FERAL

Dr. Thomas Kuhn

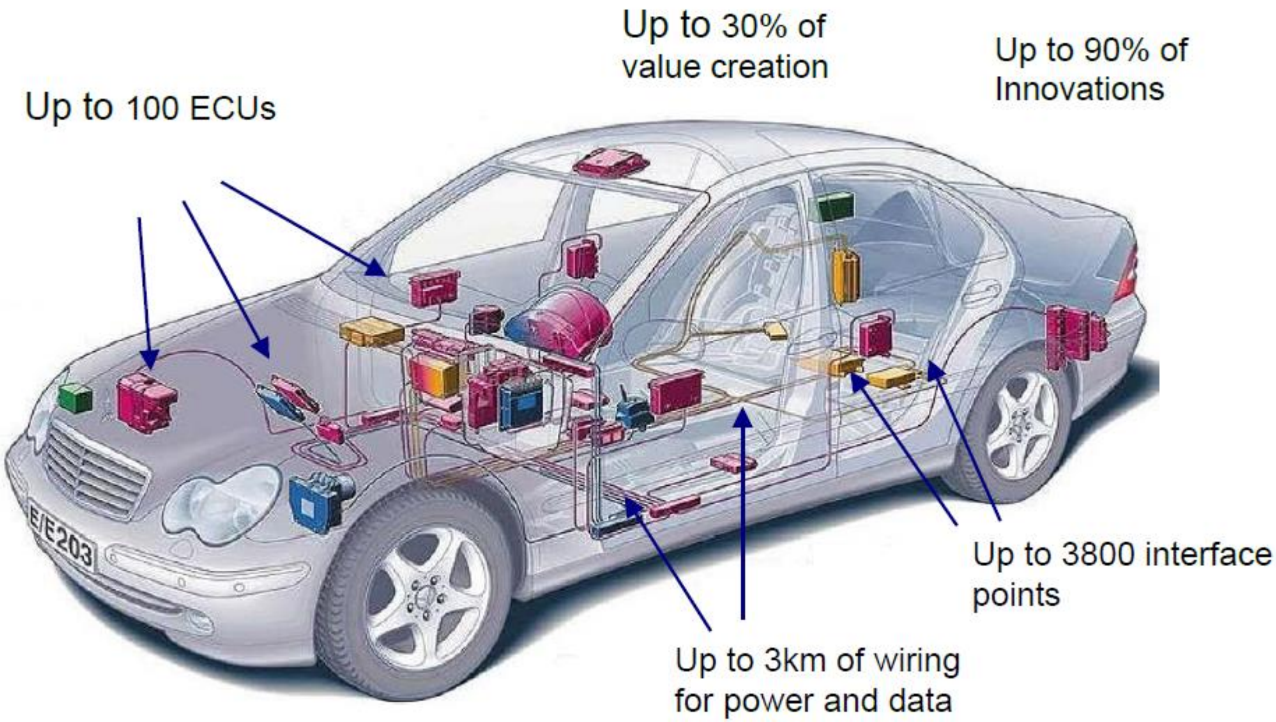
Thomas.Kuhn@iese.fhg.de

Fraunhofer IESE

innovationszentrum applied system modeling

Motivation – Embedded Software Development

Growing importance of Embedded Software



Fraunhofer IESE

innovationszentrum applied system modeling

Motivation – Embedded Software Development

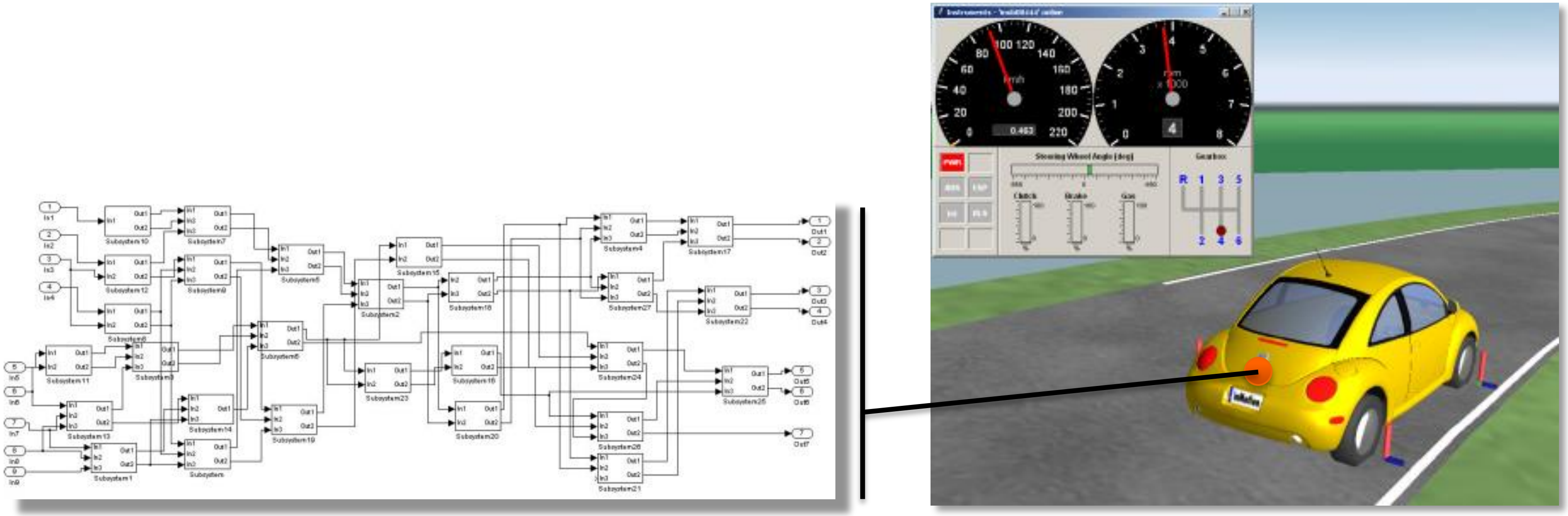
Challenges when developing and testing embedded software

- Embedded (Control) Software is developed using Model Driven Development approaches
  - Simulink, ASCET, Scade
- Tightly integrated with other system components
  - Interacts through sensors and actuators with the environment
  - Interacts with other software components
  - Shares platform and network resources with other software components
- Testing of embedded software needs to take this into account
  - Existing simulation solutions enable virtual testing of embedded software
  - Virtual platforms, Environment simulators, Network simulators

Motivation – Embedded Software Development

Simulators

- Enable early evaluation of embedded software in realistic context
- Provide accurate, yet specialized environments

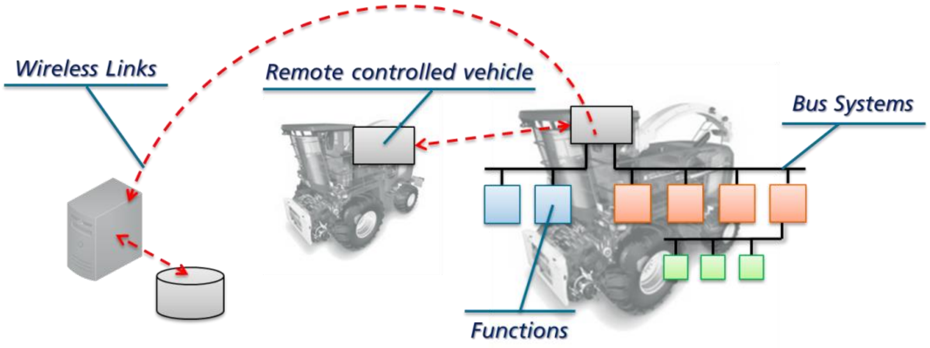


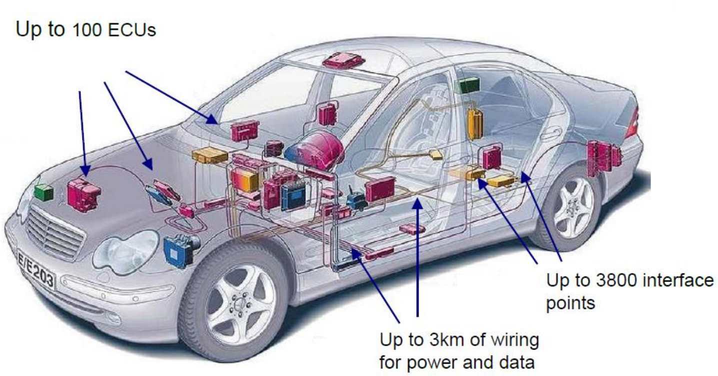
Motivation – Embedded Software Development

But: Embedded Software complexity is increasing

- Open Systems of Systems
  - Wireless links raise safety and security concerns
- Consolidation of functions
  - Potential for significant cost savings
  - How to ensure that concurrently executing functions do not interfere?

} System level concerns





System Level Design and Testing

Evaluating E/E Architecture properties

- Current solutions for embedded systems focus on component development and testing
  - Functional development of individual components
  - Board level + mandatory devices for evaluating behavior of system under test

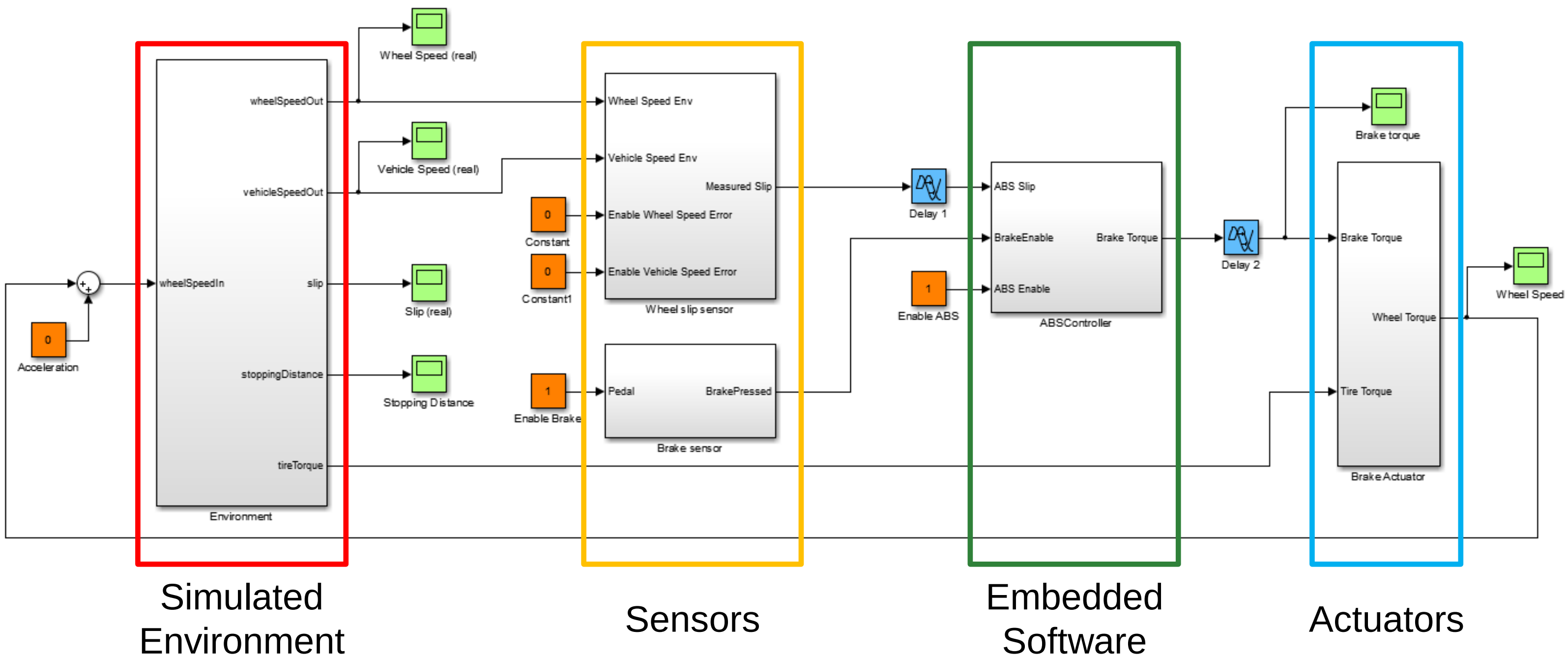
- Next generation embedded systems require more complex E/E Designs
  - How many ECUs are necessary for my product variants and expected growth?
  - Where to consolidate software functions?
  - How to segregate safety relevant functions on same hardware from each other?
  - Which busses are necessary? Wireless access? How to configure and to protect them?

→ System level architecture design and architecture evaluation is getting more important

System Level Design and Testing

System level architecture evaluation requires new simulation approaches

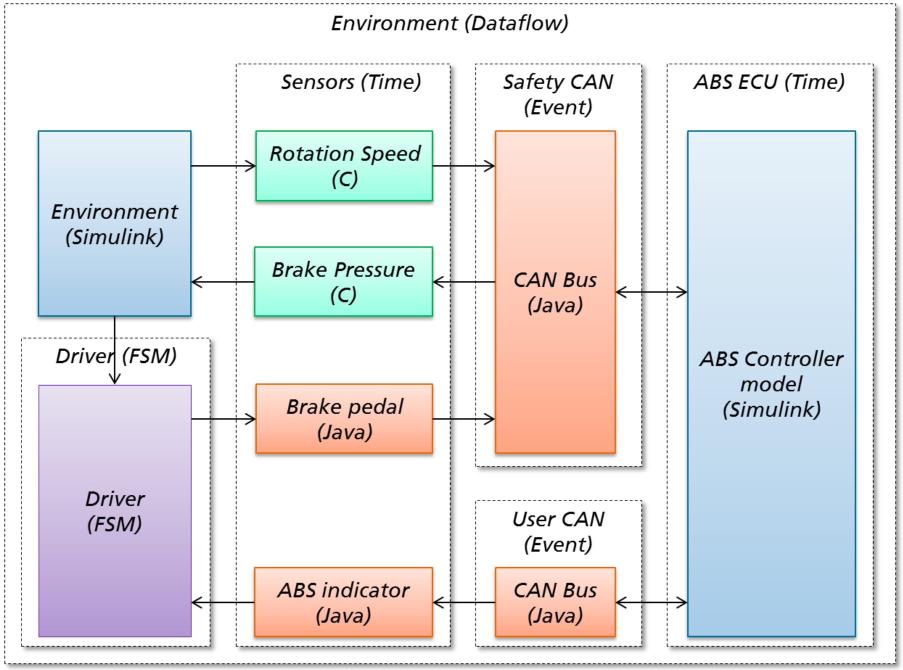
- Let’s consider a (simple) example Simulink system



Simulator Coupling

A simulated deployment of the example system requires many components

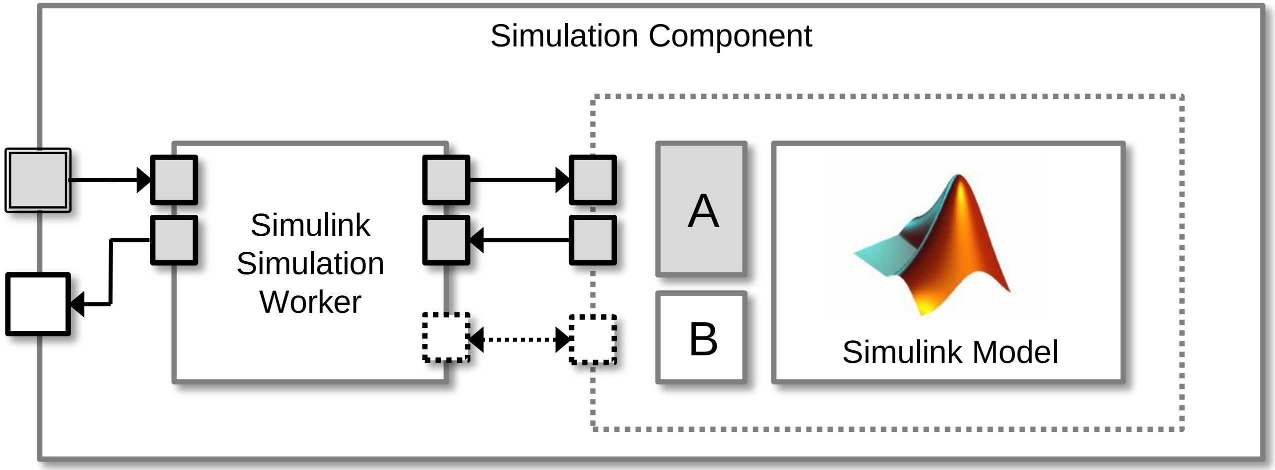
- E/E evaluation on system level requires coupling of specialized simulators
  - One integrated holistic scenario
  - Coupling on different abstraction levels must be supported to manage complexity
  - Project specific development and modeling environments
  - Possibly additional simulators
    - Wireless networks
    - Fault injection
    - …



Simulator Coupling

The FERAL simulator coupling framework

- Simulator coupling requires syntactic and semantic integration
  - Syntactic integration: Simulated network messages, value types, Simulator API
  - Specific to most simulators
  - Encapsulated as simulation components

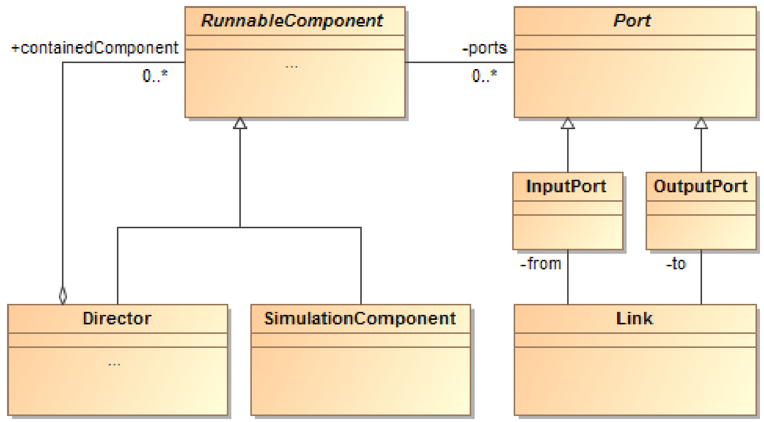


Fraunhofer IESE
innovationszentrum applied system modeling

Simulator Coupling

The FERAL simulator coupling framework

- Semantic integration is provided by directors
  - Encapsulate models of computation and communication
  - Directors may be nested - ensure proper linking of simulator semantics into one integrated scenario
  - This is supported by semantic contract between nested directors

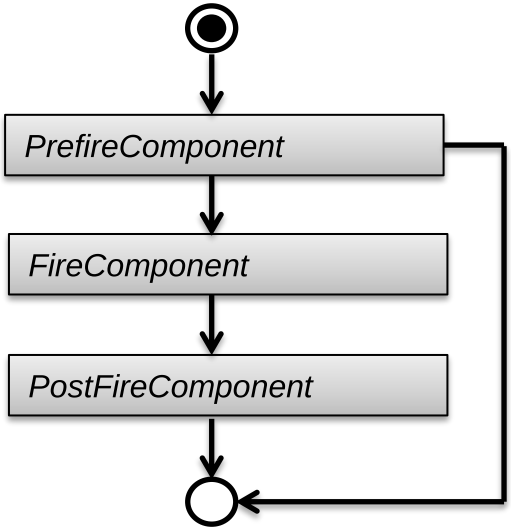


Fraunhofer IESE

innovationszentrum applied system modeling

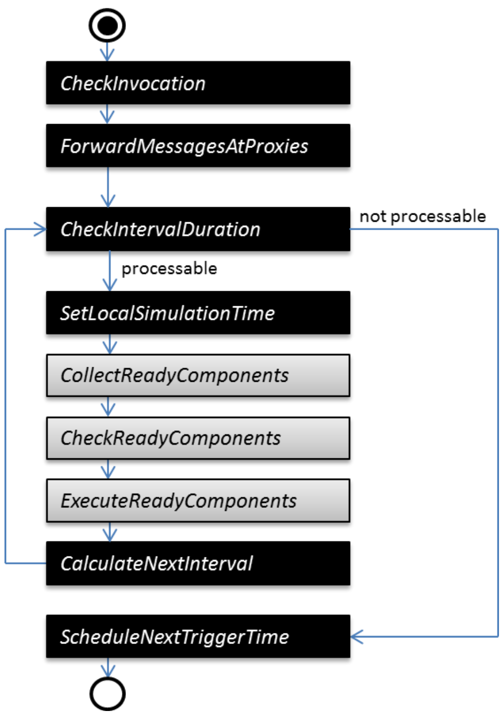
Simulator Coupling

FERAL – Execution of Components

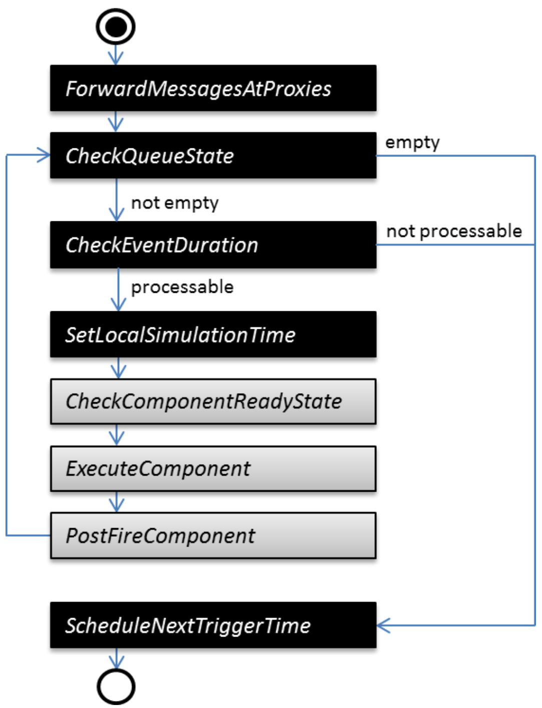


Simulator Coupling

FERAL - Time and Event based Director semantics



Discrete time director



Discrete event director

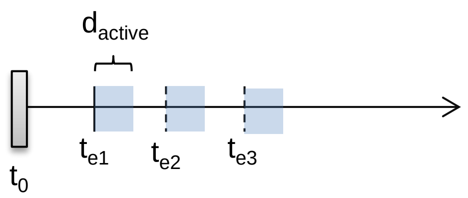
Fraunhofer IESE

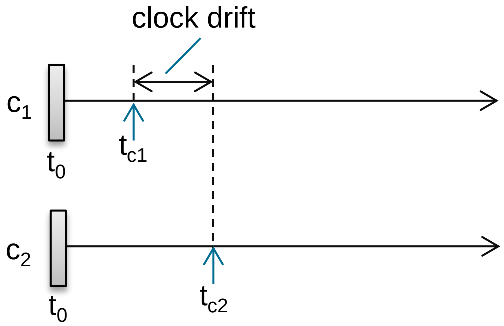
innovationszentrum applied system modeling

Simulator Coupling

Simulator coupling challenges

- Accuracy vs. efficiency
  - Simulator coupling is resource intensive due to synchronization overhead
  - Parts of a scenario require tight coupling, other parts allow a less tight integration
- Feral simulation model is based on Events and active periods
  - Foundation for all directors

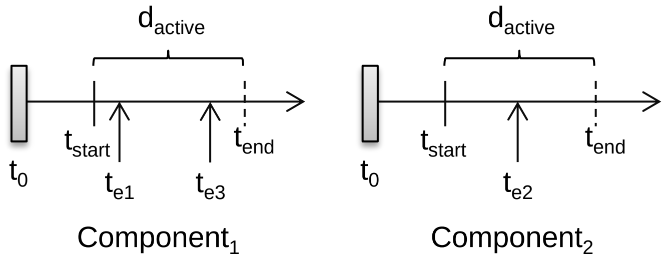


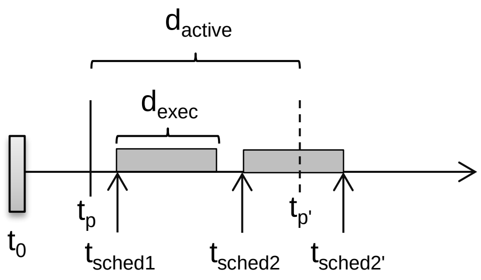


Simulator Coupling

Simulator coupling challenges

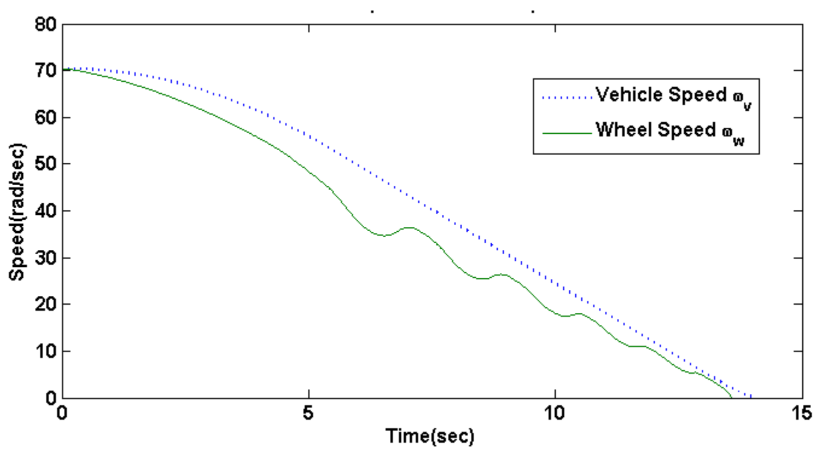
- Clock drift between simulators is permitted inaccuracy
  - Significantly reduces synchronization overhead
  - Enables components to process their active period without interferences
  - Foundation of distributed simulations
  - Deferring of events that exceed active period

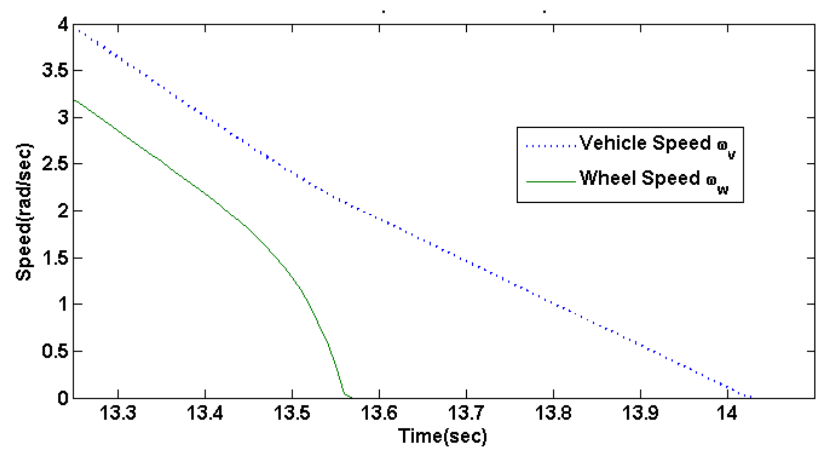




Evaluation

Impact of simulated network behavior to one function



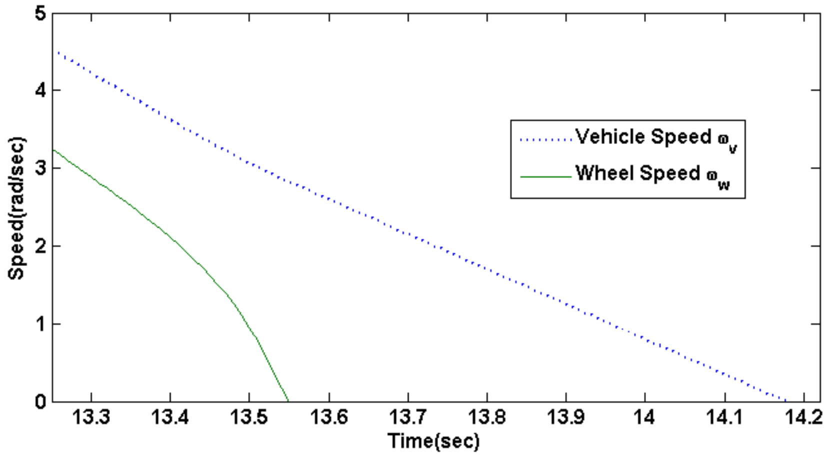


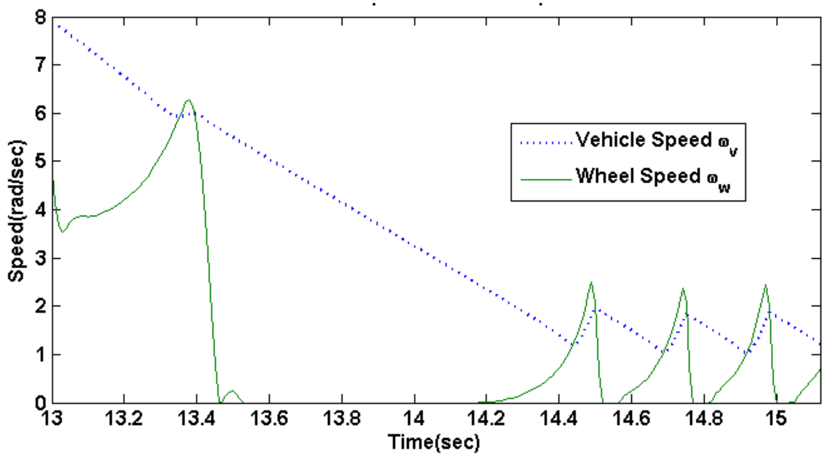
Fraunhofer IESE

innovationszentrum applied system modeling

Evaluation

Impact of simulated network behavior to one function





Conclusion

- Simulations are state of the art in embedded systems development
  - Individual and focused simulators
  - Early evaluation of system level decisions require simulator coupling
- Fraunhofer FERAL enables integration of simulators into holistic scenarios
  - Enables early validation of system behavior or function behavior in system context
  - Predict system behavior in realistic conditions
- Benefits
  - Prediction of communication performance
  - Evaluation of safety concepts
  - Substantiating architectural decisions