

Graduate Theses, Dissertations, and Problem Reports

2016

Application and Control Aware Communication Strategies for Transportation and Energy Cyber-Physical Systems

Ehsan Moradi Pari

Follow this and additional works at: https://researchrepository.wvu.edu/etd

Recommended Citation

Moradi Pari, Ehsan, "Application and Control Aware Communication Strategies for Transportation and Energy Cyber-Physical Systems" (2016). *Graduate Theses, Dissertations, and Problem Reports*. 6255. https://researchrepository.wvu.edu/etd/6255

This Dissertation is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Dissertation in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Dissertation has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

Application and Control Aware Communication Strategies for Transportation and Energy Cyber-Physical Systems

Ehsan Moradi Pari

Dissertation submitted to the Lane Department of Computer Science and Electrical Engineering at West Virginia University in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical Engineering

> Yaser P. Fallah, Ph.D., Chair Muhammad A. Choudhry, Ph.D. Parviz Famouri, Ph.D. Vinod K. Kulathumani, Ph.D. Jason N. Gross, Ph.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia 2016

Keywords: Cyber Physical Systems; Cooperative Vehicle Safety; DSRC; Safety Applications; Connected Vehicle; Event-Based Mechanism; Smart Grid; ©Copyright 2016 Ehsan Moradi Pari

ABSTRACT

Application and Control Aware Communication Strategies for Transportation and Energy Cyber-Physical Systems

Ehsan Moradi Pari

Cyber-Physical Systems (CPSs) are a generation of engineered systems in which computing, communication, and control components are tightly integrated. Some important application domains of CPS are transportation, energy, and medical systems. The dynamics of CPSs are complex, involving the stochastic nature of communication systems, discrete dynamics of computing systems, and continuous dynamics of control systems. The existence of communication between and among controllers of physical processes is one of the basic characteristics of CPSs. Under this situation, some fundamental questions are: 1) How does the network behavior (communication delay, packet loss, etc.) affect the stability of the system? 2) Under what conditions is a complex system stabilizable?

In cases where communication is a component of a control system, scalability of the system becomes a concern. Therefore, one of the first issues to consider is how information about a physical process should be communicated. For example, the timing for sampling and communication is one issue. The traditional approach is to sample the physical process periodically or at predetermined times. An alternative is to sample it when specific events occur. Event-based sampling requires continuous monitoring of the system to decide a sample needs to be communicated. The main contributions of this dissertation in energy cyber-physical system domain are designing and modeling of event-based (on-demand) communication mechanisms. We show that in the problem of tracking a dynamical system over a network, if message generation and communication have correlation with estimation error, the same performance as the periodic sampling and communication method can be reached using a significantly lower rate of data.

For more complex CPSs such as vehicle safety systems, additional considerations for the communication component are needed. Communication strategies that enable robust situational awareness are critical for the design of CPSs, in particular for transportation systems. In this dissertation, we utilize the recently introduced concept of model-based communication and propose a new communication strategy to address this need. Our approach to model behavior of remote vehicles mathematically is to describe the small-scale structure of the remote vehicle movement (e.g. braking, accelerating) by a set of dynamic models and represent the large-scale structure (e.g. free following, turning) by coupling these dynamic models together into a Markov chain. Assuming model-based communication approach, a novel stochastic model predictive method is proposed to achieve cruise control goals and investigate the effect of new methodology.

To evaluate the accuracy and robustness of a situational awareness methodology, it is essential to study the mutual effect of the components of a situational awareness subsystem, and their impact on the accuracy of situational awareness. The main components are estimation and networking processes. One possible approach in this task is to produce models that provide a clear view into the dynamics of these two components. These models should integrate continuous physical dynamics, expressed with ordinary differential equations, with the discrete behaviors of communication, expressed with finite automata or Markov chain. In this dissertation, a hybrid automata model is proposed to combine and model both networking and estimation components in a single framework and investigate their interactions.

In summary, contributions of this dissertation lie in designing and evaluating methods that utilize knowledge of the physical element of CPSs to optimize the behavior of communication subsystems. Employment of such methods yields significant overall system performance improvement without incurring additional communication deployment costs.

To my parents

The reason of what I become today

ACKNOWLEDGMENTS

Firstly, I would like to express my sincere gratitude to my advisor Dr. Yaser P. Fallah for the continuous support of my PhD study and research, for his patience, motivation, and immense knowledge. His innovative ideas and encouragement helped me in all the time of research and writing of this dissertation.

Besides my advisor, I would like to thank my dissertation committee: Prof. Muhammad Choudhry, Prof. Parviz Famouri, Dr. Vinod Kulathumani, and Dr. Jason Gross not only for their insightful comments and encouragement, but also for their questions which incented me to widen my research from various perspectives. I greatly thank Prof. Famouri for the opportunities he made for me to show my capabilities in GTC project.

I wish to acknowledge the Lane Department of Computer Science and Electrical Engineering for the opportunity I had the past four years to pursue my PhD in an educating, competitive and supportive environment.

I would like to thank my fellow labmates for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the fun we have had in the last four years. Also, I would like to thank all my friends at West Virginia University.

Last but not the least, I would like to thank my family: my parents, my brothers and my lovely sister for their unconditional and endless love and supporting me spiritually throughout the last four years and my life in general.

TABLE OF CONTENTS

Chapter

ABSTRACT	ii
1 Introduction and Literature Review	1
1.1 Communication Approaches for Cyber-Physical Energy Systems	2
1.2 Communication Approaches for Control of Transportation Cyber-Physic	al
Systems	6
1.2.1 Existing work on communication based vehicular systems	
1.3 Modeling Real-Time Situation Awareness Component of Networked Cr	ash
Avoidance Systems	10
1.4 Organization of the Dissertation	
2 Communication Mechanisms for Cyber-Physical Energy Systems	15
2.1 Embedded Communication Network Simulator	
2.1.1 Sensor Control Modules (SCM)	
2.1.2 Communication Simulator Module	
2.1.3 Abstracting Commonly Used Protocols	
2.2 Modeling and Design of Communication Strategies	
2.2.1 Periodic Generation of the Information (Periodic Polling)	
2.2.2 On-Demand Strategy for Event-Driven Communication	
2.2.3 Remote Estimator	
2.3 Implementation and Evaluation of Communication Strategies	
3 Communication Mechanisms and Uncertainty in Connected Vehicle Applica	tion . 36
3.1 System Description	
3.2 Safety Application	
3.2.1 Forward Collision Warning	
3.2.2 Intersection Movement Assist	
3.3 Analysis of Impact of communication loss on application performance	
4 Utilizing Model-Based Communication Design for Connected Vehicle Appl	ications
4.1 System Description	49
4.2 Model Development	51
4.2.1 ARX approximation	52
4.2.2 Parameters of online-a hidden Markov model	
4.2.3 Challenges of HMM derivation	55
4.3 Application to Cooperative adaptive cruise control	59
4.3.1 Stochastic Model predictive design for CACC	61
4.4 Simulation Results	
4.4.1 Ten vehicle CACC	
4.4.2 Comparison with Conventional cruise	
4.4.3 Comparison with model predictive controller when BSM is commu	nicated.
4.4.4 Comparison of predicted values when models are used with when B	SM is
held by zero order	

5	5 Towards Unified Modeling of Communication Based Connected Vehicle Systems 70			
	5.1	Sys	tem Description	70
	5.2	Mo	del Description for the Fully Connected Network	72
	5.2.	1	Communication component model	72
	5.2.	2	Modeling the Estimation Process	77
	5.2.	3	Hybrid Model	79
	5.3	Mo	deling Hidden Node Phenomenon	81
	5.4	Mo	del Verification	82
	5.4.	1	Fully Connected Network Scenario	
	5.4.	2	Hidden Node Scenario	85
6	Con	clus	sion and Future Work	88
RI	EFERE	ENC	ES	
AI	PPENI	DIX	A	101

LIST OF TABLES

 Table 2-2
 34

 Table 5-1
 83

LIST OF FIGURES

Figure Pag	ge
Figure 1-1 Communication simulation modules (CommSim and Sensor Control Modul	le,
SCM) within PSCAD simulation environment, shown in our IEEE 13-node test feeder	
model.	4
Figure 1-2 Broadcasting the information in CVS systems.	6
Figure 1-3 control-aware model-based communication overview	8
Figure 2-1 Communication simulator design within PSCAD; showing how Sensor	
Control Module (SCM) and Comm-Sim modules are connected. PSCI: power system	
communication interface, modeling protocols such as DNP3.	16
Figure 2-2 Communication network architecture coupled with PSCAD continuous time)
domain.	17
Figure 2-3 Comm-Sim module components and how they can be used to have different	t
network topologies.	20
Figure 2-4 All communication layers contributing to latency in DNP3 based	
communication over TCP/IP or UDP/IP.	21
Figure 2-5 Schematic overview of periodic polling strategy (mechanism).	23
Figure 2-6 Block diagram illustrating how error-dependent policy is realized.	24
Figure 2-7 Comparison between two transmission modes (a) transmitted data through	
the network using periodic polling (20Hz), (b) transmitted data using error-dependent	
policy with window length=2 and eth = 0.05 , (c) reconstructed data at the receiver for	
periodic polling, (d) reconstructed data at the receiver using the proposed strategy, (e)	
performance of periodic polling for transmission rate of 20Hz, (f) performance of error	-
dependent strategy when window length=2, $eth = 0.05$, (g) voltage of the grid for	
periodic polling with transmission rate of 20Hz, (h) voltage of the grid for error	
dependent policy with window length=2 and eth = 0.05 .	29
Figure 2-8 Effect of different thresholds (a) performance of battery in compensation o	f
wind turbine, when $eth = 0.1$, (b) voltage of the grid when $eth = 0.1$, (c) performance	of
battery in compensation of wind turbine, when $eth = 0.4$, (d) voltage of the grid, when	n
eth = 0.4, (e) performance of battery in compensation of wind turbine, when $eth = 0.9$	Э,
(f) voltage of the grid, when $eth = 0.9$.	31
Figure 2-9 Simulated real power system for fault monitoring process	32
Figure 2-10 Schematic overview of fault monitoring procedure, while error-dependent	
strategy of transmission.	33
Figure 2-11 Fault monitoring using on-demand communication strategy.	34
Figure 2-12 Comparison of transmission modes (a) transmitted data through the network	rk
when periodic polling (10Hz), (b) transmitted data through the network using error-	
dependent policy with window length=2 and eth = 0.25 , (c) reconstructed data at the	
receiver when periodic polling, (d) reconstructed data at the receiver when periodic	
polling, (e) performance of periodic polling with transmission rate of 10Hz, (f)	
performance of error-dependent strategy when window length=2, eth = 0.25 , (g) volta	ige
of the grid when periodic polling with transmission rate of 10Hz, (h) voltage of the grid	d
for error dependent policy with window length= 2 and $th = 0.1$.	35

Figure 3-1 Functional blocks of CAS system	37
Figure 3-2 Communication logic: (top) Periodic Beaconing, (bottom) Network- Aware	.
Error-Dependent.	39
Figure 3-3 Schematic overview of IMA situation utilizing V2V	43
Figure 3-4 Effect of communication loss on PTE for different transmission strategies. I	ED
with low threshold and high rate, and PB with the same rate as ED, (6Hz)	45
Figure 3-5 Effect of communication loss on PTE in use of transmission rate adaptation	n
policies. Both using the same error threshold.	46
Figure 3-6 Example of a warning timestamps and separation distance when packet error	or
rate is:(top) 0, (middle) 0.6, (bottom) 0.8.	46
Figure 3-7 Comparison of warning timestamps for different PER values for time-based	1
and distance-based FCW, PER=0 (top), PER=0.4 (middle), PER=0.7 (bottom).	47
Figure 4-1 Block diagram of the in-vehicle unit, including the communication	
components and their functionality.	50
Figure 4-2 SHS designed for model based communication	53
Figure 4-3 Demonstration of CACC platoon using model based communication.	60
Figure 4-4 Ten vehicles Platoon using the proposed CACC.	63
Figure 4-5 Comparison of the conventional CACC (left) and the proposed method usin	ng
Model-based communication (right)	66
Figure 4-6 Histogram comparison of the conventional CACC (left) and the proposed	
method using Model-based communication (right) in terms of spacing error for different	nt
lead vehicle maneuvers.	67
Figure 4-7 Comparison of frozen-time MPC CACC (when raw data is transmitted) and	d
the proposed stochastic MPC (stochastic models are available).	68
Figure 4-8 comparison of prediction when models are utilized and zero order holding of	of
acceleration for 1hz. (a) comparison with the original values, (b) histogram of errors for	or
zero order holding, (c) histogram of errors when model based is utilized	68
Figure 4-9 comparison of prediction when models are utilized and zero order holding of	of
acceleration for 2hz. (a) comparison with the original values, (b) histogram of errors for	or
zero order holding, (c) histogram of errors when model based is utilized.	69
Figure 4-10 comparison of prediction when models are utilized and zero order holding	; of
acceleration for 5hz. (a) comparison with the original values, (b) histogram of errors for	or
zero order holding, (c) histogram of errors when model based is utilized.	69
Figure 5-1 Two different components of RTSAC.	71
Figure 5-2 Probabilistic timed-automata model for sender at vehicle j in the saturation	L
condition	74
Figure 5-3 Probabilistic timed-automata model for shared channel.	75
Figure 5-4 Timed-automata model for receiver part at vehicle.	76
Figure 5-5 Schematic overview of Error-Dependent policy.	78
Figure 5-6 Hybrid model for CVS system at vehicle j.	79
Figure 5-7 Schematic overview of hidden node phenomenon.	81
Figure 5-8 Receiver model for vehicle j.	82
Figure 5-9 Probability of success vs. number of vehicles - all in range scenario.	84
Figure 5-10 Probability of success vs. transmission rate - all in range scenario	85
Figure 5-11 Probability of success vs. transmission range - Hidden node scenario.	86

Figure 5-12	Probability of success vs. transmission rate and range - Hidden node	
scenario.		87
Figure 5-13	Effect of threshold error on transmission rate.	87

1 Introduction and Literature Review

Cyber–Physical Systems (CPSs) are a generation of engineered systems in which computing, communication, and control components are tightly integrated. CPSs consist of many heterogeneous and inter-connected sub-systems. The heterogeneity of components and the complexity of system dynamics and its interaction with communication and control pose many design challenges, requiring a new approach to study and modeling of system components. The dynamics of CPSs are complex, involving the stochastic nature of communication systems, discrete dynamics of computing systems, and continuous dynamics of control systems. In this work we focus on modeling some example CPSs, and then introduce a new approach to communication for such systems. The following are examples of complex CPSs and their applications:

- Effective road traffic control with significantly low congestion and delays, and zero fatalities and minimal injuries in traffic accidents.
- Effective power grids with real-time cooperative control of protection devices that facilitate blackout-free electricity generation and distribution.
- Critical physical infrastructures that call for preventive maintenance.
- Energy-aware buildings.
- Highly automated manufacturing systems.

In this dissertation we study how the communication subsystems should be configured to achieve significant overall system performance improvement without incurring additional communication deployment costs. The challenges related to traffic control and smart power grids; in particular focusing on situational awareness and its associated communication approaches are investigated. The impacts of communication component on situational awareness and performance of different physical processes are studied and new approaches to enhance the existing methodologies are introduced. Note that the more accurate the situational awareness is, the better the performance of physical process and actuators would be. This chapter is divided into four sections. Section 1.1 discusses the literature and challenges of creating situational awareness in smart power grids. Literature review and introduction to smart transportation CPS are presented in section 1.2. Section 1.3 is dedicated to the challenges of modeling CPS. Organization of the dissertation is discussed in section 1.4.

1.1 Communication Approaches for Cyber-Physical Energy Systems

The application of communication technologies in power systems is viewed as the enabling factor for smart-grid [1][2]. Realizing the benefits of such technology integration requires new control mechanisms that can take advantage of the possibility of remote/distributed sensing and actuation [15][16][18]. Employing such communication based control mechanisms will inevitably couple the performance of the overall system to the performance of its cyber (communication and computing) component, as well as the physical power systems. This results in a Cyber-Physical Energy System (CPES) with tight coupling of communication, control and power system components [1][3]. Such coupling is in particular more critical at transient level, where the increased communication load at critical events, generated by control and monitoring processes, may hinder the performance of the communication network and in turn disrupt the control schemes that rely on it [3][4]. For example, a fault in one feeder will trigger a series of messages from many connected devices that will observe current increase at different physically related locations. The flood of messages will disrupt the underlying communication network and creates the possibility of long delays preventing possible protection actions to be taken in time [4]. As a result, it is important to study the performance of the communication protocols and strategies that are used for distributed or centralized control in smart grid [5]. It must be emphasized that the communication issue is mostly relevant when hard real-time actions are concerned (at levels related to the transient behavior of circuits); applications that purely relate to monitoring smart meters or demand-response applications that can tolerate delays of tens of seconds can normally be handled well by existing communication strategies.

In this dissertation, the issue of designing communication strategies for hard real-time monitoring and control in smart power systems is considered. It must be emphasized that the communication strategies refer to strategies that describe when and what needs to be communicated for real-time applications. The question of how to communicate such data is answered by the design of user level power system communication protocols (such as Distributed Network Protocol, DNP3, or IEC 61850) and lower level communication technology (e.g., TCP/IP/Ethernet, etc.). This study addresses the need for strategies (also called mechanisms) that describe how existing or future protocols can be used for efficient real-time communication. The real-time efficiency is in particular of interest in this works, since it allows significant performance improvement for network control schemes [6]. An on-demand communication strategy is introduced as the first contribution of this dissertation. This strategy allows real-time tracking of dynamical systems over networks that may suffer from loss and excessive delay.

To evaluate distributed control schemes (which are intrinsically communication based), in particular at power system transient levels, it is also important to include the effect of communication schemes at transient level analysis. This is the second contribution of this approach that builds on the early work in [7]. For this purpose, a new approach is presented by providing an embedded simulation environment inside the widespread commercially available tool PSCAD. The objective is to build the communication module where transient analysis for power systems takes place, i.e. in tools such as PSCAD. By embedding the communication module inside PSCAD, synchronization issues in integration of a discrete time simulator and a continuous time domain tool are avoided. The possibility to study communication vulnerabilities on power systems without going outside PSCAD allows power engineer and researchers to quickly convert their current models to the ones that consider the issue of communication in detail. For this purpose, a communication model is introduced that can be configured to simulate the behavior of several protocols such as DNP3 [8]-[10] or IEC 61850 [11]-[13].

Figure 1-1 shows the test network considered in our evaluations, which is modeled based on the IEEE 13-Node Test Feeder [14]. The communication protocol used to specify how communication strategies are implemented is DNP3. To have the best performance of communication module with respect to the number of transmission in PSCAD, different rates of transmission for communication strategies feasible under DNP3 and IEC 61850 are studied.



Figure 1-1 Communication simulation modules (CommSim and Sensor Control Module, SCM) within PSCAD simulation environment, shown in our IEEE 13-node test feeder model.

Smart grid provides the possibility of integrating new renewable energy sources such as solar and wind power [15]. Smart generation, metering of electricity consumption and smart control of distribution systems introduces the necessity of communication between elements in the system. Therefore, whenever smart grid ideas are being considered for electric power system reform [16], it is necessary to investigate communication technologies and their impact on power systems [17][18]. As a result, the communication component becomes a necessary part of control strategies for critical smart grid applications. Examples of such critical applications include emergency control of power system [19], protection of the system [20], voltage regulation [21], restoration of power [22] and auto-recovery distribution panels [23] in all multi agent system architectures as well as smart energy management [24].

One method of investigating the effect of communication on power systems is through transient level simulation. Commonly used power simulation tools, like EMTDC/PSCAD, work in continuous time domain (though realized through small discrete steps), while most communication simulators are designed as discrete event systems. Several recent efforts have focused on coupling of these simulators to handle communication in power systems [25]-[31]. In [26] authors propose to address the effect of network by simulating the desired network in OPNET [32] and employing the resulting traces in power system

simulation, to measure the performance of an active distribution power system in presence of communication network. To investigate the effect of the communication processes on reliability and cascading behavior of a power system, the power grid and control/communication systems are coupled in [27]. The authors formulate an optimization problem and study the vulnerabilities of control/communication systems by adding new constraints to the optimization problem. Hybrid system approach for simulating the interaction between the physical and computational elements of a large power system is presented in [28]. The authors propose a single framework to integrate continuous and discrete events instead of using specific tools separately focused on continuous or discrete event dynamics. In [29], a power/network co-simulation framework has been proposed, which integrates power system dynamics simulator and network simulator using an accurate synchronization mechanism outside both simulators.

Network simulators use different stochastic and deterministic processes to imitate the real world communication systems and simulate the communication process as close to reality as possible. In coupling a power simulator and a network simulator, a list of communication events should be executed jointly with continuous time domain simulation of the power system simulator. To deal with this coupling challenges, several methods have been proposed in literature [25][29][30][33]. In [25], authors suggest using federation which will have a considerable overhead due to component synchronization issues. It also may lead to a mismatch between the real dynamics and interaction, and the simulation as explained in [29]. The co-simulation framework proposed in [29] is designated to remove the accumulating errors by preforming the simulation globally in a discrete event-driven manner. Using this method, without the need for defining explicit synchronization points, the simulations will be synchronized. A co-simulation platform is proposed in [30] to solve the integration issue. The proposed co-simulation platform works on discrete event driven conditions. An event scheduler is considered to be a global timing coordinator by checking a global event list (power system iteration plus communication network events and their timestamps). An improvement to the simulation interfacing issues (synchronization and data exchange) is presented in [33], proposing an efficient federation between the agent simulators, power system simulators, and communication simulator interfaces.

In this study, instead of interfacing a power system simulator and a communication simulator, a communication simulator is embedded inside the power system simulator (PSCAD). The existing time steps in PSCAD transient power simulator are used and the discrete event-based simulation of communication networking module at these time steps is synchronized. Thus, the novelty of this work is to couple the communication simulator with power system simulation by implementing the network simulator within PSCAD and trying to imitate discrete event-based behaviors using PSCAD capabilities.

1.2 Communication Approaches for Control of Transportation Cyber-Physical Systems

Cooperative vehicle safety (CVS) systems are examples of intelligent transportation systems (ITS) that utilize communication amongst vehicles for the purpose of improving safety and efficiency of the transportation network. CVS systems are designed based on broadcasting information through a shared channel and providing mechanisms for each vehicle to estimate and track the state (e.g., position, speed) of other vehicles. Vehicles broadcast their state information in a neighborhood around them as shown in Figure 1-2. To implement the CVS applications, vehicles need to have a real-time updated map of their neighboring vehicles and predict hazard situations based on that. Such a real-time map constitutes the "situational awareness" for these vehicles. CVS systems conceptually rely on vehicular communication networks which is based on Dedicated Short Range Communication (DSRC) [42] as a means of acquiring real-time situation awareness and action coordination.



Figure 1-2 Broadcasting the information in CVS systems.

Real time situation awareness component (RTSAC) of cooperative crash avoidance system is designated to share information of vehicle and track information of other vehicles received from the shared channel. CVS applications use the information received from the RTSAC of CVS system. Example CVS applications include automated crash avoidance, driver safety warning systems, and coordinated highway platooning of autonomous vehicles. The more accurate data RTSAC provides, the more precise safety applications perform [43][49].

Recent research [64][65] on a subset of CVS revealed that tight coupling of the safety application and communication component yields a vulnerable system that quickly fails in presence of communication issues or excessive application load. As a result, coordinated actions for safety purposes are not possible with a high degree of assurance in large scale systems. Therefore, there is need for developing control-aware communication strategies. An example of such strategies has recently been proposed as "content and model aware communication concept" [64][96]. In this study, we use this concept and design a framework for model aware communication for cruise control application using stochastic hybrid system models. Model-based communication works by communication of models that describe situational and control information, instead of delivering raw sensing data (Figure 1-3). This is in turn achieved by model-based representation of information using modal representation of system dynamics, for example in stochastic hybrid systems (SHS) forms.

The term model-based networked control was introduced in [103][104][105] for the first time but in these works the authors propose to perform feedback control by updating the model's state (current values of the system parameters) using the actual values of the plant which are sensed by the sensor. In our proposed model-based communication methodology, due to stochastic nature of CVS (road topology and human factor), we cannot assign a function and preform feedback control and prediction based on that function or just update the current values of the underlying functions and synchronizing them. Therefore, in this dissertation the model-based communication methodology proposes to have dynamic model generation and model transmission through the network according to the real-time conditions and situations.



Figure 1-3 control-aware model-based communication overview

1.2.1 Existing work on communication based vehicular systems

In CVS the RTSAC receives the information from the shared channel and determines the map of surrounding vehicles and deliver these information to the application/ control unit. The control unit compares the situation to the driver behavior and decides about generating relevant warning to the driver. Forward collision warning (FCW) is an example of CVS application which is design to assist drivers in prevention from rear-end collision [66][67][68]. In FCW warning is issued to the driver, when the distance needed to brake and stop the car is more than the distance between rear bumper of the front vehicle and the front bumper of the driver's vehicle.

In automated driving applications, such as cooperative collision avoidance (CCA), platooning and CACC, the control and decision making unit in each vehicle determines the desired motion of the vehicle based on information received from the neighboring vehicles (and local sensors in many cases). In a way, the received information allows for real-time situational awareness, and safe operation of controllers require accurate situational awareness. CACC and platooning are both car-following designs with some similar challenges. In car platoons the goal is to maintain very close and reasonable distances between vehicles (gap control) to increase the number of vehicles on the road while in CACC the main objective is to keep a safe distance and provide a comfort ride for

passengers. The research in the area of platooning and CACC could be categorize as follows:

- Driver behaviors and interaction with CACC such as when the driver turns the CACC on, and what is the driver's take on the relative distance with the front vehicle in different traffic situations. [69][70].
- The influence of CACC on traffic such as safety, congestion [71][72].
- Design of CACC application [72][73][74][75]
- Communication requirement and role in efficient CACC and platooning [75][76].
- String stability of CACC [77][78].

To study different aspects of CVS applications and test them for different real world scenarios, it is necessary to mathematically model these algorithms and use the model for the purpose of model checking [79][80][81][95]. [79] introduces a formal system modelling and verification methodology to study CACC and CCA. The authors propose a new hybrid approach to manage intersection when the safety information about the other vehicle is not perfect and complete in [81]. Modeling driver behavior in order to be considered in the study of CVS system is beneficial [82][83][84]. In [82] authors introduce a stochastic-switched autoregressive exogenous (SS-ARX) to imitate drivers behavior in lane-change scenario. This method uses different regression models to cover different aspects of lane change behavior.

Estimation and classification of driver behavior is an attractive area of research and several applications and approaches are introduced in the literature [85][86][87]. HMM is introduced to model driver behavior to the literature in [85]. In [85] authors use HMM to break down the driver behavior into different segments and employ Kalman filter at each state of HMM to represent the driver behavior related to each segment. [86] introduces a framework to study vehicle dynamics and driver behavior. [86] uses HMM to estimate driver behavior to observe unknown decisions made by the driver when the vehicle moves toward intersection. Coupling Support Vector Machine (SVM) with HMM was introduced

in [87] in order to classify drivers as compliant or violating and determine the driver's behavior at the intersection.

1.3 Modeling Real-Time Situation Awareness Component of Networked Crash Avoidance Systems

Crash avoidance system (CAS) is one of the most important parts of any autonomous driving system. CAS, also called cooperative crash avoidance system, is a flavor of automated crash avoidance that relies on cooperation and coordination among vehicles that are within a certain range of each other. A solution for real-time situation-awareness in systems like CAS is being currently developed under the CVS work and is expected to be operational in the near future [40][41]. This dissertation provides a detailed study and modeling approaches for this component, which will be a major component of CVS, CAS and any networked autonomous driving system.

Each RTSAC component is composed of two subcomponents, communication and estimation. Communication in RTSAC is based on DSRC technology, one of the enabling technologies of vehicular ad-hoc networks (VANETs). DSRC uses CSMA/CA (Carrier Sense Multiple Access with collision Avoidance) to share the wireless channel among neighboring vehicles (up to distances of few hundred meters). The CSMA/CA communication protocol specifies how vehicles should coordinate access to the shared wireless channel within the transmission range of each other. There are many issues with using a shared channel in vehicular broadcast networks; the scalability issue is a well-known problem and refers to the fact that in crowded networks the total capacity of the network tends to zero, failing RTSAC altogether [43]. An important reason for such failure is the hidden nodes phenomenon, which happens when two nodes outside transmission range of each other transmission range of each other them. These VANET issues will adversely affect the estimation process if the network is not properly managed.

There are currently several different designs for RTSAC. The original design of [40][41] relies on periodic sampling and communication of vehicle state; in [44] an error-dependent policy is used to reduce the load of RTSAC on the underlying DSRC network. In [43] an error-dependent policy is mixed with a transmission power adaptation scheme to ensure

the scalability issue in VANET is avoided as much as possible. Another notable solution [48] uses adaptation of the sampling and communication rate to avoid congestion in the network. A study of the overall RTSAC or CVS system reveals mutual coupling of the communication and estimation processes [45][46], which prevents a straightforward solution for the scalability issue. Therefore, modeling the mutual effects of the two components of RTSAC is a critical step in designing a high-performance RTSAC system [45][46]. Our previous works in [45] [49] have shown that the traditional stochastic models, though useful, do not provide enough insight for optimal designs. Mainly due to the fact that performance measures are presented in average forms in the above models, hiding detailed interaction between components. In this study, we offer a new approach to modeling RTSAC components, based on hybrid systems concepts. We use hybrid systems to allow for a unified framework that includes all components without excessive abstraction of important details. Our objective in presenting this modeling approach is to allow study of autonomous crash-avoidance systems or CVS without ignoring the communication and estimation processes. The new modeling approach provides the capability to jointly and directly examine the effects of the communication and estimation processes in one model, for any given vehicle trajectory and network scenario. This allows for a significantly better understanding of the system behavior, when compared to the method of separately analyzing components and then combining the averaged results [45].

Instead of the stochastic Markov-chain model of [49], or the empirical models in [45][47], we propose to model the system using techniques drawn from the hybrid automata theory. In [50], we presented a preliminary model based on probabilistic timed-automata for the networking/communication component of RTSAC. The main reason for approaching probabilistic timed-automata is that the communication component can be considered as a discrete-event system (DES)[51], because the protocol behind DSRC (i.e., CSMA/CA) operates with integer quantities such as countdown timers. Moreover, many events that control the behavior of the system operate based on instantaneous events such as sending information to other vehicles and sensing the Channel-busy status. The CSMA/CA protocol functions based on some state transitions (e.g., between sending data, receiving data or sensing and waiting for free channel. The randomness in CSMA/CA is due to

random backoff procedure as well as random packet arrival patterns and channel sensing results. We employ timed-automata [51][52][53] due to its features and properties in modeling distributed computation, presence of timing constraints, and intrinsic randomness in CSMA/CA. In [53], the effect of different computer clocks on behavior of a distributed application and weakness of timed-automata in modeling these issues are investigated.

Modeling networking/communication protocols using methods outside queuing theory has been attempted by several researchers in recent years [54]-[57]. In [54], a probabilistic timed-automata model for the IEEE 802.11 protocol is presented and studied for a twostation case. In this work, two senders contend to broadcast their information, and the authors use PRISM model checker to build and verify the model for different measures. In [55], authors employ stochastic Petri Net to model, analyze the performance, and study all aspects of the IEEE 802.11 medium access control scheme. In [56], the author models the IEEE 802.15.4 CSMA/CA protocol (ZigBee) using probabilistic timed-automata due to the behavior of this type of model in the presence of dense time, non-determinism, and probabilistic choice. PRISM is employed to simulate the model and to check its effectiveness. While these models are useful, and have inspired our current work, they mainly consider the communication protocol. In the modeling of RTSAC, the communication protocol is tightly coupled with and interacts with the estimation process. Moreover, the CVS system uses IEEE 802.11 in broadcast mode and in the presence of hidden nodes. In this mode senders have no feedback to know whether their transmission has been successful or a collision has occurred at some destinations. In this chapter we present a model that considers all the above issues using probabilistic timed-automata and hybrid automata.

Our choice of hybrid automata method is also inspired by recent literature on modeling cyber-physical systems [58][59][60]. It is generally recommended that Hybrid automata (or hybrid systems) modeling seems to be rational for modeling CPS because the physical processes could be defined in terms of differential equations, while communication and computation could be represented in terms of timed-automata, state machines, data flows, discrete events, etc.[61].

In our modeling effort, we employ the estimation process that has been introduced and verified in [43][44][46]. This method is currently under test by automotive industry to replace the original design of periodic transmission of state information [41]. The unified modeling of the components of communication and estimation facilitates a more precise and straightforward study of the characteristics and properties of the system, and provides an opportunity for separate and mutual performance analysis of each component through model checking. To model the communication component, we consider the DSRC broadcast network in two scenarios; first we consider the case of all vehicles in being transmission range of each other (no hidden node effect) to derive the basic model; we then extend the model to cover the hidden node case, which is the prevalent situation in CAS and CVS networks.

1.4 Organization of the Dissertation

The main contribution of this dissertation is to study and design new methodology and mechanisms to enhance the situational awareness and application performance for two cases of cyber physical energy and transportation systems.

In chapter 2 a new on-demand communication mechanism (strategy) is presented that describes how power applications should configure the communication subsystems (almost agnostically to the underlying communication protocols) to achieve significant performance improvement without incurring additional communication deployment costs. The on-demand communication mechanism relies on the concept of error-dependent communication for tracking dynamical systems over communication networks. Also an embedded communication network simulator design that allows integration of communication strategies and protocols inside PSCAD transient level power system simulator is presented. The integrated simulator models and abstracts commonly used communication protocols, such as DNP3, and is used to verify the performance gains of on-demand communication strategies, using services of DNP3.

Chapter 3 presents a systematic approach to study the performance of emerging communication based vehicle safety systems. Examples of such systems include FCW and Intersection Movement Assist (IMA) applications. To study of the safety application and

its underlying communication system we investigate the impact of communication uncertainties on a new variation of the FCW algorithm. In this chapter, the impact of communication loss and the choice of signal communication logic are examined and compared for two different representation of FCW. We study how communication uncertainties have effect on the accuracy of safety applications (e.g. FCW and IMA). Based on precise study of results it is shown that employing rate adaptation methods (e.g. error dependent and network aware error dependent) and in general, communication strategies will yield considerable gains in communication or accuracy of tracking and hazard detection. Network awareness in communication logic is also demonstrated to be beneficial in high communication loss situations.

In Chapter 4 we present a new communication approach which helps to increase the accuracy of situational awareness and performance of safety and automated applications. A new design for CACC based on the concept of model-based communication is introduced in this chapter. For our proposed design we introduce a new approach for modeling driver/vehicle behavior. This approach uses HMM to segmentizes different behavior of the underlying function and finds the best ARX representation for each segment. We design a new controller for CACC and use the models for prediction. We demonstrate the effectiveness of this approach on control of CACC through several simulation studies.

In chapter 5 a new modeling methodology which models both estimation and networking components of CVS in a single framework is proposed. We employ Probabilistic Timed-Automata to model the networking component of CVS; a Hybrid Automata is then used to combine and model both networking and estimation components in a single framework. This approach enables the ability to simulate different scenarios without any need for field implementation. The presented hybrid automata is extended to model the broadcast network of CVS in presence of heavy hidden node interference. The model accuracy is verified by comparison with proven ns-3 (network simulator 3) simulation models. Chapter 6 concludes the dissertation and discusses the remaining work and the possible future directions for this research.

2 Communication Mechanisms for Cyber-Physical Energy Systems

Advanced and modern communication technology is the enabling factor for distributed sensing and control mechanisms that make the promise of smart grid possible. As in many complex cyber-physical systems, the communication and control components and the physical aspects of the system are tightly coupled. The performance of the communication component has a profound effect on the performance of the control systems that manage a power system, in particular when transient level behavior of the system is concerned and critical applications such as protection are considered. While communication protocols and network designs advance in parallel to power systems, the mechanisms and strategies to intelligently employ them in power systems is a separate and fundamentally important issue, as it enables efficient use of the existing or future communication capabilities for the purpose of improved performance in power systems. In this chapter, we present on-demand communication mechanisms (strategies) that are model-based and describe how power applications should configure the communication subsystems (almost agnostically to the underlying communication protocols) to achieve significant performance improvement without incurring additional communication deployment costs. The on-demand communication mechanisms rely on the concept of error-dependent communication for tracking dynamical systems over communication networks. We also present an embedded communication network simulator design that allows integration of communication strategies and protocols inside PSCAD transient level power system simulator. The integrated simulator models and abstracts commonly used communication protocols, such as DNP3, and is used to verify the performance gains of on-demand communication strategies, using services of DNP3.

This chapter is divided into three sections. Section 2.1 discusses the system architecture and designed communication interface. Communication strategies are presented in section 2.2. In section 2.3, proposed communication strategies are evaluated.

2.1 Embedded Communication Network Simulator

In this section, the design of the proposed embedded communication simulator is described. This will enable us to study communication strategies in the next sections. The proposed communication network simulator allows simulating communication behavior between power system components in PSCAD. Figure 2-1 shows the schematic overview of the design. Application layer and lower levels of communication are designed as separate layers of the communication interface in this study. The application layer behavior and lower layer of communication are implemented in Sensor Control Modules (SCMs) and Comm-Sim Component respectively. The general architecture of the model can be observed in Figure 2-2.



Figure 2-1 Communication simulator design within PSCAD; showing how Sensor Control Module (SCM) and Comm-Sim modules are connected. PSCI: power system communication interface, modeling protocols such as DNP3.

SCMs are designed as typical PSCAD modules and utilize Comm-Modules as means of communication (sender and/or receiver). The SCMs represent smart grid modules, whether in power generation, transmission or distribution systems, which have the capability of sensing, communicating or control. They control the stability and maintain the balance of the grid based on the control parameters and other information, which they send or receive through the networking channel.

PSCAD works with continuous time signals, while communication processes are discrete and event based. This fact complicates the integration of communication simulation in PSCAD. Therefore, the embedded communication modules of SCMs need to initially sample the continuous signals, and then packetize them in an asynchronous manner. Since the transmitted data should be interpolated at the receiver side (receiver SCMs), the sampling rate must be specified and known at both sides.





In real world implementation of communication networks, sender station transmits its data packet according to a predefined communication strategy and protocol. The receiver stations follow the same strategy and protocol to decode the received data and rebuild the original signal. There are different types of strategies and many different protocols that may be used. Here protocols refer to rules that specify how communication is done (e.g., TCP/IP, DNP3, etc.), whereas strategy refers to rule governing the time (when) and content (what) of communication (e.g., periodic, event based, error-dependent, etc.). These definitions are commonly used when communication is used for control purposes, but are not standardized or universally used with the same meaning in communication literature. Nevertheless, in this study strategies and protocols are separated, for the sake of clarity.

To model the effect of communication in control loops of an energy system, one can either implement or simulate the communication network with all its bit level details, which will be prohibitively complex in environments like PSCAD, or use precise abstractions of the communication network. Such precise abstract shall maintain the major contributing effects of communication networks. At a very simple level, communication may be modeled by delay and loss patterns only; however, such models would be too simplistic and may not capture all possible interactions of power and communication systems. To avoid the impreciseness of completely abstract models and the complexity of complete implementation, this work opts for a method of actually creating the network, but modeling individual network components (links, routers, hosts, protocols and strategies) to the desired abstraction level.

Our embedded simulator allows creation of actual networks with components such as links, routers and end-hosts, and supports commonly used protocols and strategies. Individual links, routers and hosts are allowed to inflict loss or delay patterns on data passing through them; hosts also implement communication strategies and protocols that allow modeling of power system specific protocols such as DNP3. Hosts are included in the SCM modules and the communication network components are simulated in Comm-Sim module between sender and receiver SCMs. The Comm-Sim module stores the received packets and delivers them to their specific SCMs (destination) based on configurable loss and delay of its components. These delays and losses are configured based on the underlying lower level communication protocol and technologies (e.g., Ethernet, WiMAX, etc.). In this work, DNP3 protocol data formats are considered for higher level SCM communication. Schematic overview of communication interface is shown in Fig. 2-2.

2.1.1 Sensor Control Modules (SCM)

Sensor Control Modules (SCM) represent the intelligent electronic devices (IEDs) that bear the responsibility of communication between different sensing, actuation or control elements in a power system. The SCMs simulate the communication interface which has been abstracted to two major components in this study, communication protocol and communication strategy. The considered communication protocol in this work is DNP3 which can work over TCP/IP and is widely used in North American power systems; this protocol will be explained in more details in next sections. The other component of communication interface (strategy) is dependent on the type of control being used and mostly addresses the rate of message transmission. The SCM main task is to discretize the signal, decide whether to send out a signal value based on communication strategy or not, and create the packet to be sent out. Additionally, the SCM at the receiver decodes the packet based on the protocol and sends the values to the control component of the system. In PSCAD, simulation assumes continuous-time signals which should be discretized using sampling; this sampled signal is zero-order hold of samples of the original signal. The sampling rate can be specified by the user based on the dynamics of the system. This sampled, but still continuous signal, is wired using Radio Link component between communicating elements of the Power System. However, communication is a discrete, event based phenomenon which happens just at certain points of time, specified by communication strategy. Therefore, in order to emulate the event of send/receive a packet, an additional clock signal is used, which has a value of "one" when a packet is sent/received, and "zero" otherwise.

2.1.2 Communication Simulator Module

SCM, as described previously, simulates application layer of communication. The Lower Layer of communication in this study is simulated in Communication Simulator (Comm-Sim) Module. The schematic overview of this module can be observed in Figure 2-3. Comm-Sim Module gives the facility to design different network topologies for the system.

Comm-Sim Module has two distinct components named Comm-Link and Comm-Router. The Comm-Link component models and simulates a Point-to-Point (P2P) communication link between any two SCMs. This link is modeled as communication delay and loss, which is applied on the packets as part of its functionality. The value chosen for delay can be a fixed, random or a delay profile that may be obtained from real networks or from a network simulator like NS3 or OPNET. Loss is defined as whether a packet is received at its receiver or is lost during the transmission process. Loss decision can be random or a logged profile, driven from an actual experiment or from a network simulator. The Comm-Router component simulates a simple router, which gives the option to have interconnected SCMs. Interconnected SCMs means, one SCM can send to a set of other SCMs connected to the Comm-Router or broadcast to all of them. This will give the ability to have more complex network topologies in addition to simple P2P connections.



Figure 2-3 Comm-Sim module components and how they can be used to have different network topologies.

2.1.3 Abstracting Commonly Used Protocols

In addition to lower layer protocols that are simulated in Comm-Sim module, power system specific protocols that run in the higher layers of the communication stack (such as DNP3, Figure 2-4) need to be modeled and simulated. While DNP3 is selected as a representative protocol, due to its dominance in legacy SCADA systems in North America [8][18], the simulator design relies on abstracting the behavior of DNP3 rather than emulating all its internal interactions. This abstraction helps in reconfiguring the simulator design, the idea is to simulate the effect of using DNP3, rather than completely implementing the protocol, to avoid unnecessary complexities that do not directly matter in power system analysis. For this purpose, the design relies on studies that derive the performance metrics of DNP3 [4] to provide meaningful range of parameters in the simulator.

Simulating a power-system specific communication protocol, such as DNP3, requires understanding the effects of using such protocols on communication strategies, and on loss and delay patterns. According to DNP3 specifications, DNP3 enabled devices can either work in event-driven mode or in non-event-driven mode (i.e., polling based) [8][34][100]. The non-event-driven mode of operation requires frequent polling and requires the communication to be initiated by the receiver (the master device), and includes at least two communication transactions. The event-driven method allows unsolicited messages to be sent from slaves based on their determination of when communication is needed. The event-driven method provides a lower communication overhead and significantly lower

latency, in particular for protection applications. Both these modes of operation are already supported by the SCM design, which allows either periodic polling, polling based communication, or event-driven communication. The simulation of the two modes of operation requires consideration of the communication timing (event-driven, or periodic), delay (single or multiple round-trip times) and loss (higher loss probability for method involving multiple transactions).



Figure 2-4 All communication layers contributing to latency in DNP3 based communication over TCP/IP or UDP/IP.

Simulation of the loss and delay patterns requires knowledge of the lower layer communication protocols. For example, when TCP/IP is used, instead of UDP/IP, the loss is reduced at the cost of increased delay. Such scenarios are simulated by allowing the user to choose a few profiles (DNP3/UDP/IP or DNP3/TCP/IP, etc.).

To properly consider the effect of using a protocol such as DNP3, its internal workings should be studied to understand the sources of latency in delivery of information. The work in [4] provides a study of the latency of DNP3 based communication. They consider four DNP3 functions (Polls of class 0/1/2/3), including integrity poll (sending all data on the slave device), exception scan (only responding when data changes), unsolicited response (spontaneously sending data out of the slave device) and data change (changing slave data according to commands). This work correctly concludes that non-event-driven based

communication is too costly and slow for most critical power system applications; it recommends that only event-driven mechanism be used. The result of this study provides a break-down of the communication latency into components of 1) lower layer communication delay (IP and below), 2) Transport protocol below DNP3 (TCP or UDP), 3) DNP3 Transport/Link Layer, and 4) DNP3 Application layer delays. Table 2-1 shows the result from [4].

Table 2-1			
Delay ratios for components in the message delivery,			
reported from [4]; CB is circuit breaker.			

application delay	Transport function delay		lower layer delay
	DNP3Transport/	TCP	
	Link Layer		
Relay to CB	63%	14%	6%
Controller: 17%			
Report from Relay	36%	17%	8%
to Center: 39%			
Report from CB	37%	17%	8%
Controller to			
Center: 40%			

It is found in [4] that the overall message delivery latency is dominated by the processing delays in the DNP3 stack. The communication delay from TCP layer and below are smaller and also mostly from the TCP stack; the use of UDP reduces this figure further. The processing delays are known to be directly related to the CPU speed and architecture of the embedded computing devices that control the power devices. As a result, the values reported in [4] should be accordingly scaled when a different embedded computer is used. For this purpose a default computing latency profile is provided for certain CPU speed (300 MHz) in each outgoing link of a SCM and users are allowed to specify scaling factors. The TCP and lower delays are implemented in Comm-Sim modules according to the used communication technology (in most cases the delay will still be much smaller than the computing delay for DNP3).

Another observation from [4] is that the dominant DNP3 processing delay is due to its reliability mechanisms (transport/link layers), which are redundant when TCP is used as the transport protocol above IP and below DNP3.

2.2 Modeling and Design of Communication Strategies

As it was mentioned before, communication under DNP3 and IEC 61850 is possible in two modes:

- Polling based (receiver periodically requests data)
- Event-driven (reports are generated when needed)

Therefore, these modes are considered for message generation, and communication strategies that build on these modes are studied.

2.2.1 Periodic Generation of the Information (Periodic Polling)

In this method, each agent has a sampler, which samples the continuous signal that it wants to share with other agents based on a fixed rate of sampling; the samples are then used to generate messages at each sample time. Figure 2-5 shows an overview of this system. The left hand side shows the sender and the right hand side indicates the receiver.



Figure 2-5 Schematic overview of periodic polling strategy (mechanism).

The simple strategy for transmitting the information at the sender agent is as follows:

transmission strategy at sender := $\begin{cases} transmit sample value \\ do not transmit \end{cases}$ at sample point between sample points

Other agent's estimator block is considered in each receiver to reconstruct the original signal based on received data. In this method receiver assumes the data to be constant between each two received samples. This way, a discretized signal (which can be presented to the controllers as a left continuous signal) is generated based on the discrete transmitted data. This strategy can be described as follows:

reconstruction strategy \coloneqq {update with new value at sample point hold the existed value between sample points

2.2.2 On-Demand Strategy for Event-Driven Communication

In this mode, communication is only initiated if an event of interest is declared by the application. For some applications these events are obvious (like a user request), but for many control applications, a signal has to be continuously monitored and a remote or distributed controller has to continuously respond to the change in sensed values. This is where the on-demand strategy proposed in current work demonstrates its strength. Considering such sensing and communication as a problem of tracking a dynamical system over a network, an on-demand strategy (also called error-dependent policy [6][44]) is proposed for communication. This policy operates by scheduling message generation only when the difference between the actual current sensed value and what is perceived as the estimated value at remote agents (Remote Estimator data) reaches a certain threshold. This notion is shown in Figure 2-6, where a "remote estimator", in the sender, repeats the same process as the estimator that is running at the receivers. In a way the sender simulates receiver's behavior. The sender can at any time read the output of the "remote estimator" $(\tilde{x}(t))$ to know the estimation error (of its own signal, x(t)) at the remote agents. This estimation error, calculated as Euclidean norm $e(t) = ||x(t) - \tilde{x}(t)||_2$, is then used in a message generation scheme to decide whether a new message should be sent or not.



Figure 2-6 Block diagram illustrating how error-dependent policy is realized.
This decision could simply be based on a e(t) crossing a threshold e_{th} Therefore, the transmission strategy, executed in small time steps (at the signal sampling rate), is as follows:

Transmissi on strategy := $\begin{cases} \text{transmit } x(t) & \text{if } e(t) > e_{th} \\ \text{do not transmit} & \text{if } e(t) \le e_{th} \end{cases}$

2.2.3 Remote Estimator

Remote estimator uses recent information transmitted through the network to predict the next value of the state variable. Therefore, the objective is to build models that are good representation of sampled signal, for use in the receiver's estimator and the sender's remote estimator.

The only available data in sender are the sampled signal and sampling rate, moreover no unique mathematical function can be assigned to these sample values (different types of sample values in different senders). Thus, the signal can be assumed as a general time varying process. Linear prediction [35][36] is aimed to model the sampled signal because it recursively represents sampled signal based on past values of the signal. Since no specific type for the input of the system in SCMs is assumed, an auto-regressive model (all poles) is applied to fit on the sample values as follows:

$$x(t) = a_1 x(t-T) + a_2 x(t-2T) + \dots + a_n x(t-nT)$$
(2-1)

where in Eq. 2-1, x(t) is the is the sampled signal at time t is sampling time, a_1, a_2, \dots, a_n are coefficients of linear prediction and n indicates the degree of the model. To find the best fit on the sample times, two steps should be considered; selecting the model degree (n), and predicting the sampled value for time steps ahead based on the coefficients of the linear predictor.

One method of finding the best coefficients for linear prediction model is least square error method. Noting that any estimation includes estimation error, for time t-1, Eq. 2-1 is rewritten as follows:

$$x(t-T) = a_1 x(t-2T) + a_2 x(t-3T) + \dots + a_n x(t-(n+1)T) + e(t-T)$$
(2-2)

where in Eq.2-2, e(t - T) is the estimation error. A vector representation for Eq.2-2 is given by:

$$x(t-T) = \varphi^T(t-T)\theta + e(t-T)$$
(2-3)

where φ is the data vector, and θ is the coefficients vector. The vector φ and θ are given by

$$\varphi(t-T) = [x(t-2T) \quad x(t-3T) \quad \cdots \quad x(t-(n+1)T)]^T$$
$$\theta = [a_1 \quad a_2 \quad \cdots \quad a_n]^T$$

If the sample values are available from t - (m + n)T to t - T in the remote estimator, Eq.2-2 can be represented as follows:

$$X = \varphi \theta + e \tag{2-4}$$

where,

$$X = \begin{bmatrix} x(t-T) \\ x(t-2T) \\ \vdots \\ x(t-mT) \end{bmatrix}, \quad \varphi = \begin{bmatrix} \varphi^T(t-T) \\ \varphi^T(t-2T) \\ \vdots \\ \varphi^T(t-mT) \end{bmatrix}, \quad e = \begin{bmatrix} e(t-T) \\ e(t-2T) \\ \vdots \\ e(t-mT) \end{bmatrix}$$

, and m is the window length. Since Least Square method delivers the best coefficients based on minimizing the energy of the estimation error term, and then the following cost function is minimized in this method:

$$J(\theta) = \frac{1}{2} \sum_{i=1}^{m} (x(t - iT) - \varphi^{T}(t - iT)\theta)^{2} = \frac{1}{2} [X - \varphi\theta]^{T} [X - \varphi\theta]$$
(2-5)

The extremum of the cost function with respect to θ should satisfy the following condition:

$$\frac{\partial J(\theta)}{\partial \theta} = [X - \varphi \theta]^T [-\varphi] = 0$$
(2-6)

Since the second derivative of the cost function is positive, the extremum point of the cost function is a minimum and the coefficients vector, which satisfies the minimum point condition, is given by

$$\theta = (\varphi^T \varphi)^{-1} \varphi^T X \tag{2-7}$$

In order to predict the next sample value, two models of computation based on availability of sample values are proposed. First, straight line $(x(t) = b_1 t + b_2)$ is selected to fit on the existing data (the last two samples) and to find the next step value at each sample time instance. In the other word, a straight line is used to curve fit on the last two samples and predict the next sample. The optimized solution for the line fitting is given by

$$X = \begin{bmatrix} 3b_1T + b_2\\ 2b_1T + b_2 \end{bmatrix}, \varphi = \begin{bmatrix} 2b_1T + b_2 & b_1T + b_2\\ b_1T + b_2 & b_2 \end{bmatrix}, \ \theta = (\varphi^T \varphi)^{-1} \varphi^T X = \begin{bmatrix} a_1 = 2\\ a_2 = -1 \end{bmatrix}$$
(2-8)

Therefore, the coefficients are fixed in this method and the strategy to estimate the next value (consider the case that the value at time t should be predicted based on previous samples) in remote estimator is given by

$$\tilde{x}(t) = 2x(t-T) - x(t-2T)$$
(2-9)

Strategy for reconstruction and estimation at the receiver side is given by:

reconstruction strategy at receiver :=
$$\begin{cases} x(t) & \text{if new value received} \\ 2x(t-T) - x(t-2T) & \text{Otherwise} \end{cases}$$

In the second model of computation (control-aware model-based approach), coefficients are not fixed and they are calculated at each sample time instance based on the available transmitted samples. To calculate the coefficients, model order and window length should be determined before the communication process and could be sent through the network to let the receiver know about the best estimation that represents the current status of the signal. It is obvious that larger window length needs more past sample values and can yield more precise estimation. If model degree is chosen too high, over parameterized estimation (matrix $\varphi^T \varphi$ becomes singular) happens. Therefore, the best strategy to solve this issue is to choose n small enough to have a good estimation. Based on Eq. 2-7, coefficients are calculated and the next value is predicted based on the coefficients received over the network (last packet content) and previous sample values. The coefficients are not fixed in this method and the method of estimation in remote estimator based on the received confidents received by the last packet is given by:

$$\tilde{x}(t) = a_1 x(t-T) + a_2 x(t-2T) + \dots + a_n x(t-nT)$$
(2-10)

We assume that the model which is considered for the estimation in the remote estimator is known in the receiver side thus, the strategy at receiver side is given by

reconstruction strategy := $\begin{cases} x(t) & \text{if new value received} \\ a_1 x(t-T) + \dots + a_n x(t-nT) & \text{otherwise} \end{cases}$

2.3 Implementation and Evaluation of Communication Strategies

To investigate the effect of different strategies on communication interface in PSCAD, the IEEE 13-node test feeder [14] is considered as the case of study. The test system consists of a 0.66MW MOD 2 type wind turbine connected to an induction generator at feeder 675 and an SCM was designated at the wind turbine as a sender. A storage component with a bidirectional converter is connected to feeder 675 [37][38] and is also assumed to have a receiver SCM. The converter is designed to work like a switch to change the direction of the flow either into the storage component or out of it. The converter compares the wind turbine power and the power from the battery and generates a reference current signal, which determines the direction of power flow across the converter. The storage component is aimed to store the excess wind turbine power and compensate the lack of power when the amount of power generated by the wind turbine decreases.

In order to simulate more complex topologies than a simple unicast (Peer to peer connection); Comm-Router is considered in communication interface inside the PSCAD. User is able to build different communication links between different SCMs to broadcast the data (transmit a data packet to all other SCMs connected to the Comm-Router) as well as peer to peer transmission of data between two SMCs.



Figure 2-7 Comparison between two transmission modes (a) transmitted data through the network using periodic polling (20Hz), (b) transmitted data using error-dependent policy with window length=2 and $e_{th} = 0.05$, (c) reconstructed data at the receiver for periodic polling, (d) reconstructed data at the receiver using the proposed strategy, (e) performance of periodic polling for transmission rate of 20Hz, (f) performance of error-dependent strategy when window length=2, $e_{th} = 0.05$, (g) voltage of the grid for periodic polling with transmission rate of 20Hz, (h) voltage of the grid for error dependent policy with window length=2 and $e_{th} = 0.05$.

20Hz sampling rate is chosen to sample the wind turbine power in the left hand side figures of Figure 2-7. Figure 2-7-a indicates the transmitted information of the wind turbine power through the communication network and Figure 2-7-c shows reconstructed signal at the receiver (based on the received values and holding them until the next sample time). Figure 2-7-e depicts the performance of the battery according to the transmitted samples from the wind turbine through the network to balance the behavior of wind turbine (blue curve is the wind turbine power and red curve indicates the battery). The battery provides the

necessary power to compensate the wind turbine and stabilize the system at a frequency of 60 Hz. At first, the wind speed is assumed to be constant at 21m/s and then stepped down to 14m/s at time 4.4 sec. After 1 sec the wind speed is stepped up to 23 m/s. These values are considered in such a way that allows investigating the effect of communication in two different cases: 1) decrease in wind speed from a high value to a low value 2) increase in wind speed from a low value to a high value. When the wind speed decreases, the wind turbine power decreases and the battery compensates the lack of power and keeps the system stable. The grid voltage is shown in Figure 2-7-g. as soon as the wind power steps up, the battery starts storing the excess power.

Right hand side of Figure 2-7 shows the same system when error-dependent policy is used (window length is 2) and with error threshold of 0.05 (per-unit). Figure 2-7-b shows the transmitted data of the wind turbine power, and Figure 2-7-d indicates reconstructed signal at the receiver side based on the proposed strategy. It can be observed from Figure 2-7-f that with error dependent policy it is possible to considerably reduce number of transmissions (to a rate of 7 Hz) and still achieve the same performance that periodic polling achieves with a transmission rate of 20 Hz. The gain will be even more for signals that do not vary much, or when a larger error threshold is acceptable. Figure 2-7-h shows the voltage of the grid. It is obvious from Figure 2-7-h that the battery compensates the wind turbine and stabilizes the grid voltage in less than 0.5sec, similar to the case with the rate of 20Hz.

To establish the effectiveness of strategy and the effect of error threshold, the performance of strategy for three different values of e_{th} are compared. In the first case, $e_{th} = 0.1$ is considered. Figure 2-8-a and Figure 2-8-b show quality of tracking and grid voltage for error-dependent policy with $e_{th} = 0.1$ respectively. Second case (Figure 2-8-c and Figure 2-8-d) uses $e_{th} = 0.4$. Form Figure 2-8-c one can see that a communication delay of 0.15sec is introduced because of the higher error threshold value and the grid voltage is affected by this delay as it is shown in Figure 2-8-d. Figure 2-8-e illustrates the performance of the compensation for $e_{th} = 0.9$. It is clear that because the sender is not able to transmit the data when the wind turbine power decreases (the difference is less than the error threshold), the battery is not able to compensate the lack of power. Figure 2-8-f depicts that the voltage of the grid is less than the cases with $e_{th} = 0.1$ or $e_{th} = 0.4$ between t = 4.5and t = 6, because the lack of generated power is not compensated by the battery. As expected, increasing the error threshold decreases the rate of information (transmission rate) and the efficiency of control system; however, a relatively high threshold of 0.4 can still produce acceptable results.



Figure 2-8 Effect of different thresholds (a) performance of battery in compensation of wind turbine, when $e_{th} = 0.1$, (b) voltage of the grid when $e_{th} = 0.1$, (c) performance of battery in compensation of wind turbine, when $e_{th} = 0.4$, (d) voltage of the grid, when $e_{th} = 0.4$, (e) performance of battery in compensation of wind turbine, when $e_{th} = 0.9$, (f) voltage of the grid, when $e_{th} = 0.9$.

In a second study the fault monitoring application from a real power system illustrated in Figure 2-9 was considered. Using Simpower Toolbox, two feeders (FA3 and FA4) are modeled in MATLAB and each of them is protected by a recloser. The circuit includes 16 switches (ZA1-ZA16), which are Cooper DAS-15 type three-phase vacuum switches with 15kVand 630A ratings. Seven of them are normally closed, and the others are supposed to

be in charge of reconfiguration and restoration applications. Transformers in each feeder are 138/12.5 KV and 33.6 MVA.



Figure 2-9 Simulated real power system for fault monitoring process

To investigate the effectiveness of the proposed communication strategy in the power grid, a short circuit is simulated in the grid between switches ZA3 and ZA5 for 0.1 seconds. In this situation, the system observes a transient state. Considering the open/close state of the switches, only feeder FA3 supplies the short-circuited point.

To guarantee an accurate monitoring of fault changes in control center, all device controllers and sensors are required to provide real-time measurement to the control center at the rate of 4800 messages per second [4]. Using the proposed on-demand (error-dependent) strategy for transmission, this rate can be considerably reduced for the communication over the network, while the control center can still be supplied a reconstructed signal with the rate of 4800 messages per second. Figure 2-10 shows the schematic overview of the fault monitoring process, when the on-demand communication strategy is used. The second model of estimation to transmit and reconstruct data at the sender and receiver is applied.



Figure 2-10 Schematic overview of fault monitoring procedure, while error-dependent strategy of transmission.

Figure 2-11 shows the original current signal (a), transmitted samples at the sender (b), and reconstructed signal at the control center (c). The window length is 3 and the threshold error is 30A. As expected, there is no communication when the signal is constant. When the fault happens, the sender transmits new sample values based on the predefined error threshold.

To compare the current signals at the control center with the original signal, the average error and max error are defined as follows:

average error = mean
$$\left(\left| \frac{\text{value at control center-origina value}}{\text{original value}} \right| \right) \times 100\%$$
 (2-11)

$$\max \operatorname{error} = \max\left(\left|\frac{\operatorname{value} \operatorname{at \ control \ center-original \ value}}{\operatorname{original \ value}}\right|\right) \times 100\%$$
(2-12)

The measurement error happens on the signal that has more variations; therefore, the phase that has the biggest changes is studied. Table 2-2 shows the error for six different error threshold values in comparison with the periodic polling method for a fault monitoring case, where the original data has 10000 samples in 0.5 sec. It is clear that when the error threshold increases, the number of transmissions decreases, and consequently, the error of communication strategy increases. Another observation is that the error-dependent policy could even perform better in term of max error.



Figure 2-11 Fault monitoring using on-demand communication strategy.

To support the idea, another scenario is shown in Figure 2-12. In this case the wind speed has a wavy pattern below for two cases of periodic poling and error-dependent policy.

$e_{th(A)}$	Number of transmissions	Max Error	Transmis sionRate	Average error
5	278	1.45%	556 <i>Hz</i>	0.38%
10	197	2.8%	294 <i>Hz</i>	0.8%
30	86	7.7%	172 <i>Hz</i>	1.5%
40	73	11.2%	146 <i>Hz</i>	2.2%
50	61	12.27%	122 <i>Hz</i>	2.7%
60	50	15.3%	100 <i>Hz</i>	3%
N A (periodic)	2500 (periodic)	4.4%	5000Hz (periodic)	0.045%

 Table 2-2

 Accuracy of proposed method for different threshold errors in compare with periodic polling method

N A= Not Applicable

In Figure 2-12, two strategies of periodic polling (blind transmission) and error dependent strategy (model-based networking) are compared. From Figure 2-12-g and Figure 2-12-h we can conclude that with less number of transmissions (rate of 3 Hz) in error-dependent



method, we can achieve the same performance as periodic polling method with transmission rate of 10 Hz.

Figure 2-12 Comparison of transmission modes (a) transmitted data through the network when periodic polling (10Hz), (b) transmitted data through the network using error-dependent policy with window length=2 and $e_{th} = 0.25$, (c) reconstructed data at the receiver when periodic polling, (d) reconstructed data at the receiver when periodic polling, (e) performance of periodic polling with transmission rate of 10Hz, (f) performance of error-dependent strategy when window length=2, $e_{th} = 0.25$, (g) voltage of the grid when periodic polling with transmission rate of 10Hz, (h) voltage of the grid for error dependent policy with window length=2 and $e_{th} = 0.1$.

3 Communication Mechanisms and Uncertainty in Connected Vehicle Application

CAS is one of the important objectives of research in automotive safety. While fully automated systems that can operate under all situations may still be years away, many flavors of collision avoidance systems are being gradually offered by the industry. CAS relies on real-time mapping of their surrounding vehicles to make decisions to either warn the driver or automatically take action to prevent a crash. The source of information for situational awareness could be local sensors (radar and lidar), or information received over a network from other vehicles (or a combination of both). Cooperative systems that rely on information exchange between vehicles have been identified as an early and low cost method of creating the situational awareness needed for hazard detection.

Analysis and development of CAS applications require study of two distinct components of the system: the vehicle safety application (hazard detection and warning) and the communication network. Since these components are not traditionally studied together, there are currently no tools that allow their simulation and analysis in a unified framework. As a result, the mutual effects of these components are rarely studied.

To fill this gap, we study the interaction between CAS applications and V2V networking component. This allows for rapid analysis of vehicle safety applications under different communication network constraints. In this chapter we explore different components of CAS system and discuss them in detail. This chapter is divided into three sections. Section 3.1 discusses the system architecture and functionality of the system. Safety applications of FCW and IMA are presented in section 3.2. In section 3.3, the effect of communication loss and rate adaptation strategies on safety applications are evaluated.

3.1 System Description

Situational awareness has three different subsystems. The first subsystem is the estimation part, which gets the position of other vehicles and makes an estimation of their next move. Each vehicle broadcasts its position to the neighboring vehicles through the network. This information and other information from sensors will be received to the situational awareness system through communication bus. The first subsystem also estimates the position of other vehicles and updates the map of surrounding area in case of network communication failure. The second subsystem of situational awareness is communication logic, which is responsible for information dissemination. after generating the map of neighboring vehicles, the system is responsible for classification of neighborhood of a vehicle. It can be done by different methods such as path history and path prediction. Generally, path history utilizes maneuvers by remote vehicles but path prediction uses the estimated future moves that host vehicle would make. In fact, these methods are applied to classify neighboring vehicles as front vehicle, right vehicle, left vehicle and behind vehicle.

The general architecture of an in-vehicle CAS system is shown in Figure 3-1. For safety applications, one of the classified areas would be the major concern. For example, FCW application, which is devoted to rear-end collision situations, investigates vehicles in front zone of the vehicle and Blind Spot Warning (BSW) considers vehicles in behind zone. CAS applications use threat zones, which are detected by the classification unit to generate proper warnings.



Figure 3-1 Functional blocks of CAS system

The second component of situational awareness is communication logic which is designed to track a dynamical system over the DSRC. A communication strategy explains how the DSRC network is used to convey vehicular information. Broadcasting information over the network has two main responsibilities. The first one is frequency of transmitting safety messages and the second one is power of sending the message. Basically it tells us when to send a message and how far this message can go. Timing of each message is not clear from rate of message. Therefore, in modeling of communication logic we consider this decision. Three different choices are selected from the literature: Periodic Beaconing (PB) [41], Error dependent strategy (ED) [44][100] and Error-Dependent Network-aware (EDN) policy [43]. Some of these methods are currently under consideration for standardization by the automotive industry.

The periodic beaconing strategy uses a simple strategy. It samples the signal periodically and transmits them over the channel without considering the content of it. Error-Dependent policy is more complex. In this approach, a sender simulates the estimation of receiver and based on that, it realizes the error of receiver for the position of sender. If this error is greater than a threshold (e_{th}), the sender will send a new message that contains the updated data. This can correct the receiver's estimation. This method is shown to reduce the rate of message generation while keeping the same estimation error at the receiver [44]. However, this method does not consider the possibility of transmitted packet getting lost. The EDN method uses feedback from the network to improve approximation of the estimation error, reducing the overall error in estimating position of other vehicles over a network. EDN requires approximating PER. This approximation can be done using the mathematical model in [44] or by monitoring lost packets of neighbors from the sequence numbers and approximating PER in the region [43].

Figure 3-2 shows the schematic overview of these methods. The periodic beaconing policy, periodically (or randomly with a predetermined rate) receives the gps data and transmits it over the channel (Figure 3-2-top). This is the baseline design for CAS.

The second aspect of communication logic is power of transmission. It was shown in [43] that power of transmission has a direct effect on the packet reception rate of other nodes in the vehicular network. In this dissertation we study the effect of PER, which includes all kind of collisions (physical and MAC layer collisions) together, on situational awareness procedure and safety application calculations.



Figure 3-2 Communication logic: (top) Periodic Beaconing, (bottom) Network- Aware Error-Dependent.

3.2 Safety Application

In this section two safety applications of Forward Collision Warning and Intersection Movement Assist are introduced. These applications are designed to warn the driver in situations that are identified as collision threat conditions. The major problem in designing these applications is to provide enough time for the driver to react based on situation and warning signals.

3.2.1 Forward Collision Warning

FCW algorithm is designed to alert the driver in a hazardous situation, where the current movement pattern of vehicles may lead to a rear-end crash. This alert should be generated early enough to include driver reaction delay. However, this shouldn't be too early to generate false alarms. False alarms may cause the driver not to trust warnings and act without considering them. Considering this balance is a complicated task and has been the subject of different studies [66][97]. In this section two types of FCW application are discussed. These two algorithms are enhanced version of conventional FCW applications such as CAMPLinear, knipling from [66][98]:

1. Time based FCW

In this approach, the time to enter collision situation and the time needed to avoid accident are driven and compared together. The time to collision (t_c) is a measure indicating the time it takes to make the distance between the front bumper of host vehicle (HV) and the rear bumper of the remote vehicle (RV) zero, if the relative velocity of vehicles stays the same for this duration of time (assuming no acceleration).

 t_c is calculated as follows:

$$t_{c} = \frac{\text{Relative Distance}}{\Delta V}$$
(3-1)

 $\Delta V = V_{HV} - V_{RV}$ is positive. Noting that this application does not consider the situations where the velocity of RV is more than the velocity of HV.

Acceleration of RV is received over DSRC every BSM and HV's acceleration is available and read from communication bus of the vehicle. Some estimation methods like Extended Kalman Filter (EKF) might be adopted if acceleration is not available or if the acceleration data is noisy. Here, we assume that we have correct acceleration values available for both vehicles.

Time to safe situation (t_s) is the time needed to make the velocity difference ($\Delta V = V_{HV} - V_{RV}$) zero assuming average braking for the host vehicle. The effect of relative deceleration (in this case we assume that HV brakes thus $\Delta decel = decel_{rqd} - decel_{RV}$) should be seen in the t_s equation. Our proposed equation to calculate t_s is:

$$t_{s} = \frac{-(V_{HV} - V_{RV})}{(\text{decel}_{rqd} - \text{decel}_{RV})}$$
(3-2)

$$t_r = 2.5 \text{ sec}$$
 (3-3)

Where $decel_{rqd}$ is a configurable value for brake deceleration rate which can be a deceleration rate of an average brake. t_r is the driver reaction time with the default value of 2.5 second in average. To make the final decision about warning generation, t_c will be compared to $t_s + t_r$. The CCA application shall generate warning if $t_c < t_s + t_r$ otherwise it doesn't generate warning.

2. Distance based FCW

In this approach instead of time to collision and safe situation time we define two factors of distance to collision (DTC), which is the relative distance of vehicles, and distance needed to either stop the host vehicle or make the relative speed zero (DTS).

Distance to stop (DTS)

The calculation of DTS shall only be made for vehicles that have been determined to be in the hazardous area of the HV (front zone). DTS is consisted of human reaction distance and stopping distance. The human reaction distance is calculated based on a configurable driver and brake system reaction time (t_r) with default value of 2.5sec as follows:

$$d_{\rm r} = (v_{\rm HV} - v_{\rm RV})t_{\rm r} + 0.5 (a_{\rm HV} - a_{\rm RV})t_{\rm r}^2$$
(3-4)

The distance required to make the speed difference between HV and RV zero ($v_{HVp} = v_{RVp}$), denoted here as DRSD, using an average brake deceleration rate (distance to safe situation) in HV is calculated as:

$$DRSD = \frac{\left\{ \left(v_{HVp} - v_{RVp} \right)^2 \right\}}{\left\{ \left(-2\left(decel_{rqd} - de_{-RV} \right) \right) \right\}}$$
(3-5)

where decel_{rqd} is a configurable deceleration value for an average brake and decel_{HV} (deceleration rate of RV) is received over the DSRC. v_{HVP} and v_{RVP} are predicted values for HV and RV and calculated as follows:

$$\mathbf{v}_{\mathrm{HVp}} = \mathbf{v}_{\mathrm{HV}} + \mathbf{a}_{\mathrm{HV}} \times \mathbf{t}_{\mathrm{r}} \tag{3-6}$$

$$\mathbf{v}_{\mathrm{RVp}} = \mathbf{v}_{\mathrm{RV}} + \mathbf{a}_{\mathrm{RV}} \times \mathbf{t}_{\mathrm{r}} \tag{3-7}$$

The distance required to stop - DTSA - (the distance both vehicles need to make their speed zero), assuming HV brakes, is calculated as follows:

$$DTSA = \frac{v_{HVp}^2}{-2decel_{rqd}} - \frac{v_{RVp}^2}{-2decel_{RV}}$$
(3-8)

The DTS is calculated as follows:

$$DTS = \max(DRSD, DTSA) + d_r$$
(3-9)

We use the maximum of the two quantities above to account for the fact that a vehicle does not continue decelerating after coming to a stop (hence DRSD becoming smaller than DTSA). To make decision about warning generation, DTC will be compared to DTS. The application should issue warning to the driver if DTC < DTS.

3.2.2 Intersection Movement Assist

IMA algorithm is designed to alert the driver in a hazardous situation, where the current movement pattern of vehicles may lead to junction crossing crashes. The IMA application is considered to issue warning to the driver of HV when it is not safe to enter an intersection due to possibility of collision with other RVs. In this sub-section we introduce a new approach for IMA crash possibility calculation.

Every 100ms new information about RV is received over DSRC, the application is triggered to evaluate the situation and check the possibility of collision situation. Figure 3-3 shows the situation when the host vehicle (white vehicle) moves toward an intersection. Conflict area is the location where we do not want both vehicles (HV and RV) to be there at the same time. The idea is to calculate the time needed for RV to reach the conflict area and the time needed for RV to exit the conflict area considering current movement values of RV (received over DSRC). If both vehicles could be in the conflict area at the same time even if the driver brakes, the application will issue warning to the driver and let the driver know about the possibility of collision to take action in advance. Algorithm 1 explains the procedure in detail.



Figure 3-3 Schematic overview of IMA situation utilizing V2V

3.3 Analysis of Impact of communication loss on application performance

Effect of communication loss and safety application can be modeled similar to the work in [99]. However, here we are considering different safety applications than in [99]. We first assume that the communication performance is independent from safety application behavior. Therefore, we investigate the communication impact on the safety application. For this purpose, the effect of communication on the position tracking error (PTE) is investigated. PTE is a system metric that is more relevant to the safety application performance. This metric measures the difference between the actual position of vehicle (RV) and the estimation of its position in the other vehicle (HV). Since one vehicle uses the real-time tracking of other vehicles to generate alerts in FCW, this metric would be more useful than PER.

Algorithm 1 Intersection Movement Assist

- 1: Receive BSM
- 2: Read the information from CAN

T

- 3: The application continues to the next step if $|elev_{HV} elev_{RV}| < \Delta elevation$ where $\Delta elevation$ is a configurable value. If $|elev_{HV} - elev_{RV}| < \Delta elevation$ is not true, the application shall go to the first step and start over.
- 4: Based on the received message over DSRC the application shall continue to the next step if the remote vehicle is moving. If $V_{RV} = 0$ is true, the application goes to the first step.
- 5: If $\Delta H_{min} \leq |h_{RV} h_{HV}| \leq \Delta H_{max}$ the application shall continue to the next step, if not there is no threat and the application jumps to the first step.
- 6: Convert the ECEF coordinates for all of the remote vehicle to East North Up (ENU) coordinates as follows:

$$\begin{split} \mathbf{X}_{ENU} &= -(X_{RV} - X_{HV}) sinLONG_{HV} + (Y_{RV} - Y_{HV}) cosLONG_{HV} \\ \mathbf{Y}_{ENU} &= -(X_{RV} - X_{HV}) sinLAT_{HV} cosLONG_{HV} - (Y_{RV} - Y_{HV}) sinLAT_{HV} sinLONG_{HV} + (Z_{RV} - Z_{HV}) cosLAT_{HV} \end{split}$$

7: Convert the remote vehicle coordinates (using rotation) to be referenced to the host vehicle as follows:

$$X_{RV(HV)} = X_{RV}cos(-h_{HV}) + Y_{RV}sin(-h_{HV})$$

$$Y_{RV(HV)} = -X_{RV}sin(-h_{HV}) + Y_{RV}cos(-h_{HV})$$

- 8: The application shall locate the conflict area by calculating its boundary points (A, B, C, and D) as $A = (-lw, Y_{RV(HV)} + \frac{lw}{2}), B = (-lw, Y_{RV(HV)} \frac{lw}{2})$ $C = (lw, Y_{RV(HV)} - \frac{lw}{2}), D = (lw, Y_{RV(HV)} + \frac{lw}{2})$
- 9: The application shall calculate the time needed for remote vehicle to reach the conflict area as follows:

$$TTI_{RVent} = \frac{|X_{RV(HV)}| - lw}{V_{RV}}$$

10: Calculate the time needed for the host vehicle to reach the conflict area as follows:

$$TI_{HV} = \frac{-V_{HV} \pm \sqrt{V_{HV}^2 + 2a_{HV} \times (Y_{RV(HV)} - \frac{lw}{2})}}{a_{HV}}$$

11: The application shall calculate the time needed for remote vehicle to Exit the conflict area as follows:

$$TTI_{RVout} = \frac{|X_{RV(HV)}| + lw}{V_{RV}}$$

- 12: The application shall continue to the next step if $TTI_{RVent} \leq TTI_{HV} \leq TTI_{RVout}$, or go to the first step otherwise.
- 13: Calculate the distance it takes for driver to stop the vehicle with a hard brake (DTS) as follows:
 - DTS= Driver and brake system reaction (D_r) + Stopping distance (D_s)

$$D_r = V_{HV} \times t_r + \frac{1}{2}a_{HV} \times t_r^2$$
, $D_s = \frac{V_{HV}}{-2 \times Decel}$

Where, t_r and *Decel* are reaction time and brake system deceleration rate. These values are configurable.

- 14: The application shall issue warning to the driver if $DTS \ge (Y_{RV(HV)} \frac{lw}{2})$.
- 15: If the remote vehicles turning signal is on, the time needed for remote vehicle to exit the conflict area as follows:

$$D_s = \frac{|X_{RV(HV)}| - lw}{V_{RV}} + \frac{\sqrt{5}lu}{V_{RV}}$$

The effect of different PER, ranging from 0 to 0.9, on PTE is shown in Figure 3-4 and Figure 3-5. We repeat the experiment for network aware and error dependent policies. Figure 3-4 and Figure 3-5 confirm that as expected increase in PER, increases the error in position tracking. From these experiments, one can model their relation as:

$$PTE = \frac{a}{Rate(1 - PER)} + c$$
(3-10)

Note that if PER is equal to one or rate is equal to zero, position tracking error could grow to infinity. The formula which is mentioned above, satisfies these two condition and also the simulation results which are depicted below.

Moreover, as it is shown in Figure 3-4 increase of PER makes periodic beaconing transmission strategy more erroneous in the sense of PTE. With the same rate of transmission, error dependent policy sends more valuable messages compared to periodic beaconing as it has a better interpretation of its own estimated position in neighboring vehicles. Thus, the PTE is always lower for ED policy. The effect of network aware communication policy in comparison with the error dependent policy is also shown in Figure 3-5.



Figure 3-4 Effect of communication loss on PTE for different transmission strategies. ED with low threshold and high rate, and PB with the same rate as ED, (6Hz)

Network aware method (EDN) has less error in PTE compared to error dependent under the same PER.



Figure 3-5 Effect of communication loss on PTE in use of transmission rate adaptation policies. Both using the same error threshold.



Figure 3-6 Example of a warning timestamps and separation distance when packet error rate is:(top) 0, (middle) 0.6, (bottom) 0.8.

Quality of network would affect safety application's performance since it dictates the HV's position estimation of the RV's. This reduction will show itself as deviations in the estimated separation distance and changes in the warning timestamps. The better the estimation accuracy is, the more timely the warning generations will be.

Figure 3-6 shows the separation distance (blue dots) and warning timestamps (red dots) when the packet error rate is 0 (top), 0.6 (middle), and 0.8 (bottom). Two blue lines were added for clearer comparison of warning timestamps. In this case the results show the changes in estimated separation distance and the moments where the alerts are generated (or missed). Alerts that are not timely will confuse the driver, who may develop a sense of ignoring alerts.



Figure 3-7 Comparison of warning timestamps for different PER values for time-based and distance-based FCW, PER=0 (top), PER=0.4 (middle), PER=0.7 (bottom).

Figure 3-7 indicates the difference between the two FCW applications of time-based and distance-based in term of the impact of communication failure. The top figures show the relative distance between HV and RV when PER is zero. The middle and the bottom show the relative distance for PER 0.4 and 0.7, respectively. As it is shown, increasing PER leads to wrong relative distances between RV and HV. It also causes generating late warning alarms. The reason is that increase in PER values leads to increased PTE. By increasing PTE, the relative distance between vehicles is calculated with some errors and

consequently, the safety application generates late or wrong warnings. Also another observation is that the distance based FCW causes sooner timestamps of warning compared to time-based FCW and this might provide more room for the driver to stop the vehicle or change the direction of movement.

4 Utilizing Model-Based Communication Design for Connected Vehicle Applications

The conventional design and different components of CAS was introduced and studied in chapter 3. While the conventional design of CAS has been shown to provide significant improvement and accuracy, its RTSAC performance degrades significantly where communication loss in the network is too high. This chapter proposes a new framework to transform the conventional implementation of RTSAC in CAS [43][101] to model-based concept and model updates be exchanged between nodes and transferred over the network, instead of raw data. Having model-based communication enabled, we propose to use stochastic model predictive control (SMPC) approach (optimization-based methods) instead of classical control methods of CAS application since behavior models (movement factors) of vehicles are available over the air.

This chapter is divided into three sections. Section 4.1 discusses the system architecture and designed in-vehicle unit. Model development methodology is presented in section 4.2. In section 4.3 the new design for CACC is introduced. Proposed framework is evaluated in section 4.4.

4.1 System Description

The concept of model-communication is shown in Figure 1-3. In this design, each vehicle identifies its movement model and broadcasts the model. These models could be in any mathematical modeling representation, which is rich enough to model vehicle movement parameters considering driver behavior (brake, throttle input). In this section, we use SHS modeling methodology to solve this problem. Each vehicle broadcast its model and state/location in the model (work point) and other vehicles use the received models for situational awareness purposes and better tracking of their neighboring vehicles in their CVS application.



DSRC channel shared by all vehicles

Figure 4-1 Block diagram of the in-vehicle unit, including the communication components and their functionality.

The functional block diagram of the proposed approach is shown in Figure 4-1. In this architecture, the communication module is responsible for sending (model-based packetizing and transmission) and receiving safety messages. Upon receiving the packet and decoding it, the extracted model will be stored in the model bank. The model bank module updates the existed model with the newest one and let the neighborhood classification and CVS application know about the new values and models. Generally, the model bank and neighborhood classification subcomponents are responsible for tracking and locating neighboring vehicles (estimation process), and CVS applications are in charge of issuing warning indicators to the user-interface subcomponent (collision detection module) and processing the automated application and drivetrain control.

These models should be represented in a way that can be communicated and then updated periodically, for example using a hybrid system model [96]. For example, a stochastic hybrid model [88] can be described as:

$$H = (Q, X, Init, f, Dom, G, R)$$
(4-1)

Where *Q* is a finite set of discrete states (e.g. accelerating, decelerating, free following, and coasting), *X* is a set of continuous states. Any pair of $(q, x) \in Q \times X$ is referred as the state of *H*. *Init* $\subseteq Q \times X$ is the set of initial values of the system (q, x(0)). $f(\ldots): Q \times X \to p(X)$ represents the dynamic of the system. *Dom*: $Q \to p(X)$ is the invariant set of the domain of each state and defines combinations of states, events and constraints for which dynamical equations are allowed also assigns an invariant open subset of *x*; here p(x) is the power set of all subsets of *x*. $G: E = Q \times Q \to p(X)$ is the set of transition guard conditions between discrete states and assigns to each $e \in E$ a guard G(e) such that:

- For each $e = (q, q') \in E$, G(e) is a measurable subset of $\partial Dom(q)$.
- For each q∈Q, the family {G(e):e = (q,q') for some q'∈Q} is a disjoint partition of ∂Dom (q).

 $R: E \times X \to P(X)$ assigns to each $e = (q, q') \in E$ and $x \in G(e)$ a reset probability kernel on X concentrated on Dom(q'). Here P(X) is a family of all probability measures on. X.

Assuming a vehicle's movement factors and parameters can be modeled as seen in Figure 1-3, model synchronization over the network can now take the form of updating state of the model by sending regular (q(t),x(t)) pairs, updating model parameters (transition probabilities, state refinements, (f,q(t),x(t)), or an update of the entire model structure .

4.2 Model Development

Based on the application requirements the model has to represent the data in a way that estimation error, error between the estimated and actual values, be within a configurable application-specified tolerable interval. To this end, the estimation errors are considered as the observation sequence elements. Since the observation sequence does not follow a unique known pattern a good idea would be to break the observation sequence down into several portions each comes from a hidden state and stochastically switch between these states. For instance, in the driving situation, we derive models corresponding to a coasting, normal accelerating, aggressive accelerating, aggressive brake, and average braking and so forth. We then classify this situations by determining which model best describes the observation sequence. In this work we use HMM to derive the adaptive SHS. HMM approach [89] has been widely used for mathematical modeling of dynamical systems. Nondeterministic nature of driving scenarios (road and traffic topologies, driver behaviors, etc.) encouraged researchers to use stochastic approaches for driver behavior modeling [82]. Note that in this work we try to find best models which represent the current status of the system in the best way, therefore we might substitute the older observations with the newest ones or give more weight to the new observations. In this section we introduce a new HMM with ARX state representation approach for the purpose of modeling movement parameters which are trained in an online manner.

HMM has a layered perspective for two interconnected stochastic processes, each of which has Markovian property. Each HMM includes three main unknown parameters of initial state probability, state transition probability, and observation probability of each state. Note that none of these probabilities are time dependent and we use continuous HMM notion to model the movement parameters.

4.2.1 ARX approximation

Here we propose that each state of the model uses AutoRegressive eXogenous (ARX) or piece-wise polynomial (PWP) model to track the designated function by training the coefficients of the mathematical model. The ARX model designated for each state is defined as follows:

$$A(q^{-1})y(t) = B(q^{-1})u(t) + \varepsilon(t), \quad \varepsilon \in N(0,\sigma^{2})$$
(4-2)

Where y(t) and u(t) are output and input of the system at time t. $\varepsilon(t)$ is the equation error, and is supposed to have a Gaussian distribution with variance σ . A and B are polynomials of order *n* and *m* in the backwards shift operator q^{-1} .

$$A(q^{-1}) = 1 + a_1 q^{-1} + a_2 q^{-2} + \dots + a_n q^{-n}$$
(4-3)

$$B(q^{-1}) = b_0 + b_1 q^{-1} + b_2 q^{-2} + \dots + b_m q^{-m}$$
(4-4)

There are different criteria to select ARX model order (n and m) in a best way to avoid overparameterization and overfitting such as 1) Akaike's Information Criterion (AIC), 2) Akaike's Final Prediction Error Criterion (FPE), and 3) Rissanen's Minimum Description Length (MDL). These criteria are used to measure relative quality of statistical models for a given set of data. When the problem is finding the driver behavior in accelerating and braking, several factors (e.g. relative distance with the front vehicle, velocity of the front vehicle) could play roles in the model. These criteria could be considered to find the best model orders to represent acceleration and deceleration in terms of those factors.

In more compact form Eq. 4-2 could be represented as:

$$y(t) = \varphi^{T}(t)\theta + \varepsilon(t) \tag{4-5}$$

where:

$$\theta = \begin{bmatrix} a_1 \cdots a_n & b_1 \cdots b_m \end{bmatrix}^T \text{ and } \varphi = \begin{bmatrix} -y(t-1)\cdots - y(t-n) & u(t) \cdots & u(t-m) \end{bmatrix}^T.$$



Figure 4-2 SHS designed for model based communication

4.2.2 Parameters of online-a hidden Markov model

We use the following definition and notation in the proposed SHS model (Figure 4-2):

- Set of states labeled by $S^{t} = \{s_i, \forall i \ i = 1, 2, ..., N^{t}\}$.
- Set of observations up to time t, is denoted by $O_{1:t} = \{o_1 \, o_2 \cdots o_t\}$. Time t is incremented every 100msec in this case of study.

In this work subscript of each parameter refers to the time instance in which that parameter is considered, and superscript indicates the length of observation sequence used for parameter estimation.

- Initial state probabilities are: $\pi_i^t = P\{q_1 = s_i \mid O_{1:t}\}, i = 1, 2, ..., N$.
- State transition probabilities, $A^t = \{a_{ij}^t\}$, where $a_{ij}^t = P\{q_{d+1}^t = s_j \mid q_d^t = s_i\}, 1 \le i, j \le N, 1 \le d \le t-1$
- Coefficients of ARX estimation (dynamical system designated for each state), $\theta_i^t = (c_1^t, c_2^t, \dots, c_{n_i}^t, d_0^t, d_1^t, \dots, d_{m_i}^t)^T$, designated to $s_i, i = 1, 2, \dots, N$.
- Set of observation probability parameters of all states is denoted by
 Bⁱ = {σⁱ_i, ∀i, i = 1,2,..., N}, where σⁱ_i is variance of the zero mean normal distribution of
 observation in state *i*, given O_{ii}.

Estimation error in each state which is a zero mean normal random variable is defined as:

$$o_t = \varepsilon_i^t = y_t - \hat{y}_i^t = y_t - (\theta_i^t)^T \varphi_{i,t}$$
(4-6)

Observation probability for each $1 \le d \le t$, $P\{o_d | q_d = s_i, \lambda^i\}$, is derived from:

$$b_{i}(o_{d}) = \frac{1}{\sqrt{2\pi\sigma_{i}^{t}}} \exp\left\{-\frac{(y_{t} - (\theta_{i}^{t})^{T} \varphi_{i,t})^{2}}{2(\sigma_{i}^{t})^{2}}\right\}$$
(4-7)

Set of all parameters of the SHS is denoted by $\lambda^t = \{\pi_i^t, A^t, B^t, \theta_i^t\}$.

4.2.3 Challenges of HMM derivation

To employ HMM and use it for the purpose of online decision making in a real time CVS application at the beginning of new model derivation we need to make sure that the model covers all aspects of the system. The best solution would be extracting all happened situations and behaviors from the past history of the driving data and include them all in the current model. Thus, the model derivation framework should potentially have evolving characteristic. Specifically, number of states might be adaptive to number of segments that data could be categorized. To generate HMM of any process three main problems should be addressed namely; evaluation problem, decoding problem, and parameter estimation.

1. Evaluation problem

This problem defines the probability of observed sequence occurrence given the last set of calculated model parameters, $P(\mathbf{0}_{1:t}|\boldsymbol{\lambda}^{t})$. Two complexity-wised equivalent solutions, forward and backward algorithms [90][91], have been proposed to address this problem.

2. Decoding problem

Finding most likely underlying state sequence based on given observation sequence is aimed in this problem. In the other words it finds a sequence of the model states with the same length with observation sequence, $Q_{1:t} = (q_1, q_2, \dots, q_t), q_i \in S^t$ which generates the observed sequence with highest probability. Solution to this problem depends on selected definition of "most likely" criterion which could be either single state based or multi-state based optimality. The Viterbi algorithm [92][93] is the most well-known solution to this problem.

3. Parameter estimation problem

Most challenging problem of each HMM design is finding best set of values for model parameters based on observation sequence. These values shall represent the observed sequence as precise as possible. Iterative likelihood maximization using expectation maximization (EM) and numerical optimizations such as gradient decent method are two well-known approaches to solve parameter estimation problem [94]. In this dissertation,

we adopt iterative likelihood maximization using EM but we need to customize the approach to the need of our system design. Two main fixes we need to make are: 1) introducing ARX or PWL model representation to each state of HMM, 2) driving an online adaptive EM frame work for accurate representation of movement parameters. Online adaptation is inevitable because we do not have complete knowledge of training sequence before deriving HMM parameters (observations are based on driving situations and maneuvers) also we need to model the current behavior of the system. Therefore, we need to trigger model generation process as soon as a new observation sequence is available. Also, because of the nature of the case study, older observation samples are less descriptive for the current model representation, therefore batch parameter estimation method does not work in this condition. Note that because of richer information set at time t + n compare to time t one could claim that $q_i^t \neq q_i^{t+n}$. To formulate the parameter estimation problem with no approximation, a complete batch online estimation is utilized after introduction of each new observation to the observation set.

Basically EM procedure is equivalent to maximizing Baum's auxiliary function

$$\arg\max_{\lambda'^{t}} Q_{B}(\lambda',\lambda'') = \arg\max_{\lambda'^{t}} \sum_{Q} P\{Q \mid O_{lx},\lambda'\} \log[P\{O_{lx},Q \mid \lambda''\}]$$
(4-8)

In each iteration and using the outcome as model parameters in the next iteration till convergence where:

$$P\{O_{1:t}, Q \mid \lambda^{t}\} = P(Q \mid O_{1:t}, \lambda^{t}) P(O_{1:t} \mid \lambda^{t})$$
(4-9)

$$P(O_{1:t} \mid \lambda') = \sum_{Q} P(O_{1:t} \mid \lambda')$$
(4-10)

Therefore,

$$Q_{B}(\lambda^{t}, \lambda^{\prime t}) = \sum_{Q} \frac{P\{O_{1:t}, Q \mid \lambda^{t}\}}{P(O_{1:t} \mid \lambda^{t})} \log \left[P\{O_{1:t}, Q \mid \lambda^{\prime t}\} \right]$$
(4-11)

To solve the optimization of Eq. 4-8, it has to be explicitly represented in terms of the model parameters. Thus,

$$P\{O_{1:t}, Q \mid \lambda^{t}\} = \pi_{q_{1}}^{t} b_{q_{1}}^{t}(o_{1}) a_{q_{1}q_{2}}^{t} b_{q_{2}}^{t}(o_{2}) \cdots a_{q_{t-1}q_{t}}^{t} b_{q_{t}}^{t}(o_{t})$$

$$(4-12)$$

$$P\{O_{1:t}, Q \mid \lambda''\} = \pi'_{q_1} b'_{q_1}(o_1) a'_{q_1 q_2} b'_{q_2}(o_2) \cdots a'_{q_{t-1} q_t} b'_{q_t}(o_t)$$
(4-13)

Hence,

$$Q_{B}(\lambda',\lambda'') = \sum_{Q} \left(\frac{\pi_{q_{1}}^{t} b_{q_{1}}^{t}(o_{1}) a_{q_{1}q_{2}}^{t} b_{q_{2}}^{t}(o_{2}) \cdots a_{q_{t-1}q_{t}}^{t} b_{q_{t}}^{t}(o_{t})}{P(O_{1_{x}} \mid \lambda')} \log[\pi_{q_{1}}^{\prime \prime} b_{q_{1}}^{\prime \prime}(o_{1}) a_{q_{1}q_{2}}^{\prime \prime} b_{q_{2}}^{\prime \prime}(o_{2}) \cdots a_{q_{t-1}q_{t}}^{\prime \prime} b_{q_{t}}^{\prime \prime}(o_{t})] \right)$$
(4-14)

$$Q_{B}(\lambda^{t},\lambda^{\prime t}) = \frac{1}{P(O_{Lt} \mid \lambda^{t})} \left\{ \sum_{q_{1}=1}^{N} \sum_{q_{2}=1}^{N} \cdots \sum_{q_{t}=1}^{N} ([\pi_{q_{1}}(a_{q_{1}q_{2}} \cdots a_{q_{t-1}q_{t}})(b_{q_{1}}(o_{1}) \cdots b_{q_{t}}(o_{t}))][\log(\pi_{q_{1}}') + \sum_{d=1}^{t-1} \log(a_{q_{d}q_{d+1}}') + \sum_{d=1}^{t} b_{q_{d}}'(o_{d})]) \right\}$$

Since we need to find the next iteration's parameter values based on the existed ones, we introduce the following definitions $\forall d, 1 \le d \le t$:

$$\begin{cases} \alpha_{d}^{t}(i) = P(o_{1} o_{2} \cdots o_{d}, q_{d} = s_{i} \mid \lambda^{t}) \\ \beta_{d}^{t}(i) = P(o_{d+1} o_{d+2} \cdots o_{t} \mid q_{d} = s_{i}, \lambda^{t}) \end{cases}$$
(4-15)

$$\begin{cases} \gamma_{d}^{t}(i) = P(q_{d} = s_{i} \mid O_{lx}, \lambda^{t}) \\ \xi_{d}^{t}(i, j) = P(q_{d} = s_{i}, q_{d+1} = s_{j} \mid O_{lx}, \lambda^{t}) \end{cases}$$
(4-16)

According to the definitions, these probabilities are derived based on the model parameters as follows (proof by induction):

$$\alpha_{d}^{\prime}(i) = \sum_{q_{1}=1}^{N} \sum_{q_{2}=1}^{N} \cdots \sum_{q_{d-1}=1}^{N} [\pi_{q_{1}}^{\prime} b_{q_{1}}^{\prime}(o_{1}) \times a_{q_{1}q_{2}}^{\prime} b_{q_{2}}^{\prime} \times \cdots \times a_{q_{d-2}q_{d-1}}^{\prime} b_{q_{d-1}}^{\prime}(o_{d-1}) \times a_{q_{d-1}q_{s_{i}}}^{\prime} b_{q_{s_{i}}}^{\prime}]$$

$$d = 1 \Rightarrow \alpha_{1}^{\prime}(i) = \pi_{i}^{\prime} b_{i}^{\prime}(o_{1})$$
(4-17)

$$\beta_{d}^{\prime}(i) = \sum_{q_{d+1}=1}^{N} \sum_{q_{d+2}=1}^{N} \cdots \sum_{q_{l}=1}^{N} [a_{s_{l},q_{d+1}}^{\prime} a_{q_{d+1},q_{d+2}}^{\prime} \cdots a_{q_{l-1},q_{l}}^{\prime} b_{q_{d+1}}^{\prime}(o_{d+1}) b_{q_{d+2}}^{\prime}(o_{d+2}) \cdots b_{q_{l}}^{\prime}(o_{l})]$$

$$d = t \Rightarrow \beta_{l}^{\prime}(i) = 1$$
(4-18)

$$\gamma_{d}^{t}(i) = \frac{\alpha_{d}^{t}(i)\beta_{d}^{t}(i)}{P(O_{1:t} \mid \lambda^{t})}$$
(4-19)

$$\xi_{d}^{t}(ij) = \frac{\alpha_{d}^{t}(i) \, a_{ij}^{t} \, b_{j}^{t}(o_{d+1}) \, \beta_{d+1}(j)}{P(O_{1t} \mid \lambda^{t})} \tag{4-20}$$

To represent the next iteration values of the model parameters recursively, the notions introduced in [89] are adopted and customized for this frame work (derivation in appendix A). The initial state probabilities, $\pi_i^{\prime\prime}$, is expected value of being in s_i at time d=1 based on observations up to time *t*. Hence,

$$\pi_{i}^{\prime t} = \frac{\pi_{i}^{t} b_{i}^{t}(o_{1}) \beta_{1}^{t}(i)}{\sum_{i=1}^{N} \pi_{i}^{t} b_{i}^{t}(o_{1}) \beta_{1}^{t}(i)}$$
(4-21)

Next iteration value for state transition probability from s_i to s_j is the expected number of transition from s_i to s_j divided by expected number of times being in s_i .

$$a_{ij}^{\prime \prime} = \frac{\sum_{d=1}^{t-1} \alpha_d^{\prime}(i) a_{ij}^{\prime} b_j^{\prime}(o_{d+1}) \beta_{d+1}^{\prime}(j)}{\sum_{d=1}^{t-1} \alpha_d^{\prime}(i) \beta_d^{\prime}(i)}$$
(4-22)

Un-biased Maximum likelihood estimation (MLE) for variance of observation distribution in different states is given by:

$$\sigma'_{i}^{2'} = \frac{\sum_{d=1}^{t} \alpha_{d}'(i) |\theta_{i}'' \varphi_{d} - y_{d}|^{2} \beta_{d}'(i)}{\sum_{d=1}^{t} \alpha_{d}'(i) \beta_{d}'(i) - 1}$$
(4-23)

Since the ARX models are zero mean normally distributed, least square method is used for finding the regressor coefficients in iterations of EM algorithm. Thus,

$$\theta_{i}^{\prime\prime} = \left(\sum_{d=1}^{t} \frac{\varphi_{d} \,\varphi_{d}^{T} \,\alpha_{d}^{\prime}(i) \,\beta_{d}^{\prime}(i)}{P(O_{1x} \mid \lambda^{\prime})}\right)^{-1} \left(\sum_{d=1}^{t} \frac{\varphi_{d} \, y_{d} \,\alpha_{d}^{\prime}(i) \,\beta_{d}^{\prime}(i)}{P(O_{1x} \mid \lambda^{\prime})}\right)$$
(4-24)

In order to learn all experienced behaviors by the latest version of the model, a policy should be defined to add any new situation to the model. At each model calculation timestamp, all currently available states from the last model version need to be explored and the best tuned parameter values should be found for each of them according to the new observation element. If at least one of these states satisfies application specified error threshold using its new parameter values, the current model assumed to be fully descriptive for total received observation sequence. But if the minimum achieved error given by the current model exceeds the required threshold, it is required to introduce a new state to represent the new segment of observation and describe the latest driver's maneuver. Hence the number of elements of state set (S^t) is determined as follows:

Protocol 1 Model Generation Policy at BSM t				
Require: $(\lambda^{t-1}, O_{1:t}, e_{th})$				
1: Find \hat{y}_i^t considering new observation (o_t) using λ^{t-1}				
2: if $\min_i y_t - \hat{y}_i^t > e_{th}$				
3: $N^t = N^{t-1} + 1$				
4: calculate λ^t for N^t number of states				
5: else if $\min_i y_t - \hat{y}_i^t \le e_{th}$				
$6: N^t = N^{t-1}$				
7: fine tune λ^t				
8: end				

The on-demand nature of Protocol 1 responds to the fact that a vehicle that has more unexpected movements requires more number of states to represent the status of the vehicle and its model switches between its states more frequently because of the identified error. On the contrary, when the identified error is small and within the tolerable range, the model tends to keep the state and this allows other vehicles to estimate its safety information by just coasting the values using the communicated ARX model.

Every vehicle should inform other vehicles about its new model by transmitting the new SHS model and current values for the underlying state.

4.3 Application to Cooperative adaptive cruise control

Consider a CACC system of a highway with n vehicles shown at Figure 4-3. Each vehicle transmit its basic safety messages (BSMs) using DSRC radio equipment. We assume that all vehicles are equipped with DSRC radio and sensors (e.g. radar, lidar) which enable the capabilities to calculate the relative distance (range), relative velocity and acceleration between themselves and their lead vehicles. In this work we assume that each vehicle

classifies its neighboring vehicles using the information of lidars and network. Two methods of path history and path prediction are applicable approaches for classifying neighboring vehicles under development and implementation of automotive industry. For CACC application purposes the vehicle considers the immediate vehicle in lane (front vehicle) and uses the information received from all vehicles in the range of communication.



Figure 4-3 Demonstration of CACC platoon using model based communication.

Assume that x_i, v_i, a_i are position, velocity and acceleration of vehicle *i* respectively. We find relative distance error in sense of *i* th following vehicle as follows:

$$x_e = x_{i-1} - x_i - hv_i - d_r \tag{4-25}$$

Where *h* is the time gap, and d_0 is the desired distance between vehicles. Note that index *i*-1 is for the lead vehicle and *i* for the following vehicle. The differential equation representing the dynamic of the *i* th vehicle [75] are considered as follows:

$$\dot{x}_{e} = \dot{x}_{i-1} - \dot{x}_{i} - h\dot{v}_{i}$$

$$\Delta \dot{v}_{i} = a_{i-1} - a_{i}$$

$$\dot{a}_{i} = f_{i}(v_{i}, a_{i}) + g_{i}(v_{i})c_{i}$$
(4-26)

where c_i is the engine/brake input (control law), which $c_i \ge 0$ models the throttle input and $c_i < 0$ represents the brake input. Using feedback linearization discussed in [75] the dynamic of the CACC system would be:
$$\dot{x}_{e} = v_{i-1} - v_{i} - h\dot{v}_{i}
\Delta \dot{v}_{i} = a_{i-1} - a_{i}
\dot{a}_{i} = -a_{i}(t) / \xi_{i} + u_{i}(t) / \xi_{i}$$
(4-27)

4.3.1 Stochastic Model predictive design for CACC

Introduce the state variables as $x = [x_1, x_2, x_3] = [x_e, \Delta v_i, a_i]^T$, the continuous-time state-space representation of the CACC system (Eq. 4-26) can be expressed as

$$\dot{x}_{1} = x_{2} - hx_{3}
\dot{x}_{2} = a_{i-1} - x_{3}
\dot{x}_{3} = -\frac{x_{3}}{\xi_{i}} + \frac{u_{i}(t)}{\xi_{i}}$$
(4-28)

To formulate the SMPC controller for the system, discrete time state space representation of the system explained in equations (4-27) is obtained as

$$\begin{cases} x_1(k+1) = x_1(k) + (x_2(k) - hx_3(k))T \\ x_2(k+1) = x_2(k) + (a_{i-1}(k) - x_3(k))T \\ x_3(k+1) = x_3(k) + (-x_3(k)/\xi_i + u_i(k)/\xi_i)T \end{cases}$$
(3-29)

The cost criteria for the stochastic model predictive optimization problem is defined as follows:

$$\min \mathbf{E}_{\{a_{i-1}(k+j)\}_{j=1}^{N-1}} \left[\sum_{n=1}^{N} \left(c_x x_1^2 (k+n) + c_u \Delta u^2 (k+n) + c_v \Delta v^2 (k+n) \right) \right]$$
s.t.

$$\Delta v (k+n \mid k) = x_2 (k+n \mid n)$$

$$\Delta u (k+n) = u (k+n) - u (k+n-1)$$

$$a_{i-1} (k+n \mid k)$$

$$a_{\min} < \frac{v(k+1+n) - v(k+n)}{T} < a_{\max}$$

$$v_{\min} < v(k+n) < v_{\max}$$

$$0 < u (k+n) < u_{\max}$$

$$(4-30)$$

Where, N is the prediction horizon of the optimization problem. The cost criterion defined in Eq. 4-30 aims at several goals such as: minimizing the expected value of relative distance error (spacing error) and keeping the desired distance, smoothing the engine/brake input to the CACC system and providing comforter ride for passengers, keeping the velocity and acceleration in a reasonable interval, and providing the velocity adaptation mechanism in case of the lead vehicle velocity changes.

Following vehicle receives the model from the network and uses these information in its stochastic model predictive optimizations. Note that from the received model in addition to the stochastic model, the following vehicle can realize the most likely estimation for the lead vehicle movement factors and the most probable estimation of those factors for the steps ahead.

4.4 Simulation Results

To simulate realistic scenarios of highway driving and verify the proposed method with different data sets, we considered Highway Fuel Economy Driving Schedule (HWFET) and the connected vehicle data sets from the Safety Pilot Model Deployment Program which are available online on the Research Data Exchange (RDE) provided by the U.S. Department of Transportation's Intelligent Transportation Systems (ITS) Program [102].

A warmed-up engine is used with HWFET and there is no stop in the cycle. Average speed is 21.45 m/s (48 mph) with a top speed of 27 m/s (60 mph) over a 16 km (10-mile) distance. The following parameters are considered for simulation:

$$\begin{split} h &= 0.7 \\ \xi_i &= 0.25 \ \forall i \\ c_x &= 1; \ c_u &= 0.1; \ c_v &= 1; \end{split}$$

For the purpose of reducing complexity of calculation in this section we consider the most probable state of the models instead of calculating expected value of all possible states. To better show the effect of the approach presented in this study, we design four scenarios. At first we investigate the effect of the proposed methodology of model-based communication in a platoon of ten vehicle. Secondly, we compare performance of the proposed method with a well-known conventional CACC [75] and also we compare the performance of

proposed strategy of communication and stochastic MPC with frozen-time MPC which the acceleration is assumed constant over the prediction horizon. At the end, we compare the proposed approach with the zero order hold method in terms of prediction errors and following the trend of underlying function in predicted values.

4.4.1 Ten vehicle CACC

In this scenario nine vehicle run CACC and follow a lead vehicle while trying to keep a predefined distance. Results for this scenario are illustrated in figure Figure 4-4. (a), (b), (c), and (d) depict velocity, relative velocity with the front vehicle, spacing error with front vehicle, and vehicle acceleration respectively. Figure 4-4 realizes that the proposed CACC performs well and the spacing error is always less than 0.6 meter even in a situation where the lead vehicle decelerates suddenly. The spacing error is almost zero when the velocity does not change rapidly. From Figure 4-4-a and c it can be realized that the following vehicles precisely track their preceding vehicles, and meet designated objectives of CACC which are minimizing the spacing error and providing a comfort ride.



Figure 4-4 Ten vehicles Platoon using the proposed CACC.

4.4.2 Comparison with Conventional cruise

In the second scenario of simulation, we compare the performances of the proposed CACC design and the conventional CACC. For the purpose of comparison we look at a group of 5-vehicles since the comparison for the case of 10 vehicles would not add any conceptually valuable information. Four vehicles follow a lead vehicle which moves based on the HWFET drive cycle.

The left hand side plots in Figure 4-5 show the results when the conventional CACC is used and the right hand side plots illustrate use of results from our proposed stochastic model predictive CACC. Comparison of Figure 4-5-a and Figure 4-5-b reveals the quality of tracking the velocity of lead vehicle in both cases. Figure 4-5-b depicts that under consideration of the proposed CACC controller and using the information provided via the model-based communication, the following vehicles brake almost immediately after their preceding vehicle braked. The amplification of the spacing error variation (Figure 4-5-e and f) for both controllers indicate the performance of each controller in terms of keeping the desired distance for the considered scenario.

In addition to HWFET, we used the connected vehicle data sets from the safety pilot model deployment program to test and verify the proposed methodology. We ran some of those maneuvers for both cases of the conventional CACC and the stochastic MPC which uses the proposed model based communication. Figure 4-6 shows the histograms of both methods in terms of the spacing error when we have a platoon of ten vehicles. The left hand side is devoted to the conventional CACC and the right hand side figures depict the proposed methodology. Note that the maximum range of errors for the proposed methodology is in the order of centimeters but when the conventional CACC is the case, it can grow to 10 meters if we study vehicles no. 7 and 8.

4.4.3 Comparison with model predictive controller when BSM is communicated

To separate the effect of new controller design (SMPC) and the impact of new communication strategy (model based communication) we compare the stochastic model predictive cruise control performance when models are available with the frozen model

predictive controller (the scenario that BSMs are communicated, acceleration is assumed constant over prediction horizon, and models are not available). Four vehicles follow a lead vehicle which moves based on the HWFET. The left hand side figures of Figure 4-7 show the results when the MPC CACC is used and the right hand side figures illustrate use of the proposed stochastic model predictive CACC. Left hand side is devoted to raw data communication an acceleration is assumed constant over the prediction horizon, and the right hand side is for model based communication. From the comparison it is obvious that the model based communication and stochastic model predictive control approaches yield better tracking of the lead vehicle, decrease deviations of spacing error.

4.4.4 Comparison of predicted values when models are used with when BSM is held by zero order

To study the effectiveness of model-based communication on prediction of future values, we transmitted models and values at lower rates (less that the original 10hz) and utilized them in prediction of remote vehicle acceleration values. We did the study for 1hz, 2hz and 5hz and results are shown in Figure 4-8 Figure 4-9 Figure 4-10 respectively.

Figure 4-8-(a) shows the comparison of model-based prediction with the case that the acceleration is zero order hold between each two consecutive transmissions of 1hz transmission rate. The absolute error of each case compared to the original values (actual 10hz data) are shown in histograms. As it was expected, when we want to predict the underlying function for the horizon of 1 second or more, the model based approach delivers better prediction in terms of prediction error and following the trend of original function compared to the conventional zero order hold approach.



Figure 4-5 Comparison of the conventional CACC (left) and the proposed method using Model-based communication (right)



Figure 4-6 Histogram comparison of the conventional CACC (left) and the proposed method using Model-based communication (right) in terms of spacing error for different lead vehicle maneuvers.



Figure 4-7 Comparison of frozen-time MPC CACC (when raw data is transmitted) and the proposed stochastic MPC (stochastic models are available).



Figure 4-8 comparison of prediction when models are utilized and zero order holding of acceleration for 1hz. (a) comparison with the original values, (b) histogram of errors for zero order holding, (c) histogram of errors when model based is utilized



Figure 4-9 comparison of prediction when models are utilized and zero order holding of acceleration for 2hz. (a) comparison with the original values, (b) histogram of errors for zero order holding, (c) histogram of errors when model based is utilized.



Figure 4-10 comparison of prediction when models are utilized and zero order holding of acceleration for 5hz. (a) comparison with the original values, (b) histogram of errors for zero order holding, (c) histogram of errors when model based is utilized.

5 Towards Unified Modeling of Communication Based Connected Vehicle Systems

This chapter introduces a new approach for modeling the estimation and networking processes of CVS in a single framework. CVS relies on real-time situation awareness, acquired through cooperation with neighboring vehicles in a wireless network, to predict and respond to hazards. The subsystem that provides situation awareness consists of an estimation process and a networking/communication process. The estimation process is the source of vehicle data that is broadcast through the underlying vehicular network (communication process). The performance of vehicle tracking (estimation process) is significantly affected by the performance of the communication network because as it was explained in chapter 3 in terms of PTE, any issues in sending or receiving the information could result in erroneous position estimates and possibly crashes. It is therefore essential to produce models that allow a clear view into the dynamics of these two components. In this chapter we introduce a new methodology to model the networking and estimation components in a single framework.

Section 5.1 discusses the system description and requirements. In Section 5.2, the model for each component is separately discussed and a hybrid model is presented to combine both estimation and communication. In Section 5.3, the hidden node phenomenon and its model are investigated. The validation of the proposed models in comparison with ns-3 (network simulator 3) results is presented in Section 5.4.

5.1 System Description

The networking/communication process is responsible for disseminating state information of vehicles, and the estimation process samples the information in the sender vehicle (message generation) and recovers communicated states at the destination vehicles (state tracking). Figure 5-1 shows these components and their tasks.

It is clear that performances of these two components are related to each other. The performance of the networking process highly depends on the load of information presented to the network by the estimation process of each vehicle (because the performance of CSMA/CA protocol degrades if the amount of load increases). The performance of the estimation process is affected by the dynamical order of its process and also the performance of the communication process because the estimation process recovers the state information based on the successfully communicated state information at the receiver.



Figure 5-1 Two different components of RTSAC.

Two different scenarios can be considered when modeling the networking component: all vehicles in each other's transmission range, and vehicles interacting only with neighboring vehicles up to a certain distance. When all vehicles are in range, the system follows a rule that situational awareness is required for all vehicles, and each vehicle is assumed to receive state information from all other vehicles (a fully connected network). This is only a hypothetical scenario and is used to create a basic model for the communication component. In the second scenario, the situational awareness is assumed to be required for all neighboring vehicles up to a certain distance; therefore the range of communication and the operation of CSMA/CA will be adjusted accordingly. The second scenario is the realistic case of a CAS or CVS and covers the hidden node phenomenon.

The estimation process of RTSAC could use many different policies; in this work we consider the promising error-dependent policy of [44] (extended in [43]), which is currently under consideration by industry for adoption in the probabilistic form presented in [43].

5.2 Model Description for the Fully Connected Network

In this section, we consider a fully connected network and develop a probabilistic timedautomata (PTA) model for each sender, receiver, and channel. A hybrid automaton is then presented, that consists of estimation process as a continuous-time process and a networking /communication process model as discrete-time process for the condition of all vehicles being within range (fully connected network).

5.2.1 Communication component model

The networking/communication component of RTSAC in CVS system is based on the IEEE 802.11 standard in broadcast mode. Since the communication process is a discrete procedure, we model the communication process as a sequence of discrete processes and events. Furthermore, the communication process is random; therefore, probabilistic timed-automata is used to model the communication part. The probabilistic timed-automata model is characterized by a six-tuple as follows:

$$PTA = \{S, T, I_0, \xi, \Sigma(s), prob\}$$
(5-1)

Where,

• S is a finite set of states (location) of the PTA models. The communication model at the sender, receiver, and channel has different states to model the transmission and reception of the information by these states. In each sender, the model starts from wait until free and ends after the model leaves the transmit state. The models simulate the actual communication process, based on the discrete jumps (transitions) between these states.

- *T* is a set of clocks. Transitions between states occur based on the clocks assigned to them. When the model enters a certain state, its clock is fired and counts from zero to the final clock value.
- *I*₀ ∈ *S* is the initial state (location). In each sender, the model starts from wait until free and ends after the model leaves the Transmit state in each try of broadcasting the data. The channel model starts from free, and the receiver model starts from the waiting state.
- ξ is a finite set of all events. Its members handle all the possible transitions and they are the guards between all the states.
- $\Sigma(s)$ is a set of feasible events at each state *s*.
- prob is a set of transition probabilities. In probabilistic timed-automata different transitions between different states take place base on their assigned transition probabilities. The probabilities of transition between states are 1 if it is not mentioned on the guards.

According to the CSMA/CA protocol, transmission procedures freeze when the channel is sensed to be busy, and transmission is allowed to resume when the channel is determined to be idle again. After a busy period, according to CSMA/CA, each sender should sense whether the channel is idle for an *AIFS* period, then it should follow the procedure from the point at which it was stopped. A random backoff waiting time is used to avoid collisions as much as possible.

Figure 5-2 shows the proposed probabilistic timed-automata model for sender of vehicle *j*. The initial location is indicated by a double circle. If the channel is sensed to be free (idle during the *AIFS* time period), the model jumps to the backoff state and randomly selects a backoff number from a uniform distribution (the contention window size is CW=16, so the probabilities are 1/CW). After passing each slot time, the backoff counter counts down until it reaches zero. The events *free* and *busy*, which are generated by the channel, are urgent events for the Sender. Note that the Model moves toward transmission of the information

by jumping to the *vulnerable* state. The *vulnerable* state and its clock timer are defined to model the time taken for sender to assess the channel and deliver its state to the MAC layer and switch from receive mode to transmission mode. Before switching to the *vulnerable* state, the sender was listening to the channel in order to determine whether the channel becomes busy, in which the procedure will be stopped. When the *vulnerable* state's clock reaches its final value, the model generates $Send_j(k)$ messages for channel and fires the transmission timer in the *transmit* state.



Figure 5-2 Probabilistic timed-automata model for sender at vehicle j in the saturation condition

Trans value is chosen to model the time taken to transmit the information through the channel in the real word. Following a transmit state, the sender restarts the process. This assumes there is always information to transmit i.e., the node and network are in saturation mode. We make this assumption here to develop the basic model, but can relax it when the estimation process is also modeled. The hybrid model is described without the saturation assumption, because the sender should send the state information when it is necessary based on the RTSAC estimation process policy.

Figure 5-3 shows timed-automata model for the channel. We consider two states for the channel, *busy* and *free*. Note that the event *end* $\hat{x}_i(k)$, generated by the transmitter of each vehicle, is an urgent event for channel. When a transmission happens (one sender leaves the Vulnerable state), the channel jumps to *busy* state and delivers its state to all vehicles by busy massages. As soon as the sender finishes the transmission, the channel returns to

free state. Therefore, in the case of all vehicles in range, they always have the same sense about the channel state (because they receive the same free or busy messages from the channel), but when there is a limited range of the transmission vehicles may have different sense about the channel state at different locations and this will yield to hidden node collision. If other transmissions happen while the first transmission has not yet completed and is still underway (this phenomenon could take place because of the *Vulnerable* sate) the channel should wait in busy state until the newest transmission finishes. We model this notion with a self-transition on the top of *busy* state, which resets the *busy* clock.



Figure 5-3 Probabilistic timed-automata model for shared channel.

Figure 5-4 shows the proposed timed-automata model of the receiver. We consider three states for the receiver to model information-packet reception. When a transmission happens and the channel switches to the *busy* state, the receiver switches from *waiting* (while there are no transmissions on the channel) to the *reception* state. Note that the events $Send\hat{x}_i(k)$, $Sends_i$ and *free*, generated by the transmitter of each vehicle and the channel model, are urgent events for the receiver too. It is also possible for multiple transmissions to occur at the same time ($Sends_i$). Therefore, we consider a transition for this condition to model these cases as collisions. If other transmissions occur while the first one is in progress, the receiver jumps to the *collision* state. After all of these events, the receiver model will go back to the waiting state when the channel sends a *free* message. The double circle indicates initial state of the receiver.



Figure 5-4 Timed-automata model for receiver part at vehicle.

One of the measures to verify a timed-automata model is reachability analysis, which indicates that the model will terminate in finite time in each independent broadcast of the information.

Problem 1 (Reachability): Given a transition system $T = (S, \Sigma, S_0, S_f)$, any state $s_f \in S_f$ is reachable from a state $s_0 \in S_0$ (initial state) by a sequence of transitions.

Proof of state reachability:

It was shown that each timed-automaton is a transition system [62]. If we apply the following reachability algorithm presented in [62]:

```
Initialization
reach_{0} = S_{0}
reach_{-1} = \emptyset
i = 0
while reach_{i} \neq reach_{i-1} do
begin
reach_{i+1} = reach_{i} \cup \{s' \in S : \exists s \in reach_{i},
\sigma \in \Sigma(s) with(s, \sigma, s')\}
i = i + 1
end
(5-2)
```

It is trivial to show that the proposed timed-automata model will terminate this algorithm after six steps, and $reach_5=S$. therefore the timed-automata is finite.

It can be concluded from this model that there are two reasons for collision when all vehicles are considered to be within range. First, if multiple transmissions occur at the same time (they have the same backoff number). Second, if two or more transmissions occur in sequence with *vuln* time duration (the difference between their backoff numbers is exactly equal to the *vuln* time slots). With very small *vuln* time slot, this type of collision can be ignored.

5.2.2 Modeling the Estimation Process

There are different policies to decide for communicating the information. It was shown in [43][44] that in the problem of tracking a dynamical system over a network, if message generation and communication have correlation with estimation error, the same performance as the periodic sampling and communication method can be reached using a lower rate of data. The reduced rate of transmission, without sacrificing performance, is a key advantage of the "error-dependent policy". In error dependent policy, each vehicle communicates its current state information (position and speed) only if the difference between its next position and what is perceived as its next position of each vehicle (remote estimator data) is greater than an error threshold. The next position of each vehicle is driven by a Kalman filter based on its current position (from GPS data), speed, and heading [63]. It was proved that a first-order kinematic model is rational for estimating a vehicle's updated position based on position and speed data already available to remote vehicles.



Figure 5-5 Schematic overview of Error-Dependent policy.

Figure 5-5 shows a schematic view of the estimation process. The remote estimator replicates the process running in other vehicles. It stores the last state information presented to the network to decide when the next transmission should occur. Self-estimation determines the next position based on a defined Kalman filter and generates a reference state vector. The remote estimator uses a first-order kinematic equation to estimate the next position of vehicle based on state information broadcast in the previous transmission:

$$\tilde{x}(k) = \tilde{x}(k-1) + \tilde{v}(k-1) \times \Delta T$$
(5-3)

Symbols with (^) on top refer to the Self Estimator's approximations and symbols with (~) on top refer to the perceived or emulated Remote Estimator's approximations. The Scheduler decides whether the difference between the predefined reference position $(\hat{x}(k))$ and estimated position has reached the threshold error or not. It calculates the estimation error based on the next position measured by the Self Estimator and Remote Estimator, then compares its Euclidean norm with the estimation error threshold as soon as it receives it. If the error calculated by the scheduler, $e(k) = ||\hat{x}(k) - \tilde{x}(k)||_2$, is more than the estimation error threshold, e_{th} , the current position and speed of the vehicle $(\hat{x}(k), \hat{v}(k))$ will be broadcast through the network.



Figure 5-6 Hybrid model for CVS system at vehicle j.

5.2.3 Hybrid Model

The hybrid automata concept is a well-known method for mixed discrete-continuous state system modeling. The hybrid model proposed here to model the RTSAC system is as follows:

$$H = (Q, X, Init, \Sigma, f, Dom, E, G)$$
(5-4)

Where Q is a finite set of discrete states, q_1 represents the estimation process and q_2 represent probabilistic timed-automata model, X is a set of continuous states. Any pair of $(q, x) \in Q \times X$ is referred as the state of H. Σ is a set of discrete input symbols and events. $Init \subseteq Q \times X$ is the set of initial values of the system $(\hat{X}_j, \tilde{X}(0))$. $f(\ldots): Q \times X \to p(X)$ represents the dynamic of the system. $Dom: Q \to p(X) \times \Sigma$ is the working domain of each state and defines combinations of states, events and constraints (here error threshold) for which dynamical equations are allowed; here p(x) is the power set of all subsets of X. $E \subseteq$ $S \times S$ is the set of edges, and $G: E \times \Sigma \to p(X)$ is the set of transition guard conditions that enable the transition between discrete states.

Figure 5-6 shows the proposed hybrid model for CVS system for vehicle j. The estimation process (q_1) , estimates the next position of vehicle according to Eq. 5-3. Scheduler determines the error of the estimations based on the information of Self Estimator and Remote Estimator and compares it with the Threshold error, and decides whether the

transmission is necessary or not. The reference vector, \hat{X}_j , contains the real position and speed $(\hat{x}(k), \hat{v}(k))$ of each vehicle at different time steps (information of Self Estimator). If the estimation error is more than the Threshold error, the system will switch to the probabilistic timed-automata model and starts the communication process (probabilistic timed-automata model) from wait until free state. After that the system leaves the Transmit state of the probabilistic timed-automata, the system will update the Remote Estimator information (position and speed) with the current position of vehicle.

Lemma 5-1: the proposed Hybrid model is non-blocking.

Proof: Based on the definitions in [62] the set of reachable state in continuous part (S1) is:

$$reach = \left\{q_1\right\} \times \left\{\tilde{x}\left(k\right) \mid e \le e_{th}\right\}$$
(5-5)

Since that the nature of the vehicle speed based on driver reactions is random (because of sudden breaking) we have two possibility for Trans (transition guard) set, that are:

$$Trans_{1} = \left\{q_{1}\right\} \times \left\{\tilde{x}(k) \mid e > e_{th}\right\}$$
(5-6)

$$Trans_2 = \left\{q_1\right\} \times \left\{\tilde{x}\left(k\right) \mid e = e_{th}\right\}$$
(5-7)

If $Trans_1$ is the case, it is trivial that $Reach \cap Trans = \emptyset$ and thus the model is nonblocking. If $Trans_2$ is the case, it is obvious that $Reach \cap Trans \neq \emptyset$. Considering the following lemma, because there is a guard between the estimation and communication process ($e \ge e_{th}$) therefore, the Hybrid model is non-blocking. Note that we proved that the communication process is finite timed-automata.

Lemma 5-2 A Hybrid automaton, *H*, is non-blocking if for all, $(q, x) \in Reach \cap Trans$ there exists $\dot{q} \in Q$ such that $(\dot{q}, q) \in E$ and $x \in G$ (q, \dot{q}) .

Proof follows [62].

5.3 Modeling Hidden Node Phenomenon

So far we have assumed a channel which sends free and busy messages to all vehicles; therefore, each vehicle has the state of the entire network. In the case of limited range of transmission, vehicles are not able to broadcast information to all other vehicles, and only vehicles within a certain distance (communication range) of a transmitting vehicle will receive the transmitted data. It should be noted that in this case vehicles do not have the information of entire network and they only can listen to the channel within their range. Figure 5-7 presents a simplified illustration. Consider a case where vehicle B has information to transmit and listens to the channel; if the channel is determined to be free, then vehicle B will attempt to broadcast its information according to CSMA/CA. But vehicle D, which is not located within the range of vehicle B, attempts to broadcast its data while vehicle B is transmitting, yet not violating CSMA/CA rules. In such a case, the packets from the two vehicles will collide at the receiver, vehicle C, who is located between and within range of both senders. These collisions could adversely affect the estimation process if the network is not properly managed. In the model of this case (limited range), each vehicle has the information of vehicles within its range, but not of the entire network. Therefore the authority to make decisions is local. Figure 5-8 shows the proposed model for the receiver part of vehicles. In this model we embed the channel model inside the receiver (because each vehicle senses the channel based on its range), and each receiver provides free and busy messages only for its transmitter. Each vehicle receives Send $\hat{x}_i(k)$ and *Sends_i* signals only from vehicles located within its range.



Figure 5-7 Schematic overview of hidden node phenomenon.



Figure 5-8 Receiver model for vehicle j.

We define a closed sphere around each vehicle to indicate the range (R_j) . In this model of receiver, we have three states namely waiting, reception, and collision. In the *waiting* state, the receiver produces free messages repeatedly, allowing for data to be broadcast. The model for the sender and the hybrid model will be the same as before. Furthermore, because these free and busy messages are not shared with other vehicles, the hidden node collision will be inevitable.

5.4 Model Verification

To evaluate the accuracy of the model, we compare the results generated by the model and Network Simulator 3 (ns-3) for several different scenarios. We consider the probability of successful transmissions with respect to the number of vehicles, range, and rate of transmission as the measure for this comparison. The model is implemented in a multithread program in JAVA. We study the model for two cases of a fully-connected network (all vehicles in range) and hidden node scenario (in a highway setting).

ns-3 is an open source discrete event simulator that is maintained by a large community of researchers and developers. It is currently in use by automotive industry for verification of

vehicular communication and their related systems. Due to its discrete event nature and the fact that RTSAC runs as a periodic (discrete) repetitive process, it is straightforward to implement all the relevant parts of RTSAC in ns-3. To verify the proposed model, we configure ns-3 in a way that it behaves similarly to the model. To this end, an ideal channel was created with no fading and propagation loss. This means that each transmitted packet is successfully delivered to receivers up to a specified distance regardless of phenomenon such as fading. This distance shows the transmission range, thus the packet will not be received at any receiver beyond it. The PHY and MAC layer in ns-3 are configured to mimic DSRC (802.11p in 10MHz) in broadcast mode (AdHoc) with a rate of 6 Mbps.

All clocks of models are scaled based on a fix slot time (time duration of 13 μ s). Hence, other time dependent model parameters (Table 5-1) are calculated based on this slot-time. The same calculation also is considered for ns-3. For example, one second of simulation time will be 76923 time-slots, the packet size in ns-3 is 388 Bytes (324 payload+64 header).

Parameter	Value
Slot Time	13 µsec
AIFS	2×slot time
Vulnerable Time	1×slot time
Transmission time	38×slot time
Contention Window size	16×slot time
Packet size in ns-3	388 bytes
Channel Capacity	2026 packets

Table 5-1System parameters

Starting the CSMA/CA procedure from sent state instead of a pre-start Backoff is the significant implementation issue of ns-3, which causes a completely concurrent collision in all of the transmitted packets. To avoid this issue, each vehicle in ns-3 should initialize its transmission by choosing a random delay drawn from a uniform distribution with the size of contention window.

5.4.1 Fully Connected Network Scenario

The probability of successful transmission is calculated by averaging over the number of successfully received packets (number of times the reception state was visited subtracted by the number of times the collision state was met by a receiver) divided by number of transmitted packets.

Figure 5-9 depicts the probability of successful transmission according to various numbers of vehicles for constant transmission rate (50 Hz). As it was expected, the probability of success decreases after the channel becomes saturated.



Figure 5-9 Probability of success vs. number of vehicles - all in range scenario.

The probability of successful transmission for different rates is shown in Figure 5-10. The results are shown for three different cases of 40, 50, and 60 vehicles. The channel behavior can be observed while it shows a saturation point for each case, which depends on the number of vehicles, and a unique pattern of decreasing.



Figure 5-10 Probability of success vs. transmission rate - all in range scenario

5.4.2 Hidden Node Scenario

In this scenario, vehicles are distributed in a highway-like pattern. To investigate effect of hidden node problem precisely, and avoid effects of edges of the topology, the middle 1/5 of the vehicles are used as the test nodes to calculate the probability of successful transmission (e.g. 50 vehicles in the center out of a highway with 250 vehicles). ns-3 uses location of vehicles to create a mobility trace file. The proposed Hybrid model also uses the same reference vector of positions for its estimation process (\hat{X}_i).

Figure 5-11 shows the probability of successful transmission for hidden node scenario with respect to different ranges of transmission. Note that the rate of transmission is fixed to 10Hz and the values are calculated over 50 vehicles in center of a highway with 250 vehicles. As the range of transmission increases, the average probability of successful transmission in a neighborhood around a vehicle increases; the reason is that more vehicles are aware of each other's transmission and cooperate with each other in a CSMA manner. It must be noted that the total capacity decreases in this case since the channel is shared by more vehicles when the range of transmission increases. It is only the probability of success for the few transmitted packets that increases.

The mutual effect of transmission rate and range on successful transmission in the case of 50 vehicles is shown in Figure 5-12. Since the capacity of channel is limited to a constant

amount of packets, increasing the rate will improve the probability of successful transmission until it reaches the saturation point of the channel. After that, increasing the rate will increase the number of hidden node collisions which results in less successful transmissions.

Another measure used to evaluate the Hybrid model is the rate of message transmission as a function of the estimation error threshold. The reference position vector, \hat{X}_j , is assumed to perform like a GPS device, which updates the speed of the vehicles every 6.5msec via a zero mean Gaussian distribution for acceleration. Therefore, every 6.5msec the speed vector will update its value.

Figure 5-13 shows the result for different values of threshold error. It can be observed that the increase of the error threshold affects the rate of transmission inversely (as expected).



Figure 5-11 Probability of success vs. transmission range - Hidden node scenario.



Figure 5-12 Probability of success vs. transmission rate and range - Hidden node scenario.



Figure 5-13 Effect of threshold error on transmission rate.

6 Conclusion and Future Work

The communication component is an essential part of the emerging cyber-physical systems. The performance of the communication component has a profound effect on the overall performance of the control algorithms and eventually on the behavior of the physical system. The main contributions of this dissertation is the study and design of communication approaches for two domains of energy and transportation cyber-physical applications.

Two contributions to the design, modeling and simulation of communication mechanisms and protocols for CPES were presented in chapter 2: 1) A novel communication mechanism has been proposed for efficient use of communication protocols and technologies in CPES, 2) Models and simulation tools for communication strategies, protocols, and technologies have been developed to add communication simulation and modeling capability to the industry's commonly used power system transient level simulator, PSCAD.

The presented on-demand mechanism is verified by our PSCAD embedded network simulator that simulates communication network components in PSCAD. The simulator is designed in a way that it allows integration of new communication strategies and abstract forms of communication protocols, such as DNP3 and IEC61850. According to DNP3 and IEC 61850, communication enabled devices can either work in event-driven mode or in non-event-driven mode (i.e., polling based). The non-event-driven mode of operation works based on periodic and frequent polling of sampled signal. The non-event-driven mode of communication is supposed to be initiated by the receiver. On the other hand, in event-driven method of communication, messages are generated and sent from senders based on their determination of when communication is needed. Since the event-driven method does not work based on a fixed rate of transmission, it can provide a lower communication overhead and significantly lower latency, for fault detection application.

While event-driven method is specified in the standard, the strategy that creates the events needs to be designed for each application. We presented one such strategy in chapter 2. However, following the same concepts in chapter 3 and 4, one could possibly design a

model-based strategy for CPES. Using this concept, the remote estimator is able to track and estimate the dynamical system using a linear prediction, piece-wise linear, or any other mathematically proven method of system identification and forecasting. One possible direction could be online adaptation and learning the behavior of underlying function, modeling the behavior, and transmitting the model which is learnt through the network. When receivers get the information, they can use this information to optimize their control and actuations in a model predictive way. The information about other agents and their behavior will form the situational awareness in a much more elaborative and control-aware form than is currently possible. Therefore, making decision about what is better to be communicated could be a promising area of research.

The current form of communication in connected vehicle (CV) applications is expected to take the form of frequent broadcast of sensed data (non-event-driven mode) to other vehicles in the proximity of a sender. The broadcast messages are used by the receivers to track the movement of the sender of information in real-time. Given the limited capacity of the channel, usually short messages with limited information about movement of vehicle are used with DSRC. This limitation degrades the accuracy and resolution of tracking, and eventually leads to constraints in the type of applications that can be developed. In chapter 3, we introduced different approaches for FCW and IMA hazard detection, and investigated the impact of communication uncertainties on the accuracy of alert generation. While closed form mathematical expressions are possible for some components of the system, the entire CAS system is very complex to be modeled mathematically. We employed a co-simulation methodology to conduct a simulation study of the impact of communication uncertainties.

Co-simulation results indicated that the performance of communication has immediate and significant impact on the accuracy of tracking and alert calculations. It was illustrated that loss of communication causes incorrect calculation of values such as separation distance, time to collision, and time to stop. Miscalculation of these values leads to warnings being generated earlier or later than they should be. Another observation we made after study of different application specifications and features is that the impact of communication on different choices of CAS application is not the same. For example, it was shown in chapter

3 that the two approaches of FCW, i.e., time-based and distance-based, have notably different accuracy levels (up to 5-8 % difference) in the case that remote vehicle is decelerating and host vehicle moves toward it.

Two communication approaches, control-aware and rate adaptation, were studied in this dissertation to enhance the situational awareness. Increasing the utility of communication, by including information that is more meaningful to the automated application, has the potential to considerably enhance the situational awareness and eventually improve the performance of CV applications. We utilized one such approach in this dissertation, namely the model communication concept, and demonstrated a realization of this concept in the context of automated cooperative applications, such as CACC in chapter 4.

The presented control-aware design, benefits from two distinctive contributions. First, we developed an HMM-based model of a vehicle and driver that allows describing the vehicle dynamics in communicable forms. Second, the communicated model was used with a stochastic model predictive controller. To cover all behaviors of the underlying function, the approach presented in chapter 4 is to break down the observation sequence into several portions each comes from a hidden state and stochastically switch between these states. Online adaptation of parameters is a novel concept presented in the proposed framework. It can be noted that the stochastic MPC control scheme for CACC is also a novel design, which utilizes the model-based communication in its optimization. The presented methods in this dissertation are applicable to other CV applications and even with further generalization to other networked systems. In particular, one possible direction to extend the proposed approach is to study the effect of this design in complicated connected vehicle scenarios to achieve more accurate and precise situational awareness.

To understand how control awareness or rate adaptation impact situational awareness, it is desirable to develop models that considers the behavior of communication component and the coupled control or physical processes. For the case of CV, the real-time situational awareness sub-system consists of two major components, estimation, which controls the transmission rate, and communication that is responsible for information dissemination. The interaction of these two sub-components has an immediate and profound impact on

the overall performance of the real-time situational awareness sub-system. To study the mutual impact of these sub-components there was a need for a unified framework to model the two sub-components. In chapter 5, the two sub-components were modeled in a single framework. A probabilistic timed-automata was presented to model the communication component and a hybrid automaton was proposed to model the entire system. Due to the randomness of the protocol (CSMA/CA) used in these systems and the discrete nature of communication process, probabilistic timed-automata was assumed to be a rational modeling strategy. A first order kinematic equation was designated to estimate the next position of vehicle based on the received state information. Two different timed-automata models for two cases of all vehicles in range and hidden node phenomenon were introduced.

The hybrid model can be extended to study other estimation processes by replacing the estimation state, while the remaining parts of the model would remain intact. For example, the probabilistic error-dependent policy can use a calculated transition probability out of the estimation state. In addition to rate adaptation strategies, the communication part can potentially be enhanced to allow power or range variations based on application of adaptive power control methods.

REFERENCES

- [1] T. Sauter and M. Lobashov, "End-to-End Communication Architecture for Smart Grids," in *IEEE Transactions on Industrial Electronics*, vol. 58, no. 4, pp. 1218-1228, April 2011.
- [2] V.C. Gungor; D. Sahin; T. Kocak; S. Ergut; C. Buccella; C. Cecati; G.P. Hancke, " A Survey on Smart Grid Potential Applications and Communication Requirements," in *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 28-42, Feb. 2013.
- [3] M. D. Ilic, L. Xie, U. A. Khan and J. M. F. Moura, "Modeling of Future Cyber–Physical Energy Systems for Distributed Sensing and Control," in *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 40, no. 4, pp. 825-838, July 2010..
- [4] X. Lu, W. Wang and J. Ma, "An Empirical Study of Communication Infrastructures Towards the Smart Grid: Design, Implementation, and Evaluation," in *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 170-183, March 2013.
- [5] Q. Yang, J. A. Barria and T. C. Green, "Communication Infrastructures for Distributed Control of Power Distribution Networks," in *IEEE Transactions on Industrial Informatics*, vol. 7, no. 2, pp. 316-327, May 2011.
- [6] Yonggang Xu and J. P. Hespanha, "Estimation under uncontrolled and controlled communications in Networked Control Systems," *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on, 2005, pp. 842-847.*
- [7] Nasiriani, N.; Ramachandran, R.; Rahimi, K.; Fallah, Y.P.; Famouri, P.; Bossart, S.; Dodrill, K.," An embedded communication network simulator for power systems simulations in PSCAD," *Power and Energy Society General Meeting (PES), 2013 IEEE*, Vancouver, BC, 2013, pp. 1-5.
- [8] K. Curtis, "DNP3 protocol primer," DNP User Group, 2005.
- [9] ABB, 630 Series DNP3 Communication Protocol Manual, 2009.
- [10] IEEE Standard for Electric Power Systems Communications—Distributed Network Protocol (DNP3), IEEE Std. 1815-2010, IEEE, 2010.
- [11] Y. Liang and R. H. Campbell, "Understanding and Simulating the IEC 61850 Standard", Technical Report UIUCDCS-R-2008-2967, University of Illinois at Urbana-Champaign, May 2008.
- [12] International Electrotechnical Commission, "Communication Networks and Systems in Substations", IEC 61850, 2003.
- [13] IEEE Standard Communication Delivery Time Performance Requirements for Electric Power Substation Automation, IEEE Std 1646-2004, IEEE, 2005.

- [14] IEEE 13 Node Test Feeder. Available at: http://ewh.ieee.org/soc/pes/dsacom/testfeeders
- [15] M. Liserre, T. Sauter and J. Y. Hung, "Future Energy Systems: Integrating Renewable Energy Sources into the Smart Power Grid Through Industrial Electronics," in *IEEE Industrial Electronics Magazine*, vol. 4, no. 1, pp. 18-37, March 2010.
- [16] C. H. Lo and N. Ansari, "The Progressive Smart Grid System from Both Power and Communications Aspects," in *IEEE Communications Surveys & Tutorials*, vol. 14, no. 3, pp. 799-821, Third Quarter 2012.
- [17] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, G. P. Hancke, "Smart Grid Technologies: Communication Technologies and Standards," in *IEEE Transactions on Industrial Informatics*, vol. 7, no. 4, pp. 529-539, Nov. 2011.
- [18] S. Mohagheghi, J. Stoupis and Z. Wang, "Communication protocols and networks for power systems-current status and future trends," *Power Systems Conference and Exposition*, 2009. PSCE '09. IEEE/PES, Seattle, WA, 2009, pp. 1-9.
- [19] D. A. Panasetsky, P. V. Etingov and N. I. Voropai, "Multi-agent approach to emergency control of power system," *Electric Utility Deregulation and Restructuring and Power Technologies, 2008. DRPT 2008. Third International Conference on*, Nanjuing, 2008, pp. 2157-2161.
- [20] F. Yang, D. Zhao, X. Zhang and J. Wu, "Research of relay protection setting calculation system for power plant based on Multi-agent," *Power and Energy Engineering Conference* (APPEEC), 2010 Asia-Pacific, Chengdu, 2010, pp. 1-4.
- [21] H. E. Farag and E. F. El-Saadany, "Voltage regulation in distribution feeders with high DG penetration: From traditional to smart," *Power and Energy Society General Meeting*, 2011 *IEEE*, San Diego, CA, 2011, pp. 1-8.
- [22] F. Ren, M. Zhang, D. Soetanto and X. Su, "Conceptual Design of A Multi-Agent System for Interconnected Power Systems Restoration," in *IEEE Transactions on Power Systems*, vol. 27, no. 2, pp. 732-740, May 2012.
- [23] M. Kezunovic, "Smart Fault Location for Smart Grids," in *IEEE Transactions on Smart Grid*, vol. 2, no. 1, pp. 11-22, March 2011.
- [24] M. Nthontho, S. Chowdhury and S. Winberg, "Smart communication networks standards for smart energy management," *Telecommunications Energy Conference (INTELEC)*, 2011 *IEEE 33rd International*, Amsterdam, 2011, pp. 1-9.
- [25] K. Hopkinson, Xiaoru Wang, R. Giovanini, J. Thorp, K. Birman and D. Coury, "EPOCHS: a platform for agent-based electric power and communication simulation built from

commercial off-the-shelf components," in *IEEE Transactions on Power Systems*, vol. 21, no. 2, pp. 548-558, May 2006.

- [26] M. Wei, Z. Chen, "Communication Systems and Study Method for Active Distribution Power Systems", NORADAC 2010 Conference
- [27] M. Rahnamay-Naeini, Z. Wang, A. Mammoli and M. M. Hayat, "Impacts of control and communication system vulnerabilities on power systems under contingencies," *Power and Energy Society General Meeting*, 2012 IEEE, San Diego, CA, 2012, pp. 1-7.
- [28] J. Nutaro, P. T. Kuruganti, M. Shankar, "Integrated Modeling of the Electric Grid, Communications, and Control", ORMMES, 2006
- [29] Hua Lin, S. Sambamoorthy, S. Shukla, J. Thorp and L. Mili, "Power system and communication network co-simulation for smart grid applications," *Innovative Smart Grid Technologies (ISGT), 2011 IEEE PES*, Hilton Anaheim, CA, 2011, pp. 1-6.
- [30] H. Lin, S. Veda, S. Shukla, L. Mili and J. Thorp, "GECO: Global Event-Driven Co-Simulation Framework for Interconnected Power System and Communication Network," in *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1444-1456, Sept. 2012.
- [31] G. N. Ericsson, "On requirements specifications for a power system communications system," in *IEEE Transactions on Power Delivery*, vol. 20, no. 2, pp. 1357-1362, April 2005.
- [32] OPNET Technologies Inc., Modeling Concepts Reference Manual Product Release 10.5. Bathesda, MD: OPNET Technologies, Inc., 2004.
- [33] X. Wang *et al.*, "Interfacing Issues in Multiagent Simulation for Smart Grid Applications," in *IEEE Transactions on Power Delivery*, vol. 28, no. 3, pp. 1918-1927, July 2013.
- [34] Triangle MicroWorks, Inc., DNP3 Overview.
- [35] J. Makhoul, "Linear prediction: A tutorial review," in *Proceedings of the IEEE*, vol. 63, no. 4, pp. 561-580, April 1975.
- [36] G.H. Riahy, M. Abedi, Short term wind speed forecasting for wind turbine applications using linear prediction method, *Renewable Energy*, Volume 33, Issue 1, pp. 35,41, January 2008.
- [37] A. Mohamed, M. Elshaer and O. Mohammed, "Bi-directional AC-DC/DC-AC converter for power sharing of hybrid AC/DC systems," *Power and Energy Society General Meeting*, 2011 IEEE, San Diego, CA, 2011, pp. 1-8.
- [38] M. Hilal, M. Maaroufi and M. Ouassaid, "Doubly fed induction generator wind turbine control for a maximum power extraction," *Multimedia Computing and Systems (ICMCS), 2011 International Conference on*, Ouarzazate, 2011, pp. 1-7.

- [39] Y.P. Fallah, P. Famouri, S. Bossart, "Why an Application Agnostic Internet Model May Not Work for Communication in Cyber-Physical Energy Systems", *NSF National Workshop* on Energy Cyber-Physical Systems, 2013.
- [40] Vehicle Safety Communications Applications (VSC-A) Final Report, Tech. Rep. DOT HS 811 492A, September 2011.
- [41] CAMP Vehicle Safety Communications Consortium, Vehicle safety communications project task 3 final report: Identify intelligent vehicle safety applications enabled by DSRC, Tech. Rep. DOT HS 809 859, March 2005
- [42] IEEE 802.11 WG, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE, Aug. 1999.
- [43] C. L. Huang, Y. P. Fallah, R. Sengupta and H. Krishnan, "Adaptive intervehicle communication control for cooperative safety systems," in *IEEE Network*, vol. 24, no. 1, pp. 6-13, Jan.-Feb. 2010.
- [44] S. Rezaei, R. Sengupta, H. Krishnan, X. Guan, R. Bhatia, "Tracking the position of neighboring vehicles using wireless communications", *Transportation Research Part C*: Emerging Technologies, Volume 18, Issue 3, pp. 335-350, June 2010.
- [45] Y. P. Fallah, C.L. Huang, R. Sengupta and H. Krishnan, "Design of Cooperative Vehicle Safety Systems based on Tight Coupling of Communication, Computing and Physical Vehicle Dynamics", in Proc. of ACM/IEEE Int. Conf. on Cyber-Physical Systems, ICCPS 2010
- [46] Y. P. Fallah and R. Sengupta, "A Cyber-physical Systems Approach to the Design of Vehicle Safety Networks," *Distributed Computing Systems Workshops (ICDCSW)*, 2012 32nd International Conference on, Macau, 2012, pp. 324-329.
- [47] Y. P. Fallah, C. Huang, R. Sengupta and H. Krishnan, "Congestion Control Based on Channel Occupancy in Vehicular Broadcast Networks," *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, Ottawa, ON, 2010, pp. 1-5.
- [48] Q. Chen, D. Jiang, T. Tielert and L. Delgrossi, "Mathematical Modeling of Channel Load in Vehicle Safety Communications," *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, San Francisco, CA, 2011, pp. 1-5.
- [49] Y. P. Fallah, C. L. Huang, R. Sengupta and H. Krishnan, "Analysis of Information Dissemination in Vehicular Ad-Hoc Networks With Application to Cooperative Vehicle Safety Systems," in *IEEE Transactions on Vehicular Technology*, vol. 60, no. 1, pp. 233-247, Jan. 2011.

- [50] E. Moradi-Pari, A. Tahmasbi-Sarvestani and Y. P. Fallah, "Modeling communication and estimation processes of automated crash avoidance systems," *Systems Conference (SysCon)*, 2013 IEEE International, Orlando, FL, 2013, pp. 681-687.
- [51] C. Cassandras and S. Lafortune, Introduction to Discrete Event Systems, Springer, 2007.
- [52] R. Alur and D. L. Dill, "A theory of timed automata," Theoretical Computer Science, 126(2):183–235, 1994
- [53] G. Rodriguez-Navas and J. Proenza, "Using Timed Automata for Modeling Distributed Systems with Clocks: Challenges and Solutions," in *IEEE Transactions on Software Engineering*, vol. 39, no. 6, pp. 857-868, June 2013.
- [54] M. Kwiatkowska, G. Norman, and J. Sproston, "Probabilistic Model Checking of the IEEE 802.11 Wireless Local Area Network Protocol," in Proc. 2nd Joint International Workshop on Process Algebra and Probabilistic Methods and Performance Modeling in Verification (PAPMPROBMIV 2002), ser. Lecture Notes in Computer Science, vol. 2399. Springer, 2002, pp. 169–187.
- [55] A. Heindl and R. German, "Performance modeling of IEEE 802.11 wireless LAN with stochastic Petri nets," Performance Evaluation, 44:139–164, 2001.
- [56] Fruth, M., "Probabilistic Model Checking of Contention Resolution in the IEEE 802.15.4 Low-Rate Wireless Personal Area Network Protocol," Leveraging Applications of Formal Methods, Verification and Validation, 2006. Second International Symposium on , vol., no., pp.290-297, 15-19 Nov. 2006
- [57] M. Kwiatkowska, G. Norman, D. Parker, and J. Sproston, "Performance Analysis of Probabilistic Timed Automata using Digital Clocks," Formal Methods in System Design, vol. 29, no. 1, pp. 33–78, 2006.
- [58] K. D. Kim and P. R. Kumar, "Cyber–Physical Systems: A Perspective at the Centennial," in *Proceedings of the IEEE*, vol. 100, no. Special Centennial Issue, pp. 1287-1308, May 2012.
- [59] P. Derler, E. A. Lee and A. Sangiovanni Vincentelli, "Modeling Cyber–Physical Systems," in *Proceedings of the IEEE*, vol. 100, no. 1, pp. 13-28, Jan. 2012.
- [60] E. A. Lee, "Cyber Physical Systems: Design Challenges," Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on, Orlando, FL, 2008, pp. 363-369.
- [61] O. Maler, Z. Manna, and A. Pnueli, "From timed to hybrid systems," in REX workshop Real-Time: Theory in Practice, ser. Lecture Notes in Computer Science, 1992, pp. 447–48.
[62] John Lygeros, Claire Tomlin, and Shankar Sastry, "Hybrid Systems: Modeling, Analysis and Control," December 28,2008. Available: http://wwwinst.cs.berkeley.edu/~ee291e/sp09/handouts/book.pdf

- [63] S. Rezaei and R. Sengupta, "Kalman Filter-Based Integration of DGPS and Vehicle Sensors for Localization," in *IEEE Transactions on Control Systems Technology*, vol. 15, no. 6, pp. 1080-1088, Nov. 2007.
- [64] http://www.nsf.gov/awardsearch/showAward?AWD_ID=1453125&HistoricalAwards=fa lse
- [65] T. Tielert, D. Jiang, Q. Chen, L. Delgrossi and H. Hartenstein, "Design methodology and evaluation of rate adaptation based congestion control for Vehicle Safety Communications," *Vehicular Networking Conference (VNC)*, 2011 IEEE, Amsterdam, 2011, pp. 116-123.
- [66] Kiefer, R., Cassar, M. T., Flannagan, C. A., LeBlanc, D. J., Palmer, M. D., Deering, R. K., & Shulman, M. A. 2003. "Forward Collision Warning Requirements Project: Refining the CAMP Crash Alert Timing Approach by Examining 'Last-Second' Braking and Lane Change Maneuvers Under Various Kinematic Conditions". Report No. DOT-HS-809-574. Washington, DC: National Highway Traffic Safety Administration.
- [67] Kiefer, R.J., LeBlanc, D., Palmer, M., Salinger, J., Deering, R., & Shulman, M. 1999. "Forward Collision Warning Systems: Development and Validation of Functional Definitions and Evaluation Procedures for Collision Warning/Avoidance Systems." Report No. DOT-HS808-964. Washington, DC: National Highway Traffic Safety Administration.
- [68] Moradi-Pari, E., Gani, S., Fallah, Y., Naserian, M. et al., "Co-Simulation of Cooperative Vehicle Safety Applications and Communication Networks," SAE Int. J. Passeng. Cars – Electron. Electr. Syst. 8(2):344-349, 2015, doi:10.4271/2015-01-0285.
- [69] Novakowski C, Shladover S, Bu F, O'Connell J, Spring J, Dickey S, Nelson D. (2010) Cooperative adaptive cruise control: testing drivers' choices of following distances. California PATH, Richmond
- [70] F. Viti, S. P. Hoogendoorn, T. P. Alkim and G. Bootsma, "Driving behavior interaction with ACC: results from a Field Operational Test in the Netherlands," *Intelligent Vehicles Symposium, 2008 IEEE*, Eindhoven, 2008, pp. 745-750.
- [71] B. van Arem, C. J. G. van Driel and R. Visser, "The Impact of Cooperative Adaptive Cruise Control on Traffic-Flow Characteristics," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 4, pp. 429-436, Dec. 2006.

- [72] VanderWerf, J., Shladover, S.E., Miller, M., Kourjanskaia, N., 2002. Effects of adaptive cruise control systems on highway traffic flow capacity. Transp. Res. Rec.: J. Transp. Res. Board 1800, 78–84.
- [73] V. Milanes, S. E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe and M. Nakamura,
 "Cooperative Adaptive Cruise Control in Real Traffic Situations," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, pp. 296-305, Feb. 2014.
- [74] Milanés, V. and Shladover, S.E. "Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data." *Transportation Research Part C: Emerging Technologies*, 48(6), 285–300, 2014.
- [75] G. Guo and W. Yue, "Sampled-Data Cooperative Adaptive Cruise Control of Vehicles With Sensor Failures," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 6, pp. 2404-2418, Dec. 2014.
- [76] M. Amoozadeh et al., "Platoon Management with Cooperative Adaptive Cruise Control Enabled by VANET," *Vehic. Commun* Volume 2, Issue 2, pp 110–123, April 2015.
- [77] G. J. L. Naus, R. P. A. Vugts, J. Ploeg, M. J. G. van de Molengraft and M. Steinbuch,
 "String-Stable CACC Design and Experimental Validation: A Frequency-Domain Approach,"
 in *IEEE Transactions on Vehicular Technology*, vol. 59, no. 9, pp. 4268-4279, Nov. 2010.
- [78] S. Oncu, N. van de Wouw, W. P. M. H. Heemels and H. Nijmeijer, "String stability of interconnected vehicles under communication constraints," *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, Maui, HI, 2012, pp. 2459-2464.
- [79] S. M. Loos, A. Platzer, and L. Nistor. Adaptive cruise control: Hybrid, distributed, and now formally verified. In FM 2011: Formal Methods, 17th *International Symposium on Formal Methods*, LNCS 6664, pages 42–56. Springer, 2011.
- [80] J. Hespanha. Modeling and Analysis of Networked Control Systems using Stochastic Hybrid Systems. *IFAC Annual Reviews in Control*, 38(2):155—170, Oct. 2014.
- [81] R. Verma and D. Del Vecchio, "Safety Control of Hidden Mode Hybrid Systems," in *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 62-77, Jan. 2012.
- [82] S. Sekizawa *et al.*, "Modeling and Recognition of Driving Behavior Based on Stochastic Switched ARX Model," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 4, pp. 593-606, Dec. 2007.
- [83] J. H. Kim, S. Hayakawa, T. Suzuki, K. Hayashi, S. Okuma, N. Tsuchida, M. Shimizu, and S. Kido, "Modeling of driver's collision avoidance maneuver based on controller switching model," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 6, pp. 1131-1143, Dec. 2005.

- [84] Okuda, H.; Ikami, N.; Suzuki, T.; Tazaki, Y.; Takeda, K., "Modeling and Analysis of Driving Behavior Based on a Probability-Weighted ARX Model," in *Intelligent Transportation Systems, IEEE Transactions* on, vol.14, no.1, pp.98-112, March 2013.
- [85] A. Pentland and A. Liu, "Modeling and prediction of human behavior," *Neural Comput.*, vol. 11, no. 1, pp. 229–42, Jan. 1999. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/9950731.
- [86] V. Gadepally, A. Krishnamurthy and U. Ozguner, "A Framework for Estimating Driver Decisions Near Intersections," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 2, pp. 637-646, April 2014.
- [87] G. S. Aoude, V. R. Desaraju, L. H. Stephens and J. P. How, "Driver Behavior Classification at Intersections and Validation on Large Naturalistic Data Set," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 724-736, June 2012.
- [88] J. Hu, J. Lygeros, and S. Sastry, "Towards a theory of stochastic hybrid systems," in Hybrid Systems: Computation and Control, LNCS 1790. Heidelberg, Germany: Springer-Verlag, 2000, pp. 160–173.
- [89] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," in *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-286, Feb 1989.
- [90] L. E. Baum and J. A. Egon, "An inequality with applications to statistical estimation for probabilistic functions of a Markov process and to a model for ecology," *Bull. Amer. Meteorol. Soc.*, vol. 73, pp. 360-363, 1967.
- [91] L. E. Baum and G. R. Sell, "Growth functions for transformations on manifolds," *Pac. J. Math.*, vol. 27, no. 2, pp. 211-227, 1968.
- [92] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," in *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260-269, April 1967.
- [93] G. D. Forney, "The Viterbi algorithm," *Proc. IEEE*, vol. 61, pp. 268-278, Mar. 1973.
- [94] W. Khreich, E. Granger, A. Miri, and R. Sabourin. "A survey of techniques for incremental learning of HMM parameters," *Information Sciences*, 197, pp. 105-130, 2012.
- [95] E. Moradi-Pari, A. Tahmasbi-Sarvestani and Y. P. Fallah, "A Hybrid Systems Approach to Modeling Real-Time Situation-Awareness Component of Networked Crash Avoidance Systems," in *IEEE Systems Journal*, vol. 10, no. 1, pp. 169-178, March 2016.
- [96] Y. Fallah, "A Model-Based Communication Approach for Distributed and Connected Vehicle Safety Systems", In Proc. of IEEE Systems Conference, 2016

- [97] K. Lee, H. Peng ,2005 "Evaluation of automotive forward collision warning and collision avoidance algorithms", *Vehicle System Dynamics*, Vol. 43, Iss. 10, 2005.
- [98] Knipling, R., Mironer, M., Hendricks, D., Tijerina, L., Everson, J., Allen, J., & Wilson, C.
 1993. "Assessment of IVHS countermeasures for collision avoidance: Rear-end crashes.", DOTHS-807-995. Washington, DC: National Highway Traffic Safety Administration.
- [99] Yaser P. Fallah, and Masoumeh K. Khandani. "Analysis of the coupling of communication network and safety application in cooperative collision warning systems." Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems. ACM, 2015.
- [100] E. Moradi-Pari, N. Nasiriani, Y. P. Fallah, P. Famouri, S. Bossart and K. Dodrill, "Design, Modeling, and Simulation of On-Demand Communication Mechanisms for Cyber-Physical Energy Systems," in *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2330-2339, Nov. 2014.
- [101] S. M. Iranmanesh, E. Moradi-Pari, Y. P. Fallah, S. Das, M. Rizwan," Robustness of Cooperative Forward Collision Warning Systems to Communication Uncertainty," In Proc. of IEEE Systems Conference, 2016.
- [102] <u>https://www.its-rde.net/data/showds?dataEnvironmentNumber=10018</u>.
- [103] L.A. Montestruque, P. Antsaklis, "Model-Based Networked Control Systems: Stability," ISIS Technical Report ISIS-2002-001, University of Notre Dame, www.nd.edu/~isis/, January 2002.
- [104] L.A. Montestruque and RJ. Antsaklis, "Networked Control Systems: A model-based approach," 12th *Mediterranean Conference on Control and Automation*, June 2004.
- [105] L.A. Montestruque and P.J. Antsaklis, "On the model-based control of networked systems," *Automatica*, Vol. 39, pp 1837-1843.

APPENDIX A

In this appendix we show the details of deriving HMM parameters for the case study. New estimations for initial state probabilities at each iteration is defined as follows:

$$\pi_{i}^{\prime t} = \gamma_{1}^{t}(i) = \frac{\alpha_{1}^{t}(i)\,\beta_{1}^{t}(i)}{P(O_{1:t} \mid \lambda^{t})} = \frac{\alpha_{1}^{t}(i)\,\beta_{1}^{t}(i)}{\sum_{i=1}^{N} \alpha_{1}^{t}(i)\,\beta_{1}^{t}(i)}$$
(4-31)

From the case study marginal conditions we have $\alpha_1^{\prime}(i) = \pi_i^{\prime} b_i^{\prime}(o_1)$, thus:

$$\pi_{i}^{\prime t} = \frac{\pi_{i}^{t} b_{i}^{t}(o_{1}) \beta_{1}^{t}(i)}{\sum_{i=1}^{N} \pi_{i}^{t} b_{i}^{t}(O_{1:t}) \beta_{1}^{t}(i)}$$
(4-32)

State transition probabilities are calculated by:

$$a_{ij}^{\prime t} = \frac{\sum_{d=1}^{t-1} \xi_d(i,j)}{\sum_{d=1}^{t-1} \gamma_d(i)}$$
(4-33)

If we rewrite the above based on $\gamma_d(i)$ and $\xi_d(i,j)$ we have:

$$a_{ij}^{\prime\prime} = \frac{\sum_{d=1}^{t-1} \frac{\alpha_d^{\prime}(i) \, a_{ij}^{\prime} \, b_j^{\prime}(o_{d+1}) \beta_{d+1}^{\prime}(j)}{P(O_{L_{i}} \mid \lambda')}}{\sum_{d=1}^{t-1} \frac{\alpha_d^{\prime}(i) \, \beta_d^{\prime}(j)}{P(O_{L_{i}} \mid \lambda')}}{\frac{1}{P(O_{L_{i}} \mid \lambda')} \sum_{d=1}^{t-1} \alpha_d^{\prime}(i) \, a_{ij}^{\prime} \, b_j^{\prime}(o_{d+1}) \beta_{d+1}^{\prime}(j)}}{\frac{1}{P(O_{L_{i}} \mid \lambda')} \sum_{d=1}^{t-1} \alpha_d^{\prime}(i) \, \beta_d^{\prime}(j)}}$$

$$= \frac{\sum_{d=1}^{t-1} \alpha_d^{\prime}(i) \, a_{ij}^{\prime} \, b_j^{\prime}(o_{d+1}) \beta_{d+1}^{\prime}(j)}{\sum_{d=1}^{t-1} \alpha_d^{\prime}(i) \, \beta_d^{\prime}(j)}}$$
(4-34)

Estimation of the next iteration for unbiased variance based on MLE is derived as follows:

$$\sigma'^{2_{i}^{t}} = \frac{\sum_{d=1}^{t} |o_{d}''|^{2} P\{q_{d} = s_{i}, O_{1:t} | \lambda'\}}{\sum_{d=1}^{t} P\{q_{d} = s_{i}, O_{1:t} | \lambda'\} - 1}$$

$$= \frac{\sum_{d=1}^{t} |o_{d}''|^{2} \alpha_{d}^{t}(i)\beta_{d}^{t}(i)}{\sum_{d=1}^{t} \alpha_{d}^{t}(i)\beta_{d}^{t}(i) - 1}$$
(4-35)

where, $|o''_{d}|^{2} = |\theta''_{i} \phi'_{i} - y_{d}|^{2}$.