delays due to communication overhead and limited availability of roadside units (RSUs). These delays can negatively impact real-time decision-making in self-driving vehicles.

V2V Communication

V2V communication offers a decentralized, cost-effective alternative by utilizing the idle computational resources of nearby vehicles for cooperative task execution. This approach can significantly reduce delays and improve overall performance.

Project Objectives

1 Simulate Real-World Scenarios

Utilize the VANET Toolbox to realistically simulate V2V communication and vehicular mobility in urban environments.

2 Develop Optimization Models

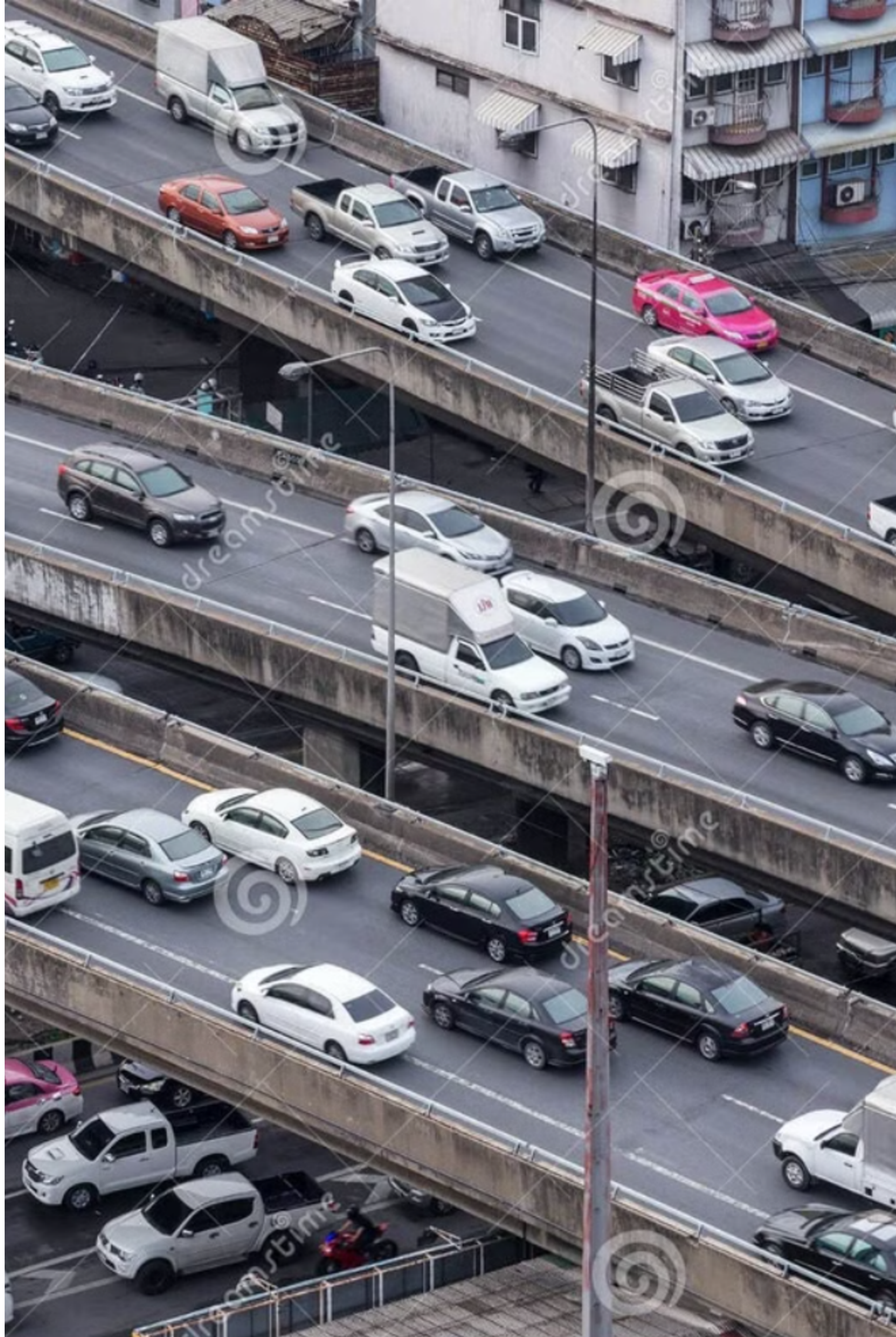
Develop a delay-optimized task offloading model for self-driving vehicles using V2V communication, taking into account dynamic vehicle positions and varying computational capabilities.

3 Implement Optimization Algorithms

Implement and compare optimization algorithms such as Max-Min Fairness and Particle Swarm Optimization (PSO) to optimize resource allocation and minimize task execution delays.

4 Validate and Analyze Performance

Validate the proposed solution through simulations, analyzing the impact of various parameters (vehicle density, task size, mobility patterns) on delay performance.



Methodology

1 System Modeling

Define a vehicular network scenario using the VANET Toolbox, including a task vehicle and multiple service vehicles. Simulate mobility using predefined models (e.g., Intelligent Driver Model) provided by the toolbox.

2 Communication Model

Use the VANET Toolbox to simulate V2V communication, including channel switching, transmission delays (tSCH, tCCH), and noise effects.

3 Computation Model

Model local computation delays (tlocal) and offloading computation delays (tedge) for various task segmentation ratios. Implement resource allocation schemes to optimize task execution.

4 Optimization Algorithms

Use MATLAB scripting within the VANET Toolbox environment to implement Max-Min Fairness and PSO for task allocation. Incorporate constraints like communication range and service time to ensure realistic simulations.

5 Simulation and Evaluation

Design urban vehicular scenarios with varying vehicle densities, mobility patterns, and task sizes. Use the VANET Toolbox visualization tools to analyze task execution times and network performance metrics.

Key Tools and Technologies

VANET Toolbox

MathWorks' VANET Toolbox enables the modeling and simulation of vehicular networks, providing tools to create realistic urban environments and simulate communication and mobility patterns.

MATLAB

MATLAB is used for scripting and implementing optimization algorithms, allowing for efficient development and execution of the algorithms within the simulation framework.

Simulink

Simulink is a powerful tool for dynamic simulations of vehicular communication and computation processes. It allows for visual modeling of the system, enabling comprehensive analysis of the interactions between various components.

Expected Outcomes and Deliverables

1. Comprehensive Simulation
2. Performance Comparison
3. Trade-off Analysis
4. Deliverables

This project will deliver a comprehensive simulation model demonstrating the effectiveness of V2V task offloading in reducing delays for self-driving vehicles. It will provide quantitative metrics comparing the performance of Max-Min Fairness and PSO algorithms. The project will also offer insights into the trade-offs between computational delay and network utilization in urban scenarios. Key deliverables include a simulated V2V-based computation offloading scenario using the VANET Toolbox, implementation of delay optimization algorithms in MATLAB, and detailed performance analysis reports showcasing the impact of vehicle density, task size, and mobility patterns on delay optimization.

Conclusion

1

V2V-Based Offloading

This project integrates the VANET Toolbox with MATLAB to model and optimize V2V-based computation offloading for self-driving vehicles.

2

Enhanced Performance

By minimizing task execution delays through cooperative resource utilization, the proposed solution contributes to the advancement of efficient and reliable autonomous vehicular networks.

3

Future Directions

Future research could explore the integration of machine learning techniques for adaptive resource allocation and the development of energy-efficient offloading strategies to prolong vehicle battery life.

Next Steps

Moving forward, the project team will focus on further refinement of the simulation model, including exploring a wider range of scenarios with different vehicle densities, mobility patterns, and task complexities. The team will also investigate the use of advanced machine learning techniques for adaptive task allocation and dynamic resource management in the V2V network. The ultimate goal is to develop a robust and scalable solution for delay-optimized computation offloading in self-driving vehicles, enabling seamless and efficient operation of future autonomous transportation systems.