3-10-2008

Mixed Reality, Now a Reality Network Virtualization for Real-Time Automotive-CPS Networks

Rahul Mangharam
University of Pennsylvania, rahulm@seas.upenn.edu

Recommended Citation

Rahul Mangharam, "Mixed Reality, Now a Reality Network Virtualization for Real-Time Automotive-CPS Networks", . March 2008.

Suggested Citation:
Mangharam, R. (2008). Mixed Reality, Now a Reality Network Virtualization for Real-Time Automotive-CPS Networks. *NSF-NCO/NITRD National Workshop on High Confidence Automotive Cyber-Physical Systems*. April 2008.

Mixed Reality, Now a Reality Network Virtualization for Real-Time Automotive-CPS Networks

Comments

Suggested Citation:
Mangharam, R. (2008). Mixed Reality, Now a Reality Network Virtualization for Real-Time Automotive-CPS Networks. *NSF-NCO/NITRD National Workshop on High Confidence Automotive Cyber-Physical Systems*. April 2008.

Mixed Reality, Now a Reality
Network Virtualization for Real-Time Automotive-CPS Networks

Rahul Mangharam
Department of Electrical & Systems Engineering
University of Pennsylvania
rahulm@seas.upenn.edu

March 10, 2008

1. Automotive-CPS Networks

Over 6.4 million automotive accidents occur in the US annually. On average there are three people involved (two drivers and one passenger). That translates to roughly 19.2 million Americans injured in car accidents. Odds of you being in an accident this year are 1 chance in 16. As most traffic accidents and incidents are of immediate local importance, inter-vehicular communication is expected to be a key driver in enhanced active safety for the vehicle of the future.

Automotive-CPS networks are a special class of networked Cyber-physical Systems where the maximum relative speeds are in excess of 80m/s, the node density can span over 9,000 vehicles/mi^2 and, most importantly, the dynamics of the vehicle, the environment, driver reaction and interaction with other vehicles need to be considered in every communication and control decision. Auto-CPS applications may be broadly classified into in three categories: (a) time-critical on-road safety alerts (b) time-critical inter-vehicle coordination/control loops (c) non-critical persistent local traffic updates and multimedia exchange for fleet communications.

The principal properties required by Auto-CPS networks are: **(i)** End-to-end delay bounds for multi-hop communication

(ii) Reliable multi-hop networking for tightly-coupled inter-vehicular control and coordination

(iii) Predictable vehicle response to traffic dynamics, environmental conditions and driver behavior

(iv) Maximum transmission concurrency among broadcasters

Unlike Internet Protocols, Auto-CPS networks experience rapid topology change with ephemeral routing tables and cannot employ unicast or multicast protocols from one node address to another. The three primary challenges with auto-CPS networks are:

1. Delivering real-time guarantees on safety alerts and inter-vehicle coordination with broadcast-only protocols.

2. Developing scalable methods to model and verify the interactions between the physical world, vehicle group dynamics, driver reaction and Auto-CPS network protocols for safe semi-automatic vehicle control.

3. Maintaining shared state across local vehicle clusters and also across large city-scale vehicle networks.

To provide such guarantees there is a need to express timeliness and node-independent coordination across the network. Furthermore, determining the appropriate local powertrain response to safety alerts and vehicle trajectories requires network abstractions which incorporate both communication and physical dynamics. To build an end-to-end model-based framework for Automotive-CPS, we require both the abstractions and tools that combine the scale of metro-area simulation with real-world test-beds.

2. Position

Our position is that Vehicular Network Virtualization, which combines large-scale simulation of virtual vehicles with real vehicles and executes the same algorithms and protocols across both vehicle types, allows for scalable model-based design of Auto-CPS networks. By modeling the mobility, communications, traffic dynamics, driver behavior and vehicle control of thousands of virtual vehicles we study the scalability and robustness of our system. By designing the same network abstractions and protocols to work in real vehicles, we evaluate the performance of the system with real-world feedback and validate the models themselves. Now by combining the virtual and real vehicles to seamlessly communicate and interact with the same protocols and algorithms on the same street map, we develop richer models and are also able to rapidly prototype Auto-CPS systems. The validated models may then be used in end-to-end model-based design for semi-automatic vehicle control.

Consider for example Figure 1, where we have only three real vehicles, $R1$, $R2$ and $R3$. While $R1$ and $R2$ are within communication range, $R3$ is over a mile away and not in range. If a Safety Alert is triggered by an air-bag deployment in $R1$, the message is only received by $R2$. If we want to see the progression of the message to all approaching vehicles, we can simulate any number of virtual vehicles on the same road with the same driving models. $R2$ sends the message over a cellular link to the simulator which will then simulate the progression of the message until another real vehicle is in the vicinity of the virtual vehicles. The message is then received by $R3$ over the cellular link but as if it were from $R1$ across multiple hops.

This way we can vary the number of virtual vehicles to study the performance of the protocols and network algorithms under various densities, driving conditions and street topologies. As more experimental vehicles become available, we can gracefully increase the realism and validation of our models. Network virtualization provides us with the best of both model-based design and real-world validation with rapid prototyping. We mask the cellular link's latency by speeding-up the communication across the virtual vehicles.

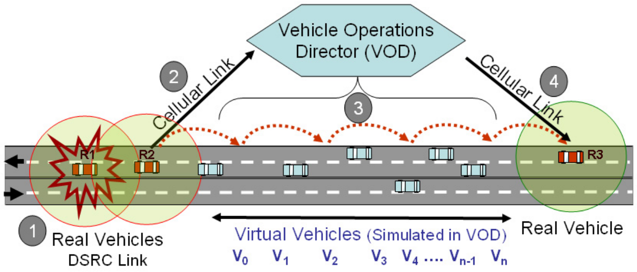

Figure 1. A hybrid simulation with real vehicles communicating across multiple simulated vehicles

Such Network Virtualization will allow us to answer questions such as: Under what driving conditions and market penetration of communication vehicles will Application A achieve the desired performance? How does the probability distribution of Model M compare with the real-world? Is the resultant powertrain response safe and under what conditions is it unsafe?

3. Fundamental Limitations

The complexity of Auto-CPS is beyond the state of current real-time theory and practice in reasoning about large-scale mobile network protocols with multiple dynamical inputs. Domain-specific communication protocols such as CAN-bus and FlexRay are typically limited in geographic extent and confined to single-vehicle networks. Our current rate-based and aperiodic event models need to be extended to events whose priorities change based on environment factors. Finally, the operating systems in each vehicle today operate at a fixed operating point. In the Auto-CPS context, the dynamic range of environment variables and vehicle group dynamics is large and requires RTOSes to scale beyond multi-modal operation to adaptive responses.

4. Research Challenges

Challenge 1- *New Class of Real-Time Network Protocols:* Auto-CPS networks introduce new constraints on network protocols that distinguish them from conventional point-to-point connection and connectionless protocols. The high rate of change in network topology and the fact that vehicles just entering or exiting a highway drastically alter the routing tables, make path-based and hierarchal address-based route discovery infeasible. If we rely on broadcast-based protocols, there is a need to introduce spatio-temporal structure for per-hop packet scheduling. In cases where traffic incidences result in disrupting traffic patterns for extended periods of time, the protocols must ensure the event alert is persistent in the region of interest. Finally, for bulk data exchange and stream-based communication, the protocols must support 'suspend-resume' operations due to frequent connectivity changes in the network. The major challenge is in developing a suite of high-confidence protocols even though the underlying physical network is unreliable.

Challenge 2- *New Models for Run-Time OS and VM:* Today's embedded RTOSes are (a) both logically and physically linked to a single vehicle and (b) function at only a single set of operating points which are determined at design-time. In order for groups of vehicles to efficiently adapt to environment conditions and share different degrees of state, adaptive RTOSes need to be developed with parametric and programmable control of runtime services. This enables *fluid networks* which, based on network and environmental factors, adapt the level of coupling between nodes.

Challenge 3- *Distributed and Adaptive Scheduling Theory:* Real-Time models and scheduling theory must be advanced in many ways. As vehicles trigger Auto-CPS safety alerts based on the local state, the receiver must consider the vehicle state and the world state both before and after the response. To coordinate chassis control with network-based active safety, navigation information and the context of other vehicles, it is necessary to determine the task priorities dynamically based on the environment.

5. Innovations and Abstractions

A bold new research agenda in real-time software and system design must be set to create innovations and abstractions to meet these challenges.

Vehicular Network Virtualization An architecture for network virtualization allowing for seamless model-based design, testing, validation and code migration between virtual and real vehicles is needed. As both vehicle types are based on the same assumptions of street topologies, driving terrains, network protocols and vehicle control algorithms, we are able to balance the complexity of wireless communication, group coordination and adaptive RTOSes with the reality of a slow and gradual roll-out of experimental vehicles. Such an approach integrates systematic system design, allows for logical abstractions across groups of vehicles and facilitates network management across large populations of vehicles. Large-scale, always-on, open-source vehicular network testbeds in association with fractional-rental services such as PhillyCarShare and FlexCar will allow for collaborative and competitive Auto-CPS network design.

Adaptive Real-Time Network Protocols As the underlying physical network is unreliable, network protocols must not associate routes with nodes but with primitives such time, position, driving direction and average speed. For example, a spatio-temporal grid structure may be overlaid on the street map to determine communication activity such that messages are delay-bounded along highways. By tightly synchronizing vehicles, we can derive reliable logical abstractions of the network through wireless interference control and end-to-end spatio-temporal schedules across a range of vehicle densities and street topologies.

Distributed VM architectures While the concept of an adaptive RTOS is not new (e.g. utility functions, variable rate-based execution, statistical task models), a level of abstraction is necessary to extend network-wide services such as runtime schedulability analysis, runtime software attestation, dynamic task migration between nodes, policy negotiation, on-line fault diagnosis, protocol adaptation, algorithm activation and data migration. To facilitate this, a distributed virtual machine across tightly coupled nodes in a fleet network will allow for efficient shared state and coordinated decisions across vehicles in the group.

6. Conclusion

The automobile of the future will be a *Programmable Car* where functionality may be purchased and enabled on-the-fly via software services. The Programmable Car will be networked for network-based active safety, coordinated fleet driving and mobile sensing. To realize this goal, new research in Auto-CPS protocols, network virtualization and vehicle virtual machines for parametric and programmable control is needed.

7. Biography

Rahul Mangharam is an Assistant Professor of Electrical & Systems Engineering at the University of Pennsylvania. He obtained his BS, M.S. and Ph.D. degrees from Carnegie Mellon University in 2000, 2002 and 2008 respectively. His interests are in scheduling algorithms and operating systems for embedded and wireless networks. He serves on the PC of RTAS, RTSS, INFOCOM MObile networking for Vehicular Environments (MOVE) and the International Symposium on Vehicular Computing Systems.