# On the Impact of Communication and Information Access on Wind-Farm Control and Demand Management in Smart Grid Scenarios

Madsen, Jacob Theilgaard

*DOI (link to publication from Publisher):*
10.5278/vbn.phd.engsci.00150

*Publication date:*
2016

*Document Version*
Publisher's PDF, also known as Version of record

# ON THE IMPACT OF COMMUNICATION AND INFORMATION ACCESS ON WIND-FARM CONTROL AND DEMAND MANAGEMENT IN SMART GRID SCENARIOS

BY
JACOB THEILGAARD MADSEN

DISSERTATION SUBMITTED 2016

AALBORG UNIVERSITY
DENMARK

# On the Impact of Communication and Information Access on Wind-Farm Control and Demand Management in Smart Grid Scenarios

Ph.D. Dissertation

Jacob Theilgaard Madsen

Aalborg University
Department of Electronic Systems
Fredrik Bajers Vej 7A
DK-9220 Aalborg

Dissertation submitted: October, 2016

PhD supervisor: Professor Hans-Peter Schwefel
Aalborg University, Denmark

Assistant PhD supervisor: Associate Professor Tatiana Kozlova Madsen
Aalborg University, Denmark

PhD committee: Professor Rafal Wisniewski (chairman)
Department of Electronic Systems
Aalborg University, Denmark

Professor Dr. Hartmut Schmeck
Institute AIFB
Karlruhe Institute of Technology, Germany

Professor Thomas Skjødeberg Toftegaard
Department of Engineering
Aarhus University, Denmark

PhD Series: Faculty of Engineering and Science, Aalborg University

ISSN (online): 2246-1248
ISBN (online): 978-87-7112-824-6

Published by:
Aalborg University Press
Skjernvej 4A, 2nd floor
DK – 9220 Aalborg Ø
Phone: +45 99407140
aauf@forlag.aau.dk
forlag.aau.dk



Printed in Denmark by Rosendahls, 2016

# Abstract

The Smart Grid utilizes the communication infrastructure and the ability to have two-way communication between entities in the power grid, in order to improve the control in the power grid. The Smart Grid involves several different technical disciplines which must interact with each other in order to fulfill the vision of the Smart Grid. Targets of applications of the Smart Grid include to stabilize the frequency, ensure voltage quality, and balance the power consumption/production by utilizing information gathered from entities in the power grid. Many of these applications utilize information that is not locally available, and it is thus important to consider the impact of communication networks and data access procedures on such an application.

The thesis investigates three different Smart Grid applications, which are optimized for data access scheduling based on communication networks. The three scenarios are chosen as they cover different time horizons in terms of control loop duration, from less than a second to minutes. The first scenario is a wind farm scenario, where a central controller regulates geographically distributed wind turbines. The second scenario concerns demand management of supermarket temperature controlled entities. Finally the thesis considers the interplay between several Smart Grid controllers in order to provide insights on how these controllers would jointly affect the Smart Grid and how the latter is impacted by the different performance metrics of the remote information access.

In order to evaluate the impact of communication networks on controller performance a co-simulation framework has been created. This framework allows simulation of control and communication networks together in the same simulation. The thesis starts by showing how communication network and information access delay can cause decreases in performance of the controllers in both scenarios. Once the impact of communication networks on control simulations has been analyzed, information access and scheduling strategies have been designed to help alleviate the performance decrease of the controller experienced due to the communication network.

The last part of the thesis is a simulation of the interplay between several controllers in a Smart Grid scenario, specifically in a MV/LV grid. The simulation analyzes the joint impact of multiple smart grid controllers while also including a time-varying price. Finally, the impact of the communication network performance for the communication

of the price signal on the overall Smart Grid behavior is investigated for different types of market realizations.

# Resumé

Smart Grid udnytter kommunikationsinfrastrukturen og evnen til at have to-vejs kommunikation mellem enheder i elnettet, med henblik på at forbedre kontrollen i elnettet. Smart Grid involverer flere forskellige tekniske discipliner, der skal interagere med hinanden for at opfylde visionen om Smart Grid. Målet med anvendelsen af Smart Grid omfatter at stabilisere frekvensen, sikre spændings kvalitet, og balancere strømforbruget/produktionen ved at bruge oplysninger indsamlet fra enheder i elnettet. Mange af disse programmer udnytter oplysninger, der ikke er lokalt tilgængelige, og det er derfor vigtigt at overveje konsekvenserne af kommunikationsnet og dataadgang på en sådan ansøgning.
Afhandlingen undersøger tre forskellige Smart Grid applikationer, som er optimeret til dataadgang planlægning baseret på kommunikationsnetværk. De tre scenarier er valgt eftersom de dækker forskellige tidshorisonter med hensyn til kontrol periode varighed, fra mindre end et sekund til minutter. Det første scenarie er et vindmøllepark scenarie, hvor en central styreenhed regulerer geografisk distribuerede vindmøller. Det andet scenarie vedrører styring af efterspørgslen af supermarked temperatur kontrollerede virksomheder. Endeligt dækker afhandlingen samspillet mellem flere Smart Grid kontrolenheder for at give indsigt i, hvordan disse kontrolenheder vil i fællesskab påvirke Smart Grid og hvordan sidstnævnte er påvirket af de forskellige effektivitetsmålinger af dataadgang planlægningen.
For at vurdere virkningen af kommunikationsnet på kontrolenheders ydeevne er en co-simulerings struktur blevet skabt. Denne struktor giver mulighed for simulering af kontrol- og kommunikationsnetværk sammen i samme simulering. Afhandlingen starter med at vise, hvordan kommunikationsnetværk og information adgang forsinkelse kan forårsage fald i effektiviteten af regulatorer i begge scenarier. Når virkningen af kommunikationsnet på kontrol simuleringer er blevet analyseret, er information adgang og planlægning strategier blevet designet til at hjælpe med at lindre faldet i effektivitet kontrolenehden oplever på grund af kommunikationsnetværket ydeevne.
Den sidste del af afhandlingen er en simulering af samspillet mellem flere regulatorer i et Smart Grid scenarie, specifikt i et medium og lavt spændings nevieau af elnettet. Simuleringen analyserer de samlede virkninger af flere smart grid-controllere samtidig,

herunder en tidsvarierende pris. Endelig er virkningen af kommunikationsnetværket ydeevne for kommunikationen af prisen signal på det overordnede Smart Grid adfærd undersøgt for forskellige typer af markedets realiseringer.

# Contents

# Thesis Details

**Thesis Title:** On the Impact of Communication and Information Access on Wind-Farm Control and Demand Management in Smart Grid Scenarios
**Ph.D. Student:** Jacob Theilgaard Madsen
**Supervisors:** Prof. Hans-Peter Schwefel, Aalborg University
Assoc. Prof. Tatiana Kozlova Madsen, Aalborg University

The main body of this thesis consist of the following papers.

[A] Jacob Madsen, Mislav Findrik, Tatiana Madsen, and Hans-Peter Schwefel, "Optimizing Data Access for Wind Farm Control over Hierarchical Communication Networks," *International Journal of Distributed Sensor Networks*, Open access journal: http://dsn.sagepub.com/content/12/5/5936235.abstract, 2016.

[B] Rasmus L. Olsen, Jakob G. Rasmussen, Hans Peter Schwefel, Jacob Theilgaard Madsen, "On the use of Information Quality in Stochastic Networked Control Systems," *Submitted to the Elsevier Journal of Computer Networks*, 2016., 2016.

[C] Jacob T. Madsen, Tomasz Minko, Hans-Peter Schwefel, "Impact of Communication Network Performance on Supermarket Control," *Submitted to Conference on Networked Systems*, 2016.

[D] Jacob Theilgaard Madsen, Thomas le Fevre Kristensen, Rasmus L. Olsen Hans-Peter Schwefel, Luminita C. Totu, "Utilizing Network QoS for Dependability of Adaptive Smart Grid Control," *Energy Conference (ENERGYCON), 2014 IEEE International* pp. 859-866, 2014.

[E] Author, "Resilience of Urban Smart Grids Involving Multiple Control Loops," *Proceedings og IEEE Second International Smart Cities Conference (ISC2 2016)*, In press, 2016.

In addition to the main papers, the following publications have also been made.

[1] Jacob T. Madsen, Jose de Jesús Barradas-Berglind, Tatiana K. Madsen and Hans-Peter Schwefel, "On the Impact of information access delays on remote control of a wind turbine," *PowerTech, 2015 IEEE Eindhoven*, pp. 1-6, 2015.

[2] Jacob T. Madsen, Mislav Findrik, Domagoj Drenjanac and Hans-Peter Schwefel, "Investigating Wind Farm Control over Different Communication Network Technologies," *DA-CH Conference on Energy Informatics*, Springer International Publishing, no. X, pp. 129-140, 2015.

[3] Jacob T. Madsen, Mislav Findrik, Tatiana K. Madsen, Rasmus L. Olsen and Hans-Peter Schwefel, "Scheduling Data Collection for Remote Control of Wind Turbines," *Energycon 2016*, 2016.

This thesis has been submitted for assessment in partial fulfillment of the PhD degree. The thesis is based on the submitted or published scientific papers which are listed above. Parts of the papers are used directly or indirectly in the extended summary of the thesis. As part of the assessment, co-author statements have been made available to the assessment committee and are also available at the Faculty. The thesis is not in its present form acceptable for open publication but only in limited and closed circulation as copyright may not be ensured.

# Acknowledgements

There are many whom have contributed to ensuring that I finished my PhD, and i am thankfull to all of them. First and foremost I would like to thank my supervisors Hans-Peter Schwefel and Tatiana K. Madsen for ensuring the quality of my research, and for helping me finish the thesis in these last weeks and months of my PhD. I would especially like to thank Hans-Peter for being willing to spend part of his weekends proof reading my papers and thesis to ensure that they would be ready for the deadline a few days later. I am also thankfull for the discussions I have had with Tatiana who would provide a different, and some times much needed, perspective, espcially in the thesis writing.
I would also like to thank my family and friends, for being supportive of my work and understanding in the last months of it when I was almost always busy with writing. Their understanding and continued support is part of what ensured that I could finish my thesis this year.
Lastly I wish to thank my friends and co-workers at Aalborg University, who helped make the last three years more educational, interesting and enjoyable. This goes espcially to my room mates whom have ensured that there always is something fun to talk about when the Phd work got too serius or boring. A very special thanks goes to Dorthe Sparre who always had the answer, or could find it, when there was some practical information that was missing.

**EDGE Project acknowledgement** This work was partially supported by the Danish Council for Strategic Research (contract no. 11-116843) within in the 'Programme Sustainable Energy and Environment', under the "EDGE" (Efficient Distribution of Green Energy) research project.

Jacob Theilgaard Madsen
Aalborg University, October 12, 2016

# Part I

# Introduction

# Chapter 1

# Motivation and Problem Statement

## 1 Evolution of Energy Grids

The energy grid was originally designed with a few central power plants which supplied power to the entire grid. It was split into different voltage levels, to make the transporting of the energy cheaper by reducing loss in the power lines. This hierarchical structure of the energy grid allowed assumptions to be made regarding the distribution of power. The grid was designed to have the power flow away from the power plants, and transformers and power lines were designed with this in mind. Furthermore the design of the grid held assumptions regarding the type of consumption that was at the consumers and when this consumption was occurring. However, as time has progressed the nature and timing of consumption has changed, partly due to more entities using electricity in consumers homes, and this has led to the consumption being more unpredictable. In addition to this, the nature of the power production has also changed, becoming more distributed with fewer large power plants. Some consumers have started producing part of their power themselves, which can lead to problems for the power utility companies and the power grid in general. The grid units in the power grid are not designed for this distributed nature of production, and with a greater unpredictability of the power consumption, this has lead to the requirement of a redesign of the energy grid. This redesign is normally called the Smart Grid, and the vision is to make the units operating in the power grid smarter, allowing more communication between entities, and changing parts of the physical grid objects, thus bringing the energy grid into the 21'st century. There are many definitions of the Smart Grid, depending on the context of what each Smart Grid project focuses on. One such definition is from the European technology platform, an industry-led stakeholder forum, and says that the Smart Grid is: "electricity networks that intelligently integrate the behaviour and actions of all users connected to it - generators, consumers and those that do both - in order to efficiently deliver sustainable, economic and secure electricity supplies” [3]. This is one of many such definitions. One common theme of these definitions is that there is a need for a bi-directional communication network for the Smart Grid to work. In the definition above it is implicitly defined from the part that says "of all users connected to it".

The Smart Grid is expected to be an evolution of the current power grid, with more intelligent systems allowing for better regulation of electricity. The hope is to ensure reliability of the power supply, by regulating power balance, frequency control among other things. By having better regulation and control of all power grid entities it becomes easier to ensure stability of the power grid. This does require more intelligent controllers, both among entities in the power grid, as well as the controllers that regulate the entities at the distribution system operators or transmission system operators.

An important part in designing more intelligent controllers is ensuring that the controllers have access to the information they require. This is achieved by the communication networks. These networks must allow the passing of informations between all entities in the Smart Grid, to ensure that controllers have the information required to make correct decisions. While communication networks at co-located entities are not difficult to design, when the location of entities are spread over a wide area it can become significantly more difficult to design an optimal network.

When considering the communication network of the Smart Grid it should be decided whether to use the existing infrastructure or if a new infrastructure should be made. The main benefit of creating a new communication network infrastructure for the Smart Grid is that it will allow for private networks, without interference from users that do not use the Smart Grid. The main issue with creating a new communication network infrastructure is costs. And this issue is rather large, as the Smart Grid covers a large geographical area with many entities the cost of deploying a complete new infrastructure is very large. If the existing infrastructure is used, however, the communication network is shared with other users.

As the Smart Grid covers a wide area, both location wise as well as technology wise, the communication networks may vary greatly in terms of what communication speeds are achievable as well as how reliable the communication networks are. It may also mean that the communication network uses a combination of several different communication network technologies which requires interfaces between them to ensure communication from end to end is possible. A large communication network may also have dynamic quality of service, and as such the network quality of service parameters, such as delay, bandwidth, packet losses and jitter, may change over time.

This means that a static model for the communication networks is not realistic. A static model would have deterministic delays and information losses. These are significantly easier to take into consideration when designing controllers, and makes it easier to de-

termine information access strategies.
It is important to ensure that the controllers that are designed are still stable when a communication network is added. Increased delays in information may lead to the controller taking decisions based on old or incomplete information, which may lead to sub-optimal control, or in worst case scenarios unstable controllers. As such, when one considers communication networks in a controllers setup, one must determine the impact on the controllers. Furthermore, it should be considered that the information may also be lost, either leading to the controller not having the required information, or it may be that the actuators do not receive signals from the controller, leading to the system not reacting to external input. In worst case scenarios, in the Smart Grid, this may lead to black-outs in part of the power grid. These losses and delays can be caused by many different factors, e.g. there may be other entities utilizing the communication network.
Part of the delays will stem from what in this thesis is called information access strategies. These are strategies that determine when information is gathered, when it is sent, how entities react to receiving or sending messages (pass them along instantly, or buffer them for later), and even how long a unit should wait from receiving a signal to acting on it.

# 2 Smart Grid Scenario Utilizing Distributed Energy Resources

In this thesis the Smart Grid scenario that is considered is related to the EDGE (Efficient Distribution of Green Energy) project. The considered scenario is concerned with the usage of green energy (or sustainable energy) in the power grid, and to ensure that the penetration of green energy is increased. This is done by investigating how green energy sources can contribute to the power grid, and which problems they may cause. The scenario involves wind farms, supermarkets, electric vehicles (EV), photo voltaic (PV), combined heating & power plants (CHP), households, an electricity price market and a communication network. The MV grid contains a medium voltage grid controller (MVGC), which regulates the MV grid, and similarly the LV grid has a low voltage grid controller (LVGC). A general overview of this scenario can be seen in Figure 1.1.

There are several challenges with increasing the penetration of green energy. The main challenge is that the green energy is not always available when the energy is needed, i.e. the wind not blowing or the sun not shining. To ensure stability of the power grid there must either be energy reserves ready to produce energy when there is not enough energy in the renewable resources, or it must be possible to shift the consumption of energy for when the energy is available. Having energy reserves ready can be rather expensive, and as such another option may be less costly. One such option

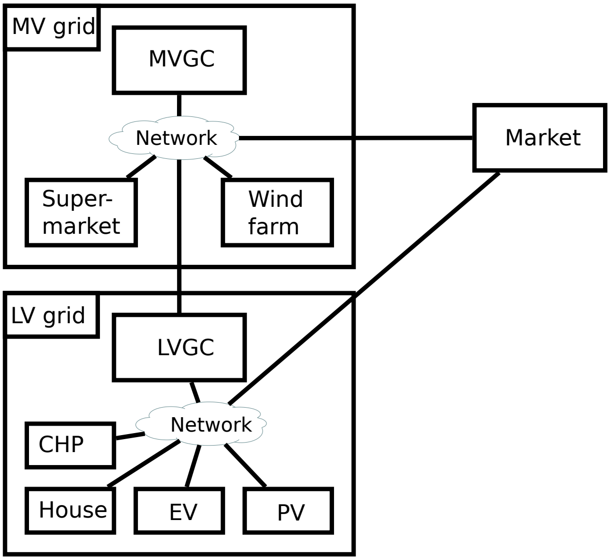



**Fig. 1.1:** Overview of EDGE scenario

is to use the thermostatic loads already present in the grid, which all have a small buffer of thermal energy that can be used to increase or decrease consumption. The idea is to shift the energy load. This is mainly used in peak reduction scenarios, where there is a predicted spike in energy consumption, and hours prior the thermostatic loads increase consumption, such that when the spike arrives it is reduced as the thermostatic loads require less energy.

Other challenges in increasing green energy are, for example, that green energy sources usually means more distributed power generation, which the current power grid is not designed to handle. In these distributed generation cases one may see power flowing from the LV grid to the MV grid or the high voltage (HV) grid, which can cause problems for grid units such as transform stations. Another challenge is that the current grid balancing algorithms are designed for a hierarchical structure, where the Smart Grid leans more to wards distributed power generation structures, at least within each part of the grid.

Solving these challenges require exchanges of information between many of the entities in the Smart Grid, whether it is to do peak reduction or make a distributed control strategy. This in turn means that there is a need for a communication network infrastructure, which allows the passing of the information needed. The EDGE project considers the control of a wind farm, a supermarket, a low voltage (LV) grid control and a medium voltage (MV) grid control, each placed in a LV or MV grid. In each of these control use-cases there is a need for communication between the entities. This communication is handled by a communication network. Furthermore there is a communication flow between each controller, which the communication network must also enable. A further part of the communication network as determined by the EDGE project, although not heavily considered in this thesis, is the monitoring of network traffic to make inferences on the QoS of the communication network. This information could then be used to improve the performance of the controllers in the Smart Grid.

# 3 General Controller Structure

Controller design often follows a specific structure, of the controller sending control signals to a plant, the plant reacting on the control signal and external disturbances, and sending sensor information back to the controller. If one assumes a perfect communication network this would allow the controller to always have the newest possible information, as there would be instantaneous transmission between the controller and plant.
When implementing an imperfect communication network to a control scenario which was previously assumed to have a perfect communication network there are two impacts on the information sent. There is a chance for the information to be lost, via packet losses. This will cause a controller to try to make decisions on information it does not have. If the controller is not designed to handle such a situation, it cannot act. The second impact is that information can be delayed. This can have two impacts, depending on the length of the delay and the information access strategy. One is that the information is delayed past the controllers decision horizon, $C_i$ in Figure 1.2, and thus it is the same situation as for lost information. The second option is that the information arrives in time, however, it is now delayed by $T_{Offset}$ as shown in Figure 1.2. $T_{Offset}$ is a design parameter introduced in order to allow for network delays. The impact of this delay depends on how fast the information sent changes compared to the length of the delays. If the information changes quickly, then the controller may often make decisions based on incorrect information.

When designing an information access strategy there are several choices that should be made. These choices includes design parameters such as $T_{Offset}$, how often plant information should be sent to the controller, how often the controller should send control signals to the plant, and how many times the same information would be sent. To design such an information access strategy can require a lot of information regarding the system considered. This information access strategy can be dependent on the specific controller used. A perfect information access strategy would always know the exact delay of the communication network, and could schedule $T_{Offset}$ to be exactly the duration of the delay. It would further know if the information about to be sent is the same information that was sent previously, and as such it does not need to send

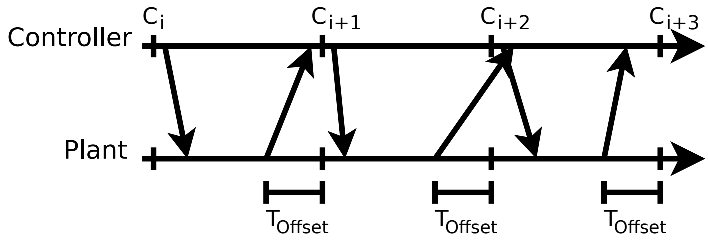



**Fig. 1.2:** Simple message sequence diagram of general purpose control scenario

the information. Creating such an access strategy would require perfect information regarding plant information dynamics, controller design and the communication network. Plant information dynamics covers how the plant acts on control signals and disturbances for the plant, e.g for a wind turbine this could be changes to the wind speed. When considering controller design information, it is in this thesis assumed to be timing requirements, computational delays and underlying algorithms for calculating the control signal. The communication network parameters includes information regarding cross traffic, physical medium disturbances, network stack protocols and much more. This information is not necessarily readily available, such as disturbances to the plant, and it is usually modelled instead. Thus one can only use generalised or modelled information regarding the plant information dynamics and the communication network. This is, for the communication network, among other things, packet loss probabilities, average delays, delay distributions and behaviour over time. For the plant information dynamics this covers how sensor information behaves over time, on which time scales and models of disturbances. The information access strategy design is usually influenced by controller design choices, i.e. the control period duration .
There is a need for a joint optimization involving the controller, the information access strategy and the communication network, as an optimization that only takes one of these factors into account would not be optimal over all factors. The goal of this joint optimization is to first find a communication network scenario which is feasible and realistic to the control scenario. The next goal is to then find optimal parameters for the controller and network access strategy based on this communication network. This is a computationally complex problem which requires extensive analysis to determine the optimal parameters. Furthermore the performance metric of the controller may change from scenario to scenario which makes generalizations very difficult. This raises the question if there is some other metric which could be used to help determine the optimal parameters, which does not require the same extensive analysis and which may be generalized for other scenarios. To be independent of the controls scenario, such performance metrics should be on below the control application. However, such metric

could still put network performance into perspective to the system dynamics, i.e. relating time-scales of delays to time-scales of changes of sensor information. Information quality metrics can achieve this; one example of these is the so-called mismatch probability (mmPr), defined as the probability that the view on sensor information values at the controller at time of control execution is deviating from the actual sensor value at that moment in time. Reference [1] introduces and discusses such mismatch metric for different information access strategies, periodic, event-driven, and reactive

# 4 Problem Statement

The main question of this thesis is: **How do imperfect communication network performance and design choices in the information access strategies impact networked Smart Grid controllers?**
When investigating this two subproblems will be discussed. These subproblems are:

- How does information delay and loss due to imperfect communication network impact a controller.
- How to design an optimal information access strategy based on the known information regarding the controller design, plant information dynamics and the communication network behaviour.

To showcase these subproblems, three use-cases from a Smart Grid scenario are identified and a communication network and information access strategy is implemented in a control scenario. As a perfect information access strategy and a perfect communication network cannot be achieved in reality these will be what in this thesis is denoted imperfect, thus making the use-cases more realistic. The three use-cases are a wind farm control use-case, a load shifting using thermostatic loads use-case, and the overall Smart Grid scenario detailed in Section 2. In the wind farm use-case it will be discussed if a performance metric called mismatch probability is applicable to the scenario and if it can be used to simplify the problem of determining the optimal parameters. Once these two use-cases have been analyzed and discussed, it will be discussed how they would fit into a larger Smart Grid scenario involving multiple control loops, and discuss how imperfect network conditions may further impact design choices.

## Assumptions and Challenges

When implementing a communication network to control scenarios it can cause two things for the system; delay and information loss. These are added to the information sent between the controllers, or between the controller and sensors/actuators. This raises the problem on how to add a communication network in a control scenario. When

considering delays in this thesis it will concern delays caused by queuing, transmission, adding/removing headers, converting between protocols. Bandwidth is not considered important in this thesis, as the information being sent is not large files but rather small packages, and in the thesis it is found that timely arrivals are more important. There may be specific scenarios where it is worthwhile to consider a private communication network infrastructure, but in general it is assumed in this thesis that the Smart Grid will use already existing communication network infrastructures. As static models are not considered realistic enough for the given scenarios this thesis considers more realistic models where the delays and information losses are probabilistic. This does, however, also make determining an optimal information access strategy more difficult.
There are aspects of these problems that this thesis does not cover, due to time constraints and complexity issues. It is chosen due to the complexity of the used controllers, to only do adaptation of the actual controller based on communication network performance for a single controller, as can be seen in Paper D. The other scenarios and use-cases considered in this thesis had controllers with more complex control signals, making direct modification to the control signal impractical. Furthermore, to do modifications of the inner controller workings were deemed too complex a problem without having the controller designed present.

# 5 Use-case Descriptions

In this thesis two main use-case are considered. These use-cases are chosen as they showcase different challenges in the overall Smart Grid scenario. The first use-case is a wind farm control scenario which operates on short time scales which gives interesting dynamics with delays. The second is a demand response scenario involving temperature controlled assets which demonstrates how a controller working on slower time scales is impacted by imperfect communication networks.

## Wind farm control

The penetration of wind energy is steadily increasing, especially in Denmark, and thus a larger portion of the energy production comes from wind farms. When wind farms become such large factors in modern power production it is important to ensure that the wind turbines produce the energy that is expected of them, and to ensure that the wind turbines have a high operation life-time. The controller that is considered in this thesis reduces the damage a wind turbine would sustains during normal operation, while following a power reference.
Controlling a wind farm can be done from a central controller which harmonizes the wind farm's power generation by sending set-points to each wind turbine, thus a control is performed via a communication network. This way a central controller ensures that

each wind turbine produces the required amount of power. For deployment of the wind farm communication network it is possible to use different technologies, however, each technology has different behaviour, hence it is important to analyse the impact of these communication technologies on the controller's performance.

## Load shifting and peak reduction using thermostatic loads

Utilizing thermostatic loads in a Smart Grid scenario can greatly help performing peak reduction in power consumption. For normal grid operation there are typically two spikes in power consumption during a single day. However, it is not always the case that it is during these times of day that the renewable energy sources are at their peak production. As such, by reducing the consumption during these peaks, it is possible to increase the usage of renewable energy.

Intelligent control of flexible loads in the power grid can help stabilize the power grid through load shifting. This load shifting can benefit users economically, as it would be possible to have a lower consumption when the electricity price is high, with a higher consumption when the price is low. One example of how this can be achieved is through temperature controlled entities, e.g. freezers, refrigerators, AC units. Usually when considering temperature controlled entities, the precise temperature is not important, as long as it is kept within a temperature band. By regulating the temperature in this band it is possible to shift the load of the entity, by increasing consumption for a while, and then having it work less later, thus reducing consumption. In order to accomplish this, it is important to know when there are peaks in power production, or alternatively to know when prices are high or low.

In this use-case two different scenarios are considered. One scenario is a supermarket controller, which utilizes a supervisory controller to regulate the temperature in freezers and refrigerators in the super market, based on sensor readings of the temperature and based on information on time-varying energy prices. This controller aims to reduce the cost of operation for the supermarket by reducing the power consumption in high price periods. The other scenario considers a controller which attempts to perform peak reductions by regulating the on and off state of a number of refrigerators in a LV grid. This controller, contrary to the previous controller, does not take the cost of operation into account, but instead attempts to reduce peak consumption regardless of price. The controller uses a prediction of power consumption, and a power profile generated which should be followed. It attempts to reduce consumption in two different peak consumption periods during a single day, by increasing power consumption prior to the peaks.

Both controllers uses thermostatic loads, and are impacted by imperfect communication networks which causes loss or delays of information.

## 6 Overview of Contributions

The rest of the thesis is structured as follows. Chapter 2 will cover state of the art and background information regarding the use-cases considered. Chapter 3 will cover the evaluation methodology, which will discuss the co-simulation framework that is used in the thesis, and an overview of performance metrics. The main result of this chapter is the co-simulation framework which can be used to investigate the impact of imperfect communication networks and information access strategies on controller performance.

Chapter 4 discusses the wind farm scenario, describing the overall scenario, how to evaluate the performance of the controller, and how to optimize the information access strategy. This chapter show insights into how the wind farm controller is affected by an imperfect communication network and information access strategy. This chapter also shows an evaluation approach one can use to evaluate the impact of imperfect communication networks by detailing which parameters are important to consider during such an investigation. Based on results from a complimentary thesis [2], regarding the modelling of sensor data using Markov chains to determine mmPr, it is shown how to use mmPr to optimize the controllers information access strategy. Lastly in this chapter the validity of mmPr as a control performance metric is investigated, and it is found that it can be used to optimize controller performance in the given control scenario. The chapter is based on Paper A and Paper B.

Chapter 5 is based on Paper C and D, and details a scenario concerning load shifting of thermostatic loads. This chapter also gives insights into the effect of imperfect communication networks on the controller, and detail an evaluation approach for a thermostatic load scenario. Furthermore, the chapter shows how the controller used in one of the scenarios can be modified to incorporate communication network parameters to improve performance in packet loss scenarios.

The last analysis chapter is Chapter 6 which is based on Paper E, and here the interaction of multiple Smart Grid controller is studied under the effects of an imperfect communication network. This chapter details the evaluation of the Smart Grid scenario detailed in Section 2. This scenario is more complex than the two scenarios detailed in the previous chapters.

Finally the thesis is concluded in Chapter 7, which gives a summary and an outlook on future work. These chapters conclude Part I of the thesis. Part II consists of the five papers, Paper A to E, which the thesis is based on.

# References

[1] M. Bogsted, R. L. Olsen, and H.-P. Schwefel, “Probabilistic models for access strategies to dynamic information elements,” *Performance Evaluation*, vol. 67, no. 1, pp. 43 – 60, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166531609001175

[2] M. Findrik, “Network-aware adaptation framework and its application to smart grid controllers,” Ph.D. dissertation, Aalborg University Doctoral School, 2016, phD thesis.

[3] E. technology platform, “European technology platform for electricity networks of the future,” 2013, http://www.smartgrids.eu.

# Chapter 2

# Background

## 1 Networked Control Systems

Networked control systems (NCS) concern situations where controllers may experience delays or losses of information from sensors. Usually when investigating NCSs it is mainly delay from the sensors to the controller that is considered, as explained in Reference [1]. In this discipline the controller is designed under the assumption that the sensor information that it needs is delayed. The controller will then use a predictor, such as a smith predictor or a Kalman filter, to determine what the information would have been had it not been delayed. This method requires a model of the system state that the controller uses, and a caching of previous system states to make the prediction as precise as possible. However, this prediction also requires that the controller knows what the information delay is, which would require network monitoring or at least use of time-stamps and synchronized clocks.
Another issue for the NCS discipline is when the information is lost instead of delayed, or if the information is delayed past the control computation deadline. In this case the predictor does not have the newest information to make the prediction from, which would require it to make the prediction from the previous received message. This does inherently make the prediction less precise. This problem can become more severe in high packet loss scenarios, where the information may be lost for several control cycles. This means that the controller must take a decision, based on predictions on old information. This leads to greater uncertainty on the precision of the predictors.
NCS covers a great variety of applications. This thesis considers NCS for a wind farm scenario, a thermostatic load scenario, and a more complex Smart Grid scenario, but there are many other scenarios where a similar analysis and discussion would be beneficial to the field. These areas include swarm robots [2], or distributed controllers [3].
The primary concern for NCSs is the stability of the system. Stability in this case means

that a given controller can ensure that the system being controlled does not come into zones which the components are not designed for. An example could be for a wind turbine, there are certain maximum speeds and position which the blades can handle, and the controller should not force the system into such a position. For more information regarding ensuring that a NCS is stable consider [4] or [5]. [6] discusses, among other points, the issue of stability for a wind turbine control system.
Once stability of the controller in a NCS is assured, the next step becomes ensuring the performance of the controller. This is generally done under the assumption of specific communication network conditions, and the controller will then be optimised based on these conditions. [7] looks at performance of controllers in a NCS scenario from the perspective of the controller. In this thesis, we focus instead on a controller designed for perfect communication networks and information access strategy, and then a communication network is added. The analysis is from the perspective of the communication network rather than the controller, and as such we look first at the stability and impact on performance of the controller when adding a imperfect communication network.

# 2 Control of Wind Farm

Part of the energy production is currently supplied by wind farms, and there are several plans to increase the percentage of power production coming from wind. This, however, comes with some inherent problems, some of which are that power production from wind is not easy to predict, thus making it difficult to plan the production, making the grid more susceptible to power fluctuations. Work done within the field of control of wind farms and turbines, lies in predicting the power production of a wind farm. This is described in [8], where the authors attempt to predict the production of a wind farm up to 36 hours ahead. This is done based on forecasts of the Danish Meteorological Institute.
Other work within this field lies in ensuring the an entire wind farm is able to follow a power reference signal. This is investigated in [9], which uses a central controller to send set-points to specific wind turbines. The wind turbines then have a local controller which is used to ensure that the set-point received is achieved at the wind turbine. This requires an investigation into the impact of aerodynamics interaction between the wind turbines, which is discussed in [10].
Another problem is the maintenance and repairing of wind turbines. Reducing the damage to wind turbines will reduce the down time, and thus lead to an increased production in power overall. Some work in this respect is done in [6]. For this thesis this problem is the one that is analysed further for investigating the impact of a communication network on the controller.
This thesis applies a communication network and an information access strategy to a wind farm scenario. There was little complimentary work found in this field in terms of the impact of communication networks on performance. [11] analyses a ethernet commu-

nication network in order to determine upper limits for communication delays. However, they do not consider the delays impact on controller performance. A different approach is to use ZigBee, such as in [12], where the focus is on the ability to monitor a wind turbine, and an entire wind farm, remotely.

# 3 Load Shifting Using Thermostatic Loads

Part of demand side management in Smart Grids concerns load shifting [13]. The concept of load shifting is to move the consumption of entities in the Smart Grid in time by changing when the entities have high and low periods of consumption. By changing the timing of consumption for certain entities in the grid it is possible to reduce the stress on the power grid in terms of consumption, by performing peak reductions. An example is in [14] where it is assumed that there are two periods during a day when there is a drastic increase in power consumption in LV grids. By changing the timing of consumption of some of the entities in the grid these peaks can be reduced.

Load shifting generally concerns thermostatic loads as these are readily available at consumers already. Another target for load shifting could be batteries of EVs, as described in [15], however this thesis is mainly concerned with thermostatic loads as thermostatic loads are already present in abundance in the LV and MV grid, while EVs and batteries are still being deployed. This thesis considers two scenarios that uses thermostatic loads for load shifting.

One scenario is a supermarket controller. [16] analyses the challenges of modelling the actual temperature of refrigerated goods, and analyses how different foodstuffs can be used in load shifting. The scenario considered in this thesis is a controller which uses price signals to perform load shifting with the goal of reducing cost of supermarket operation [17] and [18]. The focus of the thesis is to add a communication network and information access strategy to this scenario and investigate the impact on controller performance.

Another target of demand side management is control of thermostatic loads in the LV grid. As opposed to the supermarket scenario, where it is a few rather large thermostatic loads being controlled, this scenario is concerned with multiple relatively small thermostatic loads. The scenario considered in this thesis utilizes thermostatic loads in a LV grid control scenario where the controller uses the fridges present in the LV grid to reduce peaks in power consumption [14]. Reference [19] introduces a similar scenario with thermostatically controlled loads, however it is a more general setup and uses more feedback from the regulated entities, and it does not consider the impact of an imperfect communication network.

# 4 Access Strategies in Control Systems

Generally speaking there are three different types of access strategies that are considered when talking about information access. The first is *Reactive*, which means that the controller will poll the plant whenever it needs new information, and the plant will respond. The next is *Proactive - Event driven* where information is t whenever there is a change in the system. The last is *Proactive - Periodic* where the plant sends updates periodically to the controller. Each of these have their pros and cons. The reactive case can cause longer delays, as a message first has to get to the plant before the plant responds, however there is only traffic whenever the controller needs it. The *Proactive - Event driven* strategy can cause the communication network to be flooded with information if a lot of events occur withing a short time span. It is, however, the strategy with the best chance of the controller having the newest information when it needs it. The *Proactive - Periodic* is a mix of the two strategies in pros and cons. There is no risk of flooding the network, as the plant only sends one message per period, however there is a risk if the information is delayed too much that the controller does not have it when it needs it.
Figure 2.1 shows a simple *Proactive - Periodic* information access strategy. In this strategy the controller will send control information periodically to the plant, and the plant will send information back periodically. The time when the plant information is needed by the controller is denoted as $C_i$, and can be seen as the controller computation deadline. This is the time when the controller would start the computation of new control information. The time denoted $T_{Offset}$, denotes the time prior to $C_i$ that the information is sent from the plant. In this given scenario it is assumed that the controller and plant run on the same period length, although this may not always be the case. As can be seen on the figure, the information from the plant may be delayed past the computation deadline, thus requiring the controller to either not compute or to use older information if this is stored.

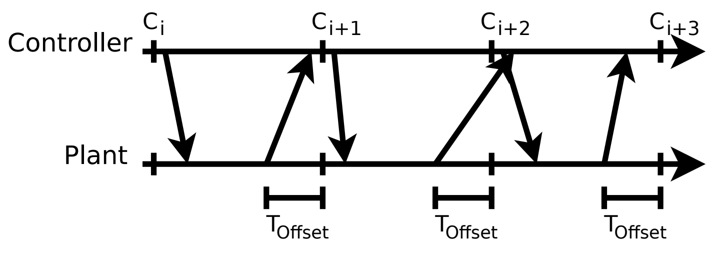



**Fig. 2.1:** Simple message sequence diagram of general purpose control scenario

# 5 Mismatch Probability as Quality Metric

MmPr, as defined in [20], is the probability that the value of the information used at the controller does not match the value of the information at the plant. It is also noted that the consequence of a mismatch is dependent on the control application considered. MmPr is used to describe the probability of correctness of information at the controller. There have been other investigations into mmPr and how to use it in specific scenario contexts. Reference [21] models the mmPr based on the communication network. It was used in the context of accessing location information based on the communication network in [22] and [23]. In this thesis we investigate if this metric can be used to optimize the information access strategy in the wind farm use-case.
As the consequences of mmPr changes depending on the specific scenarios considered the use of mmPr also varies based on the scenarios. In Reference [23], the metric was used to find optimal location-based relay policies in mobile networks, while in [24] it was used to optimize the information access strategy based on mmPr. MmPr was used in context subscription management systems for effective configuration of context access strategies in order to maximize the reliability of context information in [25]. In this thesis mmPr is used to optimize the information access strategy for the wind farm use-case.

# References

[1] Fei-Yue Wang and Derong Liu. *Networked control systems*. Springer, 2008.

[2] B. Anthony Kadrovach and Gary B. Lamont. A particle swarm model for swarm-based networked sensor systems. In *Proceedings of the 2002 ACM Symposium on Applied Computing*, SAC '02, pages 918–924, New York, NY, USA, 2002. ACM.

[3] X. Wang and M. D. Lemmon. Event-triggering in distributed networked control systems. *IEEE Transactions on Automatic Control*, 56(3):586–601, March 2011.

[4] Wei Zhang, Michael S. Branicky, and Stephen M. Phillips. Stability of networked control systems. *Control Systems, IEEE*, 21(1):84–99, 2001.

[5] Gregory C. Walsh, Hong Ye, and Linda G. Bushnell. Stability analysis of networked control systems. *Control Systems Technology, IEEE Transactions on*, 10(3):438–446, 2002.

[6] Jose de Jesus Barradas Berglind. *Fatigue-Damage Estimation and Control for Wind Turbines*. PhD thesis, Aalborg University Doctoral School, 2015. PhD thesis.

[7] W.P.M.H. Heemels, Andrew R. Teel, Nathan van de Wouw, and Dragan Nesic. Networked control systems with communication constraints: Tradeoffs between transmission intervals, delays and performance. *Automatic Control, IEEE Transactions on*, 55(8):1781–1796, 2010.

[8] L. Landberg. Short-term prediction of the power production from wind farms. *Journal of Wind Engineering and Industrial Aerodynamics*, 80(1–2):207 – 220, 1999.

[9] Anca D. Hansen, Poul Sørensen, Florin Iov, and Frede Blaabjerg. Centralised power control of wind farm with doubly fed induction generators. *Renewable Energy*, 31(7):935 – 951, 2006.

[10] K. E. Johnson and N. Thomas. Wind farm control: Addressing the aerodynamic interaction among wind turbines. In *2009 American Control Conference*, pages 2104–2109, June 2009.

[11] M. A. Ahmed, W. H. Yang, and Y. C. Kim. Simulation study of communication network for wind power farm. In *ICTC 2011*, pages 706–709, Sept 2011.

[12] Mohamed A Ahmed and Young-Chon Kim. Hybrid communication network architectures for monitoring large-scale wind turbine. *J. Electr. Eng. Technol*, 8(6):1626–1636, 2013.

[13] M Alparslan Zehir and Mustafa Bagriyanik. Demand side management by controlling refrigerators and its effects on consumers. *Energy Conversion and Management*, 64:238–244, 2012.

[14] Luminita C Totu, John Leth, and Rafael Wisniewski. Control for large scale demand response of thermostatic loads. In *2013 American Control Conference*, pages 5023–5028. IEEE, 2013.

[15] MA López, S De La Torre, S Martín, and JA Aguado. Demand-side management in smart grid operation considering electric vehicles load shifting and vehicle-to-grid support. *International Journal of Electrical Power & Energy Systems*, 64:689–698, 2015.

[16] Tobias Gybel Hovgaard, Lars FS Larsen, Morten J Skovrup, and John Bagterp Jørgensen. Analyzing control challenges for thermal energy storage in foodstuffs. In *Control Applications (CCA), 2012 IEEE International Conference on*, pages 956–961. IEEE, 2012.

[17] S. Ehsan Shafiei, Henrik Rasmussen, and Jakob Stoustrup. Modeling supermarket refrigeration systems for demand-side management. *Energies*, 6(2):900, 2013.

[18] Seyed Ehsan Shafiei, Jakob Stoustrup, and Henrik Rasmussen. A supervisory control approach in economic mpc design for refrigeration systems. In *European Control Conference*, pages 1565–1570, 2013.

[19] S. H. Tindemans, V. Trovato, and G. Strbac. Decentralized control of thermostatic loads for flexible demand response. *IEEE Transactions on Control Systems Technology*, 23(5):1685–1700, Sept 2015.

[20] Martin Bogsted, Rasmus L. Olsen, and Hans-Peter Schwefel. Probabilistic models for access strategies to dynamic information elements. *Performance Evaluation*, 67(1):43 – 60, 2010.

[21] M. Findrik, J. Groenbaek, and R.L. Olsen. Scheduling data access in smart grid networks utilizing context information. *Smart Grid Communications (SmartGridComm)*, pages 302–307, Nov 2014.

[22] J.J. Nielsen, R.L. Olsen, T.K. Madsen, and H. Schwefel. On the impact of information delay on location-based relaying: A markov modeling approach. In *Wireless Communications and Networking Conference (WCNC), 2012 IEEE*, pages 3045–3050, April 2012.

[23] Jimmy Jessen Nielsen, Rasmus Løvenstein Olsen, Tatiana K Madsen, Bernard Uguen, and Hans-Peter Schwefel. Location-quality-aware policy optimisation for relay selection in mobile networks. *Wireless Networks*, pages 1–20, 2015.

[24] Mislav Findrik, Thomas Fevre Kristensen, Thomas Hinterhofer, Rasmus L. Olsen, and Hans-Peter Schwefel. Ad-hoc, mobile, and wireless networks: 14th international conference, adhoc-now 2015, athens, greece, june 29 – july 1, 2015, proceedings. pages 317–329, 2015.

[25] Ahmed Shawky, Rasmus L. Olsen, Jens Pedersen, and Hans-Peter Schwefel. Network Aware Dynamic Context Subscription Management. *Computer Networks*, 58:87–101, December 2013.

# Chapter 3

# Evaluation Methodology

Given the difficulty and complexity of creating a joint analysis of controller performance under imperfect network conditions, it is not feasible to make a purely theoretical approach to determine optimal information access strategy parameters. In order to evaluate these complex scenarios it was deemed necessary to simulate the controllers to determine the effects of imperfect communication network conditions on performance. Simulations allow perfect information in regards to what is being simulated while allowing specific conditions to be tested. This chapter will give an overview of the co-simulation framework that have been created to perform the simulation for the results in this thesis. It will also discuss the metrics used for the results.
Other ways of analysing this problem is via testbeds. There have been several projects which have focused on communication networks in Smart Grid testbeds. One such example is Reference [1] which demonstrates a testbed that includes wired and wireless technologies. It is based on Ethernet and ZigBee networks. Reference [4] introduces a cognitive radio network in a Smart Grid testbed (specifically a micro-grid), with the main focus on the security of the Smart Grid. In Reference [3] ZigBee is used as the communication technology, and is implemented in an office environment with the goal to optimize the energy costs for the offices connected to the Smart Grid. Using a Smart Grid testbed was not an option for this thesis, primarily due to time and budget limitations. However, some testbed results regarding communication network delays and packet losses are used in the wind farm scenario. These results come from a complementary thesis [2].

# 1 Motivation for Evaluation Framework

The goal of the simulation is to make a joint evaluation which takes the controller settings, communication network parameters and information access strategy into account. Herein lies the challenge, as while there are extensive developed approaches and tools for investigating these items separately there are few options for making a joint simulation framework. To optimize controller settings simulation tools such as MATLAB and Simulink are often used. Communication networks are simulated in tools such as NS2, NS3, OMNeT++ or NetSim. For access strategies the literature usually describes custom made simulations. This still leaves the problem of running these simulations together.

# 2 Design of Co-simulation Framework

We have developed a co-simulation framework which allows a communication network and information access strategy to be simulated with a controller simulation. The first step in developing the co-simulation framework was to decide which simulation tools to use. For the controller simulator MATLAB was chosen, as this tool is widely used for simulating controllers, and it has a large library of tools and development packages. It was chosen to use OMNeT++ as the communication network and access strategy simulation tool. OMNeT++ has all the necessary features required by a communication network simulation tool, and a sufficient level details in the protocol stack implementation. It furthermore has a user interface that helps a user to visualize a developed network, which may also be helpful if other people are to use the framework. OMNeT++ also already contains models and modules for simulating all the layers in the network stack, as well as implementations of UDP or TCP and many other communication protocols.

The next step in the creation process was determining and defining the interfaces between MATLAB and OMNeT++. These can be seen in Figure 3.1. An issue here is that MATLAB is a Java based simulator while OMNeT++ is a C++ based simulator. Thus a communication protocol between the two simulators should be designed. Furthermore MATLAB and OMNeT++ have different ways of handling time. While both simulators are event driven OMNeT++ has the most crucial requirements in terms of time, as MATLAB would only care for specific time steps while OMNeT++ would need to know if a message was delayed in milliseconds.

These challenges have been solved in the implementation of the co-simulation framework by addressing the issues described in the paragraph above. This has been done by ensuring that both simulators have the same view of time in the simulations. The framework was designed such that OMNeT++ handles all matters in terms of timing.

This means that it is OMNeT++ that informs MATLAB that it can now simulate e.g. a sensor or the control algorithm. The framework was implemented such that it is not necessary for OMNeT++ to know what information is was sending in the network, as long as it knew the size of the information in bytes. This means that all the MATLAB simulation information is available in the same workspace, and as such it is important that the controller is fully aware what information it actually has access to. This means that the controller, whenever something is simulated, would need to inform OMNeT++ when in the simulation this is simulated, such that OMNeT++ could forward this information to the controller to ensure that the controller did not access information it is not allowed to access.

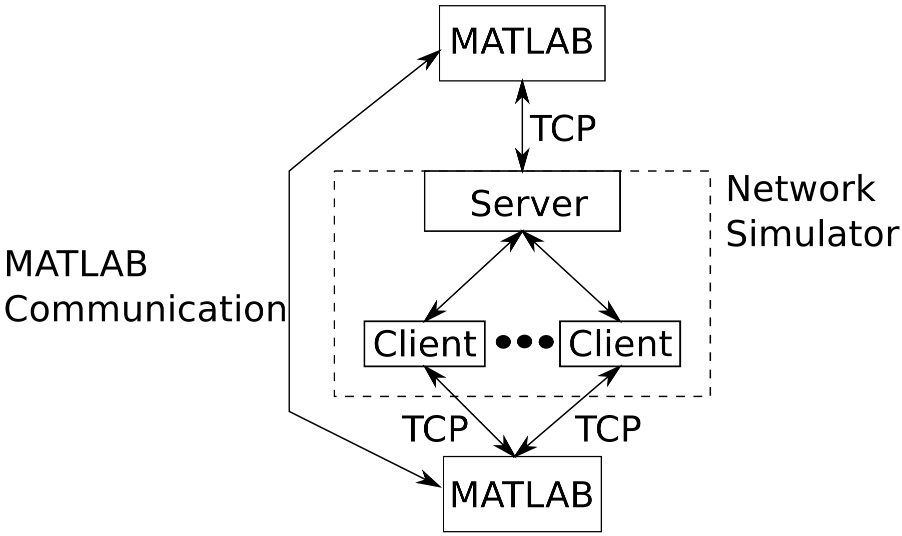



**Fig. 3.1:** Interfaces of the simulation framework

The co-simulation framework was designed to the MATLAB simulations to run with minimal required modifications to the controller. The modifications required for the MATLAB simulations are mainly to split the simulation it up into different functions. One function would simulate the control algorithm, another would simulate the plant, an initialization function etc. These functions allows OMNeT++ to only call the specific part of the simulation that should be simulated at the given time.

Because the controller part of the simulations were run in MATLAB, all the information required for post processing of data was available in the MATLAB workspace. This allowed for easy transfer of data into a MATLAB data processing script. The compartmentalization approach for the MATLAB part of the framework allowed for a relatively quick integration of the controller into the framework. Combined with OMNeT++

ability to make a fast simple network allowed for a relatively quick minimum working example (MWE). This MWE would give a good overview of how the controller acts when introduced to a communication network, and is a great help for further implementing information acccess strategies or communication networks in the scenario. Furthermore the OMNeT++ graphic interface allowed for easier problem detection when running longer simulation with more complex communication networks, as it was possible to see each packet being transfered if one wishes to. The OMNeT++ part of the framework could run into problems when running simulation with large amounts of cross traffic, as OMNeT++ creates events for each packet of cross traffic. When having large amounts of cross traffic many events occurred which would not significantly increase the simulation time, which greatly increased the length of the simulation runs.

# 3 Evaluation Metrics and Approach

When studying scenarios using this evaluation framework there are some common metrics which are considered for performance. When considering a communication network the first metric considered is the network performance. This is often described in terms for delays or information loss, among other terms. These are the terms considered in this thesis, and the network is often characterized by the performance characteristics it is wished to investigate. As such, if a certain loss probability's effect on controller performance is investigated, the communication network would be designed with this loss probability in mind. Another metric considered is information age. This relates both to the communication network performance, but also in terms of the information access strategy for when the information is accessed. The age of the information when it reaches a plant or controller can be crucial to the stability of the system. How important it is, is in part determined by the system itself, as a system with very slow dynamics a high information age can have lower impact.

A metric which describes both the network performance and the information age is mmPr. mmPr allows for comparison between performance of different communication network parameters, or information access strategies, without the requirements of using specific controller performance metrics. As it incorporates both network performance and information access it can be used as a single metric when having both these parameters in the same simulation. mmPr is in this thesis defined as the probability that the is a difference between the information the controller has available and the actual information that is at the sensor. In order to use mmPr in the simulations there was a problem of discretization. When considering the real world many values are continuous, and thus the mmPr would always be 1 as there would always be minute differences. In the simulations, the granularity of the sensor data was so high that it was necessary to further discretize the sensor data when calculating mmPr. This was done using a $\epsilon$ value, which determined by how large a difference there should be between two data

points before there is a mismatch.
There are some performance metrics which are generalized over most control scenarios, however these also contain variations in how they are implemented at the controller. The first is cost minimization. As a performance metric for this one can look at the cost of operation, or the saving in operations. Another metric would be regards to reference tracking, where one can make a performance metric based on the distance between the actual output and the reference. Here there is some room for variation, as it may be there is a higher penalty for going above the reference than going below it as an example.
The output performance metric can also depend highly on the controller or scenario being simulated, as the performance metric is decided by the controller. To test performance in these cases, one could run several simulations using a perfect communication network and an ideal access strategy. This would give a baseline performance, and assuming the controller has prior been optimized to a perfect communication network and an ideal information access strategy, would give the optimal performance of the controller. Subsequent simulation runs could then be performed using different imperfect communication networks, or non-ideal information access strategies, which would then most likely produce different results in terms of controller performance. A comparison between the optimal performance and the performance using imperfect communication networks, or non-ideal information access strategies, would then give the degredation of the performance. It is important to note that for a simulation run one might see an improvement in performance when expecting a decrease, which may be due to randomness in the simulations.

A single simulation run from this framework would allow the user to define the distribution for packet delays, and the distribution and probability for dropping a packet. It also allows the defining of the access strategy in terms of access type (push/pull/reactive), offsets, number of messages. Once this is determined, simulations are run using different seeds for the random number generators, in order to generate several simulation runs. This allows the user to quantify the variability that may occur in the simulations, due to randomness, by using confidence intervals.

# References


[1] B. Bhattarai, M. Levesque, M. Maier, B. Bak-Jensen, and J. Radhakrishna Pillai, "Optimizing electric vehicle coordination over a heterogeneous mesh network in a scaled-down smart grid testbed," *Smart Grid, IEEE Transactions on*, vol. 6, no. 2, pp. 784–794, March 2015.

[2] M. Findrik, "Network-aware adaptation framework and its application to smart grid controllers," Ph.D. dissertation, Aalborg University Doctoral School, 2016, phD thesis.

[3] I. Georgievski, V. Degeler, G. Pagani, T. A. Nguyen, A. Lazovik, and M. Aiello, "Optimizing energy costs for offices connected to the smart grid," *Smart Grid, IEEE Transactions on*, vol. 3, no. 4, pp. 2273–2285, Dec 2012.

[4] R. Qiu, Z. Hu, Z. Chen, N. Guo, R. Ranganathan, S. Hou, and G. Zheng, "Cognitive radio network for the smart grid: Experimental system architecture, control algorithms, security, and microgrid testbed," *Smart Grid, IEEE Transactions on*, vol. 2, no. 4, pp. 724–740, Dec 2011.

# Chapter 4

# Evaluation of Wind Farm Control

In this chapter it will be shown how the wind turbine control is affected by imperfect communication networks and information access strategies. The chapter starts by describing the scenario that is considered, and then show how information delay and loss impact the controller. It will then be shown how to optimize the information access strategy based on the communication network. Following this the chapter discusses how the mmPr metric can be used to simplify the optimization of the information access strategy. Lastly it is validated that mmPr can be used as a performance metric to optimize controller performance.

## 1 Problem Description and Assumptions

The choice of a the wind farm use-case gives a scenario where the delays of the communication network are very important. This is due to the short time horizon of the controller (150 ms) compared to the delays that are expected to be present in the communication network. The overall wind farm scenario is described in Chapter 1, and some of the more specific parts of the scenario become rather important when determining the communication network and the access strategy. When first considering the communication network one should determine which technologies are feasible for the given scenario. For this wind farm scenario there were two different communication networks that should be considered, what was called a sensor network and an IP network, as shown in Figure 4.1. Once the feasible technologies are determined, the access strategy should be considered. The access strategy is shaped by, among other parameters, the timing requirements of the system, the amount of entities in the system, the dynamics of the sensor information and the communication network technology.

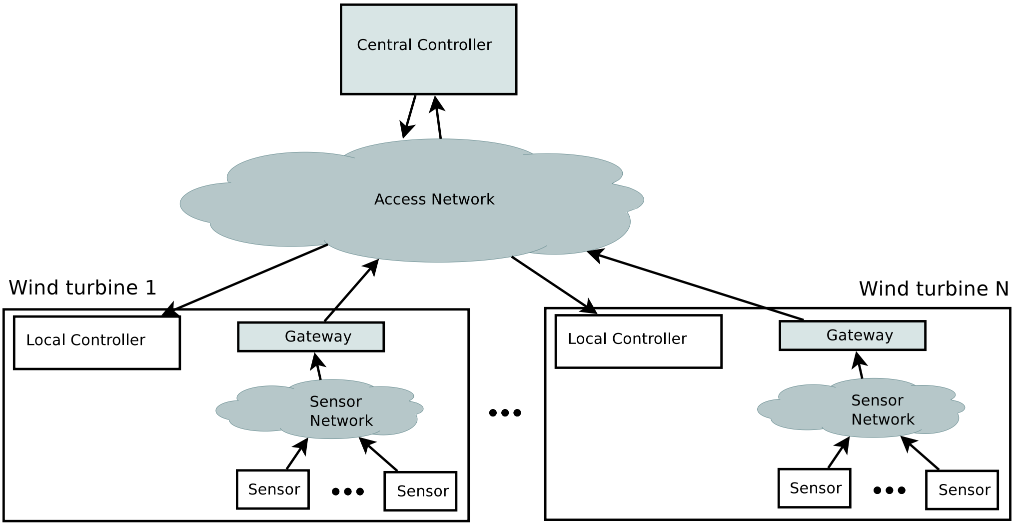



**Fig. 4.1:** Overview of the wind farm scenario, Paper A

For this wind farm scenario it was found that several technologies were possible to enable the communication network, both wired and wireless. These can be seen in Table 4.1, along with a list of pros and cons. In this Thesis we investigate how some of these technologies impact the performance of the wind farm controller using measurements of networks, and simulate others of these networks.

**Table 4.1:** The technologies considered for the wind farm scenario

| Technology | Pros | Cons |
|---|---|---|
| Fiber optics | Reliable, fast, private network | Expensive |
| Ethernet | Reliable, fast, private network | Expensive to install |
| PLC | Easy to install, existing infrastructure | Noisy medium, slow |
| 2G | Easy to install, existing infrastructure | Public network, slow, cross traffic |
| 3G | Easy to install, existing infrastructure | Public network, cross traffic |
| WLAN | Easy to install | cross traffic, low range |

Some of these technologies are simulated while on others measurements have been performed which are then used in the simulations. The technologies are PLC, 2G, 3G and WLAN, and the measurements have been done at the FTW testbed in Vienna. These measurement results where used to generate delay traces and probability density functions which were used in some of the simulations to simulate the communication network delays.

### Performance Metric of Wind Farm

In order to determine how the communication network affects the performance of the controller, a performance metric must be obtained. As performance metric this thesis considers the accumulated damage of the wind turbine, which is which is calculated from perfect information about the wind-turbine state in the simulation (as opposed to estimating it from delayed and noisy observations). An estimate of the accumulated damage is used by the controller to compute set-points. The higher this value is, the worse the performance of the controller is. The central controller aims to reduce the fatigue load of the shaft in the wind turbine. It does this while tracking a desired power reference. To achieve this the central controller requires an input from a fatigue estimator. This estimator requires sensor measurements of the wind turbine, which are the physical measurements of torque, as well as the angle rotation of the wind turbine. These measurements are run through different hysteresis relays with different weights. The dependency between the offset parameter and the control performance is evaluated in simulated experiments. In fact, the offset parameter together with network delays was determining whether the information collected from sensors at execution times of the controller has drifted from the real value, and how it has an impact on the control performance. In order to model aforementioned effect we utilize the metric mmPr as defined in Reference [1]. MmPr considers the aspect of real time information access, capturing impact of access delays and access strategies on information accuracy in a distributed system. Let us assume $X_s(t)$ is the state of sensor's information (represented by a discretization of the value range) and $X_c(t)$ is the state of the sensor known to the controller. $X_c(t)$ is sent over the communication network and then the controller updates $X_C(t+d)$ after receiving the message. In this thesis the mmPr is defined at the discrete time instances at which the controller is being executed at control time $(C_i)$.

## 2 Evaluation of Imperfect Communication Network on Wind Farm Control

In order to determine if it is possible to optimize controller access strategy it is first important to understand the impact an imperfect communication network and information access strategy has. It was first determined that the performance of the controller is impacted by the delays and losses of a imperfect communication network. Paper A and [3] show that the accumulated damage increases as delays increase. These papers also determined that the information access strategy also plays a role in the decrease in the performance of the controller, as varying offset impacted the performance. [5], and expanded on in Paper A, shows that choosing an optimal information access strategy is

not a trivial task, and that choosing it heuristically may give a sub-optimal information access strategy.
Using the delay and packet loss traces found via the testbed measurements it was possible to determine which of the technologies considered were feasible for the communication network of the wind farm controller. It was found in Paper A and [4] that PLC and 2G both have delays that were so large that the controller either became unstable or the performance decreased significantly (by at least a factor 2). Further it was found that PLC had too high packet loss, mostly due to queuing, to be feasible as a communication network technology. On the other hand it was found in these papers that both 3G and WLAN had delays that were low enough for the controller to run without becoming unstable, and an optimal information access strategy to be determined, as described in the following section.

# 3 Simulation Based Optimization of Information Access Strategy

Paper A shows that it is possible to determine an information access strategy that optimizes the controller performance based on the delays and losses of the communication network by using extensive simulations of the controller under various imperfect communication network scenarios. It was discovered during the analysis that the sensor information was changing several times during the control period and that the sensor information could change significantly during the duration of the network delay. In these fast changing scenarios the choosing of the optimal information access strategy becomes a trade-off between the ensuring that sensor information is received before control computation, but that it has not changed since the information was sent. The main part of the information access strategy being considered here is the offset, $T_{Offset}$. This value determines the maximum delay allowed before the information is delayed past the control computation time. However, it may be, and this will be discussed further below, that the information at the sensor has changed by the time the information is used by the controller. It was found in Paper A that if $T_{Offset}$ was too large the fast changing nature of the sensor information caused the controller to use outdated information. Paper A shows that choosing the optimal offset is not a trivial task. So far simulations of the entire system have been considered, which requires extensive and long simulations. A different method for choosing the $T_{Offset}$ which does not require such extensive simulations may be beneficial, and one such method could be the use of models of the sensor information dynamics to determine mmPr, which may in turn be used to determine the optimal offset.

### Validation of Modelled MmPr to Simulation MmPr

As previously stated choosing $T_{Offset}$ is not a trivial task. In Paper A it is found that choosing the value $T_{Offset}$ is a trade-off between ensuring the offset is large enough that there is a high probability the message is received before the next control computation instance, while it should be small enough that the sensor information does not change between the message is sent and the control computation instance. The paper describes a model which should balance these two aspects. The first aspect is primarily influenced by the end-to-end communication network delays, while the second aspect is primarily influenced by the sensor information dynamics. An analytic model for the controller performance is however very complex, hence the approach is to instead optimize the information quality, defined by the mmPr.
In order to describe the sensors' information dynamics as a continuous time Markov chain (CTMC), a transition matrix Q has to be derived from a finite number of discretized sensor values. The process of doing this is described in Paper A, and more detailed in [2]. Part of the model parameters are the number of states in the CTMC, denoted $N$. It was shown in Paper A, Section 4, that the model results for mmPr closely resembled those for the simulation results. It was also shown the absolute values of the mmPr curves are dependent on the choice of N, which is the number of states in the Markov Model, or on $\epsilon$, which is described in Chapter 3, and relates to the simulation sensor information discretization. Changing the $N$ or $\epsilon$ values also modifies how sensitive the mmPr curves are to changes in the information, which may cause the mmPr curves to have broader or more narrow minimum plateaus. By using the mmPr model it was possible to determine an $T_{Offset}$ in the information access strategy that was the same as the one the simulations found to be the optimal offset.

## 4 Suitability of MmPr as Performance Metric

Paper B analyzes in more detail whether mmPr is a suitable metric to use to optimize controller performance, rather than using the performance metrics inherent in the controller. This would allow a more generalized approach to optimizing controller performance based on communication network parameters.

Two hypotheses are introduced in Paper B:

1. Hypothesis-1: A significant difference in control performance causes a significantly different behavior of mmPr with the adequate monotonicity relation (i.e. higher control performance causes lower mmPr).

2. Hypothesis-2: A significant difference of mmPr causes a significantly different behavior of control performance with the adequate monotonicity relation (i.e. lower mmPr causes higher control performance).

A significant difference here refers to non-overlapping 95 % confidence intervals from the simulations. These hypotheses are then tested for all combinations of pairs available. This means that the mmPr is being used to optimize across parameter tuples (delay and offset), while we in the previous work only looked into varying once of these.

In Paper B it is shown that the relative frequency of occurring simulation results pairs that confirm Hypothesis 1 respectively Hypothesis-2 for different thresholds $\epsilon$. As described in Chapter 3, $\epsilon$ determines the granularity of when a mismatch occurs. It was shown, on Figure B.5 in Paper B, that for $\epsilon$ values less than $2.5 \cdot 10^{-4}$ Hypothesis 1 is true over 80 % of the cases, while Hypothesis-2 is true over 60 % of the cases. For larger values of $\epsilon$, this probability reduces strongly for both hypotheses. It is important to note that for high enough $\epsilon$ values there would never be a mismatch, and for very low values there may always be a mismatch. These are the primary concerns for when choosing $\epsilon$ values, and it is one of the primary reasons that there are such large variations in the results for low and high $\epsilon$ values.
Values of around 80% and 60% of observing Hypothesis-1 and Hypothesis-2, respectively, in the data, do not imply that otherwise inverse (wrong) relations of the corresponding metrics are observed. There is also the possibility that no significant difference is observed in the simulation experiment, so not giving evidence of a wrong Hypothesis. Figure B.6 in Paper B plots the relative frequencies that the corresponding performance metrics in the Hypotheses show a significantly wrong difference, which shows probabilities of less than 14 % at the highest probability. This shows that for over 85 % of $\epsilon$ values, utilizing mmPr to optimize control performance does not decrease control performance.

# 5 Conclusion and Outlook

In this chapter the impact on the controller performance when introducing an imperfect communication network and information access strategy was quantitatively analyzed. It was determined that it was possible to reduce the impact by choosing an optimal information access strategy i.e. control performance shows a maximum over varying offset values, however choosing the strategy is not a trivial matter. The chapter shows that it is possible to model the sensor data, and using these models to find an optimal information access strategy. Lastly it was determined that in the wind farm scenario, for over 85 % of the $\epsilon$ values, optimizing the information access strategy based on mmPr does not decrease controller controller performance.
Further work in this field could be to investigate how information regarding the communication network state might be used by the controller, in order to further decrease the negative impact of an imperfect communication network on controller performance.

# 6 Included Articles

There are two articles that forms this part of the Thesis. **Paper A: Optimizing Data Access for Wind Farm Control over Hierarchical Communication Networks** Paper A is a journal paper expanding on the work done in the earlier articles: *On the Impact of information access delays on remote control of a wind turbine* [3], *Investigating Wind Farm Control over Different Communication Network Technologies* [4], *Scheduling Data Collection for Remote Control of Wind Turbines* [5].
In this paper the impact of a imperfect communication network and access strategy on the performance of a wind farm controller was investigated. The communication network was simulated and also measurements of delay and loss from a testbed were used. Based on the performance of the controller an optimal information access strategy was found. These results were then compared to results where the sensor information dynamics were used to determine the mmPr of the system, and an optimal offset based on the mmPr was found. It was found that it was possible to use the mmPr of the sensor information dynamics to determine an optimal offset.

**Paper B: On the use of Information Quality in Stochastic Networked Control Systems**
Paper B is a journal paper which investigates if there is a correlation between mmPr as a control performance metric, and control performance. First the performance of a wind farm controller as a realistic example case for networked control systems is analyzed. This is where my contribution to this journal paper is focused, and is described in Section 2 in Paper B. Many simulations were performed of this scenario, where it is possible to determine both controller performance and mmPr. Using these results a hypothesis was confirmed, that when there is a significant change in control performance there is a significant change in mmPr in the same direction. Another hypothesis was that the inverse is also true, a significant change in mmPr leads to significant change in control performance in the same direction. As the simulation is subject to stochastic variability, and it required a large amount of computational effort the paper introduces a very simplified control scenario, and models this scenario using Markov chain models. This allows the paper to analyse the mmPr through numerical solutions rather than via extensive simulations. The paper considers both ideal information access as well as periodic information access and non-zero delays. This is used to evaluate in which scenarios that mmPr is useful for optimizing control performance.

# References

[1] M. Bogsted, R. L. Olsen, and H.-P. Schwefel, “Probabilistic models for access strategies to dynamic information elements,” *Performance Evaluation*, vol. 67, no. 1, pp. 43 – 60, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166531609001175

[2] M. Findrik, “Network-aware adaptation framework and its application to smart grid controllers,” Ph.D. dissertation, Aalborg University Doctoral School, 2016, phD thesis.

[3] J. T. Madsen, J. de Jesús Barradas-Berglind, T. K. Madsen, and H.-P. Schwefel, “On the impact of information access delays on remote control of a wind turbine,” in *PowerTech, 2015 IEEE Eindhoven.* IEEE, 2015, pp. 1–6.

[4] J. T. Madsen, M. Findrik, D. Drenjanac, and H.-P. Schwefel, “Investigating wind farm control over different communication network technologies,” in *DA-CH Conference on Energy Informatics.* Springer, 2015, pp. 129–140.

[5] J. T. Madsen, H.-P. Schwefel, M. Findrik, R. L. Olsen, and T. K. Madsen, “Scheduling data collection for remote control of wind turbines,” *Energycon 2016*, 2016.

# Chapter 5

# Load Shifting and Peak Reduction Using Thermostatic Loads

This chapter will discuss the impact of a imperfect communication network and information access strategy on the control of thermostatic loads. The chapter will discuss the two different scenarios concerning load shifting of thermostatic loads that are considered in the thesis, a supermarket scenario and a LV grid scenario. It then looks at how it is possible, using network monitoring, to modify the control signals of the controller in order to reduce the impact if packet loss on the controller performance.

## 1 Problem Description and Assumptions

Two different scenarios are considered in this use-case. As both deal with temperature controlled entities the nature of the information sent changes slowly compared to controller cycles, and the controller cycles are usually rather long. The main focus of these cases will be investigation into packet loss, and how packet losses affect the controller performance.
In the first scenario a supermarket controller is investigated, as described in Paper C. This scenario uses prices to determine the actions of the supermarket, and it was investigated how the loss of local sensor information could cause the supermarket controller to increase the cost of running the supermarket. The second scenario considers the control of fridges in a LV grid scenario, described in Paper D. Here is was investigated how packet losses could lead to the controller being unable to perform peak reduction by performing load shifting. Furthermore it was in this specific scenario considered how

to modify the controller information based on the measured packet loss to reduce the impact of packet loss on the controller performance.
A general overview of these scenarios can be seen in Figure 5.1. There is a market, which sends prices or prediction of behaviour of prices to the controllers. The controllers regulate the thermostatic loads, fridges or freezers in the case of the two scenarios, and the thermostatic loads responds with local state information. What is investigated for both scenarios, is the impact of imperfect communication network on performance of the controllers. For the LV grid scenario it is considered how network monitoring would allow for modifications of the controller, in order to reduce the impact of the imperfect communication network. The focus is on the communication network and determining information access strategy in the area between the controller and the fridges, and not on the communication network between the market and the controller (see Chapter 6 for a discussion of the latter).

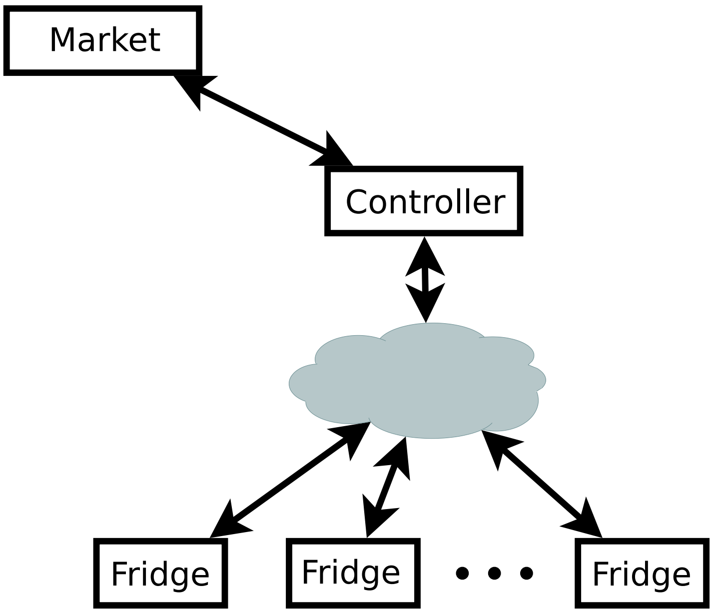



**Fig. 5.1:** Overview of a general thermostatic load control scenario

The two chosen scenarios within this use-case both regulate temperature controlled entities. The controllers attempt to shift the load of the thermostatic loads based on information they receive from the market. The supermarket controller receives price data, and attempts to fit its consumption in such a way that the price paid for the electricity is reduced. The LV grid controller receives a power reference which it must attempt to make the grid follow. While the supermarket controller has complete control over the fridges, the LV grid controller can only send encouragements for the fridges to change status.

## Communication Networks and Access Strategy in Supermarket Scenario

In this scenario a supermarket supervisory controller is considered, which attempts to shift the load of the supermarket by controlling the temperature in its freezers and refrigerators. It is assumed in this scenario that the price of electricity varies within time-intervals of 15 minutes. These variable prices are, however, assumed known one day in advance and are communicated to the supervisory controller. These assumptions are motivated by experience from real deployments, and predictions of future implementations. The supervisory controller regulates the temperature such that when a price peak is imminent it increases consumption prior to the peak, to lower the consumption during the peak. As the supermarket knows (due to the price signal) when the price will be high or low it can perform load shifting in advance of peaks in price. A supermarket consists of several zones denoting different requirements for heating or cooling, each zone containing a local controller. The local controller regulates temperature based on set-points received from the supervisory controller, as well as information received from local sensors. The general supermarket setup in this paper is assumed to have a zone for each type of heating and cooling, which means a zone for freezers, a zone for refrigerators, a zone for the heating at the entrance, for the heating at the fruit department etc. This means that there are multiple local controllers in the supermarket, and the supervisory controller must balance all these controllers. This is also the reason that the supervisory controller cannot be co-located with the local controllers.

The physical location of the supervisory controller in relation to the local controller can have a large influence on the communication network, both in determining feasible technologies and the conditions of the communication channel. When adding a communication network to a control scenario, the access strategy for the system is also determined. The access strategy determines the timing of when messages are generated, and when they are sent, it determines how and when messages are received are handled and processed. Are they for instance used immediately when they are received or are they cached for later use. This is in addition to the delays and risk of losses that a communication network imparts when it is implemented. The supervisory controller uses a predetermined price signal, which covers a 6 day period. The system has two other inputs which it has no control over, one being the indoor temperature. This input is modelled as a sinusoidal function which fluctuates between 19 and 21 degrees Celsius. The local controller tracks the reference signal provided by supervisory controller and sends information to the supervisory controller every minute, and the supervisory controller responds with set-points every 15 minutes. The local controller system is, from our point of view, a black box system, and as such it was not feasible to consider the effects of communication networks within this subsystem.

### Performance metric

As a performance metric the normalized price for running the supermarket is used. The normalization is done compared to the supermarket running with an ideal communication network and access strategy. It determines the cost of running the supermarket by multiplying the total power consumption over an entire day with the price at the given time of that day. The sum of this gets a single value for each day. There will in most cases be variation between each day, as the price and temperature signals are different for each day, which can also be seen on the results with each day having similar but different behaviours. Two different communication scenarios are shown, one where the delay of the packets is increased, and another where the packet loss is increased of the communication network between supervisory controller and local controller. The results shown in Paper C are not averaged over the days that are simulated, as each day did show significant changes in the price data, and as such it was found that averaging across days could hide relevant information for specific days.

## Communication Networks and Access Strategy in LV Grid Scenario

This scenario is based on a controller that attempts to regulate the measured power consumption in a grid, by regulating refrigerators. The regulation depends on an estimate of future power consumption in the grid. In this scenario it is assumed that two spikes in power consumption will occur. The controller attempts to increase or decrease power consumption by turning fridges in the grid on or off, and it does this by sending a single value to all fridges in the grid, indicating a certain percentage of fridges that should turn either on or off. The fridges then decide locally, based on local information regarding temperature and compressor state, if the fridge will turn on or off. The scenario is simplified by only having two different entities in the grid, a number of non-controllable units designated as lights, and a number of controllable units designated fridges.
The communication network in this scenario is assumed to be PLC. The controller broadcasts the control signal the downstream communication, and as such is not very demanding. As there is only 100 units, and a 1 minute interval the upstream communication is neither very demanding, as all information transmitted is an id and a bit indicating if the fridge is on or off.

### Performance Metric

The controller attempts to perform peak reduction while following a predetermined power reference signal. As a performance metric for this controller it was chosen to use a performance metric based on several simulations of the same day under ideal network conditions. These results are then averaged for each minute of the day, giving a expected total power consumption. Any results found using non-ideal communication networks can then be compared to these values, by measuring the deviation from the expected total power consumption.

### Network Monitoring

In this work it was decided to utilize a simple scheme for measuring the packet losses the controller would experience. This was done by having each fridge only respond when it receives a message from the controller, and then counting the number of responses at the controller. This gives a two-way packet loss, as the controller is assumed to know the number of fridges, which makes it possible to calculate the one-way packet loss, under the assumptions of independent and symmetrical packet loss rates.

## 2 Evaluation of Supermarket Scenario

In order to understand the impact of packet losses, the simulations are run with different packet loss drop rates in three different communication network scenarios. The first results for the simulation with packet loss are for downstream packet loss only, while upstream communication is fully reliable. Packets are not dropped and are delivered instantaneously, i.e. delay is zero. It is shown in Paper C that for low packet losses the normalized price is close to the the ideal communication network scenario, however as the packet loss increases the normalized price increases as well. This shows that there is an impact on the controller performance due to imperfect communication networks. The results show the similar behaviour for the upstream communication. The results in Paper C show that the upstream packet loss impacts the overall system performance more strongly as compared to the downstream packet loss. The results, however, also show a high variability, which leads to wide confidence intervals for the shown relative mean energy cost.

## 3 Modifying Control Signals Based on Network Conditions

For the LV grid control scenario, Paper D shows that the impact of packet losses leads to a decrease in the controller's ability to perform peak reduction and load shifting. The controller uses a control signal between -1 and 1, with the same value broadcast to all fridges in the system. The control signal indicates a probability to turn either on or off, with negative numbers indicates probability to run off, positive on (i.e. a -0.3 would mean that fridges should turn off with 30 % probability). The fridges acts on the control signal dependent on their own internal state, i.e. if the temperature is close to the upper bound in the fridge it will not turn off.
In Paper D it was shown that the controller makes assumptions based on the number of fridges it receives information from, as well as the number of fridges that received the control signal. As there is a risk of loosing information, the information that the controller uses may not be accurate. It was chosen to make a simple network monitoring scheme, that counts the number of packets received from the fridges by the controller. As the fridges were designed to only respond if they receive a control signal, this is a two-way packet loss that is measured. The one way packet loss is calculated, as this is the information used to modify the control signal.
The control signal was modified by using the one-way packet loss of the last control signal transmission. The modification was done by increasing the probability that the controller sends to the fridges, ensuring the more fridges react to the control signal. It was found that by making this modification it was possible to reduce the impact of even large packet loss percentages, although it was not possible to completely remove the impact of packet loss on performance.

## 4 Conclusion and Outlook

It was found for both scenarios that imperfect communication networks and information access strategies there is a decrease in controller performance. It was further found, in Paper D, that it was possible to implement a network monitor and based on the measured packet loss it was possible to modify the control signal of the controller. It was found that performing this modification the performance increased significantly compared to the performance without modifications.
Future work would be in investigating the supermarket scenario further, either by doing similar work as was done on the LV grid controller, or by using a different supermarket controller and performing the same analysis and simulations. It would also be interesting to more reliable protocols like TCP to compensate for the losses.

# 5 Included Articles

There are two articles that forms this part of the Thesis. **Paper C: Impact of Communication Network Performance on Supermarket Control**
This paper details the impact of an imperfect communication network on the performance of a supermarket controller. It focuses on the impact of packet loss, both in upstream and downstream directions. It found that the upstream communication was more susceptible to packet loss than the downstream communication.

**Paper D: Utilizing Network QoS for Dependability of Adaptive Smart Grid Control**
This paper introduces a low voltage fridge control scenario where the controller aims to perform peak reductions. It shows the impact of packet loss on the performance of the fridge controller. It details how to perform network monitoring on the communication network. It then shows how it is possible to modify the controller to take packet losses into account when calculating the control signal, thus greatly reducing the impact of packet loss on the controller performance.

# Chapter 6

# Performance Analysis of Multiple Smart Grid Controllers over Communication Network

In the previous parts of this thesis it was shown how two different Smart Grid controllers are impacted by imperfect communication network and information access strategies. In this chapter it is considered how these controllers would tie in to a more complex Smart Grid setup, such as the one described in Chapter 1. It is also investigated how to simulate multiple interacting Smart Grid controllers, and a communication network, in a single simulation framework.

## 1 Problem Description and Assumptions

This scenario is considered as it incorporates the two use-cases that were detailed previous in this thesis, and can thus show how these controllers would work in a Smart Grid scenario. The scenario is load frequency control (LFC) of a modern wind power dominated system, as seen on Figure 6.1. The LFC consists of medium and low voltage grid units participating in power balancing control. This MV grid contains a distributed wind farm, supermarkets, distributed CHPs, and a low voltage grid. In this scenario the HV grid is seen as a limited size energy buffer. The grid entities are connected to the electricity market.

The communication network considered is the communication network between the

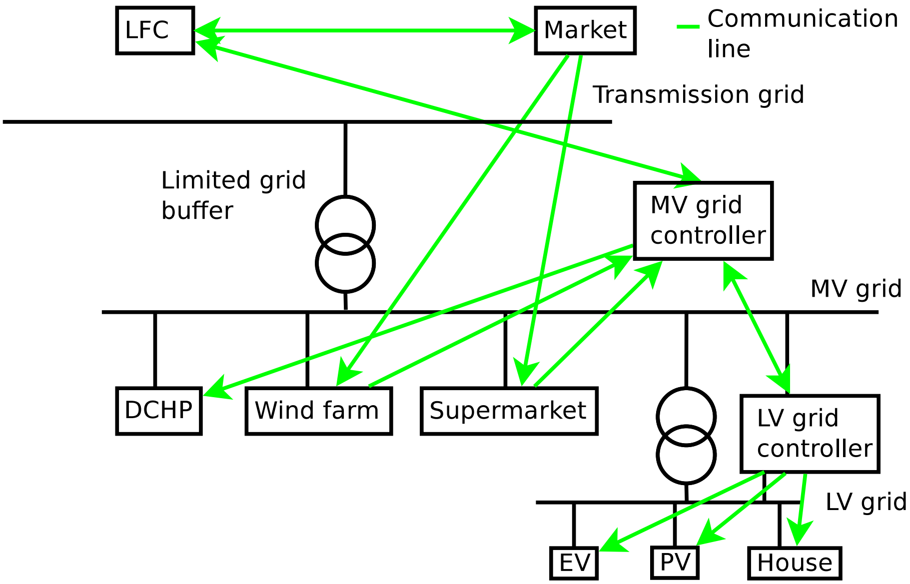



**Fig. 6.1:** Overview of the scenario, Paper E

market, wind farm, supermarket, LFC and MVGC. This work is done in two steps. In the first step it is investigated what impact extremely poor communication network performance (long delays, high losses) would have on the systems ability to balance the power grid. This poor performance could be caused by malicious attacks on the ICT infrastructure. In the second step it is investigated whether using locally modified prices based on the power balance helps balance the grid. This is again investigated with potential attacks on the ICT infrastructure.

The Market is assumed to be a single nation-wide entity which dictates the electricity price every hour based on the current power production and consumption in the whole nation-wide grid, and predictions of future power production and consumption. This price is communicated to each flexible entity in the grid. By setting the price it will influence the controllers of the Smart Grid, as they produce or consume based on, among other things, the price of electricity.

The wind farm controller is an simplified and abstracted version of the controller described in Part II. It takes decisions based on a trade-off between the cost of producing energy and the price of the electricity.

The supermarket controller is similar to the one described in Part III, although it is abstracted and simplified. The controller is set to increase or decrease consumption/production based on the price of electricity, if at all possible.

The first target for the LFC is to stabilise the frequency, and as a secondary priority

attempts to balance the power consumption based on system flexibility to the lowest cost of the system. It is assumed it achieves this by calculating set-points for the aggregated generation units, medium voltage grid controller (MVGC) and low voltage grid controller (LVGC). It takes in measurements from the power grid on frequency and voltage, as well as information regarding state of the distribution grid from the MVGC/LVGC, and uses this when generating set-points.
The coordinated MVGC/LVGC must follow the set-points from the LFC, and uses these to regulate the consumption and generation of power at the consumers. The MV grid involves distributed wind turbines and small/medium CHP units and flexible consumption loads such as supermarkets. The LV grid contains prosumers with different flexibilities, among these are electric vehicles (EV), photo voltaic units (PV) and households.

# 2 Evaluation of Smart Grid Scenario Under Imperfect Communication Network

Paper E considers two different market dynamics for this scenario. The first is what is in the paper called a nation-wide market, where the local energy grid has no impact on the prices of the market. The second market dynamics is where the prices are modified based on the local power balance, and this is called locally modified prices. First it is considered how the grid behaves when the nation-wide market dynamics and ideal communication network is used. It was shown that the different controllers behaves as expected, and that the grid seems to be balanced.
The system is then considered using locally modified prices based on the power balance in the grid, and it is investigated how the behaviour of this scenario changes as the communication network degrades in performance, as shown in Paper E. Based on a histogram of the price in the ideal communication network scenario, a high and low price threshold is set. These thresholds are set based on values where approximately 30 % of prices lie within the threshold, and was found to be $65 DKK/KWh$ as the low price threshold, and $165 DKK/KWh$ as the high price threshold. The consumption in the time period when each of these thresholds are exceeded is then averaged. It is hypothesized that when there is packet loss or substantial communication delays, the system's ability to regulate the consumption based on price will decrease. If this happens the average consumption in low price periods will decrease, as the entities in the system are not aware that they should increase consumption. Conversely, the average consumption in the high price periods will increase, as the system does not know that it should decrease consumption.

It is shown that the system is rather resilient to packet loss, with needing more than 80 % packet loss to reduce performance of the system when using locally modified prices, and the results for delays show similar behaviour. The average consumption in the defined price periods do not change significantly until the packet loss exceeds 80%. It is shown that over 80% packet loss the consumption in low price periods decreases, and it increases in high price periods, as expected. Similar for delay the graphs cross at slightly over 2 hours delays.

# 3 Conclusion and Outlook

It was found that the framework is able to simulate both the communication network and the controllers together in one simulation framework. It was found in Paper E that while there is some benefit to using locally modified prices, it is not a large improvement in power balance for the grid. Better models for the local market may improve this, by taking predictions of future wind or solar radiation into account. It is further found that it took extreme degradation in communication network performance before there are noticeable effects on the performance of the controllers in the grid.
Future work would be in updating the market module, and in adding a communication network between the MVGC and LVGC.

# 4 Included Articles

This Part of the Thesis is based on Paper E. **Paper E: Resilience of Urban Smart Grids Involving Multiple Control Loops**
This paper shows a co-simulation framework which allows multiple Smart Grid controllers to be simulated together with a communication network. It shows that the controllers in the framework behave as expected under fixed parameter conditions. It was further shown that the system was very resilient to packet losses or delays in the communication network. Under extreme cases it was found that there were decreases in the performance of the consumers.

# Chapter 7

# Conclusion

The main problem statement of the thesis, as stated in Chapter 1.4, is: **How do imperfect communication network performance and design choices in the information access strategies impact networked Smart Grid controllers?** This thesis quantifiable shows that for two different use-cases that the controller performance is decreased when an imperfect communication network and information access strategy is imposed on the controller.
The thesis detailed a co-simulation framework which allows a communication network to be simulated together with a controller. This helped investigate scenarios where the complexity of the control simulation made a purely theoretical analysis of the impact of the communication network infeasible. This framework was what allowed for the analysis of the impact of imperfect communication network and information access strategies on the controllers.
For the wind farm use-case it was investigated what the impact of imperfect communication networks was on the performance of the controller. The thesis showed that the performance of the controller was decreased when imposing a communication network on the simulation of the controller. It further shows that it is possible to improve performance of the controller by choosing and optimal information access strategy. For the wind farm use-case it was found that mmPr can be used as a metric to select an optimal information access strategy, and this was validated over several different communication network scenarios.
For the thermostatic load use-case, and considering the communication network parameters used in two of the use-cases, it can be concluded that this use-case is more resilient to delays in information. This is due to the slower nature of the plant dynamics for the thermostatic loads. There was still an impact of imperfect communication networks on the performance, as shown in Chapter 5. The thesis also shows that it is possible to modify a controller based on measurements of packet loss in the communication net-

work, and by doing so improve the controller performance in packet loss scenarios.
In the last use-case, which considered a more complex Smart Grid scenario with multiple control loops interacting, it was found that it took extreme conditions of packet loss or delays before the ability of the grid to regulate based on price signals was impacted.
The are several questions that arise from this work which could be interesting to conduct further work on. The one this author considers most interesting is: how does one modify the more complex controllers based on the communication network performance? This would start by doing modifications to specific controllers, either my changing their control signals or possibly by modifying the control algorithm itself, and possible move on to make a generalized method for optimizing controller performance based on communication networks. MmPr could play a significant role in this as it was designed to take several network parameters and give a single value to represent all of them. To solve this one would need some form of online network monitoring system, which would raise further questions on its own.

# Part II

# Papers

# Paper A

## Optimizing Data Access for Wind Farm Control over Hierarchical Communication Networks

Jacob Madsen, Mislav Findrik, Tatiana Madsen, and Hans-Peter Schwefel

The paper has been published in the
*International Journal of Distributed Sensor Networks* Vol. 2016(2016), Open access journal: http://dsn.sagepub.com/content/12/5/5936235.abstract, 2016.

## Abstract

*In this paper we investigate a centralized wind farm controller which runs periodically. The controller attempts to reduce the damage a wind turbine sustains during operation by estimating fatigue based on the wind turbine state. The investigation focuses on the impact of information access and communication networks on the controller performance. We start by investigating the effects of a communication network that introduces delays in the information access for the central controller. The control performance as measured by accumulated fatigue is shown to be significantly impacted by communication delays and also by the choice of the time instances at which sensor information is accessed. In order to optimize the latter, we introduce an information quality metric and a mathematical model based on Markov chains, which are compared performance-wise to a heuristic approach for finding this parameter. This information quality metric is called mismatch probability, mmPr, and is used to express quantitatively the information accuracy in a given scenario. Lastly measurements of different communication technologies have been performed in order to carry out the analysis in a practically relevant scenario with respect to the communication network delays. These measurements are done in regards to packet loss and communication delays, and the simulations are rerun using either the traces from the measurements or scenarios constructed from the delay parameters.*


# 1 Introduction

Over the last few years we have seen an increase in deployment of renewable energy generation. Due to increased deployment and the fluctuating nature of renewable power generation, there is a greater need for controlling the balance between energy consumption and production, by means of Smart Grid deployments. One large contributor to renewable generation are wind farms. Controlling a wind farm can be done from a central controller which harmonizes the wind farm's power generation by sending set-points to each wind turbine, thus a control is performed via a communication network. This way a central controller ensures that each wind turbine produces the required amount of power. For deployment of the wind farm communication network it is possible to use different technologies, however, each technology has different behaviour, hence it is important to analyse the impact of these communication technologies on the controller's performance.

In this paper, we consider a central wind farm controller gathering information from wind turbines distributed over an area. We analyse control performance under communication network conditions, and determine which parameters of the information access have a strong impact. In order to optimize the parameter we determine an information quality model, expressing quantitatively the information accuracy as difference between the value used by the controller and the real physical value. This information access model therefore needs to include information dynamics, which we obtain empirically from a simulation of the system. In order to perform the analysis in a practically relevant scenario with respect to communication delays, we perform measurements of different communication technologies, where we measure packet loss and delays, and use these measurements when simulating the wind farm controller. The comparison of the simulation results of control performance metrics provides a quantitative basis for the suitability of communication technologies for this distributed control scenario.

In recent years attempts have been made in creation of Smart Grid test-beds that integrate electrical assets (or their models) with communication technologies (or communication network simulators) in order to study interdependencies between the cyber-physical system [1]. One such example is Reference [2] which demonstrates a testbed that includes wired and wireless technologies. It is based on Ethernet and ZigBee networks. Reference [3] introduces a cognitive radio network in a Smart Grid testbed (specifically a microgrid), with the main focus on the security of the Smart Grid. In Reference [4] Zigbee is used as the communication technology, and is implemented in an office environment with the goal to optimize the energy costs for the offices connected to the Smart Grid. Measurements of WLAN technology in other scenarios confirm the order of magnitude of the observed delays in our experiments. However, in the presence of stronger cross-traffic, the shape of the delay distribution will change, see Reference [5]. Reference [6] details different communication technologies available for wind farm communication, among these are WLAN and ethernet based solutions. Another communication technology for a wind farm is fiber optics, as described in Reference [7]. Unlike previous work that analysed communication technologies for wind farms, our methodology is based on measurements obtained from real communication networks in a scaled down manner compared to the real environment. Subsequently, we use the measurement results by plugging the traces into a co-simulation framework which is similar to the framework used in Reference [8].

The use of sensor information in distributed systems is subject to two types of inaccuracies: Measurement errors at the sensor may propagate through the whole computation chain, see, e.g., [9] for work characterizing such errors and their impact. For distributed real-time systems, a second cause of inaccuracies requires attention: while the sensor data is being transmitted and processed, the actual physical value changes so that it deviates from the value used in the processing. In this paper, we focus on the second aspect and utilize a information quality metric, called mismatch probabilit (mmPr). This metric and mathematical models for certain base cases was introduced first in Reference [10]. MmPr considers the aspect of real-time information access, capturing impact of access delays and access strategies on information accuracy in a distributed system. So far, this metric has been used in the following context: (i) in Reference [11] mmPr was used in context subscription management systems for effective configuration of context access strategies in order to maximize the reliability of context information;

(ii) in Reference [12], the metric was used to find optimal location-based relay policies in mobile networks. In this paper, we investigate whether optimization of the information quality is a means of improving the control performance of the hierarchical wind farm controller.
The work in this paper builds upon the conference publications in References [13], [14], and [15]. Compared to these conference papers, additional results are included, showing the following new aspects: (1) The simulation results use longer traces and drop the transient phase of the simulation; that way better statistical significance of the results is achieved; (2) The communication network structure is now hierarchical and performance measurements from a CAN network simulation are used to characterize the impact of this hierarchical structure; (3) The analytic model from Reference [14] for the information quality metric has been newly developed to now explicitly distinguish control actions and system evolution without control. It now fully captures the qualitative behaviour of the corresponding simulations.

# 2 System overview

This paper considers a system consisting of a central controller, communicating with $N$ wind turbines over a communication network, each wind turbine containing K sensors. There is a bi-directional information flow, consisting of sensor measurements from the sensors to the central controller, and set-points from the controller to the wind turbine. The architecture of the system is shown in Figure A.1 and explained further in Section 2.1.

The central controller has the objective to follow a set-point for the wind farm's power generation while reducing the fatigue a wind turbine experiences in order to prolong the lifetime of the wind turbine. The central controller should ensure that each wind turbine in the farm produces a certain level of power to ensure that the entire wind farm reaches the given set-point for the wind farm, motivating the central control architecture of Figure A.1. The sensors send information regarding the state of the wind turbine to the central controller. The central controller uses the sensor values and its internal state to compute set-points for the wind turbine. These set-points are then forwarded to the wind turbine, where the local controller uses them to adjust the wind turbine. An overview of this scenario can be seen in Figure A.1, where the grayed boxes denote places that introduce delay in the information flow. In this paper, we assume that the central controller is executing periodically with a fixed determinstic period of duration $T_S$.

## 2.1 Communication network architecture

The end-to-end communication path between a sensor and the central controller consists of two commnication networks, which can be seen in Figure A.1, with conceptually

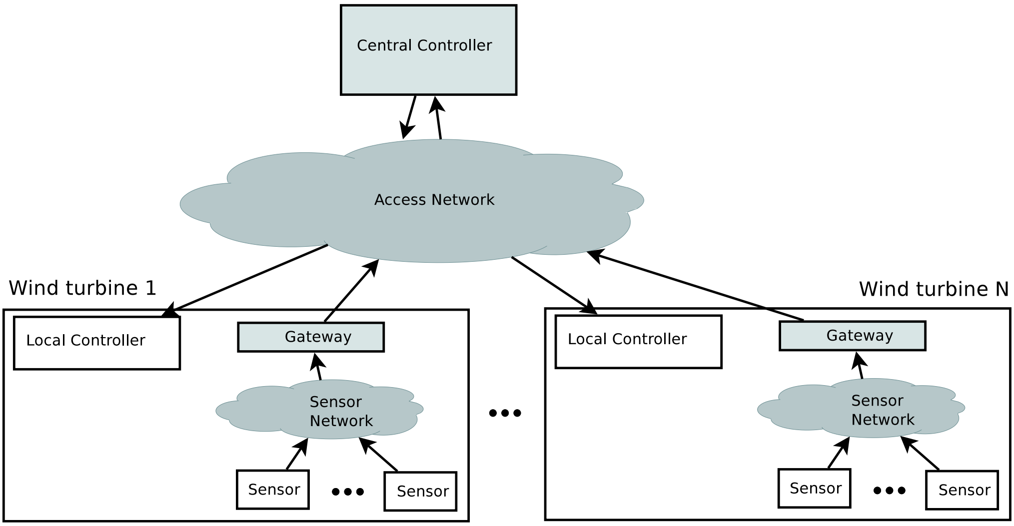



**Fig. A.1:** Overview of the communication architecture

different properties; a Sensor Network and an Access Network. The Sensor Network is a communication network internally on the wind turbine, and can be implemented by different communication technologies, for example a fieldbus. It handles the communication between the sensors on the wind turbine and contains a gateway that handles communication from the wind turbine to the central controller. The Access Network is the network between a wind turbine and the central controller, and varies in size based on the size of the wind farm. An off-shore windfarm can have lengths of over 20 km with hundreds of wind turbines, which must be covered by the given communication network. The network could be implemented by technologies such as 2G, 3G, WLAN, fiber optics, PLC or others.

Cellular is already deployed in many regions and hence can be used without further infrastructure. Whether 2G or 3G fulfils the requirements of this scenario is part of the analysis. WLAN is an off-the-self technology readily available and covering a few hundreds of meters which could support wind farms in a geographically smaller area. PLC in principal could be interesting, as the central controller and the wind turbine controller may be connected via power lines, which would mean that except the PLC modems no additional infrastructure is needed. However, unlike the hard-wired communication infrastructure, PLC and WiFi technologies can have additional drawbacks impacting their reliability. For instance, PLC communication is not possible if the power line is cut, while WiFi technology has a shared spectrum possibly reducing the performance. The fiber optics gives the best QoS, but may not be available to already deployed wind farms, and may require additional costs to implement.

Figure A.2 depicts a message sequence diagram of a control period with message delays. The figure shows that during a given control period the central controller will calculate a set-point for the wind turbine. The set-point will be sent to the wind turbine, which will then act upon it. At some point during the control period, defined by the value $T_o$, denoting the offset time prior to control computation $C_i$, the sensor will take a reading, and will send it via the sensor network to the gateway that forwards the sensor reading via the access network to the central controller. For simplicity, Figure A.2 only shows the communication endpoints; the delays indicated by the tilt of the messages are end-to-end delays.

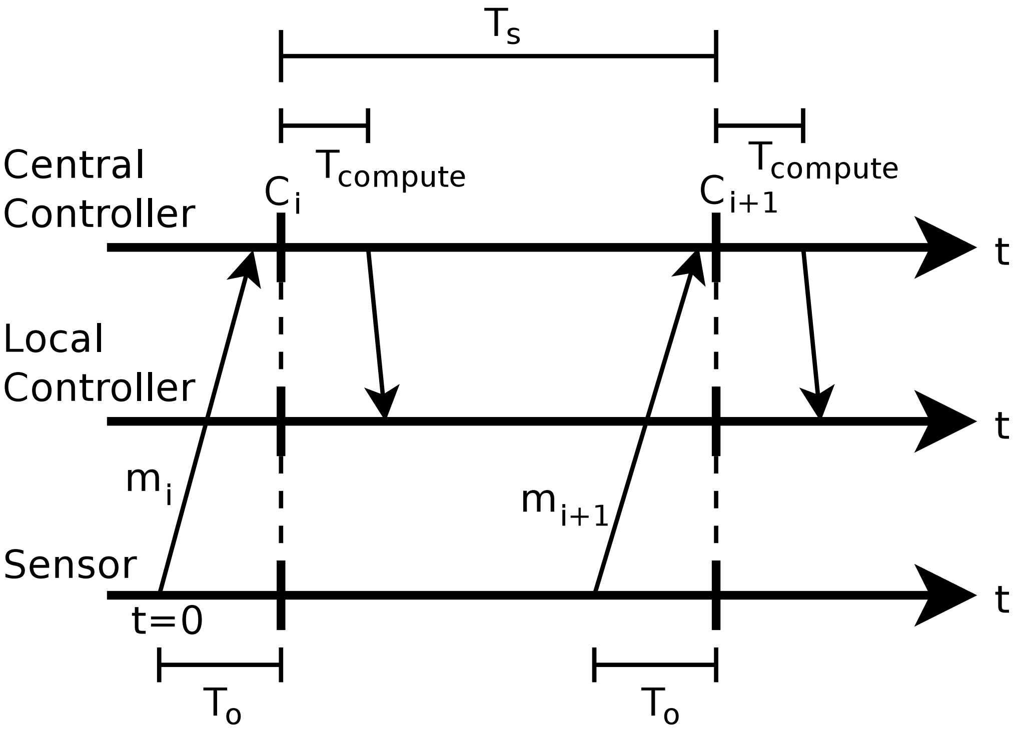



**Fig. A.2:** Message sequence diagram between the sensor and central controller for a single wind turbine

The information access strategy thus is that the controller sends one control set-point during each control period, and the sensors and the sensor sends a measurement once each control period, determined by the offset $T_o$, with sensor readings. The gateway will forward any packet it gets from the sensors immediately to the controller, and is thus not shown in this figure.

## 2.2 Controller design

The central controller aims to reduce the fatigue load of the shaft in the wind turbine. It does this while tracking a desired power reference. To achieve this the central controller requires an input from a fatigue estimator. This estimator requires sensor measurements of the wind turbine, which are the physical measurements of torque, as well as the angle rotation of the wind turbine. These measurements are run through different hysteresis relays with different weights. For details on estimator synthesis and the controller see [16].

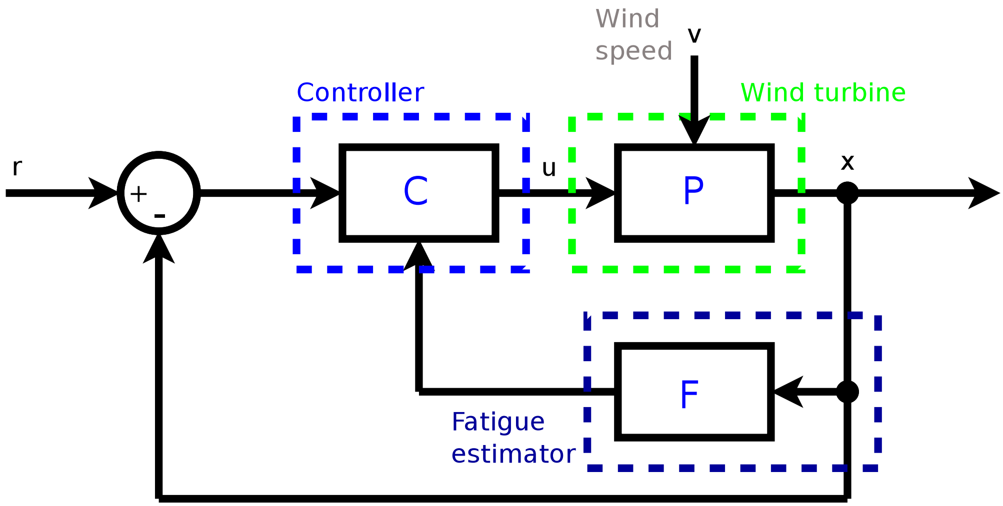



**Fig. A.3:** Centralized controller scheme without network effects

The control scheme without network effects is depicted in Figure A.3. In this paper we assume the fatigue estimator (F) and the controller (C) are co-located in a central controller. This controller describes the procedures for an individual wind turbine, while the same procedure, with coordination on the generation share of each wind turbine, is applied for the whole farm, which requires a centralized controller. For control performance, we later analyse a single wind turbine as we focus on the fatigue as the KPI describing control performance, while the communication scenario is shaped by the whole wind farm.

The controller cycle, $T_s$, is determined by the design of the controller, and is in this scenario set to $150ms$ and cannot easily be changed. The central controller is assumed to have a $50ms$ computation time. We consider a total of 30 wind turbines, each with 3 sensors, in the wind farm, which constitutes a small to medium sized wind farm.

# 3 Controller performance under communication network

We start by investigating if a delay in the communication network, and a specific access strategy, have an impact on the performance of the central controller. To do this a co-simulation framework has been developed, in which MATLAB is used to simulate the wind turbine controller, and OMNeT++ is used to simulate the communication network. An interface has been developed for the controller allowing it to interact with the OMNeT++ simulation. The OMNeT++ part of the simulation is in charge of timing requirements, i.e. when to simulate the controller, wind turbine or fatigue estimator, and is thus in charge of handling the information access strategies. In the simulations, time is handled by an internal clock, ensureing perfect synchronization between all modules in the simulation. The wind turbine is modelled by the differential equation linearised around a chosen operating point:

$$X(t+1) = A_d \cdot X(t) + B_d \cdot U(t) + E_d \cdot V(t) \tag{A.1}$$

where $X$ is the state of the wind turbine, $U$ is the control set-point and $V$ is the wind, which is real wind data. How these interact is determined by the matrices $A_d$, $B_d$ and $E_d$, and we refer to Reference [16] for details for determining these matrices. Here we let $t$ be the discretized time.

## 3.1 Scenario parameters

Some assumptions regarding the controller and its physical representation are taken in order to simplify the simulations. It is assumed that there will be end-to-end communication delays in the communication network due to propagation delay, queues and computation delays. We also assume that the controller and the sensors have perfect clock synchronization. In regards to the controller, it is assumed that both the fatigue estimator and the computation of the wind turbine set points have an internal computation delay, as summarized in the value given in Table A.1. This computation delay is used by the fatigue estimator to estimate the fatigue of the wind turbine based on the wind-turbine state, and by the controller to calculate a new set-point based on the output of the fatigue estimator. At any give time there is a change in the wind turbine state it is assumed that the sensors are able to get a reading of this instantly. An example of this could be a new control set point or a change in wind speed. The wind turbine itself is assumed to act on a set point as soon as it is received, and we assume that this is done instantly, and that the change has an impact on the state of the wind turbine instantly. The wind data which have been used in the simulations have been downloaded from Reference [17]. The wind data used have a measurement every 12.5$ms$, leading to 12 samples per control period.

Table A.1 details the relevant parameters for the simulation scenario.

| | |
|---|---|
| Controller cycle, $T_s$ | $150ms$ |
| Controller computation delay | $50ms$ |
| Number of wind turbines | 1 |
| Number of sensors | 3 |

**Table A.1:** Parameters on the overall wind-farm topology and parameters for the control execution

We start by investigating the performance of the central controller over increasing delays. We assumed a periodic access strategy as described in section 2.1, where the offset is set to $75ms$. As we are mainly interested in the impact of the non-ideal networks in the access to the sensor information, we assume for the downstream communication from central controller to wind turbine a constant $1ms$ delay. For the 'upstream' end-to-end communication between sensor and controller, we assume a shifted exponential distribution, with a fixed minimum of $1ms$ and we change the exponential rate in order to change the mean of the upstream delays.

Figure A.4 depicts the accumulated damage over increasing total mean delays, with the dashed lines depicting the 95 % confidence interval. It shows as the average delay increases, the accumulated damage increases, and thus the performance of the controller decreases. For the further analysis in this section, we focus the parameter regime of delays that are well below the control period duration of $150ms$. We therefore choose to use a shifted exponential distribution with a mean of $31ms$ in the remainder of this section. We next attempt to determine the effect of different offsets on the performance of the controller.

The results in Figure A.5 show that the accumulated damage of the wind turbine decreases as the offset increases, until it reaches a minimum at $T_o \approx 90ms$. A small choice of $T_o$ has as consequence that a large fraction of the update message from the sensors does not arrive in time to be considered for the control computation. As a consequence, the controller would use the 'old' value from the last received update message as basis of its control computation. The poor values of control performance to the left of Figure A.5 show that such missed updates have a negative impact on control performance. After $100ms$, the accumulated damage drastically increases up to a value close to that of the lower offset values. The reason for this sudden increase is to be found in the central controller computation time of 50 ms. This means if the offset is higher than 100 ms, the newest control set-point has not yet been received on the wind turbine. The arrival of the set-point causes a noticeable change in the sensor data, which means that the data sent right before and right after the arrival of the set-point are noticeably different as well.

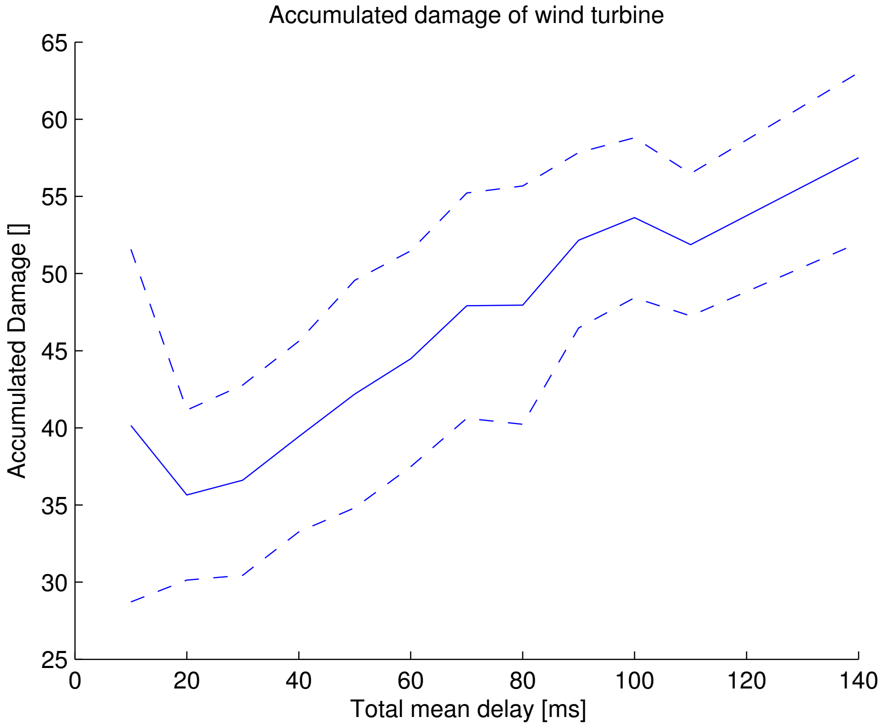



**Fig. A.4:** The accumulated damage of the wind turbine over increasing delays with 95 % confidence intervals, the X-axis denoting the mean of the upstream end-to-end delays between sensors and central controller

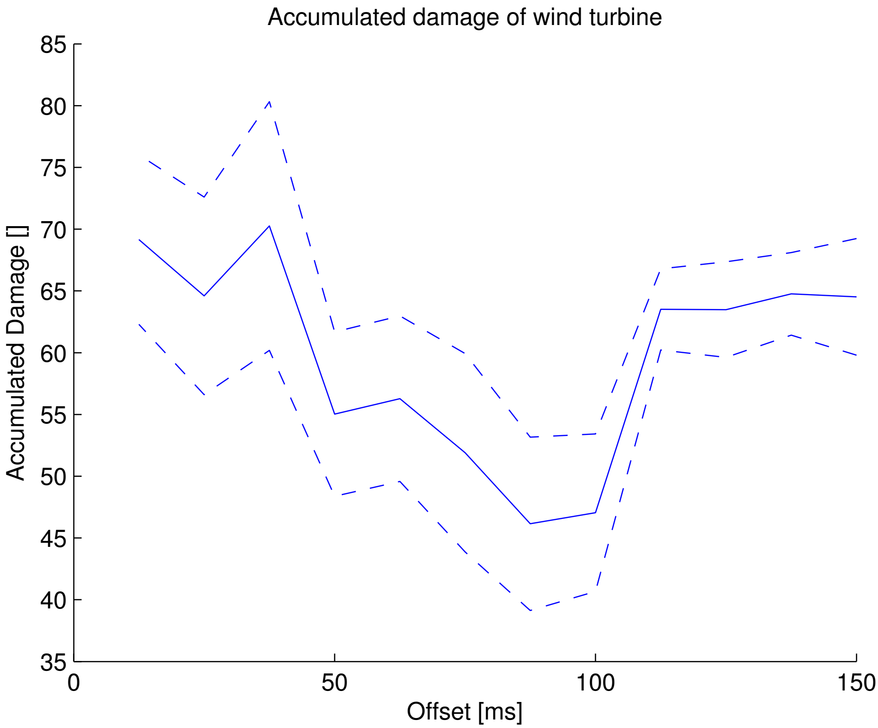



**Fig. A.5:** Accumulated damage of the wind turbine with varying offset with 95 % confidence intervals

## 3.2 Performance results of heuristic cases

As performance metric we consider the accumulated damage of the wind turbine. The higher this value is, the worse the performance of the controller is. How these values are determined is explained in detail in Reference [16]. Since we discovered in Section 3.1 that too small a choice of offset may negatively affect control performance, since a larger fraction of update message from the sensor do not arrive in time at the controller, one way of heuristically choosing an optimal offset is by taking a $p\%$ percentile of the end-2-end delay distribution between sensor and central controller. We investigate such heuristic in the following for percentiles of 80 %, 90 %, 95 % and 99 % Given the shifted exponential end-to-end delay distributions, these percentiles map into values $T_o = 48ms, 69ms, 90ms$ and $138ms$. We compare these heuristics with a baseline case where the offset is chosen, for each control period, at random from a uniform distribution between $0ms$ and $150ms$. The results can be seen in Figure A.6, which shows the performance for each of the cases.

The values for the controller performance, as well as the 95% confidence interval, found from 10 repetitions of 400 control cycles each, can be found in Table A.2. The

| Case name | Accumulated damage | Case number |
|---|---|---|
| Baseline case | 73.1 ± 7.4 | 1 |
| Heuristic 99% | 64.5 ± 4.7 | 2 |
| Heuristic 95% | 47 ± 6.4 | 3 |
| Heuristic 90% | 46.1 ± 7 | 4 |
| Heuristic 80% | 55 ± 6.6 | 5 |

**Table A.2:** The accumulated damage of the six different cases, showing accumulated damage and 95 % confidence interval

results show that performance-wise the baseline case and the heuristic 99 % case are clearly worse than the heuristic 95 % and 90 %, with the heuristic 80 % case being in between. We see that the heuristic cases 95 %, 90 % and 80 % have overlapping confidence intervals, and we can as such not say which one is the better option. There is however still the problem of choosing the best heuristic case as the heuristic is parameterized with the percentile $p$ of the delay distribution. Obviously, $p$ can be obtained by a detailed simulation of the control performance as done in Figure A.5. However, we would in that case not need a heuristic any more, as the minimum can be read off from Figure A.5 right away. Therefore, we will in the next section search for computationally less costly ways of optimizing $T_o$ via a model-based approach.

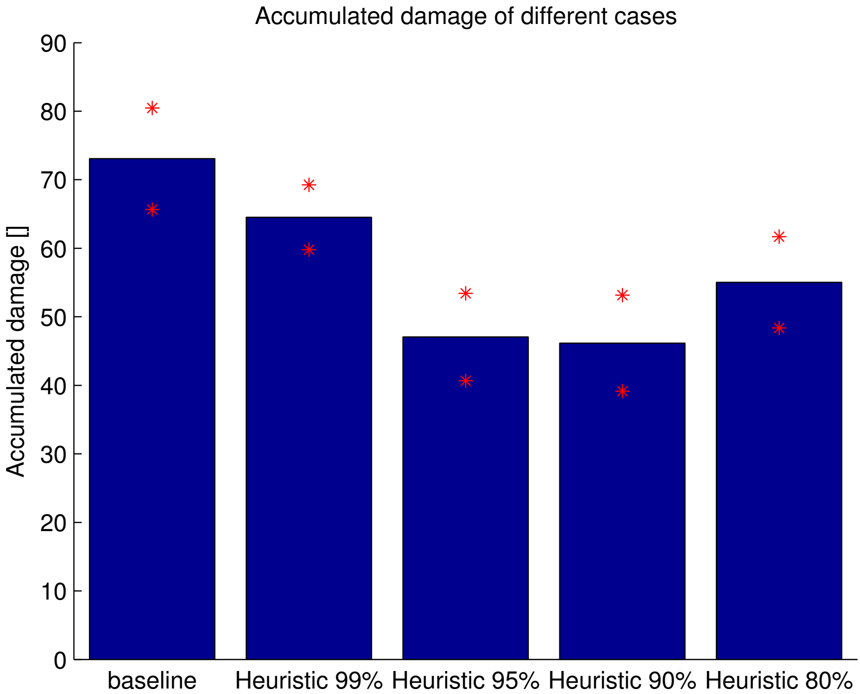



**Fig. A.6:** Controller performance in terms of accumulated damage of respectively the baseline case and the heuristic cases

# 4 Model-based offset optimzation

As it was shown in the previous section, the offset of the sensor information update should be chosen in a way that (1) the offset is large enough such that the probability that the update message is received at the controller before it starts its control computation is increased while (2) the offset should be small enough so that changes of the sensor values after generating the update message are not impacting control performance. As the two criteria are in trade-off, a model will be needed to balance this trade-off. The first aspect is thereby mainly influenced by the end-2-end delay distribution, while the second aspect is influenced by the information dynamics of the sensor information. An analytic model for the controller performance is however very complex, hence the approach is to instead optimize the information quality, defined by the match between the controller's view of the system at the start of the control computation in comparison to the ground truth system state in that time instance.

## 4.1 Information quality metric

Mismatch probability (mmPr), as defined in Reference [10], is a metric which fulfils our requirements for an information quality metric. Let us assume $X_s(t)$ is the state of the sensors's information and $X_c(t)$ is the state of the sensor known to the controller. $X_c(t)$ is updated through the communication network utilizing the periodic update messages generated at offset $T_o$ before the control computation start. The mmPr is defined for the time instances, $C_i$, see Figure A.2, at which the controller starts its computation of the set-points:

$$mmPr(\epsilon) := Pr(|X_s(C_i) - X_C(C_i)| > \epsilon). \tag{A.2}$$

The parameter $\epsilon$ here corresponds to some 'discretization' threshold for continuous sensor value domains; changes of the continuous sensor value are only counted if they exceed this threshold $\epsilon$. The mmPr metric is mainly influenced by the following factors: (1) the freshness of information at the time the controller is calculating new set-points, (2) how quickly information changes to a mismatching value, i.e. the information dynamics. Note, that the information freshness depends on the network delay, the data access strategy, as well as sensor sampling frequency. The sensor dynamics is modelled with a discrete time Markov chain (DTMC), since the mathematical framework for handling DTMCs is well known and suitable for the mmPr calculations. The following subsection describes how sensor values are used for DTMC fitting and defines analytical models for *mmPr* calculations.

## 4.2 Markov Model of the Controlled System

We approximate the controlled system by a discrete time Markov model; we thereby simplify to assume the state of the system is described by the time series of one of the sensors, denoted by $S_1, S_2, ...$; unless mentioned differently, we assume these values represent Sensor 3 in the results shown later on. The continuous values $S_i$ are thereby mapped in an equidistant manner to the discrete states $1, ..., N$ of the Markov model. When analyzing a simulated trace of $S_i$, it shows a different behavior in time-steps when new set-points are communicated as opposed to time steps when no action of the central controller is imposed. Therefore we use two different transition matrices, $P$ and $C$, to describe the state transition probabilities in intervals without and with central control action, respectively. The matrices $P$ and $C$ can be easily obtained from the observed transition counts from the simulated system as the exact moments of set-point adaptations are known. Note that the model is obviously a simplification as it does not consider the values of the set-points obtained by the controller.

Too avoid too many technicalities, we here assume that the computation delay and the downstream delay from the central controller to the local controller are determinstic. Due to the periodic nature of the controller, the system behavior as a Markov chain within one control cycle is then determined by the transition matrix $M = C \cdot P^{c-1}$, where $c$ is the number of discrete time steps in one control cycle. The eigenvector $\pi = \pi \cdot M$ for eigenvalue 1 of the matrix $M$ is then the steady state probability distribution of the Markov chain, when observed just before the control action implementation in each cycle.

## 4.3 Analytic model for the mismatch probability

Assuming that the controlled system is adequately described by the Markov model in the previous section, the information quality metric, mismatch probability from Section 4.1, can be calculated from the markov model and from the cumulative distribution function, $F_D(t)$, of the upstream end-2-end delay as follows:

$$1 - mmPr = F_D(T_o) \sum_{k=1}^{N} \pi_u(k) \cdot Pr(\text{Markov chain is in state } k \text{ at time instant } C_i \mid \text{Markov chain has been in state } k \text{ at time instant } C_i - T_o) + (1 - F_D(t_o))F_D(T_o + T_c) \sum_{k=1}^{N} \pi_u(k) \cdot Pr(\text{Markov chain is in state } k \text{ at time instant } C_i \mid \text{Markov chain has been in state } k \text{ at time instant } C_i - T_o - T_c)0 \quad \text{(A.3)}$$

Thereby, $\pi_u$ is the steady state probability vector of the Markov model at the moment just before the updates are sent by the sensors. The first term considers the case that the last update message reaches the controller in time before the start of the control computation; the second term considers the case that the last update does not arrive in time, but the previous update does. The above equation yields an upper bound for the mismatch probability as other cases, $n$ last updates do not arrive in time, but $n+1$th does, for $n >= 2$ are not considered in detail, but all these cases are mapped into a mismatch (hence the resulting upper bound).
When implementing Equation A.3, the state $\pi_u(k)$ and the detailed composition of the conditional probabilities (which basically are the diagonal elements of products of the form $P^i \cdot C^j \cdot P^k$) depend on the cycle duration, the offset $T_o$ and the computation and downstream delay. Therefore, the Markov transition probabilities in the sums can be obtained from adequate products of the matrices $P$ and $C$, we skip the technical details here.

## 4.4 Offset optimization

The above analytic model can be used to obtain optimal offset choices by simple numeric minimization with respect to $T_o$. Figure A.7 shows the result of the analytic mmPr model from the previous section, when applied to the scenario analyzed in Section 3; the corresponding control performance graph is in Figure A.5 and showed a minimum around an offset choice of $90ms$. The Markov model with different state-sizes $N = 2, ..., 12$ is fitted to a trace of Sensor 3 that has been obtained from a closed-loop control simulation with ideal network. The analytic mismatch metric caclulations show a minimum (marked by a cross) around an offset value of 50 ms (when using N=8 states or less) and this minimum shifts to 37.5 ms for N=9,...,12 states. The shift of the minimum to the left can be intuitively explained as follows: The more states are used in the Markov model, the more the mismatch probability is influenced by small changes in the sensor information, so the mismatch metric become more sensitive to information age. With smaller offset, the information at the controller is fresher given that it arrives in time. So this gain in freshness here seems to overcompensate the effect of late arrival of messages for some range of offsets. As Figure A.7 shows, the absolute value of the mismatch metric depends on the number of states in the Markov model; however the benefit of the model is not the absolute value, but that it allows to determine an optimal offset.

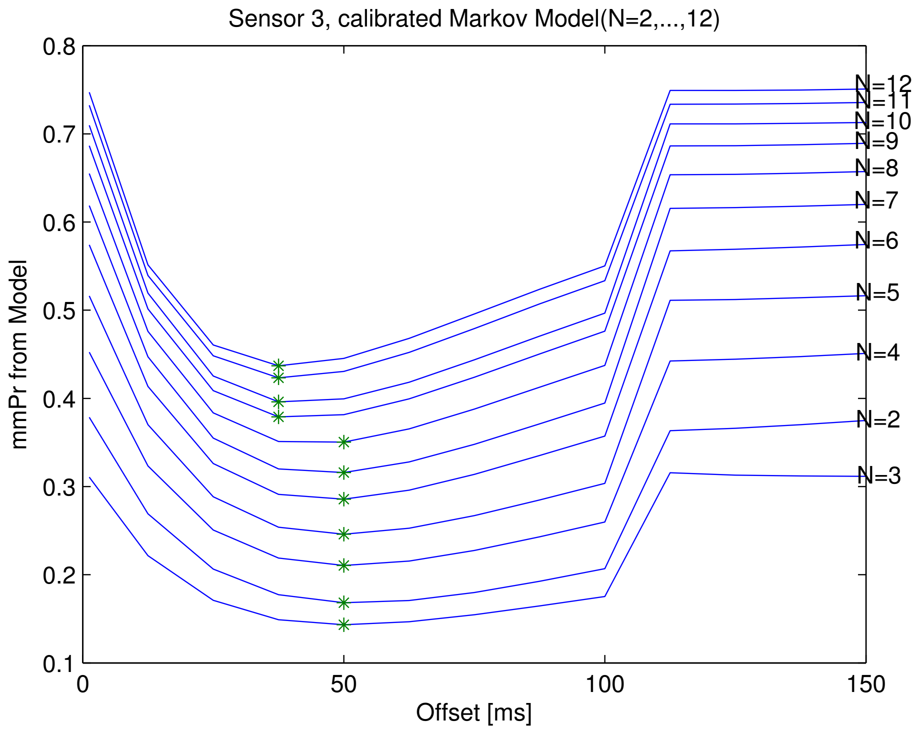



**Fig. A.7:** mmPr of sensor 3 for different number of Markov States

## 4.5 Sensor mmPr simulation results

Figure A.8 shows the mmPr of sensor 3 for different offsets found both via the full co-simulation (solid blue curve) and from the model of the sensor information dynamics (dashed red and green curve). All curves are simulated using the same scenario parameters, as described in Section 3.1. The red curve is found from a Markov chain containing

$N = 5$ states, while the green curve contains $N = 4$ states. These are chosen as they are the number of states that match the blue curve most closely, both in mmPr values and in optimal offset.

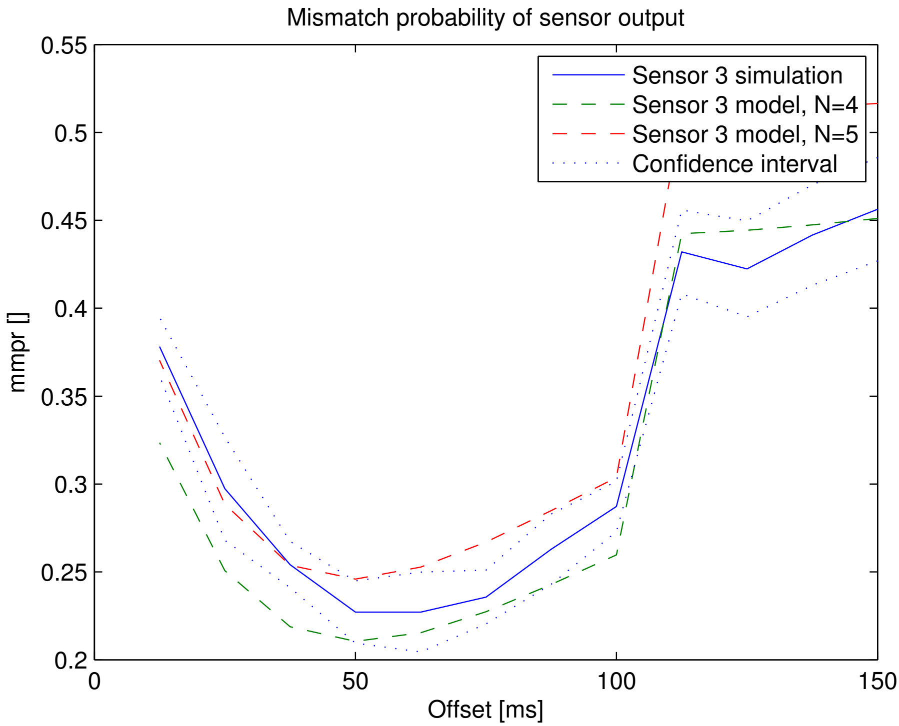



**Fig. A.8:** mmPr of Sensor 3 for different offsets: the solid curves shows the simulation result together with the 95 % confidence interval (dotted), the two dashed curves show the model-based result for Markov models with N=4 and N=5 states

Note that the simulation of the mmPr also takes a discretized version of the sensor space as basis, i.e. $mmPr(\epsilon) := Pr(|X_s(C_i) - X_C(C_i)| > \epsilon)$. A smaller discretization level $\epsilon$, would lead to higher mismatch probabilities. The simulation takes the whole value space of all sensor values for all offsets and uses an $\epsilon$ which is $1/20$ of this range. Note that this does not correspond to a Markov chain with 20 states, as the Markov chain is fitted to only one simulation run (with an ideal network) and hence the value range is smaller for that fit. It turns out that a Markov chain with between $N = 4$ and $N = 5$ states has the similar discretization level as assumed for the simulated mmPr.

The control performance that is analyzed in Figure A.5 is however influenced by the information quality of all three sensors. Therefore Figure A.9 shows the average

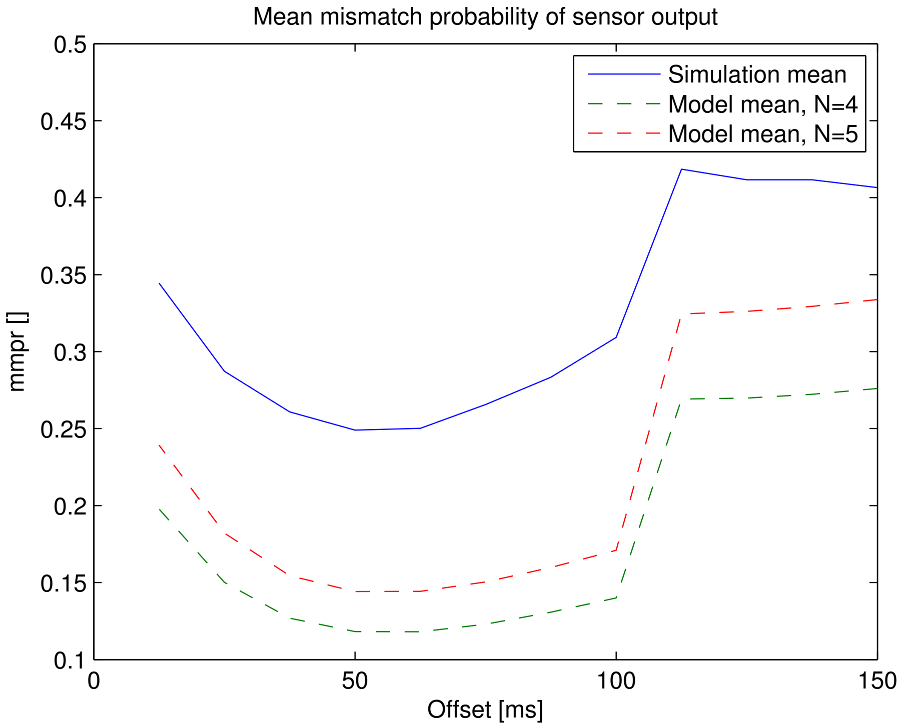



**Fig. A.9:** mean mmPr of all sensors for different offsets: the solid curves shows the simulation result, the two dashed curves show the model-based result for Markov models with N=4 and N=5 states mismatch probability, where the average is taken over the simulated and model-based values for all 3 sensors when using the same offset. Note however, that different sensors with different dynamics and different end-2-end delays could also use different offset values. In this paper however, we restrict ourselves to the analysis of the case when the same offset is applied to all three sensors. The number of states is the same for each sensor, and these values are chosen because they matched the curve on the previous figure. We see that while the curves still match in terms of optimal offset, there is a distinct difference in the mmPr values that each curve take. The optimal offset for $N = 4$ and $N = 5$ in Figure A.9 is in both cases at $62.5ms$.

Figure A.10 compares the accumulated damage of the wind turbine of the offset chosen from the model and the offset found via heuristic methods. Cases 1 to 5, as described in Table A.2, are also shown in Figure A.6, Case 6 (yellow colour) denotes the offset found via the mean mmPr of all 3 sensors, while for case 7 (green colour) the offset is found based on the mmPr of sensor 3, in both cases for $N = 5$. We see that for the mean case the accumulated damage is slightly lower, however the two cases are still within each others confidence interval.

# 5 QoS parameters of communication technologies

We have in the previous section made assumptions regarding the end-to-end delay, which was modeled by a shifted exponential distribution with a mean of 31ms. However, for different communication technologies in the Access and Sensor Networks, the delays experienced may vary, and also other types of delay distributions as well as correlations between subsequent delay values may occur. In this section we investigate the delays of different communication technologies, and afterwards use these delay traces to determine the end-2-end delay in the co-simulations.
The aim of the measurement setup for the Access Network was to mimic the communication pattern generated between the central controller and 30 gateways on the wind farm communication network in order to characterize the network properties in terms of delays and packet losses. For each of the technologies ICMP ping measurements are performed with 64 bytes of application layer payload in the uplink (i.e. for the sensor measurements) and downlink (i.e. for the control commands) roughly corresponding to an actual size of the messages [18]. In further subsections, details of measurement setups for each of the technologies are given while Section 5.4 describes the measurement setup for the Sensor Network.

## 5.1 2G/3G measurement setup

The test-bed setup for conducting cellular 2G/3G measurements is comprised of three Huawei E392 USB modems [19] equipped with SIM cards from Austrian telecom provider YESSS. Three SIM cards connected to the modems are provisioned to allow both GPRS

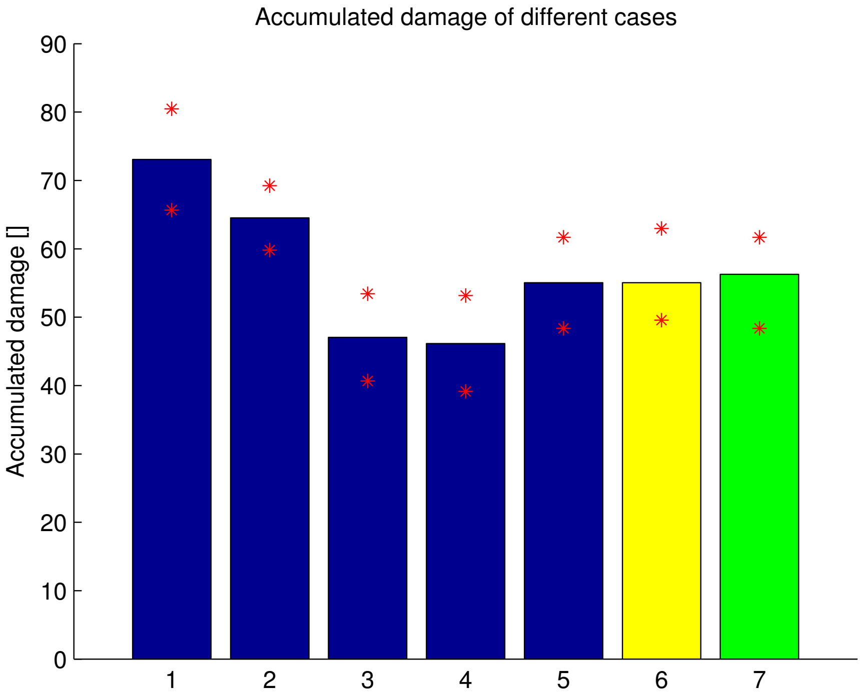



**Fig. A.10:** Controller performance in terms of accumulated damage of respectively the baseline case, the heuristic cases and the model cases.

and UMTS transmission access. Furthermore, the equipment is placed indoors, in the same room on the second floor of an eight floor office building. First, the modems are configured to have access only to a GPRS network. Through each modem ten ICMP ping connections were executed in parallel with a time interval of 150ms between the probes, thus 30 connections in total were running in order to mimic the communication pattern between the gateway and the central controller. Subsequently, the same procedure is repeated for the 3G network by configuring the modems for UMTS transmissions. In both cases, around 4000 measurements are collected from each ICMP ping connection, during a week day between 11am and 12am. The resulting distribution of the measurements for 2G and 3G respectively can be seen in Figure A.14 and A.11. The distance between the modems and GPRS/UMTS base transceiver stations (BTS) is around 30m, since the stations are located on top of the same building.

## 5.2 PLC measurement setup

Powerline communication technology is evaluated with narrow-band Devolo G3-PLC 500k PLC modems [20], having a total throughput capacity of 240 kbps. Five such PLC modems are connected via power-lines of approximately 1 meter length in total, which is much shorter than an actual implementation where they would potentially be hundreds of meters long, making this a best case scenario. In an initial test, three parallel ICMP ping connections are ran from three different PLC modems (9 connections on the PLC network), with 150ms time interval. It is shown later that for nine connections running over a PLC network, the network performance is impractical for the wind farm controller due to high delays and packet losses even in this best case setup with short distances.

## 5.3 WiFi measurement setup

WiFi technologies are tested using two commercially available embedded boards "ALIX 3D2" [21], which allow different IEEE 802.11 hardware modules to be added onto the system. For this study, we connected the 802.11g (2.4 GHz band) modules of type CM9-GP Wistron [22] onto the boards. The WiFi points are placed indoors within the same room, on the distance of approximately 3 meters, line-of-sight, which again is shorter than an actual implementation, making this a best case setup. In the vicinity of the WiFi points 23 other networks were scanned, 16 of them were running on 2.4GHz band, thus adding some interference to the signal; the transmission power is set to 15 dBm. Over such WiFi link, 30 parallel ICMP ping connections were executed during a work day between 3pm and 5pm, with the same interval as before. The resulting delay distribution observed on the link is shown in Figure A.12.

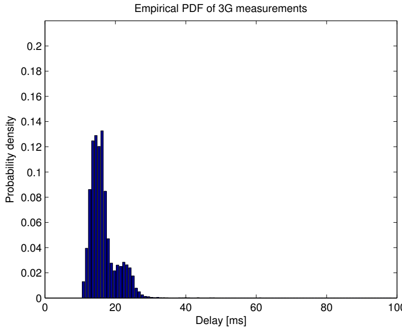



**Fig. A.11:** PDF of the 3G RTT measurement results

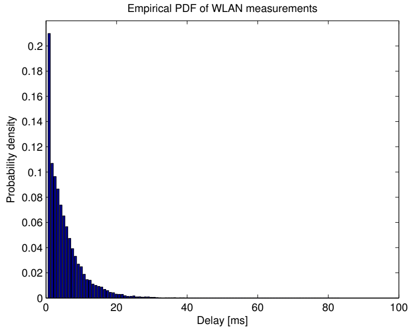



**Fig. A.12:** PDF of the WLAN RTT measurement results

## 5.4 CANBUS simulation setup

The CANBUS is setup with 3 sensors and 1 gateway connected to the CANBUS, with the gateway having the highest priority, then sensor 1 followed by sensor 2 and 3 respectively. The CANBUS is transmitting with 500 kbps, and 0 % packetloss on the connection. There is no other modules connected to the BUS to generate cross-traffic. Each sensor is set to transmit at the same offset $T_o$ once each control period (every $150ms$). The CANBUS is implemented in OMNeT++ and simulated.

## 5.5 Delay Results

As explained previously the delays measured on the different technologies, and thus the results shown in this section, are all RTT delays, and can be seen in Figure A.11 to Figure A.14, and in Table A.3. The results in Table A.4 are, however, one-way delays. As can be seen in Figure A.11 there are two peaks to the 3G measurements. The first peak is at roughly 15 ms, and the other, smaller, peak is at 20-22 ms. We have outliers of the data as high as 98.5 ms, which can be caused by spikes in traffic on the base station. In contrast to the measured delay distribution for 3G, the delays of WLAN in Figure A.12 by shape appear like an exponential distribution with a small shift. Similar results, delay wise, can be seen in [5], however with a slight different distribution, most likely to be attributed to cross-traffic.

From the 30 2G connections we take the data from a single connection on sim-card 2 and plot it as a time series in Figure A.13. The measurements of the other connections show the same trend as we see in Figure A.13. We see low delays at start of the measurements, which may be caused by not all sim cards and/or connections being setup right away. We see a few spikes in the delays between 0 and up until sample number 4250, which can be caused by spikes in traffic on the base station. After measurement 4250 we start to see a steady increase in delays, most likely caused by increased traffic leading to longer queues on the base station and in the sim-cards. For

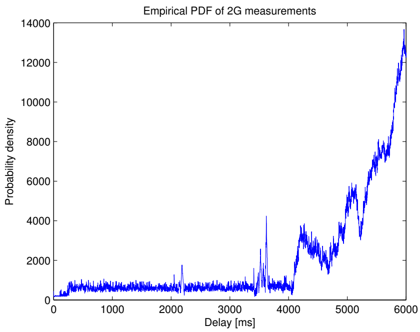



**Fig. A.13:** Time series of the 2G RTT measurement results

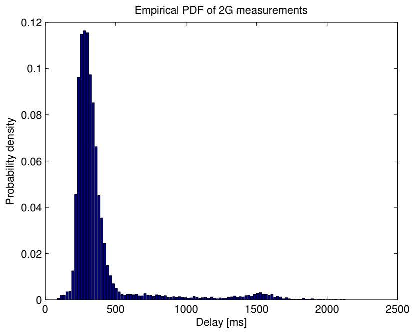



**Fig. A.14:** PDF of the the 2G RTT measurement results

the evaluation later, we only use the samples from 250 to 4250, in order to avoid the large instability period that occurs after. The subsequent table and Figure A.14 show the distribution of these 4000 samples.

Results from the PLC measurements show that the average delay is several control cycles long, and the packet loss is 78 %, even for only 9 parallel connections. This is rather problematic for the controller, as it will lead to instability and the controller being unable to generate set-points. For this reason these results are not used further in this paper, as it is clear from the statistics that the solution is not feasible. The statistics for all the measurements results can be seen in Table A.3 below.

| Technology | Minimum delay | Average delay | Maximum delay | Packet loss |
|---|---|---|---|---|
| 3G | 10.5 ms | 16.7 ms | 98.5 ms | 0 % |
| WLAN | 0.5 ms | 5.4 ms | 83.2 ms | 0 % |
| 2G | 84.5 ms | 385.2 ms | 2131 ms | 0 % |
| PLC | 50.4 ms | 1504 ms | 3400.8 ms | 78 % |

**Table A.3:** QoS results of the RTT measurements

The delays from the CANBUS simulation show deterministic delays for each sensor, with sensor 1 having the lowest delay as it has the highest priority. Each delay is the one-way delay and there were no packet losses experienced. The delay values are summarized in Table A.4.

The results of the measurements are used as a time series in the co-simulation framework in order to determine the delays of individual messages. The measurements we have access to are, as explained above, RTT ping messages, except for the CANBUS

| Sensor | Delay |
|---|---|
| 1 | 5 ms |
| 2 | 10 ms |
| 3 | 15 ms |

**Table A.4:** Simulation results for CANBUS one-way delay

delays. The delays we are interested in are, however, one-way delays. Therefore we assume that the uplink and downlink channel properties are identical and we can thus divide the RTTs by 2 to get one-way delays. This represents an approximation as many technologies differentiate in their handling of uplink and downlink data streams.

# 6 Wind-farm control using cellular communication technologies and CANBUS

In this section we use the delay traces found in Section 5 to create different simulation scenarios. To start we create scenarios where only the Access Network delays are used, and we simulate using the traces of the technologies for this network. We then create a combination scenario, where we use both the traces from the 3G measurements, as well as the delays incured by the CANBUS in the Sensor Network. This scenario thus entails both networks, creating a simulation of the entire system. The performance of the controller is described by two metrics: reference tracking and accumulated damage of the wind turbine. Analog to Section 3, we focus on the second metric, the accumulated damage. The simulations are done using the same framework as described in Section 3, where the end-to-end delays both in uplink from sensor to the central controller and in downlink from the central controller to the local controller are obtained now by the traces from the technology measurements in Section 5.

## 6.1 Control Performance over Cellular Access Technologies

The accumulated damage for the 3G trace (Figure A.15) shows an increase for low offsets when offsets get smaller than $\approx 40ms$, and a sharp increase at an offset of $100ms$. Both of these increases are expected, as at $100ms$ offset the computational delay of the central controller mean that the control set-point is not yet received. At low offsets there is a higher chance that the messages will be delayed past the control point $C_i$, and thus old data will be used. The optimal offset is between $37.5ms$ and $100ms$ where no statistically significant differences between those offset choices can be observed.

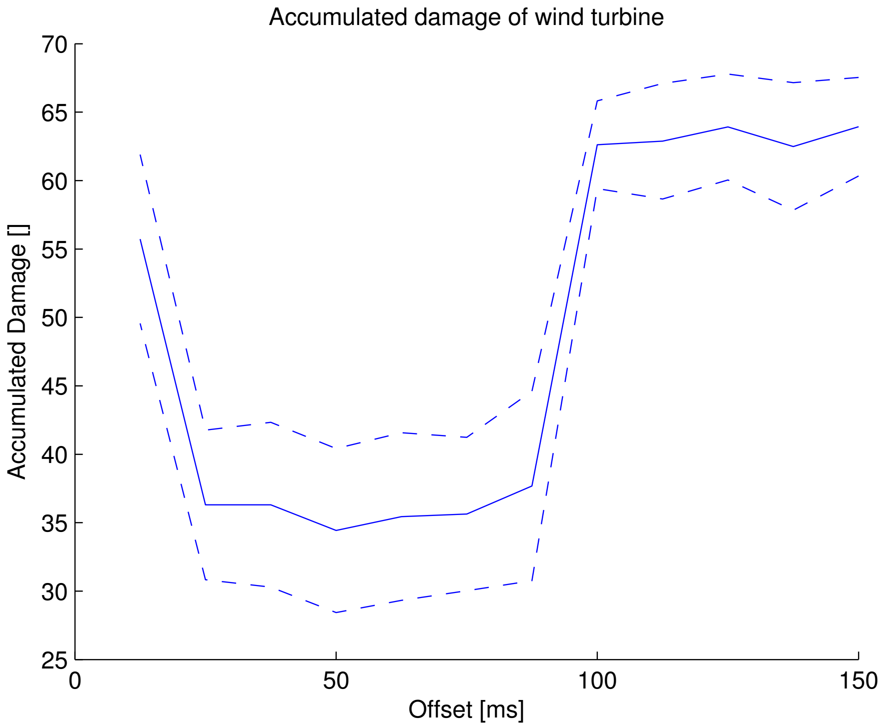



**Fig. A.15:** Accumulated damage of wind turbine based on 3G measurements with 95 % confidence intervals

Figure A.16 shows the control performance when using the 2G data that was the basis for the pdf in Figure A.14. As 2G delays are generally rather long, on average more than 1 control cycles, the controller will in this case most of the time use sensor measurements which are several control cycles old. Figure A.16 shows that in this case the control performance is about twice as bad as in the better 3G cases; furthermore, the detailed choice of the offset within the control period is irrelevant as anyways the sensor measurement will be outdated by the time it arrives at the controller. Therefore, there is no offset optimization benefitial for this 2G case given the short control periods of the central controller.

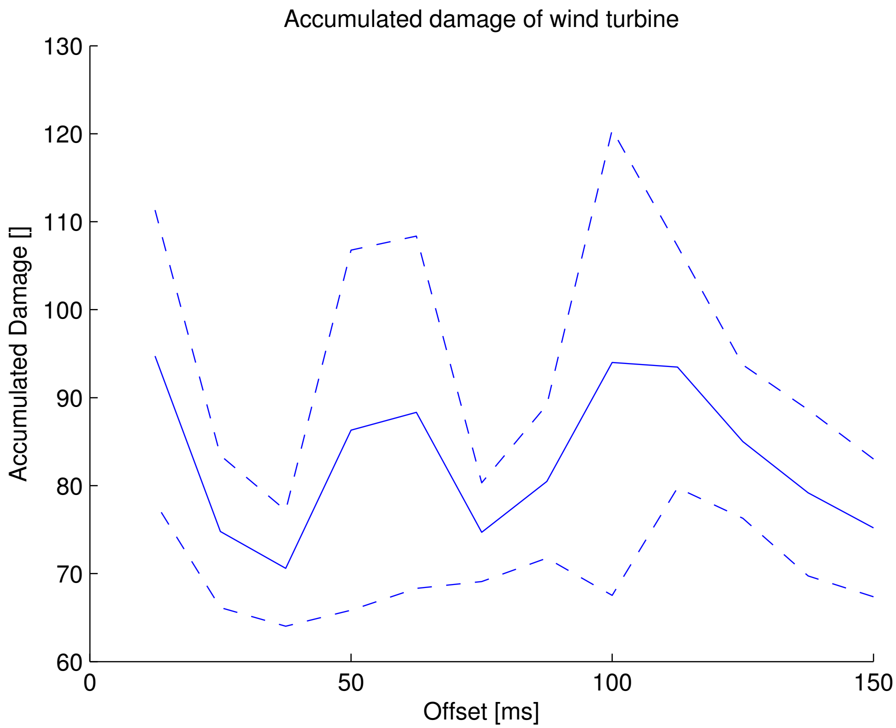



**Fig. A.16:** Accumulated damage of wind turbine based on 2G measurements with 95 % confidence intervals

For comparison to the other technologies a simulation with an ideal network, meaning a network with 0 ms delays and 0 % packet loss, has also been executed. The result of this can be seen in Figure A.17. This shows a relatively flat curve until an offset of 100 ms, at which point we again see an increase in accumulated damage, as expected based on previous simulation results. We have not included the figure for accumulated damage for the WLAN technology in this paper, as the results are very similar to the results for the ideal network. The biggest difference was in a small increase at offsets below $50ms$, however this increase was so slight to be within the 95 % confidence interval of the ideal network simulation.

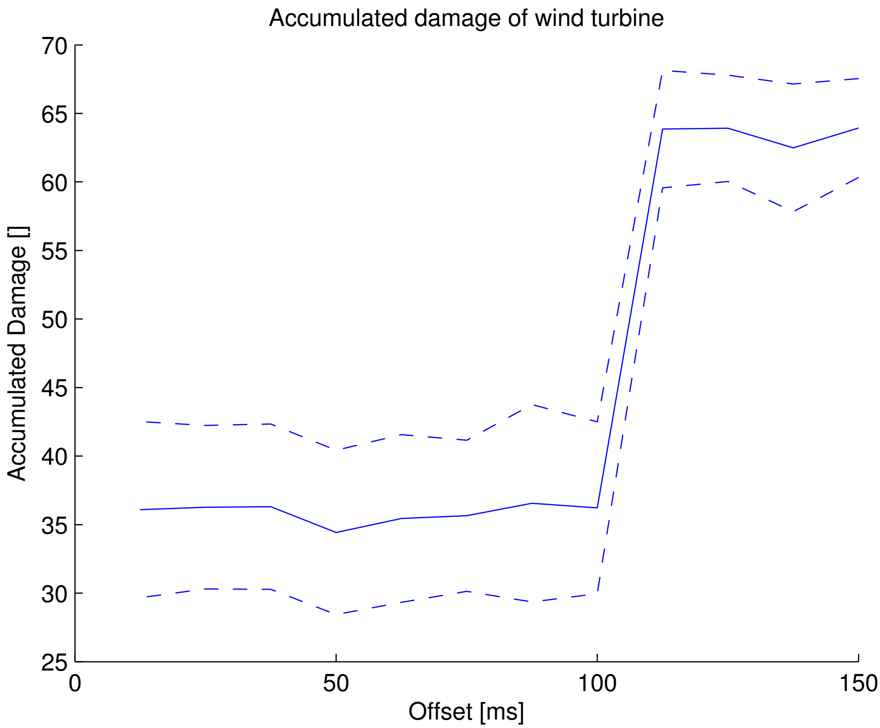



**Fig. A.17:** Accumulated damage of wind turbine based on ideal network

## 6.2 Combined 3G and CANBUS scenario

Finally, we compare the analytic model in a scenario that mimics the use of 3G access for connections to the wind turbines and a CAN network for the sensor data access with delays as stated in the previous sections. Figure A.18 shows a simulation run where the 3G trace is used, as in Figure A.15, and the deterministic delay of the CANBUS is added to the delay between the sensors and the central controller. The figure shows a sharper drop in the accumulated damage than for the pure 3G simulation, yet is otherwise very similar within the 95 % confidence intervals.

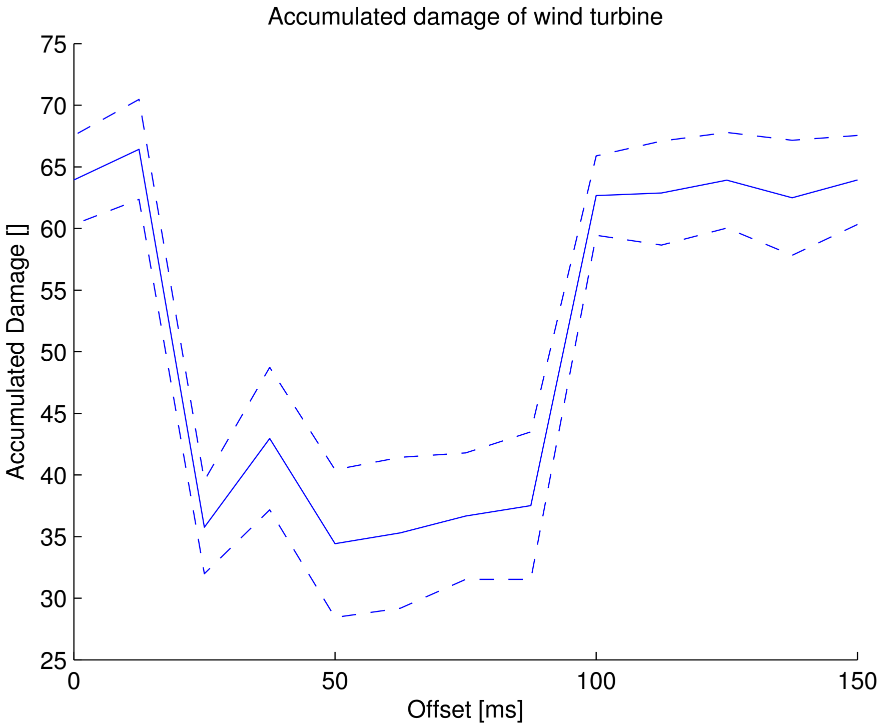



**Fig. A.18:** Performance of controller under combined 3G traces and CANBUS delays

In order to analyze the impact of the downstream delay from the central controller to the wind turbine, another simulation run like in Figure A.18 has been conducted, but now with a determinstic downstream delay with the same mean ($8.35ms$). The resulting control performance curve did not show any visual difference and any deviations were minor compared to the confidence intervals. As a conclusion, the determinstic assumption for the dowstream delay, as taken for the analytic model, does not change the control behavior in this scenario. We also ran a simulation of the scenario in Figure A.18, where the correlation in the UMTS delays used in the upstream (from sensor to controller) was destroyed by scrambling the delay trace. Also in this case, the control performance did not significantly change. Hence, the assumption of a renewal process for the upstream delays in the analytic model does not have any impact for this scenario.

As the analytic model assumes a shifted exponential delay distribution for the access to the sensor data and a deterministic downstream delay, and since we want to see the differences caused by other model assumptions, we however do not use the trace of the 3G delay measurements, but rather fit a shifted exponential distribution to the parameters of Table A.3; and analogously we use also in the simulation a deterministic downstream delay with the parameter matching the mean 3G delay. We call this the exponential approximation of the combined 3G and CANBUS scenario.

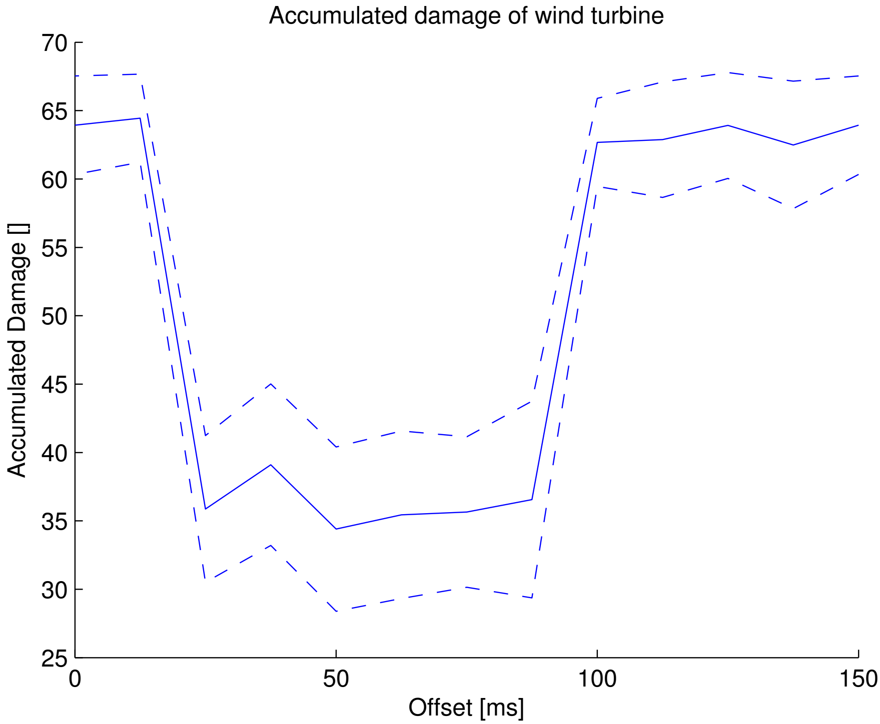



**Fig. A.19:** Performance of controller using deterministic downstream delay, and a shifted exponential fitted to the 3G traces

Figure A.19 shows the accumulated damage as resulting from such a simulation run with varying offset $T_o$ for the sensor information access. Figure A.19 shows a clear minimum for offsets between approximately $25ms$ and $87.5ms$, taking the 95 % confidence intervals (marked dashed) into account. Figure A.18 and Figure A.19 show very similar performance, the main difference being the accumulated damage at $T_o = 37.5ms$, with Figure A.19 having higher accumulated damage, however still within the confidence intervals of each other. The relative flat curves for offset between $50ms$ and $87.5ms$ is due to the rather low variability of the 3G delays.

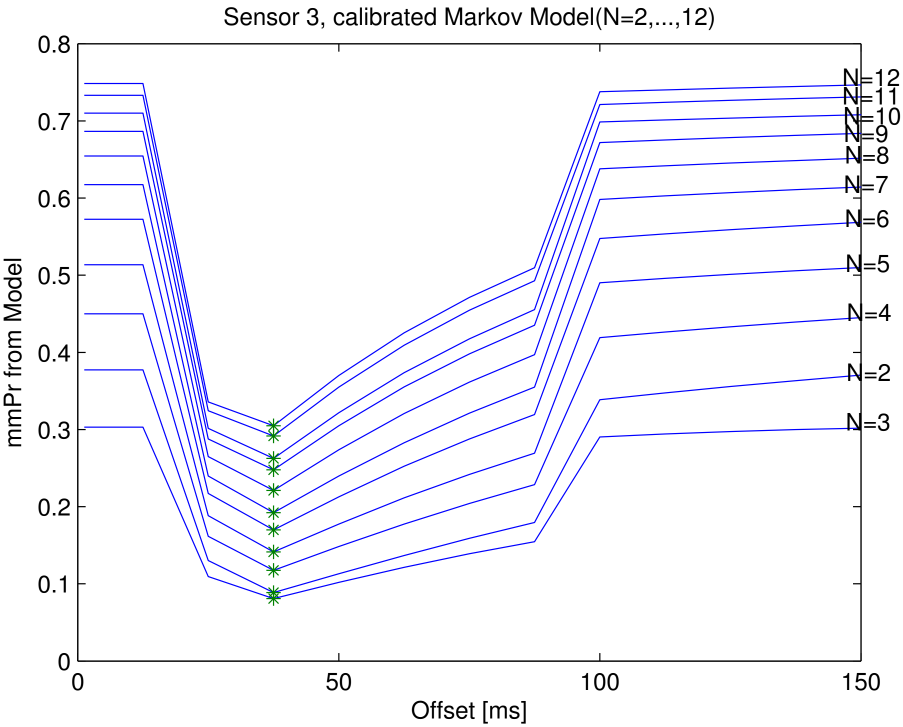



**Fig. A.20:** mmPr over changing offsets for sensor 3 in the exponential approximation of the combined 3G and CANBUS scenario

Figure A.20 now takes the Markov models for Sensor 3 (with the same calibration as earlier, as this calibration is independent of network delays) of sizes $N = 2, ..., 12$, and applies the mmPr calculations to these Markov models. The curves show minima around $40ms$ while, for small $N$, they are rather flat in the interval $40ms \leq T_o \leq 90ms$.

Control performance of the system simulated in Figure A.19 is however influenced by all 3 sensors. In the CAN setting, the 3 sensors have different CAN delays. The analytic model can also be used to calculate mismatch probabilities for each of these sensors. Figure A.21 shows the resulting average over the 3 sensors for the mismatch probability. The curves show the same shape as Figure A.19, while the optimal offset remains at around 40ms. Nevertheless, by comparison of Figures A.20/A.21 with the simulated control performance in Figure A.19, it becomes clear that the approach to

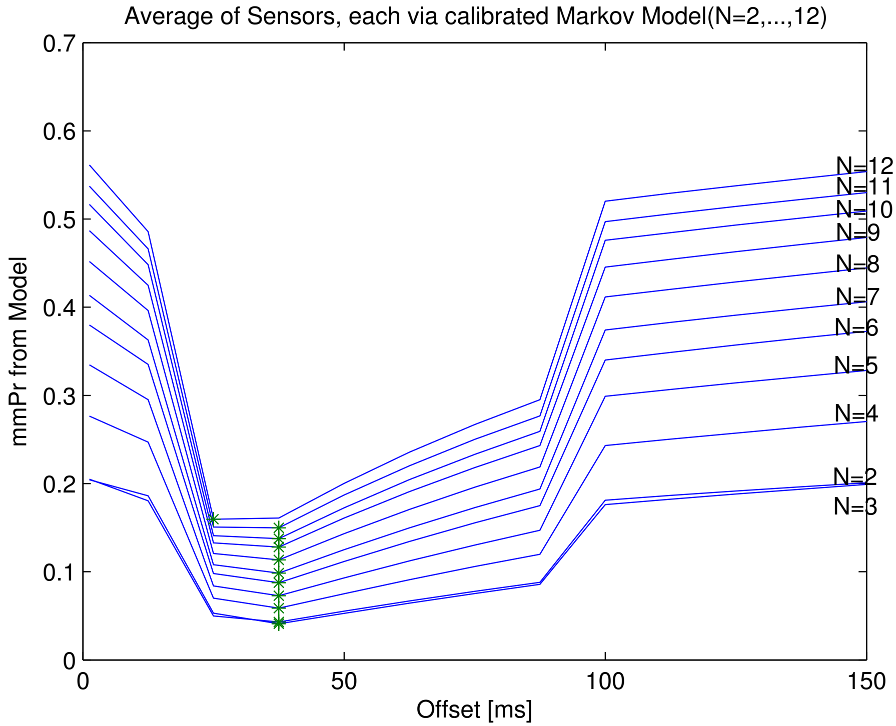



**Fig. A.21:** Average mmPr of all 3 sensors over changing offsets in the exponential approximation of the combined 3G and CANBUS scenario use data quality metrics, here mismatch probability, has general potential for optimizing control performance in periodic control systems.

# 7 Conclusion

In this paper we investigated the impact of end-to-end communication delays and sensor access scheduling optimization on the performance of a wind turbine controller as part of a Smart Grid use-case. We started by determining in Section 3 that there indeed was an impact of the communication network delays, and that an optimized choice of the offset, defined as the time interval between sensor information update generation and controller computation, can be used to improve controller performance. We found that using a simple heuristic to determine the offset ran the risk of choosing a suboptimal offset, and that determining the optimal offset is not a trivial task. In section 4 we modelled the sensor information dynamics to use an analytical approach to determining the optimal offset, instead of relying on extensive simulations of the controller and network together. We found that it was possible to use the mmPr metric to determine an offset, and that this chosen offset performed as well, within a 95 % confidence interval, or better than the considered heuristic methods. Lastly, we investigate the impact of imperfect communication on controller performance using different communication technologies in Section 5. The delays found from this analysis were then used to simulate the performance of the wind turbine controller once again, and the analytical models were used to find the mmPr of the sensors. We find that due to the low variability of the 3G data there is a rather large offset range in which to choose the optimal offset. The measurements of WLAN data had even lower variability, and would such not lead to a narrower offset range. Alternatively, the delays of the 2G data were too long for offset optimization to make sense in the chosen scenario. However, for controllers with longer control periods the 2G data might be more interesting to make offset optimization for, due to the increased variability of the delays.

Future work will look at the impact of different control period choices: slower network technologies like 2G may still be utilized when running the controller less frequently; however the trade-off of modifications in the control period given a certain end-2-end delay will need to be analyzed and understood. For the analysis of this paper, we assumed that the exchange of sensor values to the central controller and of set-points to the local controller is executed via a very lean protocol, containing just the actual values and a few bytes of information type. The analysis of the impact of intermediate layers like TCP [8] and [23], or of protocols that try to achieve real-time bounds and reliability [5] or just improve performance behavior of off-the-shelf shared transmission media [24] for the setting of the wind-farm control is interesting for future work.

An assumption made in this paper was perfect clock synchronization between the central controller and the sensors. This may not be the case, and could cause the offset $T_o$ chosen at the sensors to drift time wise. This could be fixed by inlcluding time

synchronization packets on the network, however this would increase traffic and could increase the delays on the communication network.
This work has been done for a specific Smart Grid scenario, however in future work it would be interesting to expand to cover other scenarios, for example a voltage control scenario as in [25].

## Conflict of Interest

The authors declares that there are no conflict of interests regarding the publication of this manuscript.

## Acknowledgment

The authors would like to thank José de Jesús Barradas-Berglind for lending his wind turbine control simulation and allowing us to work with it.
This work was partially supported by the Danish Council for Strategic Research (contract no. 11-116843) within in the 'Programme Sustainable Energy and Environment', under the "EDGE" (Efficient Distribution of Green Energy) research project, and by the SmartC2Net research project, supported by the FP7 framework programme under grant No318023.

## References

[1] K. Mets, J. Ojea, and C. Develder, “Combining power and communication network simulation for cost-effective smart grid analysis,” *Communications Surveys Tutorials, IEEE*, vol. 16, no. 3, pp. 1771–1796, Third 2014.

[2] B. Bhattarai, M. Levesque, M. Maier, B. Bak-Jensen, and J. Radhakrishna Pillai, “Optimizing electric vehicle coordination over a heterogeneous mesh network in a scaled-down smart grid testbed,” *Smart Grid, IEEE Transactions on*, vol. 6, no. 2, pp. 784–794, March 2015.

[3] R. Qiu, Z. Hu, Z. Chen, N. Guo, R. Ranganathan, S. Hou, and G. Zheng, “Cognitive radio network for the smart grid: Experimental system architecture, control algorithms, security, and microgrid testbed,” *Smart Grid, IEEE Transactions on*, vol. 2, no. 4, pp. 724–740, Dec 2011.

[4] I. Georgievski, V. Degeler, G. Pagani, T. A. Nguyen, A. Lazovik, and M. Aiello, “Optimizing energy costs for offices connected to the smart grid,” *Smart Grid, IEEE Transactions on*, vol. 3, no. 4, pp. 2273–2285, Dec 2012.

[5] B. Malinowsky, J. Gronbaek, H.-P. Schwefel, A. Ceccarelli, A. Bondavalli, and E. Nett, “Timed broadcast via off-the-shelf wlan distributed coordination function for safety-critical systems,” *Dependable Computing Conference (EDCC), 2012 Ninth European*, pp. 144–155, May 2012.

[6] Siemens, “Siemens ag,” 2015, http://www.industry.siemens.com/verticals/global/en/wind-turbine/communication/pages/default.aspx.

[7] I. Contemporary Control Systems, “Managed ethernet switches ensure reliable communications for a wind farm in inner mongolia,” December 2010, http://www.ccontrols.com/enews/1210story2.htm.

[8] J. Madsen, T. le Fevre Kristensen, R. Olsen, H.-P. Schwefel, and L. Totu, “Utilizing network qos for dependability of adaptive smart grid control,” in *Energy Conference (ENERGYCON), 2014 IEEE International*, May 2014, pp. 859–866.

[9] T. Brade, S. Zug, and J. Kaiser, “Validity-based failure algebra for distributed sensor systems,” *Reliable Distributed Systems (SRDS), 2013 IEEE 32nd International Symposium on*, pp. 143–152, Sept 2013.

[10] M. Bøgsted, R. Olsen, and H.-P. Schwefel, “Probabilistic models for access strategies to dynamic information elements,” *Performance Evaluation*, vol. 67, no. 1, pp. 43–60, 2010.

[11] A. Shawky, R. L. Olsen, J. Pedersen, and H.-P. Schwefel, “Network Aware Dynamic Context Subscription Management,” *Computer Networks*, vol. 58, pp. 87–101, Dec. 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128613003629

[12] J. J. Nielsen, R. L. Olsen, T. K. Madsen, B. Uguen, and H.-P. Schwefel, “Location-quality-aware policy optimisation for relay selection in mobile networks,” *Wireless Networks*, pp. 1–20, 2015. [Online]. Available: http://dx.doi.org/10.1007/s11276-015-0986-0

[13] J. Madsen, J. de Jesus Barradas-Berglind, T. Madsen, and H.-P. Schwefel, “On the impact of information access delays on remote control of a wind turbine,” *PowerTech, 2015 IEEE Eindhoven*, pp. 1–6, June 2015.

[14] J. Madsen, M. Findrik, D. Drenjanac, and H.-P. Schwefel, “Investigating wind farm control over different communication network technologies,” *DACH Energie Informatics*, November 2015.

[15] J. Madsen, M. Findrik, T. Madsen, and H.-P. Schwefel, “Scheduling data collection for remote control of wind turbines (abstract accepted, full paper submitted),” April 2016.

[16] J. Barradas-Berglind, R. Wisniewski, and M. Soltani, “Fatigue damage estimation and data-based control for wind turbines,” *IET Control Theory & Applications*, 2015.

[17] K. S. Hansen, “Database on wind characteristics,” March 2015, http://www.winddata.com.

[18] M. A. Ahmed and Y.-C. Kim, “Communication network architectures for smart-wind power farms,” *Energies*, vol. 7, no. 6, p. 3900, 2014. [Online]. Available: http://www.mdpi.com/1996-1073/7/6/3900

[19] 4GLTEmall, “Huawei e392 4g lte fdd tdd multi-mode data card,” June 2015, http://www.4gltemall.com/huawei-e392-4g-lte-multi-mode-data-card.html.

[20] Devolo, “Devolo g3-plc modem 500k,” June 2015, http://www.devolo.com/at/SmartGrid/Produkte/devolo-G3-PLC-Modem-500k.

[21] P. Engine, “alix3d2,” June 2015, http://www.pcengines.ch/alix3d2.htm.

[22] Mini-Box, “Wistron cm9-gp minipci card,” June 2015, http://www.mini-box.com/Wistron-CM9-GP-Atheros-miniPCI.

[23] M. Findrik, R. Pedersen, C. Sloth, E. Hasenleithner, and H.-P. Schwefel, “Test-bed assessment of communication technologies for a power-balancing controller (to be published in april 2016),” *ENERGYCON2016*, April 2016.

[24] T. Hinterhofer, H. Schwefel, and S. Tomic, “Rd2: Resilient dynamic desynchronization for tdma over lossy networks,” *Reliable Distributed Systems (SRDS), 2012 IEEE 31st Symposium on*, pp. 231–236, Oct 2012.

[25] M. Findrik, T. Fevre Kristensen, T. Hinterhofer, R. L. Olsen, and H.-P. Schwefel, “Ad-hoc, mobile, and wireless networks: 14th international conference, adhoc-now 2015, athens, greece, june 29 – july 1, 2015, proceedings,” pp. 317–329, 2015. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-19662-6_22

# Paper B

## On the use of Information Quality in Stochastic Networked Control Systems

Rasmus L. Olsen, Jakob G. Rasmussen, Hans Peter Schwefel, Jacob Theilgaard Madsen

The paper has been submitted to the
*Elsevier Journal of Computer Networks*, 2016.

*The layout has been revised.*

## Abstract

*Networked control is challenged by stochastic delay that is caused by properties of communication networks and also by the time instances of accessing information about system state or actuation set-points. This paper investigates, which design choices of networked control systems can be optimized based on an information quality metric, here using the so-called mismatch probability (mmPr). The benefit of this mmPr metric is that it captures in a single scalar value the combined impact of access protocols and delays from the communication network and puts those in relation to the dynamics of the accessed information. The hypothesis of the paper is that the optimization of certain parameters of networked control systems targeting the mmPr metric is equivalent to the optimization targeting control performance, while the former is practically much easier to conduct. The hypothesis is first analyzed in simulation models for the example system of a wind-farm controller. As the simulation analysis is subject to stochastic variability and requires large computational effort, the paper in the remaining part develops a Markov model of a simplified networked control system and uses numerical results from the Markov model analysis to demonstrate, the mmPr based optimization can improve control performance for a subset of system parameters.*

# 1 Introduction

Networked control systems have become increasingly relevant in recent years; main examples include the management and operation of geographically distributed infrastructure such as telematics systems [1] and smart energy grids [2]. In such systems, the control function uses input from distributed sensors which have to be collected via a communication infrastructure to compute set-points which are in turn sent to actuators, again via a communication infrastructure. In large-scale systems, not only does the communication infrastructure add delays and potential message loss, but the information access and set-point distribution may also be subject to further data management procedures, such as caching, aggregation and preprocessing in hierarchical information architectures. In order to facilitate the design of control applications, the information access and set-point distribution is frequently handled via an intermediate layer, e.g. a data management middleware or context management system [3].

Target of the networked control realization is the optimization of overall system metrics, i.e. those are influenced by the interplay of different system parts. Control performance, hereafter referred to as Quality of Control (QoC), is hence an overall metric, which is influenced by the controller design, but additionally also influenced by the resulting properties of the information access and set-point distribution approach and by properties of the communication network. The analysis of QoC is therefore complex and the optimization of the information access approach is frequently done in order to meet worst-case bounds [4], e.g. using network calculus [5], or via time-consuming (co-) simulation experiments [6]. The analysis of networked control systems targets in many cases stability, see [7, 8]. Other performance metrics of networked control systems are depending on the specific control scenario; we here will utilize the example of wind-farm control that is addressed in [9].

A more efficient approach to optimize the information access approach and the communication network configuration can be via the use of information quality metrics: the basic idea here is to abstract from the specific controller implementation but rather target an optimization of the controller's view about current system parameters at the time of the execution of the control computation. The benefit of this is to decouple controller design and network optimization and thereby simplifying design of networked plant control. This paper shows the benefits of information quality metrics for the parameter optimization of networked control systems and also provides an understanding of their limitations.

The use of sensor information in distributed systems is subject to two types of inaccuracies degrading information quality: Measurement errors at the sensor may propagate through the whole computation chain, see, e.g., [10] for work characterizing such errors and their impact. For distributed real-time systems, a second cause of inaccuracies requires attention: while the sensor data is being transmitted and processed, the actual physical value changes so that it deviates from the value used in the processing. In this paper, we focus on the second aspect and utilize an information quality metric, called mismatch probability (mmPr) which is defined as the probability of information when being used is not matching the true value at the remote sensor. This metric and mathematical models for certain base cases were introduced first in Reference [11]. MmPr considers the aspect of real-time information access, capturing impact of access delays and access strategies on information accuracy in a distributed system. So far, this metric has been used in the following context: (i) in Reference [12] mmPr was used in context subscription management systems for effective configuration of context access strategies in order to maximize the reliability of context information; (ii) in Reference [13], the metric was used to find optimal location-based relay policies in mobile networks. Reference [14] investigates whether optimization of the information quality is a means of improving the control performance of the hierarchical wind farm controller. The simulation-based use-case example in this paper is based on the latter, while additional analysis is performed in order to support the claims and insights of this paper.

The mismatch probability metric which was introduced in [11] is defined by a deviation between information at a client and a server, and depends highly on 1) end-to-end delay, 2) information dynamics and 3) the access method used. This metric hereby captures an otherwise complex relation between these three factors into a single, scalar value that enables designers to make decision on which strategy and configuration to take. Intuitively, reducing network delays may obviously benefit the performance of networked controllers in most cases; however, when there is a choice between a low update rate in conjunction with a low network delay, or a higher update rate at the price of a higher network delay, this choice cannot be resolved based on intuition only. Secondly, in order to judge the impact of network delays and update intervals, it is not only the mean value but the whole distribution which will affect overall performance of networked controllers. Therefore, capturing the impact of these different factors in conjunction with the dynamics of the observed system in a single scalar metric, like the mmPr, can be very valuable.

The paper first analyzes the performance of a wind farm controller as a realistic example case of networked control systems in Section II. This analysis is done via simulation models. In addition to the insights into the system behaviour, the analysis also illustrates the challenges of using simulation based analysis for the problem of this paper. Therefore Section III introduces a very simplified control scenario that can be modelled as a Markov chain and therefore control performance as well as mismatch probability can be analyzed via numerical solutions of the steady state Markov process probabilities. In Section IV, we provide the Markov model of the system with ideal information access (with mismatch probability always equal to zero). Section V provides an extended model including periodic information access and non-zero network delays; this section also provides the algorithm to numerically compute mismatch probability in this model. In Section VI we provide numerical results for the different models and evaluate in which scenarios the mismatch probability is useful for optimizing control performance. Finally in Section VII we summarize and conclude the paper.

# 2 Simulation analysis of wind-farm control

Control of wind farms [9] targets two objectives: generation of a target profile of desired produced power (reference tracking), while minimizing fatigue of the individual wind turbines. As the former control objective requires coordination across all wind-turbines in the wind-farm, a centralized controller with bi-directional communication to the wind turbines can be deployed. The system and analysis approaches we use here are based on [15] while we here are specifically interested in which case mismatch based optimization leads to optimization of control performance and vice versa.

In this example scenario we consider a system consisting of a central controller, communicating with several wind turbines over a communication network, each wind turbine containing 3 sensors. There is a bi-directional information flow, consisting of sensor measurements from the sensors to the central controller, and set-points from the controller to the wind turbine as shown in Figure B.1.

The objective of the central controller is to follow a set-point for the wind farm's power generation while reducing fatigue of the a wind turbine to prolong its lifetime. The sensors send information regarding the state of the wind turbine to the central controller. In this example, we assume that the central controller is executing periodically with a

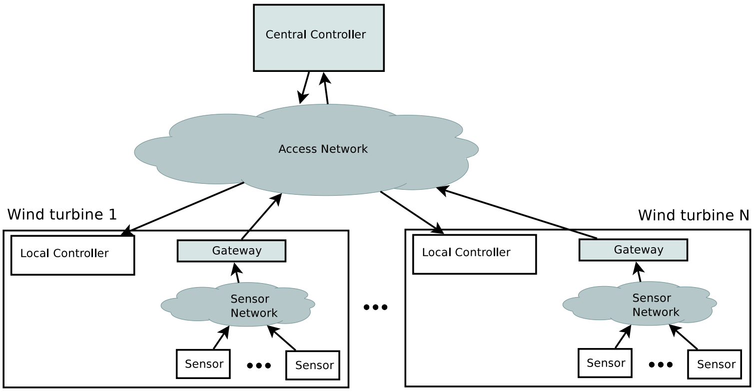



**Fig. B.1:** Overview of the communication architecture

fixed deterministic period of duration $T_S$.

## 2.1 Communication network architecture

The end-to-end communication path between a sensor and the central controller consists of two communication networks, which can be seen in Figure B.1: a Sensor Network and an Access Network. Figure B.2 shows a message sequence diagram of a control period with message delays. At a certain point in time, $T_{C,i}$ the central controller will calculate a set-point for the wind turbine. The controller sends this set-point to the wind turbine, which will then act upon it. At some point during the control period, defined by the value $T_o$, denoting the offset time prior to control computation $t_C$, the sensor will take a reading, and will send it via the sensor network to the gateway that forwards the sensor reading via the access network to the central controller. Figure B.2 only shows the end-to-end communication relations, where the delays indicated by the tilt of the messages are end-to-end delays.

The sensor information in this scenario is continuous. We therefore use a discretization threshold to identify the mismatch case as follows:

$$mmPr(\epsilon) := Pr(|S_Q^C(t_C) - S_Q(t_C)| > \epsilon) \tag{B.1}$$

where $S_Q^C(t_C)$ is the sensor value that the central controller has received from the sensor, while $S_Q(t_C)$ is the actual sensor value at control decision time $t_c$. We thereby define a mismatch as any difference in those two values larger than $\epsilon$. By decreasing $\epsilon$ the mmPr will be more susceptible to small variations of the physical value within the time duration since accessing and transferring it.

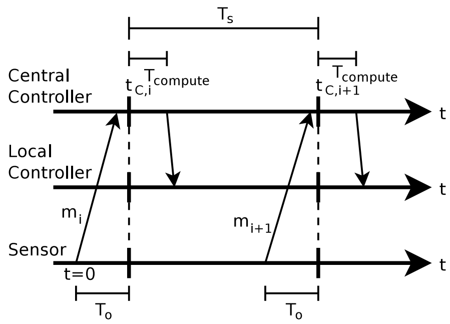



**Fig. B.2:** Message sequence diagram between the sensor and central controller

## 2.2 Controller design

The central controller sends set-points to a wind turbine with the aim of reducing the fatigue load of the shaft in the wind turbine. It does this while tracking a desired power reference. To achieve this the central controller requires an input from a fatigue estimator. This estimator requires sensor measurements of the wind turbine, which are the physical measurements of torque, as well as the angle rotation of the wind turbine. These measurements are run through different hysteresis relays with different weights. For details on estimator synthesis and the controller see [15]. In this paper we assume the fatigue estimator and the controller are co-located in a central controller as shown in Figure B.1. This controller describes the procedures for an individual wind turbine, while the same procedure, with coordination on the generation share of each wind turbine, is applied for the whole farm, which requires a centralized controller. For control performance, we analyse a single wind turbine as we focus on the fatigue as the KPI describing control performance, while the communication scenario is shaped by the whole wind farm.

## 2.3 Simulation results

The performance of the wind-farm scenario is analyzed in a co-simulation framework from [16]. In fact, the actual numerical result values for mismatch probability and control quality are from the analysis scenarios from the latter reference, while this section develops and applies a different analysis methodology in order to investigate whether the mismatch metric is a suitable indicator, through which control performance can be optimized in the given setting.

Before approaching the actual analysis of the latter question, we illustrate the behavior of the wind-farm controller by repeating some results from [16] in Figures B.3 and B.4. Both curves show qualitatively similar behavior, giving a first intuitive indication that mmPr may be a good indicator of expected control performance. However, vari-

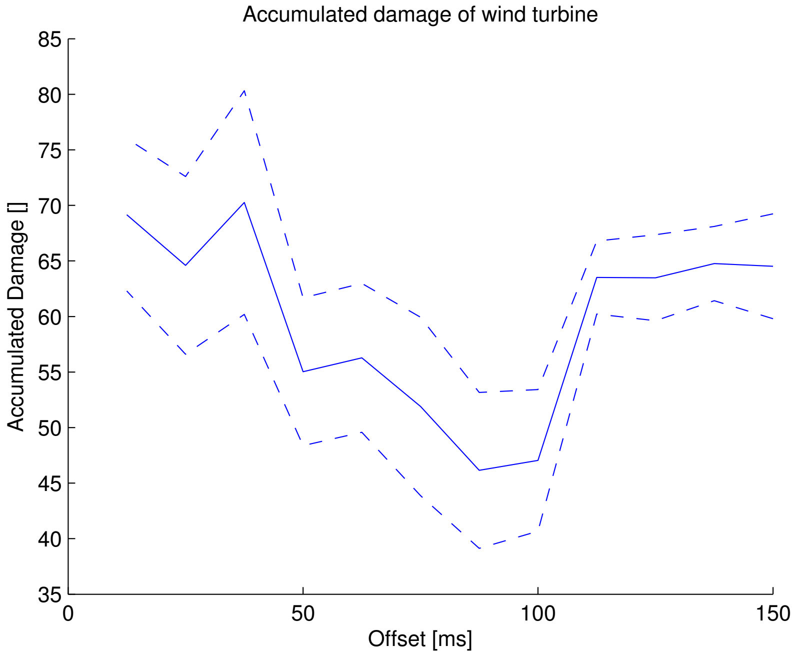



**Fig. B.3:** Wind farm control performance and 95% confidence interval for the simulation estimate for varying offset choices, from [16]

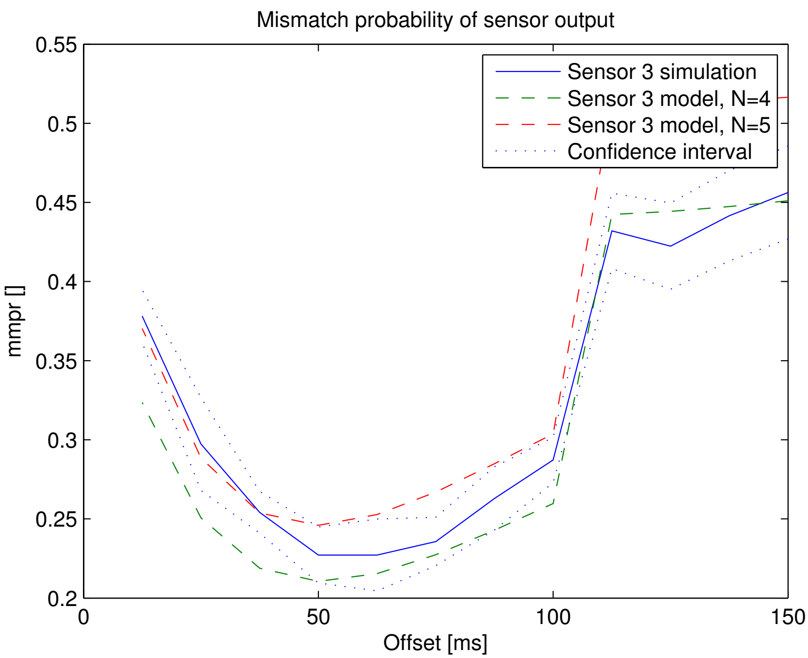



**Fig. B.4:** Mismatch probability in the wind-farm control scenario for one of the three sensors (solid line) and 95% confidence intervals. The figure is repeated from [16], while the actual model results in the dashed lines are not relevant for this paper.

ability indicated by the size of the confidence intervals needs to be taken into account. Similar results can be generated for varying network delays, see [16].

We now investigate the question for what parameter tuples of delay and offset, the mmPr can be used to optimize control performance, we run several simulations for five different communication network scenarios: 3 scenarios with network delays from exponential distributions with different means and measured network delay traces from 3G networks and WLAN (see [16] for the detailed description of the measurement setup). Each control simulation for the five different network models is executed for 12 different offset values ($T_o$), resulting in a total of 60 different control performance and mmPr pairs. The overall simulation parameters are summarized in Table B.1.

| | |
|---|---|
| Controller cycle, $T_s$ | $150ms$ |
| Controller computation delay | $50ms$ |
| Controller cycles per simulation | 400 |
| Independent replications | 10 |
| Network Delay Scenarios | 3 Exponential, 3G, WLAN |
| Offset Values $T_o$ | 12.5ms, 25ms, ..., 150ms |

**Table B.1:** Parameters on the overall wind-farm topology and parameters for the control execution

The drawback of simulation analysis as opposed to numerical results that are used in later sections is that the results are subject to stochastic variations. In order to address the latter, we introduce two hypotheses:

1. Hypothesis-1: A significant difference in control performance causes a significantly different behavior of mmPr with the adequate monotonicity relation.

2. Hypothesis-2: A significant difference of mmPr causes a significantly different behavior of control performance with the adequate monotonicity relation.

A significant difference here refers to non-overlapping 95 % confidence intervals from the simulations. These hypotheses are then tested for all combinations of pairs available.

Figure B.5 shows the relative frequency of occurring simulation results pairs that confirm Hypothesis 1 respectively Hypothesis-2 for different thresholds $\epsilon$. We see that for $\epsilon$ values less than $2.5 \cdot 10^{-4}$ Hypothesis 1 is true over 80 % of the cases, while Hypothesis-2 is true over 60 % of the cases. For larger values of $\epsilon$, this probability reduces strongly for both hypotheses.

Values of around 80% and 60% of observing Hypothesis-1 and Hypothesis-2, respectively, in the data, do not imply that we otherwise observe inverse (wrong) relations of the corresponding metrics. There is also the possibility that no significant difference is observed in the simulation experiment, so not giving evidence of a wrong hypothesis. Figure B.6 plots the relative frequencies that the corresponding performance metrics in

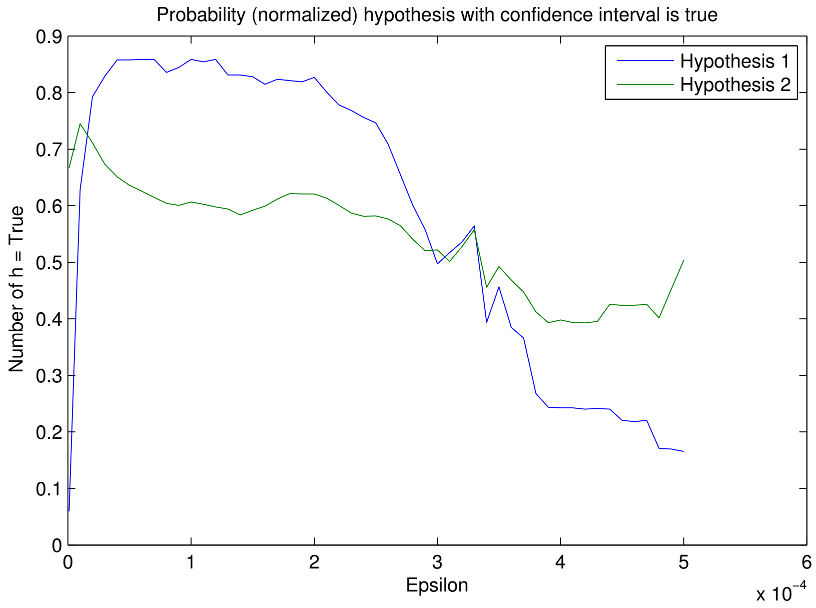



**Fig. B.5:** Probability that H1 and H2 are true

the hypotheses show a significantly wrong difference. These probabilities are rather low for the choice of $\epsilon < 2.5e - 4$.

Finally Figure B.7 sheds some light on what happens for larger thresholds $\epsilon > 3 * 10^{-4}$, the probability that the mmPr observed in the simulations is zero rapidly increases. Hence, the mmPr metric becomes too insensitive to changes.

In summary, the simulation results show that mmPr is a suitable metric to optimize control performance for the considered cases of wind-farm control, given that the threshold for the mmPr definition is not chosen too low or too high; in the latter case, the metric turns not useful as it becomes too insensitive (is most of the time equal to zero); in the former case the information would be seen as new information, even if the change of the sensor information is not having a relevant impact for the future controller behavior.

The use of mismatch probability allows to optimize offset choices according to the delay and information dynamics. The software that carries out this adaptation does no need to know about the controller, but only needs to model the mismatch probability. The example in the previous results used three sensors and the results for the mismatch probability showed the sensor that had the strongest impact on control performance. This approach can be generalized to other control systems, i.e. to identify and optimize based on the mmPr for the most relevant input information. Alternatively, also vector

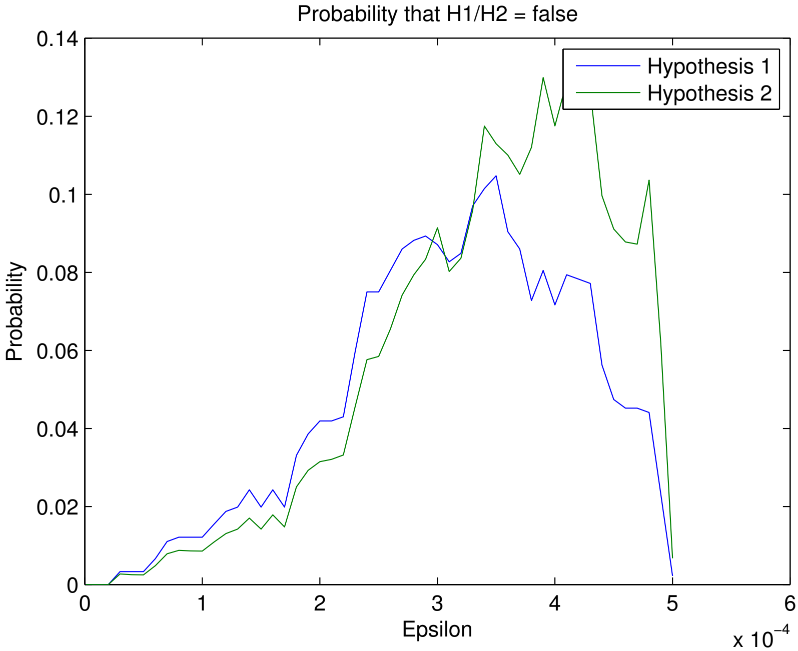



**Fig. B.6:** Probability that H1 and H2 are false

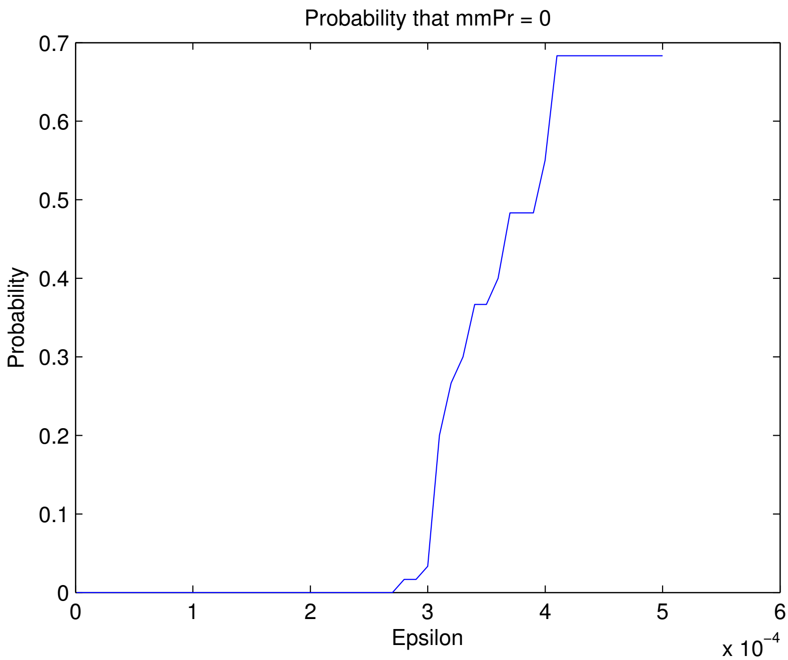



**Fig. B.7:** Impact of an increasing threshold epsilon on the resulting mmpR values: for $\epsilon > 3 * 10^{-4}$, the fraction of cases in which mmPR is zero increases rapidly.

mismatch metrics can be defined based on observation vector inequalities; however for a large number of sensors, such vector mismatch metrics would quickly approach the value of 1, so this approach may reach its limits for complex systems with many observations.

The simulation analysis in this section was computationally quite expensive as it requires execution of a co-simulation framework in Matlab/Simulink and Opnet. Collecting the estimators and confidence intervals for the 60 input value tuples required close to one week of simulation time. Even more replications, and hence even more simulation time may be needed to reduce the size of the confidence intervals and that way produce more samples contributing significantly to the analysis in this section.

Due to the drawbacks of the simulation approach, the next sections develop and apply a method based on numerical solutions of Markov Chain models.

# 3 Simplified control scenario and base system model

Since the simulation analysis in the previous section demonstrates the well-known difficulties regarding computational time and stochastic variability, we now come up with an abstracted and simplified system that can be modeled by a Markov chain model and hence can be evaluated via numerical solution of the steady-state probabilities.

Figure B.8 illustrates the message exchange of the considered control loop system with three control cycles. The given plant is sampled periodically by a local sampling process and the observation is forwarded via a communication infrastructure to a remote controller. At periodic intervals, the controller uses the last received observation to determine a setpoint, which it sends back via the communication infrastructure to be executed by a local actuator affecting the plant. In some cases due to delay and information age at the time of the control decision calculation, the plant may have changed state and which ultimately may lead to a different control decision than if the true plant state was known to the controller at that point in time. This we call a mismatch and is impacted by network delays, sampling/update and control processes as well as the information dynamics (the plant state change rates). We assume the plant state is discretized into $N$ distinct plant states $S_Q(t) \in \{1, 2, ..., N\}$ which are sampled periodically. The observed state is sent via a communication network, which imposes stochastic delays to the transmitted updates. We describe messages in transit with a state variable $S_M \in \{m_0, m_1, ..., m_{Mn}\}$ for $Mn$ messages in transit and with $m_0$ being zero messages in transit. When the controller receives an update message, it internally changes controller state $S_C(t) \in \{c_1, c_2, ..., c_{Cn}\}$ according to the latest received information update. At certain points in time, $t_c$, when control is being executed, the controller state is mapped to a control action $C$ which subsequently is sent to the plant where an actuator executes the control action. The control action leads to a change of one of the $An$ possible actuation states $S_A(t) \in \{a_1, a_2, ..., a_{An}\}$. Hereby the system is fully described by the tuple $[S_Q, S_A, S_C, S_M]$. Due to message delay and caching of updated information at the controller, decisions made for changing actuator states may be wrong which is illustrated in Figure B.8 in control period 3 (right most case). Here, the plant changes state such that it invalidates the controller's state view; and thereby the decision the controller takes ultimately leads to a wrong actuation.

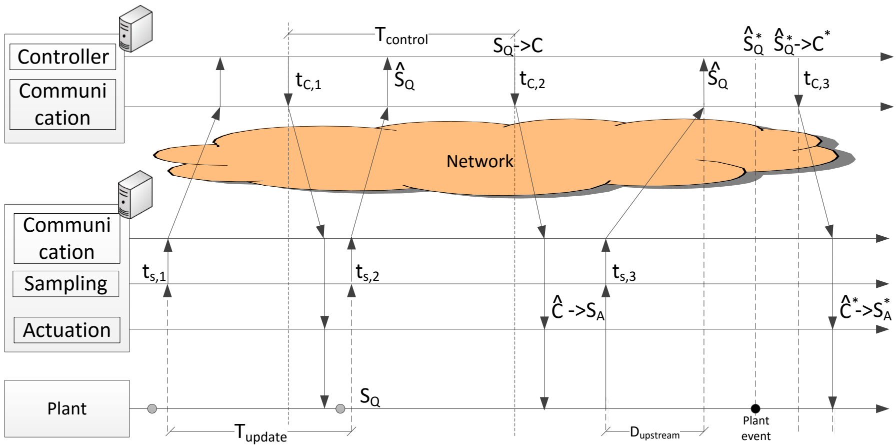



**Fig. B.8:** The system, components and their interaction in a timing diagram of the remotely controlled system

We define our metric mismatch probability as the probability that the controller at time $t_c$ makes a different actuation decision compared to the one that it would have made if it had known the correct system state at the computation time. Subsequently, we analyze in which cases this mismatch metric is useful to optimize such a control loop. To answer these questions we first model and analyse control performance of a plant under ideal information access conditions and then compare this to non-ideal information access and draw conclusions on cases in which this metric is useful for designing networked control.

## 3.1 General assumptions on the system

For the control and sampling process we will assume that the time intervals are not deterministic but suffers from some randomness due to e.g. clock drift, so we model the time intervals as an exponential distributed random variable with rate $\tau = 1/T_{update}$ as well as control rate $\alpha = 1/T_{control}$. This choice of exponential distributions simplifies the later modelling.

**Table B.2:** List of symbols

| Symbol | Meaning |
|---|---|
| $T_{update}$ | Time interval between updates send to controller |
| $T_{Control}$ | Time interval between control decision |
| $D_{upstream}$ | Upstream delay |
| $t_{s,n}$ | Point of time for sample $n$ |
| $t_{c,m}$ | Point of time for control decision $m$ |
| $S_A$ | Actuator state |
| $S_Q$ | Plant state |
| $S_C$ | Controller state |
| $\hat{C}$ | Control action at actuator |
| $C$ | Control decision |
| $C^*$ | Wrong control decision due to event at plant |

## 3.2 Plant representation as Birth-Death Process

We consider the plant being a discrete state stochastic system, where states are totally ordered and changes occur only to neighbouring states. For instance, such model could result from the discretization of a continuous state system. In our work we focus on the detectable state-changes, and in the remaining part of this work we assume that the time intervals between these events are stochastic and can be described by exponential distributions (while other types of distributions could be in principle represented by additional states using phase-type distributions [17, 18]). The resulting plant model can then be described as a Birth-Death Markov chain, which allows us to perform the desired numerical analysis in a stochastic framework that can be combined with a model of a stochastic network. The analysis framework we provide in this paper do allow for other types of models if e.g. complex behaviour of the system needs to be captured in the model, but in order to keep the number of parameters small for the analysis here, we assume the Birth-Death process with a constant rate of upwards transitions as $\lambda$, and the downward transition rate as $\mu$. We denote the generator matrix of this Markov chain as $\mathbf{Q}$ which state at time $t$ is described by $S_Q(t) \in \{1, 2, ..., N\}$.

## 3.3 Control and actuation of the plant

The objective of the controller is to keep the plant state between some given boundaries, $B_{low} < S_Q(t) < B_{high}$. For the controller we define three control states, which are a) add additional drift to the system to wards higher state numbers (assumed to be placed to the right), b) do nothing or c) add additional drift to the system to lower state numbers (assumed to be placed on the left). In order to achieve the control objective, the following behaviour of the controller is assumed:

- $S_Q(t) < B_{low}$: add drift to the system from left to right - Left-to-Right Mode (LRM).
- $B_{low} \leq S_Q(t) \leq B_{high}$: do nothing - system is inside desired state levels - Normal Mode (NM)
- $S_Q(t) > B_{high}$: add drift to the system from right to left - Right-to-Left Mode (RLM).

The decision on which actuation state should be used is taken by the controller based on the knowledge it has of the plant state at a given moment in time which is then send to the actuator. The actuator then executes the controllers decision by imposing a drift rate $\Delta$ (and internally change actuator state) to one direction or the other in the chain. This approach enables us to define three states for the controller and actuator: $S_A(t), S_C(t) \in \{LRM, NM, RLM\}$.
To encode the actuation in the plant model we define two versions of the non-controlled plant model $\mathbf{Q}$, one for each of the actuators view of the system denoted as $\mathbf{Q}_{LRM}$ and $\mathbf{Q}_{RLM}$. This yields the following structure of the variants of the generator matrix, $\mathbf{Q}$

$$\mathbf{Q}_{LRM} = \begin{bmatrix} -(\lambda+\Delta) & \lambda+\Delta & 0 & \dots \\ \mu & -(\lambda+\mu+\Delta) & \lambda+\Delta & \dots \\ .. & .. & .. & .. \\ 0 & 0 & \mu & -\mu \end{bmatrix}$$

$$\mathbf{Q}_{RLM} = \begin{bmatrix} -\lambda & \lambda & 0 & \dots \\ \mu+\Delta & -(\lambda+\mu+\Delta) & \lambda & \dots \\ .. & .. & .. & .. \\ 0 & 0 & \mu+\Delta & -(\mu+\Delta) \end{bmatrix}.$$

The controller makes the decision in which of these states the plant must be in, and leaves the actuator to ensure that these drifts are being effectively executed in the plant.

# 4 Markov model for control based on ideal information

For our initial study we focus on the control loop with ideal information access as a comparison case. For the ideal information access case we assume that the update or sample rate of the system, with mean $T_{update}$, is equal to the control time interval, $T_{control}$ and that these are perfectly synchronized in time. Any delays e.g. from processing or network delays we assume to be negligible. The synchronization assumption enables us to combine the two processes into a single one: the control process. The complete system state with ideal communication is then described by the tuple $[S_Q(t), S_A(t), S_C(S_Q(t))]$.

## 4.1 Markov model of the system with ideal information access

The approach of our Markov chain model for the ideal information access case is shown in Figure B.9. The rationale behind this approach is to allow preservation of the plant state while also keeping track of the controller state. Actuation is effectively done by jumps between the different versions of the plant model.

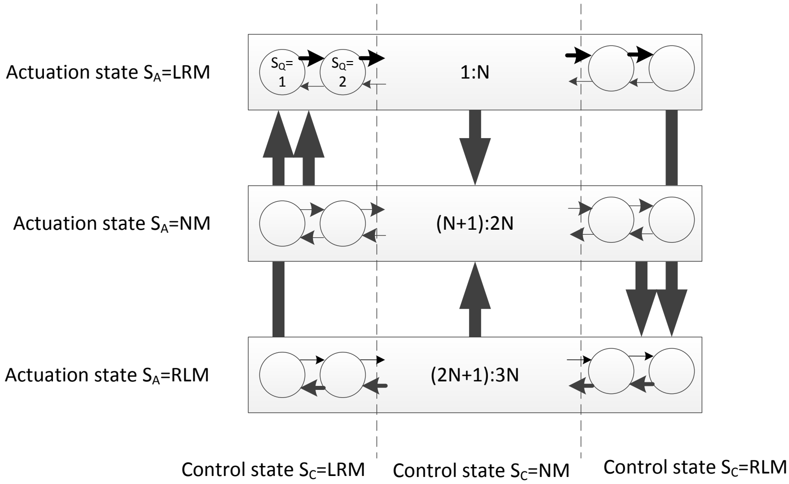



**Fig. B.9:** Block structure of the model for controlled system with update strategies

The complete Markov chain model of the controlled system is done with a modified product space representation of the system which takes into account the controllers view of the plant. The different actuation views ($\mathbf{Q}_m$, $m \in \{LRM, NM, RLM\}$) are all interlinked via the controller with rate, $\alpha$. This independent process triggers the events in the system that imposes the different control states onto the system, i.e. which of the three actuation states has to be executed. The model is created by properly inserting state transition matrices, $\mathbf{C}_m$ which maps $\mathbf{Q}_m - > \mathbf{Q}_{m^*}$ with $m$ being the old control view mode and $m^*$ the next one. These $\mathbf{C}_m$ matrices are defined by

$$\mathbf{C}_{LRM} = \begin{pmatrix} \alpha \mathbf{I}_{B_{low} x B_{low}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \qquad C_{NM} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha \mathbf{I}_{M x M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix}$$

$$\mathbf{C}_{RLM} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha \mathbf{I}_{B_{high} x B_{high}} \end{pmatrix}$$

with $M = Dim(Q) - B_{low} - B_{high}$. The dimension of any of the **C**-matrices are always $N \times N$. The different system models, $\mathbf{A}_*$ and the control action matrices, $\mathbf{C}_*$ is mapped into a generator matrix as

$$\mathbf{G}_{ideal} = \begin{pmatrix} \mathbf{Q}_{LRM} & \mathbf{C}_{NM} & \mathbf{C}_{RLM} \\ \mathbf{C}_{LRM} & \mathbf{Q} & \mathbf{C}_{RLM} \\ \mathbf{C}_{LRM} & \mathbf{C}_{NM} & \mathbf{Q}_{RLM} \end{pmatrix} \tag{B.2}$$

for which we denote the random variable describing the current system state of the ideal information access system model $\mathbf{G}_{ideal}$ as $Z_{ideal}(t)$.

### 4.2 Control performance definition and calculation

As the objective of the controller is is to keep the plant state inside a given range, $B_{low}$ and $B_{high}$, we can now define following performance metric based on the steady state solution of the system. We consider at an arbitrary time $t$ in the stationary system, what is the probability that the current plant state $S_Q(t)$ is in the state interval $\{B_{low}, \ldots, B_{high}\}$, and here from define the Quality of Control (QoC) as

$$\text{QoC} = \mathbb{P}(S_Q(t) \in \{B_{low}, \ldots, B_{high}\}).$$

This can be obtained from $Z(t)$ (we defined already $Z_{ideal}$ and later we introduce $Z_{nonideal}$) through its related steady state probability vector $\pi$ by

$$\text{QoC} = \mathbb{P}(Z(t) \in B) = \sum_{i \in B} \pi_i,$$

where $B$ denotes the set of states of $Z(t)$ corresponding to $S_Q(t) \in \{B_{low}, \ldots, B_{high}\}$.

So far we focused on the case where the controller has ideal information access. This could be cases when the control is done locally (i.e. not distributed) or if network conditions are such that delays are negligible and the sensor is aware of the periodic time instances when the controller is starting its computation and sends the system observation infinitely close to that. Alternatively, this could be achieved without clock synchronisation by an infinite fast sampling and observation transmission rate. In reality this is not possible, but does give a comparison base when later adding network impact in the system.

## 5 Markov model for control subject to non-ideal information access

In this section we address the same control of the plant as discussed in Section 3.3, but now sampling of the plant state is no longer synchronized in time with the controller and the communication network adds stochastic delay to message transmissions from sensor to controller, the upstream delay $D_{upstream}$. We do not consider any particular communication protocol but focus on an abstraction of messages being passed from the plant to the controller. From the controller to the actuator, downstream we maintain our assumption of ideal communication to keep the model simple, while the principle we use for upstream delays can also be used in this direction at the expense of model complexity.

For the communication we assume that once messages are in transit, new updates may be sent concurrently. We keep track of different content in updates by defining the state variable $S_M \in \{m_0, m_{LRM}, m_{NM}, m_{RLM}\}$ which models the network in different conditions with $m_0$ as no messages in transit. In such case, in principle more messages in transit needs to be accounted for, but to keep our model simple, we keep track of only the latest transmitted message. This means that while existing messages are in transit new one replaces those already in transit. This assumption relates to network scenarios with a single link and a preemptive data transmission discipline.

## 5.1 System description and parameters

Due to delays caused by both communication network and information access the controller will now need to make control decisions based on potentially outdated plant state information, i.e. where before $\hat{S_Q}(t_c) = S_Q(t_c)$ now $\hat{S_Q}(t_c) = S_Q(t_c - D_{upstream} - \delta)$, with $\delta$ being the time difference between the receiving time of the update at the controller and the control decision time. We consider now the sampling process and the control process as two independent processes, with exponential rates $\tau$ and $\alpha$, respectively.

## 5.2 Markov model extension

The Markov chain model of the complete system with non-ideal information access is composed by several elements, and in addition to previous model, the extended model must be able to keep track of messages in transit while maintaining the state of the plant, the controller and the actuator, that is capturing the complete tuple $[S_Q, S_A, S_C, S_M]$. Again, we model the extended system by a modified product-space representation in which **Q**, **C** and **M** contains transitions between plant state, actuation state, controller state and messages in communication state. The block structure of the model is visualized in Figure B.10.

The plant and actuation system is in the extended model here similar to what has previous been described in Section 3.

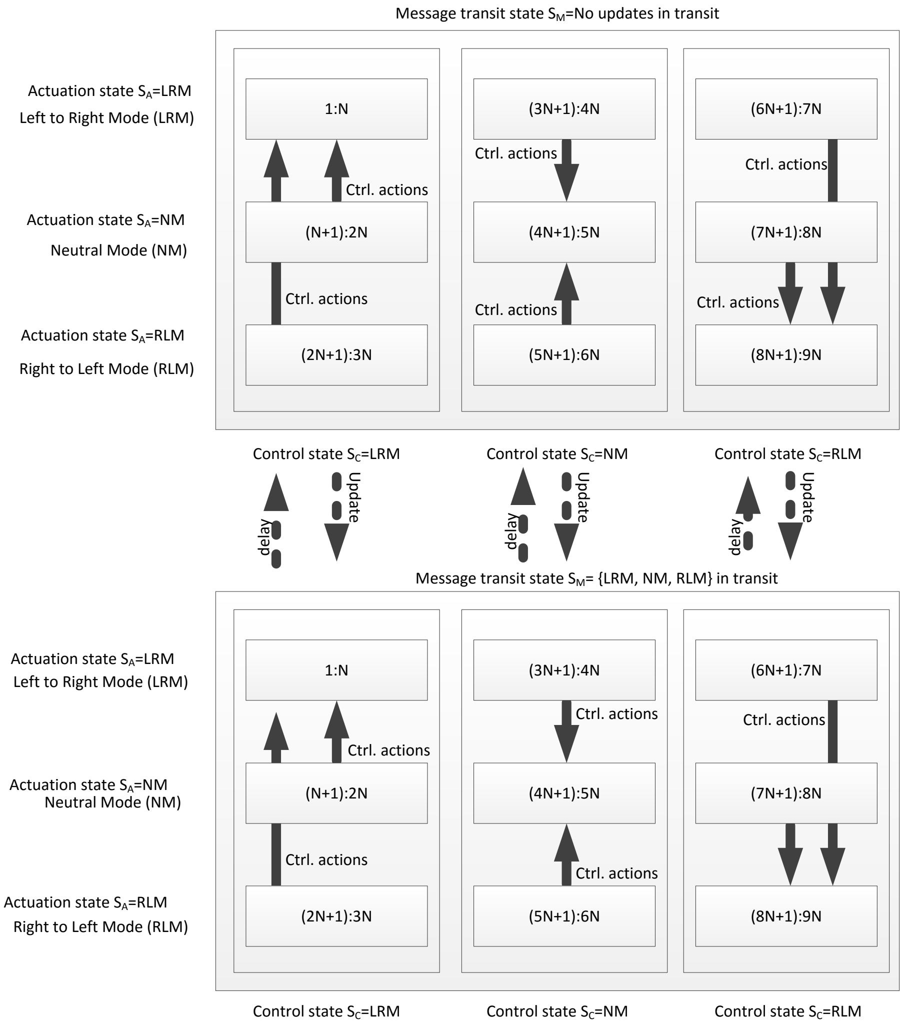



**Fig. B.10:** Block structure of the model for controlled system with update strategies

### Controller state view

Consider the controller and actuator can take any of the three different state values $LRM$, $NM$ or $RLM$, there are in total nine permutations of possible state combinations the control/actuation subsystem can take. All nine combinations needs to be represented in the Markov chain, which has been organized such that columns indicates three blocks of control views changing the actuator view such that $[S_Q, S_A, S_C]- > [S_Q, S_A*, S_C]$. A control action thereby requires a transition matrix that maps the relevant actuation states. This is done via the defined control action matrices, $\mathbf{C}$, one for each of the control views that can be taken and defines the system with no messages in transit, $S_M(t) = m_0$ with $\mathbf{M}_{m_0} = diag(\mathbf{C}_{LRM}, \mathbf{C}_{NM}, \mathbf{C}_{RLM})$.

$$\mathbf{C}_{LRM} = \begin{pmatrix} \mathbf{A}_{LRM} & \mathbf{0} & \mathbf{0} \\ \alpha\mathbf{I}_{dimQ} & \mathbf{A}_{NM} & \mathbf{0} \\ \alpha\mathbf{I}_{dimQ} & \mathbf{0} & \mathbf{A}_{RLM} \end{pmatrix}$$

For the other cases, the transition matrices becomes slightly different

$$\mathbf{C}_{NM} = \begin{pmatrix} \mathbf{A}_{LRM} & \alpha\mathbf{I}_{dimQ} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{NM} & \mathbf{0} \\ \mathbf{0} & \alpha\mathbf{I}_{dimQ} & \mathbf{A}_{RLM} \end{pmatrix}$$
$$\mathbf{C}_{RLM} = \begin{pmatrix} \mathbf{A}_{LRM} & \mathbf{0} & \alpha\mathbf{I}_{dimQ} \\ \mathbf{0} & \mathbf{A}_{NM} & \alpha\mathbf{I}_{dimQ} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{RLM} \end{pmatrix}$$

From this, we can generate the first part of the complete Markov chain considering that there is no messages in transit as $\mathbf{M}_{m_0} = diag(\mathbf{C}_{LRM}, \mathbf{C}_{NM}, \mathbf{C}_{RLM})$ which models the system without message in transitions, i.e. $S_M = m_0$.

### Communication system and message encoding

Independently of the plant state changes, an update event may occur, which brings the system into a new state, except plant, controller and actuator is required to maintain their states. The updates carries one of three message types: $LRM$,$NM$ or $RLM$, depending on which state the system were in when the update event happened.

Therefore we define three additional matrices: $\mathbf{M}_{msg:LRM}$, $\mathbf{M}_{msg:NM}$ and $\mathbf{M}_{msg:RLM}$ which are copies of $\mathbf{M}_{m_0}$ but in fact encodes that different message types are in transit. Updates are triggered with rate, $\tau$, and may happen at any state combination, and creates the links between the different versions of $\mathbf{M}$. Here we make a little trick and focus on encoding not the plant state value itself, but the decision the controller would make anyway enabling us to keep the amounts of additional blocks to the number of control actions instead of numbers of states. Thereby the update messages $S_M$ are defined by the boundaries of control, $B_{low}$ and $B_{high}$ as

$$
\begin{aligned}
\mathbf{M}_{upd:LRM} &= \tau diag(\mathbf{1}_{1,9} \otimes [\mathbf{1}_{1,B_{low}} \quad \mathbf{0}_{1,dim(Q)-B_{low}}]) \\
\mathbf{M}_{upd:NM} &= \tau diag(\mathbf{1}_{1,9} \otimes [\mathbf{0}_{1,B_{low}} \quad \mathbf{1}_{1,dim(Q)-B_{low}-B_{high}} \quad \mathbf{0}_{1,B_{high}}]) \\
\mathbf{M}_{upd:RLM} &= \tau diag(\mathbf{1}_{1,9} \otimes [\mathbf{0}_{1,dim(Q)-B_{high}} \quad \mathbf{1}_{1,B_{high}}])
\end{aligned}
$$

These transitions ensures that the state numbers are mapped properly such that the current plant state value is not affected by the update transition, and $[S_Q, S_A, S_C]$ is copied into one of the $\mathbf{M}$ blocks. Within one of these blocks, the system is operating just as before, but now in one of the encoded state sets for the particular update in transit states.

Updates are received after some delay, $\nu$, which brings the system back to the state of no transitions and thus, similar we define matrices that captures the receiving events that maps the encoded message into the relevant control state block

$$
\begin{aligned}
\mathbf{M}_{rec:LRM} &= [\mathbf{1}_{3,1} \quad \mathbf{0}_{3,2}] \otimes (\nu \mathbf{I}_{dim(Q)}) \\
\mathbf{M}_{rec:NM} &= [\mathbf{0}_{3,1} \quad \mathbf{1}_{3,1}\mathbf{0}_{3,1}] \otimes (\nu \mathbf{I}_{dim(Q)}) \\
\mathbf{M}_{rec:RLM} &= [\mathbf{0}_{3,2} \quad \mathbf{1}_{3,1}] \otimes (\nu \mathbf{I}_{dim(Q)}).
\end{aligned}
$$

Based on these model blocks in terms of transition matrices we can now connect and write the complete generator matrix for the system model with non-ideal information access

$$
\mathbf{G}_{nonideal} = \begin{pmatrix}
\mathbf{M}_{m_0} & \mathbf{M}_{upd:LRM} & \mathbf{M}_{upd:NM} & \mathbf{M}_{upd:RLM} \\
\mathbf{M}_{rec:LRM} & \mathbf{M}_{msg:LRM} & \mathbf{M}_{upd:NM} & \mathbf{M}_{upd:RLM} \\
\mathbf{M}_{rec:NM} & \mathbf{M}_{upd:LRM} & \mathbf{M}_{msg:NM} & \mathbf{M}_{upd:RLM} \\
\mathbf{M}_{rec:RLM} & \mathbf{M}_{upd:LRM} & \mathbf{M}_{upd:NM} & \mathbf{M}_{msg:RLM}
\end{pmatrix}.
$$

The total size of this model depends on a fixed number of multiple copies of the plant model, $\mathbf{Q}$, more precisely $9 \times 4 \times N$ states in total with $N$ being the numbers of states in the plant model. For all subsequent assessment we focus on a plant with 10 states, i.e. $dim(G_{nonideal}) = 360$. Transition rates between two states $i$ and $j$ enumerated according to what is shown in Figure B.10 is denoted by $g_{ij}$. Define the system state variable of $\mathbf{G}_{nonideal}$ as $Z_{nonideal}(t)$ we are now ready to define our mismatch probability.

## 5.3 Definition of mismatch metric and calculation

Consider the embedded Markov chain, i.e. the Markov chain $Y_k = Z_{nonideal}(t_k)$, where $t_k$ is the time of the $k$th event in $Z_{nonideal}$. We let $\mathbf{P}$ with entries $p_{ij}$ denote the transition matrix of this chain, and the corresponding stationary distribution (i.e. the stationary distribution at event times) be $\tilde{\pi}$.

Consider all events in $Y_k$ that correspond to an information update reaching the controller. Conditional on such an event, we let the mismatch probability be the probability that at the time the controller receives the update, this update results in a controller action which is incorrect compared to the true state of the system. More precisely, denote any transition between states (i.e.a pair of states) by $(i, j)$. Let $A_{\mathrm{up}}$ be the set of transitions, where an update reaches the controller, and let $A_{\mathrm{mm}} \subseteq A_{\mathrm{up}}$ be the set of transitions, where this update results in a mismatch. Then the mismatch probability is defined by

$$\mathrm{mmPr} = \mathbb{P}((i,j) \in A_{\mathrm{mm}} | (i,j) \in A_{\mathrm{up}}),$$

where $(i, j)$ is an arbitrary transition in the stationary embedded Markov chain. The following proposition gives simple formulas for obtaining the mismatch probability either through $\mathbf{P}$ and $\tilde{\pi}$ or through $\mathbf{G}_{nonideal}$ and $\pi$ being the steady state probability vector of $Z_{nonideal}$.

**Proposition 1.** *The mismatch probability is given by*

$$\mathrm{mmPr} = \frac{\sum_{(i,j)\in A_{mm}} \tilde{\pi}_i p_{ij}}{\sum_{(i,j)\in A_{up}} \tilde{\pi}_i p_{ij}} = \frac{\sum_{(i,j)\in A_{mm}} \pi_i g_{ij}}{\sum_{(i,j)\in A_{up}} \pi_i g_{ij}}.$$

*Proof.* Firstly note that the probability that an arbitrary event is transition $(i, j)$ is given by $\pi_i g_{ij}$. Summing up, we thus get

$$\mathrm{mmPr} = \frac{\mathbb{P}((i,j) \in A_{\mathrm{mm}})}{\mathbb{P}((i,j) \in A_{\mathrm{up}})} = \frac{\sum_{(i,j)\in A_{\mathrm{mm}}} \tilde{\pi}_i p_{ij}}{\sum_{(i,j)\in A_{\mathrm{up}}} \tilde{\pi}_i p_{ij}}, \qquad \square$$

which proves the first equality. To obtain the second equality, recall that $p_{ij} = -g_{ij}/g_{ii}$ for $i \neq j$, and $p_{ii} = 0$ for all $i$; or equivalently $P = I - \mathrm{diag}(\mathbf{G})^{-1}\mathbf{G}$, where $\mathrm{diag}(\mathbf{G})$ is the diagonal matrix with the same diagonal as $\mathbf{G}$. Inserting this into $\tilde{\pi}P = \tilde{\pi}$, we get that $\tilde{\pi}$ is a solution to $\tilde{\pi}\mathrm{diag}(\mathbf{G})^{-1}\mathbf{G} = 0$. Since $\pi\mathbf{G} = 0$, we thus have that $\tilde{\pi}\mathrm{diag}(\mathbf{G})^{-1} = \pi c$, where $c$ is a constant ensuring that the entries of $\tilde{\pi}$ sum to 1, or equivalently we have that $\tilde{\pi}_i = g_{ii}\pi_i c$. Thus

$$\begin{aligned} \mathrm{mmPr} &= \frac{\sum_{(i,j)\in A_{\mathrm{mm}}} \tilde{\pi}_i p_{ij}}{\sum_{(i,j)\in A_{\mathrm{up}}} \tilde{\pi}_i p_{ij}} \\ &= \frac{\sum_{(i,j)\in A_{\mathrm{mm}}} g_{ii}\pi_i c \left(-\frac{g_{ij}}{g_{ii}}\right)}{\sum_{(i,j)\in A_{\mathrm{up}}} g_{ii}\pi_i c \left(-\frac{g_{ij}}{g_{ii}}\right)} \\ &= \frac{\sum_{(i,j)\in A_{\mathrm{mm}}} \pi_i g_{ij}}{\sum_{(i,j)\in A_{\mathrm{up}}} \pi_i g_{ij}}. \end{aligned}$$

# 6 Numerical Results from Markov Model Analysis

In this section we perform a parametric analysis of the models developed in previous sections. The analysis is divided into two parts, first, focusing on the ideal information access cases and second on the case with non ideal information access. Table B.3 lists the parameters and ranges used as default in the analysis. Unless used as a parameter, these are the default values used in the numerical result analysis.

**Table B.3:** List of default parameter settings and ranges

| Name | Symbol | Value | Range | Unit |
|---|---|---|---|---|
| Number of states | $N_Q$ | 10 | - | * |
| Mean holding time | MHT | 1 | 0.1-10 | Seconds |
| Ratio between up/down rate | $\rho = \lambda/\mu$ | 1 | 0.1-10 | * |
| Low boundary | $B_{low}$ | 3 | - | * |
| High boundary | $B_{high}$ | 7 | - | state number |
| Control rate | $\alpha$ | 1 | 0.1-10 | 1/second |
| Update rate | $\tau$ | 1 | 0.1-10 | 1/second |
| Delay rate | $\nu$ | 10 | 0.1-10 | 1/second |

## 6.1 System with ideal access

Figure B.11 shows control performance for a plant with 10 states in the ideal communication case. From this result it is easy to see that there is no desirable control rate, $\alpha$, the higher control rate the better control quality. If an increase in control rate is followed also by an increase in controller drift rate $\Delta$, it is possible to achieve even better control performance and in the limit of infinite values of $\alpha$ and $\Delta$ control performance goes to a value of one. However, as control rate and control drift rate are often associated to a cost (although not considered in this paper) infinite (or just very high values) is not realistic and typically a trade-off has to be made.

In Figure B.11 there are two lines, one (red) going nearly vertically up with a QoC value of 0.44, indicating the resulting control performance if the system is not controlled at all. This line indicates that there are cases where controller parameters $[\alpha, \Delta]$ can be chosen such performance of the controller is actually worse than not controlling the plant, i.e. the controller in fact works against the system. For example the control parameters $[0.1, 10]$ would for this system give a QoC of apprx. 0.1 or roughly 80% less than when not controlled.

Typically when designing a control system, control parameters are selected in order to optimize some other metrics often in terms of economical sense or resource usages which in many cases can be further mapped into economics. For this reason, a second line (blue) is drawn along the surface illustrating the combination of $[\alpha, \Delta]$ where QoC is only increased insignificantly which can happen in two cases: 1) if $\alpha$ (controller rate) is being selected for a fixed value $\Delta$ (controller drift rate), at some point increasing the controller

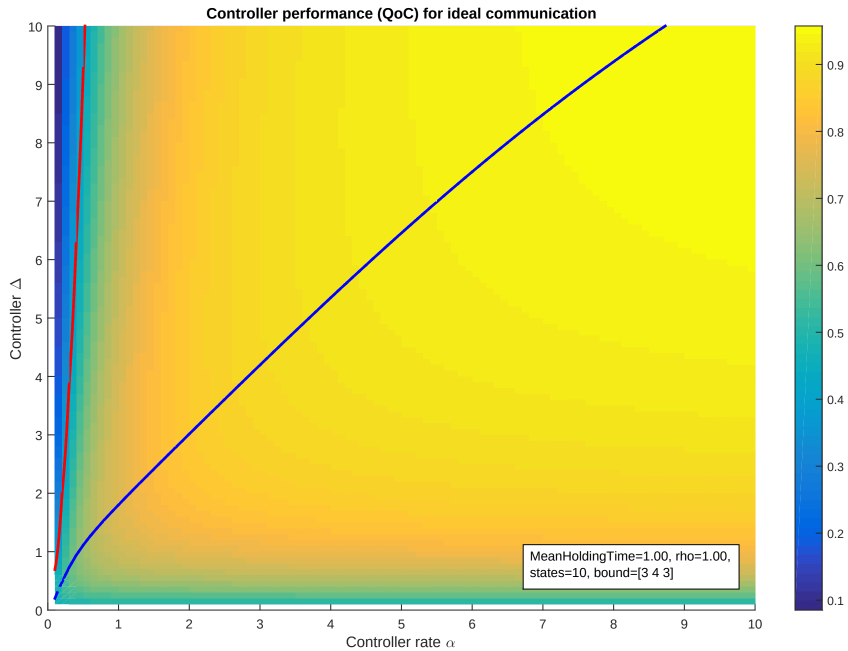



**Fig. B.11:** Control quality as a function of control rate, $\alpha$, and control drift rate, $\Delta$, with lines indicating (red) the boundary between improved or reduced QoC as a result of the controller and (blue) the optimal QoC achieved for the plant used as example.

rate only increases the QoC very little i.e. $|QoC(\mathbf{G}(\alpha)) - QoC(\mathbf{G}(\alpha = \infty)) < \delta|$ for any given parameters, and 2) if $\alpha$ is fixed and $\Delta$ can be changed, there is a possibility to select the controller drift rate either too low or too high hereby reducing the effectiveness of the controller, that is $max(QoC(\mathbf{G}(\Delta)))$ for any given parameters. To be sure that any degradation observed later on is not caused by a poor controller design we focus in the subsequent analysis on selecting $\Delta = argmax_{\Delta}(QoC(\mathbf{G}(\Delta)))$. In the next section when we assess the mismatch probability, the ideal information case shown here relates to the case when the mismatch probability is exactly 0.

## 6.2 System with remote access

In this part we now focus on the control loop with non-ideal access to plant state information. Here we look at the mismatch probability as described in Section 5 and perform an analysis of the link between this metric and control performance. The rationale for this focus as stated earlier, is that by demonstrating a clear link between mmPr and control performance we can effectively differentiate the two work processes of designing 1) control loops and 2) network configurations.

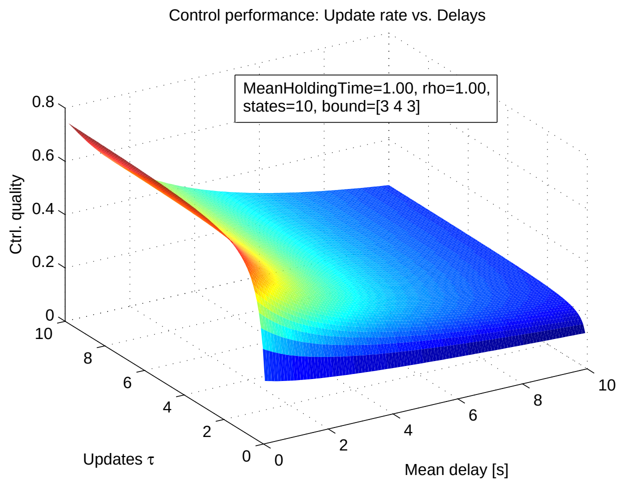
Figure B.12 shows how exponentially distributed delays impacts on mismatch probability and control quality, respectively. Not surprisingly, the higher delay, the higher mismatch probability as well as decreased control quality. In the case of ideal network, $\Delta$ was chosen so that it is optimal (which automatically means some improvement of QoC compared to no control). The case of infinite delays resembles the system without control, for which QoC is definitely degraded. The results shown in Figure B.12 appears to confirm a monotonous behaviour between mmPr and QoC for changes in network and information access parameters.

Figure B.13 shows cases when (a) the control rate, $\alpha$ and (b) the controller drift, $\Delta$ is changed. In (a) a general trend is shown that increasing the control rate also increases the control quality (as expected from previous analysis), however, the delay means that the information which the controller receives has an increased probability of not matching (an increased mismatch probability). This is clearly one case when there is no direct link between the mismatch probability and the control quality, meaning that reducing mmPr by adjusting the control rate although having an impact, is not a good idea.

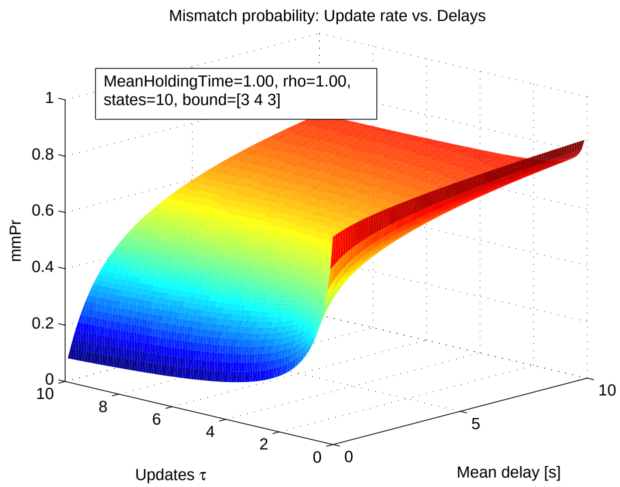
Figure B.13 (b) shows the case with varying control drift rate, $\Delta$, which also points out that not only does a maximum exists, i.e. an optimal value for the case when delay is added, but that this is very different from the ideal system. In general, this maximum goes further to the left, i.e. the control drift must be more conservative due to the delay (which is not a big surprise) as also seen that if the control drift is too high, it quickly becomes quite bad and close to the case when the system is not controlled at all. Notice that if this optimal control drift rate is not used, it is easy to end in a situation where an increased mmPr does not necessarily means a reduction in QoC, i.e. if $\Delta$ is less than the optimal value, which is what we seek to asses in the following.

## 6.3 Suitability analysis of information quality for control optimization

The main point of the work presented here is to assess the usability of mismatch probability as an effective notion for optimizing the network and access to information independent of the controller and control design. Figure B.12 shows the resulting mismatch probability and control quality as a function of the network and information access related process. In (a) the control quality is shown, illustrating not surprisingly that ideally the update rate, $\tau$ should be as high as possible and obviously, the delay $\bar{D}$ as low as possible.



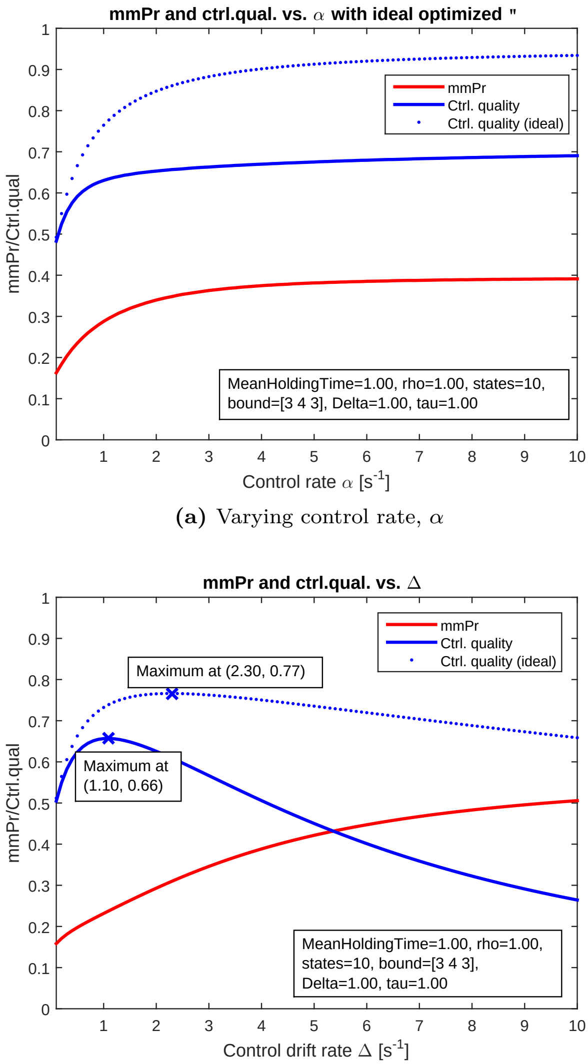

**(a)** Resulting control quality as a function of varying update rate, $\tau$ and mean network delays




**(b)** Resulting mismatch probability as a function of varying update rate, $\tau$ and mean network delays

**Fig. B.12:** Varying different control parameters for a given system over a given network.

Although there appear from these plots to be a monotonously relation between the




**(a)** Varying control rate, $\alpha$



**(b)** Varying controller drift rate ($\Delta$)

**Fig. B.13:** Varying different control parameters for a given system over a given network.

metrics the relation appears to hold only for the network parameters, i.e. an increase of update rate leads to an improved quality of control, a decrease of delay by adding quality of service to the network traffic also leads to improved quality control, while for the control parameters it is not necessarily the case.

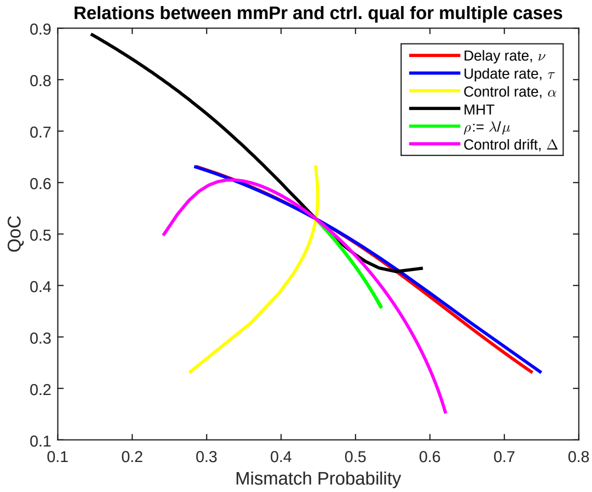



**Fig. B.14:** Multiple functions showing the relation between mmPr and control quality for varying the update rate, $\tau$, mean network delays, control rate, $\alpha$, control drift, $\Delta$, mean holding time and $\rho$.

From different cases with varying parameters, we illustrate in Figure B.14 the different relations between mismatch probability and control quality varying different parameters. As it is shown, the network related parameters, $\tau$ (update rate) and $\nu$ (delay rate) leads to a monotonous relation while for the controller related parameter $\Delta$ and $\alpha$ this is not always the case as they impact the dynamics of the plant state changes and thereby also the mismatch probability, in particular varying $\Delta$ for the specific system here always leads to a reverse non-linear relation between QoC and mmPr. This observation is also done considering changes in the plant state change rate which also in some cases reveals a non-monotonous relation between the mmPr and QoC, however, less visible. Changing the ration between the rates, $\lambda$ and $\mu$ here seems to create a monotonic relation, however, for other cases (not shown) this is not guaranteed. This leads to the conclusion that the mmPr metric is most useful for network and middle ware support designer rather than the other way of supporting the control engineer in design. However, for a given controller where a control performance is expected, a mismatch probability may then be specified for a fixed control value tuple $[\alpha, \Delta]$.

From Figure B.12 we now systematically assess the monotonicity relation by exploring the points $c_{ij} = \{(\tau_i, \nu_j)\}$ with $c_{kl} = \{(\tau_k, \nu_l)\}$ for the mmPr and QoC, respectively and $i \neq k, j \neq l$ under the constraints that $|QoC(c_{ij}) - QoC(c_{kl})| \geq \zeta$ and $|mmPr(c_{ij}) - mmPr(c_{kl})| \geq \zeta$

$$D = \frac{1}{c^2}\sum_{c_{ij}}\sum_{c_{kl}} I_{i,j}(sign(QoC(c_{i,j}) - QoC(c_{k,l})) + sign(mmPr(c_{i,j}) - mmPr(c_{k,l})) \neq 0)$$

with $c^2$ representing the total number of points $c$ which fulfils the constraints.

Due to the desirable objective of using an optimal control, we asses the failure rate $D$ as a function of deviation from optimal control. Starting from the $\Delta$ that gives the highest control quality (see Figure B.13 for which with the given delays this is at $\Delta = 1.1$) we deviate between $\pm 50\%$ of the optimal drift rate and observe the failure rates $D$. Not shown here due to space limitations, the results ranges from 0.004 to 0.014, i.e. 0.4% to 1.4% of all assessed points with a $\zeta = 10^{-3}$. This means that for even non-optimal control there are only few cases with non-monotonicity which is anyway suspected being caused by numerical issues.

Finally we check the bounds for numerical errors by changing $\zeta$. The results are shown in Figure B.15 for different cases of $\Delta$ values in relation to the optimal value.

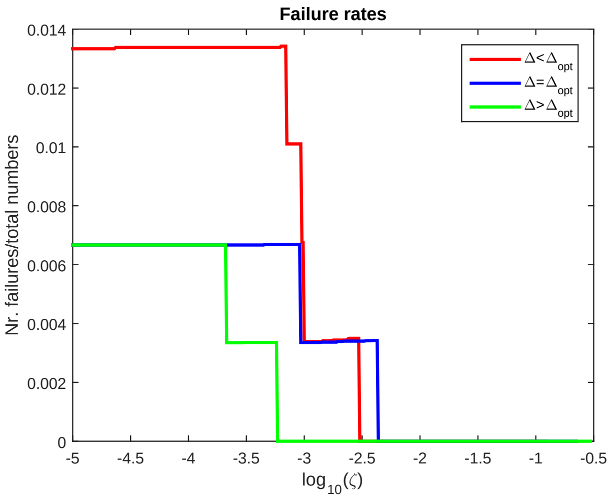



**Fig. B.15:** Monotonicity failures (fraction of all possible permutations in assessed case) with varying accuracy $\zeta$

This result confirms that for the non-optimal control there exists a theoretical monotonic relation between mismatch probability and control quality, if we eliminate numerical errors via an adequately large choice of $\zeta \geq 10^{-2.3}$ then all numerical results confirm that an improvement of mmPr is equivalent to an improvement of control performance.

## 6.4 Discussion and perspectives of results

Relevance of these findings are manifold. The main result shows that it is doable to decouple network optimization and control design effectively. For the network optimization consider that packet losses for UDP based protocols leads to dropped messages (which effectively can be seen as a decrease in update rate, $\tau$) the impact of poor network conditions (increased packet losses) leads to a reduction in control quality and increase in mismatch probability as shown in Figure B.12(a) when shifting the update rate to wards a lower value. For example, given the base rate is 10 updates per second, then only when the effective update rate is at around 2 updates/second (80% loss of update messages) starts to significantly decrease control performance. Alternatively using TCP connections, poor network quality (packet losses) are mapped into increased delays, which in Figure B.12(a) leads to a reduction of control quality over the whole range of delays. From the slopes of the surfaces this hints that the controlled system is more robust to wards packet losses rather than delay. This is reflected also by the mismatch probability as shown in Figure B.12(b). Future research will look into different system types in order to gain more generic results.
Further, based on the results shown in Figure B.13 it is also seen that the controllers optimal rate, $\Delta$, is influenced by the delay (for the case, comparing the ideal case with the case where a one second delay is present, the optimal value is reduced from $\Delta$ =2.3 to $\Delta$ =1.1). Future work will focus on determine the link between this change of optimal value relates to the delay only or if the mismatch probability can be used as well.

# 7 Conclusions

This paper is addressing the problem of optimization of information access in networked control systems. Due to the remote communication with sensors, the controller's view of the system state at the time of set-point computation may deviate from the system observations that the controller could have without remote interactions, or equivalently, a controller that would use an infinitely fast communication network and can sample the system observations at infinite rate – the latter case is called in this paper a controller with ideal information access. The probability that the controller's view does not match the current system observations at the sensors is called mismatch probability. This metric can be easier derived or measured and does not depend on the specific controller realization. The paper investigated for which type of system parameters of a networked control system, the optimization of mismatch probability is equivalent to the optimization of control performance.

In order to do this, we first performed a simulation analysis of a wind-farm controller. The results show that a choice of input parameter tuples for the so-called offset and network delay that reduces mmPr also in most cases increase control performance, measured by the resulting fatigue of the wind-farm. However, the simulations are rather time consuming and the results are subject to substantial stochastic variability. Therefore, the paper developed a simplified networked control system that can be represented as a Markov chain model. For controllers that improve system behaviour under ideal access compared to the non-controlled system, the joint optimization of communication delay and sensor sampling frequency in a periodic observation scheme for minimized mismatch probability was numerically shown to also be optimal for control performance.

The results of this paper clearly show the opportunity to use mismatch probability for automatically adapting network and access configuration to maintain an optimized quality of control, however requiring additional research in estimation of processes and understanding the impact on adaptation decisions which is left for future work.

## Acknowledgment

This work was partially supported by the Danish Council for Strategic Research (contract no. 11-116843) within in the 'Programme Sustainable Energy and Environment', under the "EDGE" (Efficient Distribution of Green Energy) research project, and by the FP7 project SmartC2Net (FP7/20072013).

## References

[1] M. consortium FP7 ICT 2011 6.7 Grant agreement no.: 318485., "Mobinet internet of mobility," 2013. [Online]. Available: http://www.mobinet.eu

[2] H.-P. Schwefel, R. L. Olsen, T. Kristensen, G. Dondossola, R. Terrugia, F. D. Giandomenico, F. Kurtz, C. Wietfeld, D. Iacono, N. Silva, and S. Marzoreto, "Smart control of energy distribution grids over heterogeneous communication networks," *Submitted to IEEE Wireless Communications*, 2016.

[3] H. van Kranenburg, M. S. Bargh, S. Iacob, and A. Peddemors, "A context management framework for supporting context-aware distributed applications," *IEEE Communications Magazine*, 2006.

[4] D.-S. Kima, Y. S. Leea, W. H. Kwona, and H. S. Parkb, "Maximum allowable delay bounds of networked control systems," *Control Engineering Practice*, vol. 11, pp. 1301–1313, 2003.

[5] H. Georg, N. Dorsch, M. Putzke, and C. Wietfeld, "Performance evaluation of time-critical communication networks for smart grids based on iec 61850," in *Computer Communications Workshops (INFOCOM)*, 2013, pp. 43–48.

[6] T. Godfrey, S. Mullen, D. W. Griffith, N. Golmie, R. C. Dugan, and C. Rodine, "Modeling smart grid applications with co-simulation," in *First IEEE International conference on SmartGrid Communications (SmartGridComm)*, 2010, pp. 291 – 296.

[7] W. Zhang, M. S. Branicky, and S. M. Phillips., "Stability of networked control systems." *IEEE Control Systems*, 2001.

[8] G. C. Walsh, H. Ye, , and L. G. Bushnell., "Stability analysis of networked control systems." *IEEE Transactions on Control Systems Technology*, 2002.

[9] J. d. J. Barradas Berglind, "Fatigue-damage estimation and control for wind turbines," Ph.D. dissertation, Aalborg University Doctoral School, 2015.

[10] T. Brade, S. Zug, and J. Kaiser, "Validity-based failure algebra for distributed sensor systems," in *IEEE 32nd International Symposium on Reliable Distributed Systems (SRDS)*, 2013, pp. 143–152.

[11] H.-P. S. M. Boegsted, R. L. Olsen, "Probabilistic models for access strategies to dynamic information elements," *Performance Evaluation...*, 2010.

[12] A. Shawky, R. Olsen, J. Pedersen, and H. Schwefel, "Network aware dynamic context subscription management," *Computer Networks*, vol. 58, no. 0, pp. 239 – 253, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128613003629

[13] J. J. Nielsen, R. L. Olsen, T. K. Madsen, B. Uguen, and H.-P. Schwefel, "Location-quality-aware policy optimisation for relay selection in mobile networks," *CoRR*, vol. abs/1505.01630, 2015. [Online]. Available: http://arxiv.org/abs/1505.01630

[14] J. Madsen, M. Findrik, T. Madsen, and H.-P. Schwefel, "Optimizing data access for wind farm control over hierarchical communication networks," *To appear in International Journal of Distributed Sensor Networks*, 2016.

[15] J. de Jesus Barradas-Berglind, R. Wisniewski, and M. Soltani, "Fatigue damage estimation and data-based control for wind turbines," *IET Control Theory & Applications*, vol. 9, no. 7, pp. 1042–1050, 2015.

[16] J. Madsen, M. Findrik, T. Madsen, and H.-P. Schwefel, "Optimizing data access for wind farm control over hierarchical communication networks," *International Journal of Distributed Sensor Networks*, vol. 2016, 2016.

[17] M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models.* Baltimore: John Hopkins, University Press, 1981.

[18] L. Lipsky, *Queueing Theory: A Linear Algebraic Approach.* New York: McMillan and Company, 1992.

# Paper C

## Impact of Communication Network Performance on Supermarket Control

Jacob T. Madsen, Tomasz Minko, Tatiana K. Madsen Hans-Peter Schwefel

The paper has been submitted to the
*Submitted to Conference on Networked Systems*, 2016.

*The layout has been revised.*

# Abstract

*Variable prices for electric energy can provide incentives for shifting power consumption in time in order to help balance the power grid. Supermarkets contain relatively large temperature controlled power consumers, which gives them some flexibility towards electricity consumption shifting. This paper considers a realization of a supermarket refrigeration system based on a hierarchical control structure where a supervisory controller interacts with local controllers via a communication network. We investigate the impact of communication network performance on the supermarket refrigeration control and resulting energy costs using simulation models. The results show that the controller is resilient to downstream information delays, however upstream delays or up- and downstream information loss can cause significant performance decrease.*


# 1 Introduction

Intelligent control of flexible loads in the power grid can support stabilizing the power grid through load shifting. Such load shifting can be achieved via time-varying prices, which create incentives to energy consumers to avoid high-price periods. Cooling devices represent one of the promising types of consumer devices for such price-based demand management as they typically target a temperature band within which they have some flexibility. Modern supermarkets contain several of such cooling devices and hence potentially can provide a large degree of aggregated flexibility, which may be economically attractive to be exploited.

This paper investigates a temperature control loop within a supermarket that is executed by a supervisory controller at the supermarket which has the knowledge about variable energy prices and interacts with multiple cooling devices. The interaction is done via a communication network and the paper investigates the impact of communication delays and message losses within this communication network on the overall performance of the control loop, measured by cost savings.

Introducing delays and information losses in the considered scenario allows us to make a more realistic performance evaluation of a controller behaviour, as well as to get a more realistic estimation of benefits that can be achieved by having a controller in a system. In case a controller and a controlled entity are physically separated by some distance and have to communicate via a communication network, the impact of information delays or losses can not be ignored, and this impact should be quantified. Typically, a degradation in controller performance is observed. However, in worst cases an ill-performing controller using stale and outdated information might perform worse compared with a case with no controller at all. The goal of this paper is to investigate how a supervisory controller in a supermarket is affected by information outdating caused by a communication network.

While this paper investigates load shifting via variable prices, other work has investigated direct load control: Reference [1] considers load-control of a large number of cooling devices directly by the utility. Reference [2] investigates a direct load control scenario with feedback from the regulated entities without considering the impact of non-ideal communication. Different controller designs can be found in the literature: Reference [3] analyses the challenges in the supermarket control scenario, and Reference [4] creates a learning based algorithm using thermostatic loads as energy buffers. The controller used in this paper is based on Reference [5], which performs load shifting on both freezers and refrigerators in a supermarket scenario, while assuming ideal communication behavior. We therefore build on top of the controller simulation from that reference and investigate how the control performance is affected when non-ideal communication networks are considered, e.g. messages may be delays and lost.
A similar approach has been realized in Reference [6], where the impact of delays on a TCP based communication via narrow-band Power-line technology is investigated for the controller of Reference [7]. Reference [8] considers a price-based general demand management system for home environments and shows that the developed approach is rather robust to delays and losses caused by the communication network. It however does not perform a detailed modeling of the controlled assets and of the local control loops associated with these assets. When looking beyond demand management, other use-cases of networked control include wind-farm control [9] and voltage control in Smart Grids [10], [11].
The rest of the paper is structured as follows. Section 2 describes the general supermarket load shifting scenario, which is further detailed in Section 3, describing the specific evaluated case study. Section 4 discusses the results of the simulations of the given scenarios.

## 2 Scenario description

In this paper we consider a supermarket supervisory controller, which uses the variable energy prices and the status of the heating and cooling devices in the supermarket in order to generate set-points for the local controllers of the cooling devices, in order to minimize the total energy costs via load shifting. We assume in this scenario that the price of electricity varies within time-intervals of 15 minutes. These variable prices are, however, assumed known one day in advance and are communicated to the supervisory controller. Figure C.1 details a general supermarket setup for heating and cooling. It consists of several zones denoting different requirements for heating or cooling, each zone containing a local controller. The local controller regulates temperature based on set-points received from the supervisory controller and based on information received from local sensors. The general supermarket setup in this paper is assumed to have a zone for each type of heating and cooling, which means a zone for freezers, a zone for refrigerators, a zone for the heating at the entrance, for the heating at the fruit department etc. This means that there are multiple local controllers in the supermarket, and the supervisory controller interacts with all these controllers. Consequence of this one-to-many relation is that the supervisory controller cannot be co-located to all local controllers, so the realization of this set up requires a communication network covering distances of some hundred meters.

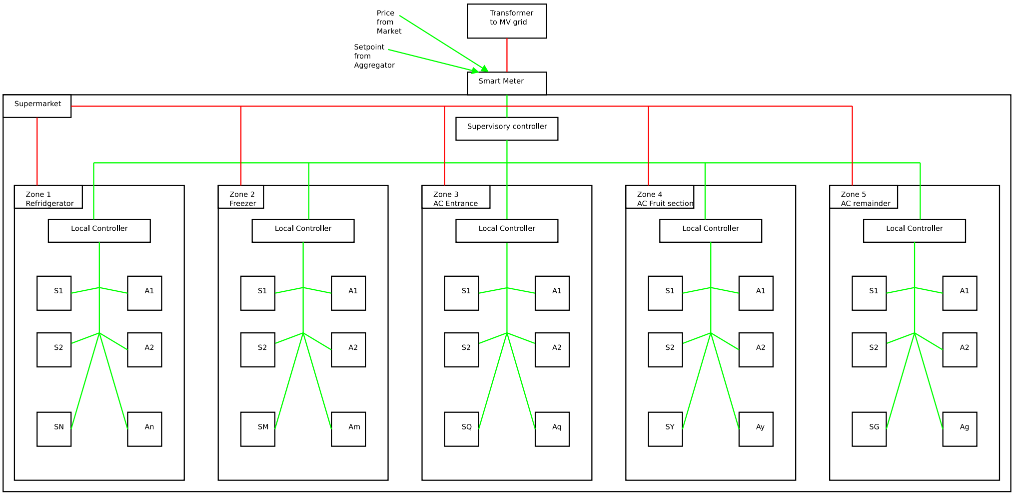



**Fig. C.1:** Overview of a general supermarket containing 5 zones, red lines denote power flow, green lines communication flow

## 2.1 Model Predictive Control Formulation

In general model predictive controllers ($MPC$) exploits the additional information provided by mathematical models of controlled systems. In addition to that they handle state and control constraints, what is very often the main motivation for using this class of controllers. In this paper we investigate the MPC for refrigeration system. In general, such system dynamics is non-linear and can be described using a set of non-linear ordinary differential equations [12] [5]. The optimal control policy (OCP) will find the best possible control policy on the time horizon $\tau := [t_0, t_\mathrm{p}]$ for a given initial point $\bar{x}_0$. The general form of the objective function can be written as (C.1). Knowing all that we can formulate the OCP problem as follows:

$$\min_{x(\cdot),u(\cdot)} \int_{t_0}^{t_\mathrm{p}} L(x(t),u(t))dt \tag{C.1}$$

subject to the initial value constraint (C.2), system dynamics (C.3) and constraints (C.4).

$$x(0) - \bar{x}_0 = 0 \tag{C.2}$$

$$\dot{x}(t) - f(x(t),u(t)) = 0,\ t \in [t_0, t_\mathrm{p}] \tag{C.3}$$

$$h(x(t),u(t)) \geq 0,\ t \in [t_0, t_\mathrm{p}]. \tag{C.4}$$

We obtain the close loop optimal control problem by introduction of instantaneous feedback and repeated solution of the open loop OCP [13], [14]. The approach taken in [5] uses discrete MPC with objective function (C.5). $L(x(t),u(t))$ consist of three terms, see (C.5). $J_{ec}$ is chosen such it minimizes the operational cost of the compressor, second term $J_{\Delta u}$ penalizes the rate of change of the set-points and the last one $\rho_\epsilon \epsilon$, penalizes the softening slack variables of inequality constraints. Systems dynamics (C.3) is put into a convex, linear form (C.6), for which an explicit Euler integrator is used. Furthermore the objective function is convex. In addition to that box constraints for states (C.7) and control inputs (C.8) are introduced, that guarantee safe operation of system devices.

$$\min_{u,\epsilon} (J_{ec} + J_{\Delta u} + \rho_\epsilon \epsilon) \tag{C.5}$$

$$\textit{subject to}$$

$$x_{t+k+1|t} = Ax_{t+k|t} + Bu_{t+k|t} + Ed_{t+k|t} \tag{C.6}$$

$$\textit{state constraints} \tag{C.7}$$

$$\textit{input constraints} \tag{C.8}$$

The acquired linear model is used in order to generate feasible state and control trajectories. After computing optimal values, non linear model plant is updated according to the receding horizon policy. For more detailed modelling and description of optimization algorithm please refer to [5] [15] [16].

## 2.2 Communication Network

In order for the supervisory and local controllers to interact, a message exchange over a communication network is needed. If the supervisory controller is located at the supermarket, then there must be a communication network covering some hunderd meters of distance. Due to deployment and operational costs, and because the supermarket does not wish to deploy its own hardware, a cloud or fog based solution may also be implemented for the supervisory controller in future scenarios. In the latter case, communication via a wide-area network may be needed in addition to the local network in the supermarket. We abstract from the specific communication network realization here in this paper and utilize large ranges of resulting delay and loss to cover both scenarios. The load on the communication network and the resulting delay of information is also depending on the specific message exhange procedure between supervisory controller and local controller. For an illustration of this we refer to the message sequence diagram in Figure C.2, which shows a supervisory controller executing with a fixed, deterministic period, marked by the short horizontal line. In the general case, the supervisory controller may receive information in real-time on energy prices from a market and it may even provide information on consumption levels to the market; we however simplify the scenario in this paper and assume that prices are known beforehand. The supervisory controller receives information from the local controllers (containing the state of the local controller, i.e. temperature measured by the unit), and it in turn sends set-points to these local controllers. The local controllers also acts with a fixed, faster period. The local controller receives information from sensors, and acts by sending information to the actuators.// Focus in this paper is on the communication between supervisory controller and local controller.

# 3 Evaluation scenario

The scenario we consider in this paper is of a supermarket containing a single zone, which contains both freezer and refrigerator systems. These are regulated by a local controller, which receives set-points from the supervisory controller. The supervisory controller uses a predetermined price signal, which covers a 6 day period. The system has two other inputs which it has no control over, one being the indoor temperature. This input is modelled as a sinusoidal function which fluctuates between 19 and 21 degrees Celsius. In a general scenario this temperature would be controlled by another local controller, however, in the scenario we are working we only consider the control of fridges and freezers. The other input is the outdoor temperature which fluctuates between 8 and 20 degrees Celsius. This temperature is real world temperature measurements, taken from the same days as the price signals are taken from.

The local controller tracks the reference signal provided by supervisory controller and sends information to the supervisory controller every minute, and the supervisory controller responds with set-points every 15 minutes. The local controller system is, from our point of view, a black box system, and as such we are unable to consider the effects of communication networks within this subsystem. We assume that whenever the local controller receives a new set-point from the supervisory controller it acts on it immediately, and whenever it requires a sensor reading to send to the supervisory controller, this information is always readily available to the local controller.

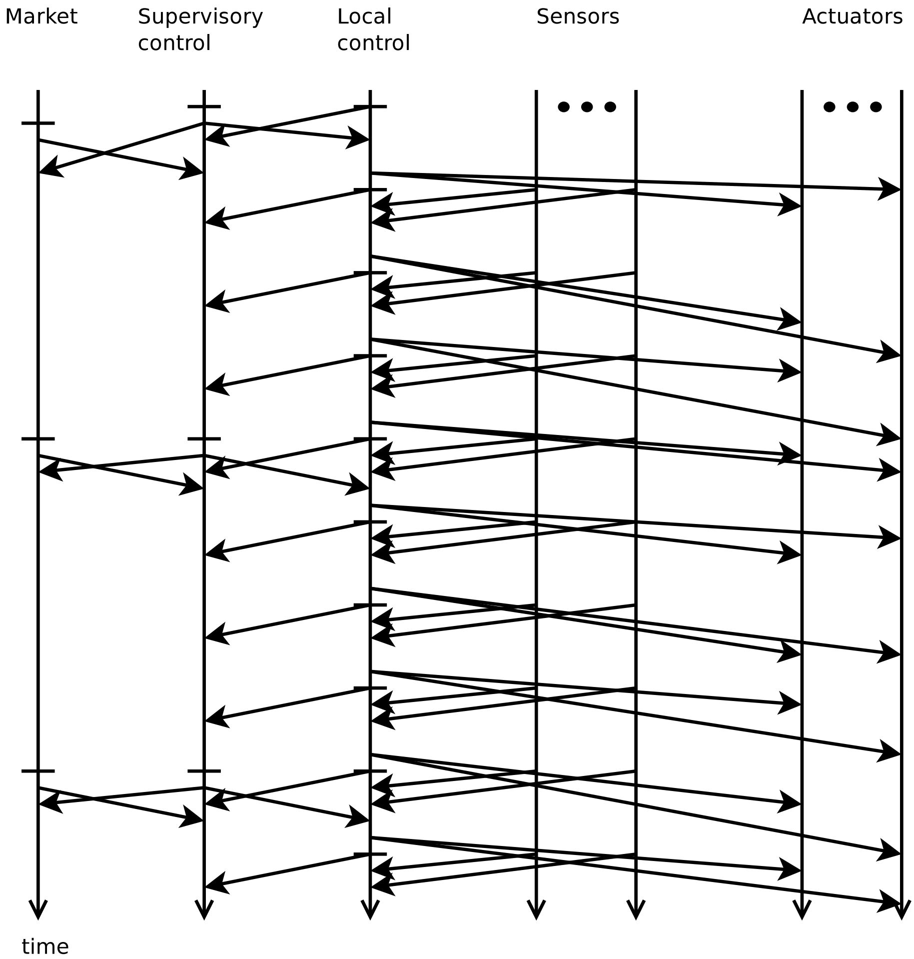



**Fig. C.2:** message sequence diagram of the full super market communication

The system modelled is a large-scale $CO_2$ refrigeration system, with 3 subsystems modelled; display cases, suction manifold and condenser. For more information on the specific controller, and its subsystems, used in this scenario we refer to Reference [5].

For the communication network we consider the following scenario for the system. The supervisory controller and local controller are connected via an IP network using UDP. We introduce either delays or packet loss to this network to simulate different network behaviours. We neglect computation times in this paper and only focus on the communication delays and delays introduced by the message patterns for interactions between local controller and supervisory controller. We assume that when the time $T_{supctrl}$ is reached, the supervisory controller performs a calculation of a new set-point, based on its current information, and send this to the local controller. The supervisory controller is executed every 15 minutes. For the local controllers we assume that they perform its control at time $T_{locctrl}$, and with a period of one minute, and based on this execution they send a new state message to the supervisory controller. We have chosen that the local controllers start $55s$ after the supervisory controller and it results in an offset between the execution of the supervisory controller and the next closest local controller execution time which is 5 seconds. We assume that the controllers are synchronized and thus, the offset remains constant. This means that if a message is delayed more than 5 seconds, the decision will be taken on the basis of older infomration available.

We determine two different information streams, and call the one from the supervisory controller to the local controller downstream, and the one from the local controller to the supervisory controller upstream.

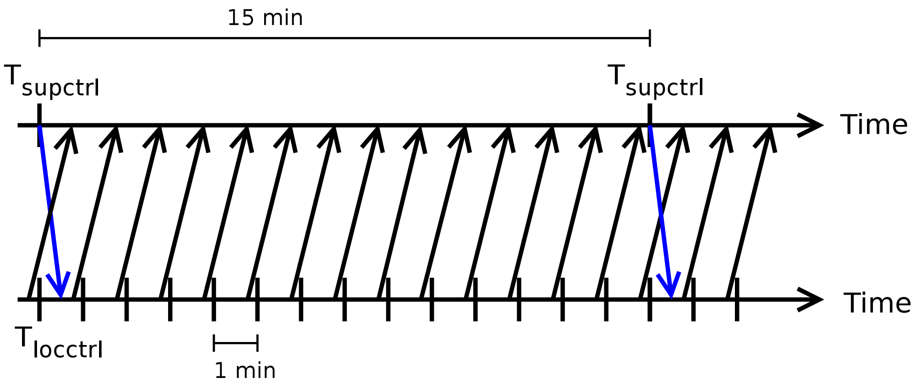



**Fig. C.3:** Message sequence diagram of the super market communication

# 4 Impact of imperfect networks on controller performance

Since the electricity cost spent on supermarket running depends greately on electricity prices and outdoor temperature, and as a consequence it is varying from one day to another, we have chosen to use another metroc, namely the normalized price for running the supermarket, for our performance evaluation. The normalization is done compared to the supermarket running with an ideal communication network and access strategy., when the necessary information is available at a controller instantaneously without delays and losses. The cost of running the supermarket is calculated as a product of the power consumption and the electricity price in a corresponding time period. The daily cost can be found by integration of the product over duration of an entire day. In the considered case, the electricity price is constant for 15 min periods, thus integration can be substituted by summation over 15 min time intervals.

We make two different communication scenarios, one where we increase the delay of the packets, and another where we increase the packet loss of the communication network between supervisory controller and local controller.

## 4.1 Delay results

The first simulation experiment is done with deterministic communication delays which are constant and identical for upstream and downstream. We simulate results for 4 different days with distinct differences in price and temperature data. We also consider the case where there is no control of the system, and we denote that in Table C.1 as $\infty s$ delay, and we see that the total costs of electricity for running the supermarket is 19 % to 36 % higher when there is no control. We then consider the intermediate levels, shown in Table C.1, by setting delay to be equal to 1 sec, 60 sec or 900 sec. For 1 sec delay case there is just a slight degradation in the performance. Indeed, in this case, messages from the local controllers are delayed for 1 sec and they arrive 4 sec before the supervisory controller should make a decision. The information is received in time, but the content of the message is 5 sec old, compared with the ideal case when the information is available instantaneously. This explains the observed slight degradation in performance. At 60 sec delays performance is decreased up to 30 % on some days, and increasing the delay to 900 sec does not decrease the performance significantly more. With no control of the system we see a further decrease in performance, as expected.

In order to investigate whether the downstream or the upstream delay contribute more significantly to the performance decrease of the simulation, we perform simulations with the same delay values as previously, however delay is only imposed on one direction, while the delays in the other direction are set to zero. The results are found in Table C.2 and C.3. The results show that the upstream delays are the largest contributor to the performance decrease. We expect the primary reason that downstream delays are

**Table C.1:** Normalized price for running supermarket with symmetric delays

| Scenario | Day 1 | Day 2 | Day 3 | Day 4 |
|---|---|---|---|---|
| Ideal network | 1.00 | 1.00 | 1.00 | 1.00 |
| 1$s$ delay | 1.03 | 1.02 | 1.08 | 1.04 |
| 60$s$ delay | 1.13 | 1.16 | 1.31 | 1.32 |
| 900$s$ delay | 1.13 | 1.17 | 1.32 | 1.3155 |
| $\infty s$ delay | 1.20 | 1.19 | 1.36 | 1.35 |

**Table C.2:** Normalized price for running supermarket with only downstream delays

| Scenario | Day 1 | Day 2 | Day 3 | Day 4 |
|---|---|---|---|---|
| 1$s$ delay | 1.00 | 1.00 | 1.00 | 1.00 |
| 60$s$ delay | 1.00 | 1.00 | 1.01 | 1.01 |
| 900$s$ delay | 1.03 | 1.03 | 1.09 | 1.05 |

so insignificant, is that the set-points from the supervisory controller have very little variation, and thus it does not have a large impact if the messages are delayed more than one supervisory control time period, ie. 15 minutes.

## 4.2 Packet loss results

In order to understand the impact of packet losses, we run the simulations with three different packet loss values. The first results for the simulation with packet loss are for downstream packet loss only, while upstream communication is fully reliable. Packets are not dropped and are delivered instantaneously, ie. delay is zero. The results can be seen in Figures C.4 and C.5. All results are shown with 95 % confidence intervals, based on 9 independent repetitions. Figure C.4 shows the normalized price for the supermarket on one specific day for increasing packet losses for the downstream communication. We see that for low packet losses the results are close to the same as for the ideal

**Table C.3:** Normalized price for running supermarket with only upstream delays

| Scenario | Day 1 | Day 2 | Day 3 | Day 4 |
|---|---|---|---|---|
| 1$s$ delay | 1.00 | 1.00 | 1.00 | 1.00 |
| 30$s$ delay, 59$s$ offset | 1.00 | 1.00 | 1.00 | 1.00 |
| 30$s$ delay | 1.13 | 1.16 | 1.32 | 1.32 |
| 60$s$ delay | 1.13 | 1.16 | 1.31 | 1.32 |
| 90$s$ delay, 59$s$ offset | 1.13 | 1.16 | 1.31 | 1.32 |
| 900$s$ delay | 1.13 | 1.16 | 1.31 | 1.32 |

communication network scenario, however as the packet loss increases the normalized price increases as well. The results are not averaged over the days that are simulated, as each day did show significant changes in the price data, and as such it was found that averaging across days could hide relevant information for specific days. We do not show each specific day in this paper, however the other three days showed similar behaviour to the day shown.

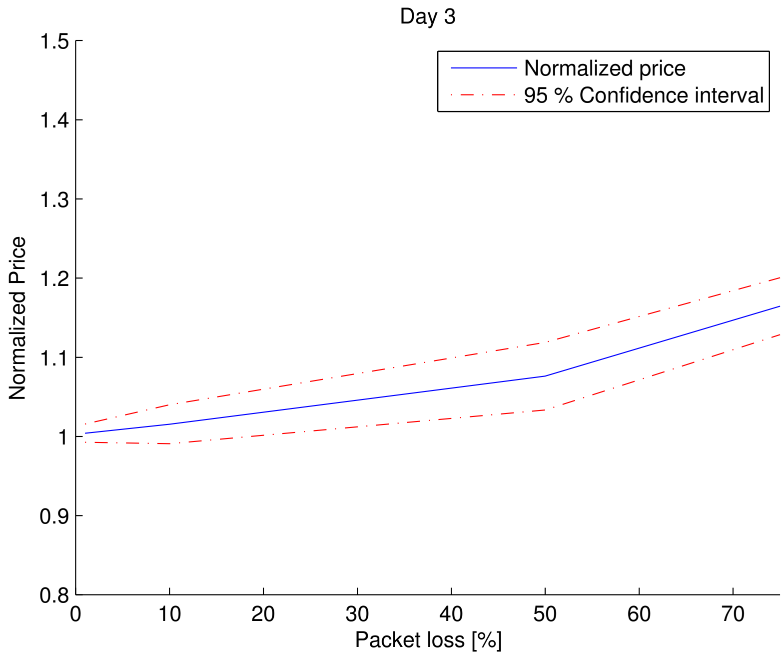



**Fig. C.4:** Result showing the increase in price, normalized to ideal case, over increasing packet losses for downstream communication, day 3

The results on Figure C.5 shows the same only for the upstream communication. The results show that the upstream packet loss impacts the overall system performance more strongly as compared to the downstream packet loss. The results, however, also show a high variability, which leads to wide confidence intervals for the shown relative mean energy cost.

### 4.3 Exponential delays

Our investigations in case of deterministic delays show that if the last packet before the time $T_{supctrl}$ was delayed past $T_{supctrl}$, this has a significant impact on the controller

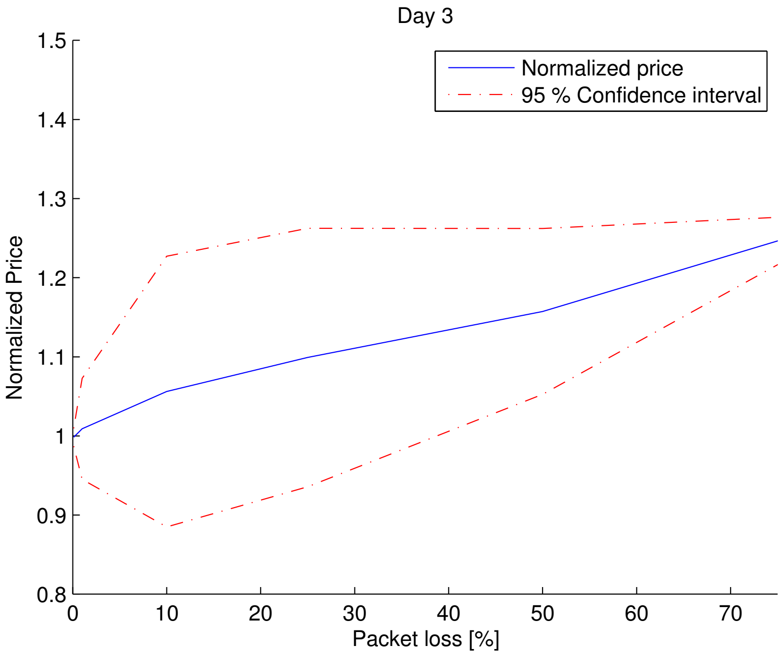



**Fig. C.5:** Result showing the increase in price, normalized to ideal case, over increasing packet losses for upstream communication, day 3

**Table C.4:** Table showing which packet losses corresponds to which mean delays for the exponential distribution

| Packet loss | 0.1 | 1 | 5 | 10 | 50 | 75 |
|---|---|---|---|---|---|---|
| Mean delay | 0.7238 | 1.0857 | 1.6690 | 3.1067 | 7.2135 | 17.3803 |

performance. This happens when the last upstream message is delayed more than 5 seconds.

The assumption of constant delay does not hold for many communication systems where packets are delivered with varying and unpredictable delays. To model this situation we introduce delays that are exponentially distributed.

To visualise the controller performance, instead of making a plot over the mean delay, we have chosen to plot the performance metric over the probability that the delay exceeds 5 seconds. Following the argumentation presented above, this corresponds the case when the last received message is delayed above the offset. Therefore, these delay plots could be compared to the upstream packet loss graphs. The probabilities and the mean delays are shown in Table C.4.

The comparison of Figure C.5 and Figure C.6 shows that when there is a 30 % probability of the last packet being delayed, the normalized price is slightly lower than when there is 30 % probability that packets are lost, however they are within each others confidence intervals. To the right end of the curves, the results match in the two graphs; therefore the impact of the exponential delay distribution is mainly due the tail probability $Pr(D > 5s)$. Other delay distributions are expected to behave similarly.

# 5 Conclusions

In this paper we evaluate the impact of communication network performance on a supermarket supervisory controller which attempts to reduce the electricity costs of the supermarket. We defined the communication flows between supervisory controllers and local controllers at the supermarket zones. We found that the supermarket controller was very robust to delays in the downstream communication, however it was rather susceptible to packet loss. We also found that for the upstream communication it was impacted greatly by delays that where larger than the offset, which is the time interval between the generation of the last update message from the local controller and the execution of the supervisory control computation. Due to the implementation of the local and supervisory controller, this offset (and subsequently the delays) could be as large as 59 seconds which would provide additional robustness to many communication scenarios. We also found that packet loss in the upstream communication was less severe than for the downstream communication, however it still impacted the performance of the controller.

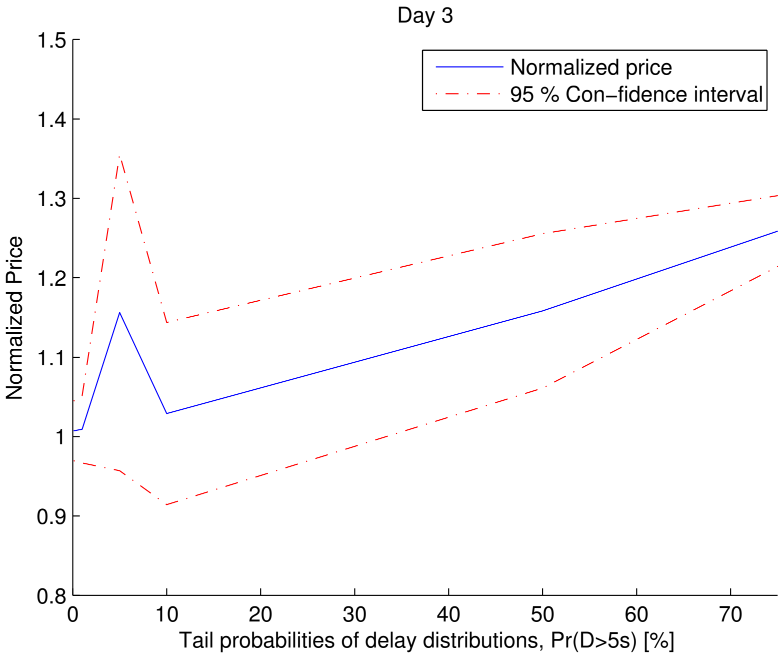



**Fig. C.6:** Result showing the increase in price, normalized to ideal case, over increasing delays for upstream communication, day 3

Future work would include finding an optimal trade-off between delays and packet loss for the upstream communication, as it can for instance be achieved via retransmission schemes such as implemented by TCP. Another topic to investigate in this area is how to ensure that communication in the supermarket is secure, and how malicious third party attacks would impact performance. A similar analysis was done for communication for voltage control in medium-voltage grids in Reference [17].

## Acknowledgment

This work was partially supported by the Danish Council for Strategic Research (contract no. 11-116843) within in the 'Programme Sustainable Energy and Environment', under the "EDGE" (Efficient Distribution of Green Energy) research project. The Authors would like to thank Ehsan Shafiei for making his work available to us.

## References

[1] L. C. Totu, J. Leth, and R. Wisniewski, “Control for large scale demand response of thermostatic loads,” in *2013 American Control Conference*. IEEE, 2013, pp. 5023–5028.

[2] S. H. Tindemans, V. Trovato, and G. Strbac, “Decentralized control of thermostatic loads for flexible demand response,” *IEEE Transactions on Control Systems Technology*, vol. 23, no. 5, pp. 1685–1700, Sept 2015.

[3] T. G. Hovgaard, L. F. Larsen, M. J. Skovrup, and J. B. Jørgensen, “Analyzing control challenges for thermal energy storage in foodstuffs,” in *Control Applications (CCA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 956–961.

[4] K. Vinther, H. Rasmussen, R. Izadi-Zamanabadi, J. Stoustrup, and A. G. Alleyne, “Learning-based precool algorithms for exploiting foodstuff as thermal energy reserve,” *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, pp. 557–569, 2015.

[5] S. E. Shafiei, H. Rasmussen, and J. Stoustrup, “Modeling supermarket refrigeration systems for demand-side management,” *Energies*, vol. 6, no. 2, p. 900, 2013. [Online]. Available: 2

[6] J. T. Madsen, T. le Fevre Kristensen, R. L. Olsen, H. P. Schwefel, and L. C. Totu, “Utilizing network qos for dependability of adaptive smart grid control,” pp. 859–866, May 2014.

[7] L. C. Totu, J. Leth, and R. Wisniewski, "Control for large scale demand response of thermostatic loads*," pp. 5023–5028, June 2013.

[8] S. Bessler, D. Drenjanac, E. Hasenleithner, S. Ahmed-Khan, and N. Silva, "Using flexibility information for energy demand optimization in the low voltage grid," in *Smart Cities and Green ICT Systems (SMARTGREENS), International Conference on.* IEEE, 2015, pp. 1–9.

[9] J. Madsen, M. Findrik, T. Madsen, and H.-P. Schwefel, "Optimizing data access for wind farm control over hierarchical communication networks," *International Journal of Distributed Sensor Networks*, vol. 2016.

[10] M. Findrik, J. Groenbaek, and R. L. Olsen, "Scheduling data access in smart grid networks utilizing context information," in *Smart Grid Communications (SmartGridComm), IEEE International Conference on.* IEEE, 2014, pp. 302–307.

[11] M. Findrik, T. le Fevre Kristensen, T. Hinterhofer, R. L. Olsen, and H.-P. Schwefel, "Information-quality based lv-grid-monitoring framework and its application to power-quality control," in *International Conference on Ad-Hoc Networks and Wireless.* Springer, 2015, pp. 317–329.

[12] T. Minko, R. Wisniewski, J. D. Bendtsen, and R. Izadi-Zamanabadi, "Cost efficient optimization based supervisory controller for supermarket subsystems with heat recovery," in *Control Conference (ECC), 2015 European*, July 2015, pp. 2321–2327.

[13] M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer, "Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations," *Journal of Process Control*, vol. 12, no. 4, pp. 577–585, 2002.

[14] J. M. Maciejowski, *Predictive control: with constraints.* Pearson education, 2002.

[15] L. F. S. Larsen, *Model based control of refrigeration systems.* Citeseer, 2006.

[16] J. M. Maciejowski, *Predictive control: with constraints.* Pearson education, 2002.

[17] G. Dondossola and R. Terruggia, "Cyber security of smart grid communications: Risk analysis and experimental testing," in *Cyber Physical Systems Approach to Smart Electric Power Grid.* Springer, 2015, pp. 169–193.

# Paper D

## Utilizing Network QoS for Dependability of Adaptive Smart Grid Control

Jacob Theilgaard Madsen, Thomas le Fevre Kristensen, Rasmus L. Olsen
Hans-Peter Schwefel, Luminita C. Totu

The paper has been published in the
*Energy Conference (ENERGYCON), 2014 IEEE International* pp. 859-866, 2014.

# Abstract

*A smart grid is a complex system consisting of a wide range of electric grid components, entities controlling power distribution, generation and consumption, and a communication network supporting data exchange. This paper focuses on the influence of imperfect network conditions on smart grid controllers, and how this can be counteracted by utilizing Quality of Service (QoS) information from the communication network. Such an interface between grid controller and network QoS is particularly relevant for smart grid scenarios that use third party communication network infrastructure, where modification of networking and lower layer protocols are impossible. This paper defines a middleware solution for adaptation of smart grid control, which uses network QoS information and interacts with the smart grid controller to increase dependability. In order to verify the methodology, an example scenario of a low voltage grid controller is simulated under imperfect network conditions.*


# 1 Introduction

Energy production in todays' power grid is mainly done using non-renewable energy sources [1], it is, however, desired to rely more and more on several types of renewable energy sources in the future. Private wind turbines and solar panels are gaining popularity at low voltage levels, meaning that energy production is changing from a more centralized system, to a highly distributed system, as well as part of the energy production moving from the high voltage(HV) and medium voltage(MV) layers to the low voltage(LV) layer. To enable the power grid to handle future requirements of a highly dynamic energy production and consumption at this level, it must be possible for the distribution grid to have better control of grid assets.

To facilitate this added control intelligence in smart grid systems, a communication infrastructure must be in place as well as functionality to allow grid asset control, which for the LV grid is not yet implemented. Since the assets are highly distributed, dependable communication is required to allow proper grid operation, however, this is not without challenges since several trade-off's have to be made, [2].

Because dependable communication infrastructures tend to be expensive, either a suitable compromise between dependability and price must be found or the dependability of economically feasible communication infrastructures must be increased. Therefore, low complexity solutions to control grid assets over existing (or low cost deployment communication networks), but imperfect communication networks are required. This paper illustrates the example of asset control in distribution grids with poor communication network performance.

The control scenario in this paper is based on a smart grid communication scenario that uses an open and heterogeneous communication infrastructure [3]. This open communication infrastructure may be a third party public communication network, where the IP stack must be used, and access to lower communication network layers is restricted. The solution proposed in this paper will, therefore, only contain communication network adaptations on layers above the transport layer. A consequence of the use of a public heterogeneous communication network is that QoS will change dynamically over time. This paper presents a solution which adapts a smart grid controller to counteract poor communication network performance, and gain improved controller performance. The conceptual approach of adapting control to network QoS is illustrated in Figure D.1; the figure shows the control loop of grid assets in the outer circle. The required communication infrastructure is monitored and information on the network performance is used to adapt the control approach via middleware functionality. The latter corresponds to the upward red arrow in the figure; the downward direction, adjusting network QoS to control demands, is treated in [3] but beyond the scope of this paper.

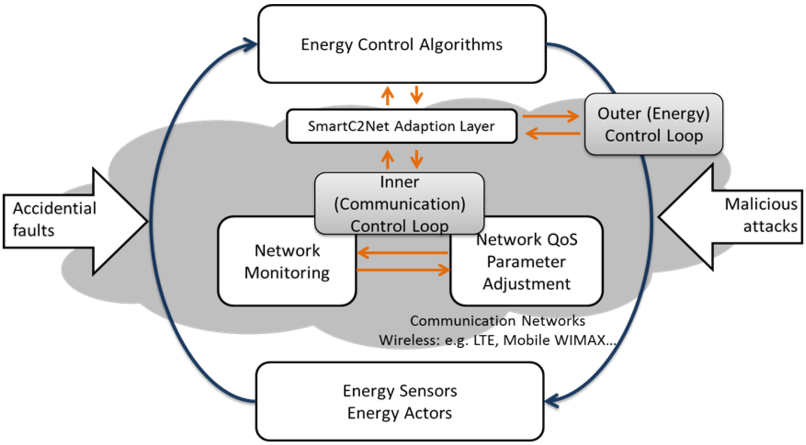



**Fig. D.1:** Conceptual dual loop for dependable smart grid operation, [3].

The proposed solution will be implemented for a given LV grid controller controlling a population of refrigerators. The adapted controller will then be simulated to illustrate the effectiveness of the adaptation method.

The rest of the paper is structured as follows; first an overview of the existing state-of-the-art is provided. Then, in section 3, an overview of the system architecture and generalized protocol stack is given. Section 4 describes the example control solution as well as the proposal for extension, which realizes the adaptation of the control system to the communication network properties. Section 6 describes the simulation approach and the evaluation results from the simulation experiments, demonstrating the benefits of the proposed adaptation scheme.

# 2 State of the art

Reference [4] gives an overview of the different technologies available for smart grid communication, and lists the relevant standards. In Ref. [5] the viability of using power line communication (PLC) is investigated through simulation. Two different topologies for the smart grid communication are proposed and then simulated. This paper sets certain requirements that the communication should adhere to, and investigates if it is possible to meet these requirements using PLC. The results show that PLC can fulfill the stated requirements in only a limited set of scenarios, and hence more relaxed communication requirements need to be deployed, or a different communication solution must be used. The poor performance of PLC is also shown in Ref. [6], where it is shown that under certain conditions, bit error rates in the order of $10^{-2}$ can be expected.

Adapting a control algorithm to the communication network QoS has been discussed in different papers, but it is scenario dependent which adaptation scheme is the most suited for the task. Adaptation should preferably be done without an extensive restructuring of the control algorithm. According to Ref. [7], it is possible to adapt the control algorithm without major changes. The idea is to modify the control loop time while leaving the parameters of the control algorithm unchanged. While this algorithm is not optimal, it can improve the overall performance if the increased control loop time leads to lower communication network traffic, thus ensuring all data in the communication network reaches its destinations before the next sample time. Furthermore, Ref. [7] shows that a control algorithm with variable control loop time will still be stable under certain conditions. A different method for adapting the control algorithm to the communication network state is presented in Ref. [8]. The distributed control algorithm is changed dynamically depending on the QoS of the communication network as estimated by the distributed controllers. This approach changes the control loop time, which either leads suboptimal control, or recalibration of control parameters. Ref. [9] proposes to adapt the network to the controller by choosing the optimal throughput to allocate to based on different power cost functions.

# 3 System architecture and use case

In this section we introduce the system architecture and the example case on which we focus our work. Furthermore, we describe the embedding in the communication stack that allow the transparent adaptation of the controller to changing communication network properties.

## 3.1 System architecture and assumptions

The adaptation approach in this paper focuses on the low-voltage grid controller, whose embedding in a hierarchical overall smart grid control architecture is shown in Figure

D.2 [10]. From higher level, the LV grid controller receives set points used as a reference for optimal power consumption. Due to the restricted number of assets in the LV grid, this consumption may not always be achievable, and so, power may flow in and out via the transformer station as needed, but following at best effort the level of a reference signal from the medium voltage (MV) grid controller. However, the in- and out-flux of power between the grid domains should be kept at a minimum for a reliable operation (the more predictable a LV grid domain is to the MV controller, the less effort it is to maintain this, and chances for energy waste is reduced). Toward the MV grid controller, the LV controller thus offers a flexibility service in terms of following set points given by the MV controller. In this paper we focus on the LV grid controller and assume that the set points from the MV grid controller are given.

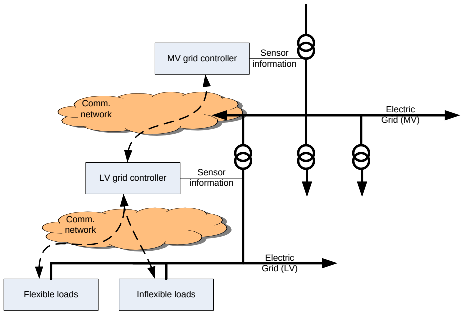



**Fig. D.2:** Hierarchical Smart Grid control architecture [10].

To achieve the goal of balancing production and consumption, the LV grid controller is envisioned to be able to interact with the assets connected to the LV grid. We distinguish between two different types of assets:

- Flexible loads: These assets may be controlled fully by the LV grid controller.
- Inflexible loads: These assets cannot be controlled, but rather show a stochastic behaviour.

We refer to Ref. [11] for more information on these types of assets. As shown, the LV grid controller needs to interact via a communication network with the assets. In the next section we propose some addition to the communication network protocol stack to improve the controller performance under imperfect network conditions.

## 3.2 Adaptive communication network functionality

We describe a middleware solution that performs adaptation of the control signal and is performed based on the QoS of the used communication network (Downstream Adaptation). Using this middleware, the controller can focus on the originally targeted control operation while the adaptation to changing network performance is taken care of by the middleware. Further, we also propose similar middleware adaptation for access to measured sensor data (Upstream Adaptation). To be able to perform this adaptation based on communication QoS, packet loss estimates must be provided by the network monitoring functions. These QoS estimations can be used to adapt the control signals. The proposed middleware will be placed on the controller side of the communication network for both downstream and upstream adaptation. An overview of the solution and how it interacts with a control system can be seen in Figure D.3.

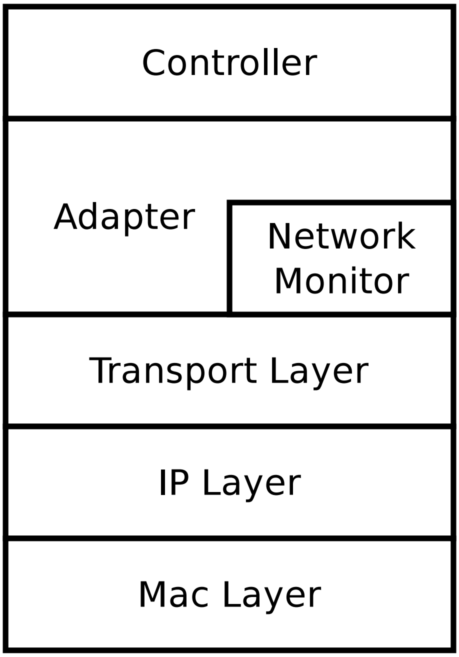



**Fig. D.3:** Illustration showing how the proposed solution is placed in the communication network stack.

Figure D.3 also illustrates where the adapter and the communication network monitor is placed in the communication network stack. As seen on the figure, the communication network monitor is placed partly between the transport layer and the adaptation layer. This allows the communication network monitoring to be done using passive network monitoring.

# 4 Example Smart Grid Control Scenario

In this section we describe an example LV grid controller scenario that will be controlling the assets as previously described. The scenario considers the control of power consumption of thermostatic loads in a LV grid scenario.

## 4.1 Regulation of thermostatic loads

A block diagram of the control scenario is shown in Fig. D.4, for which the core design without the communication network, has previously been published in [12]. This structure contains three main elements:

- A number $N$ of refrigerator units $\mathbf{F}_i$ each equipped with a modified thermostat controller $K_i$, $i \in \{1...N\}$, and each wih the power consumption $y_i$.
- The cumulated inflexible power consumption $w$ that acts as a predictable disturbance.
- The main controller consisting of the supervisor control, estimation and dispatch blocks.

The objective of the controller in this scenario is to reduce the peaks in the overall power consumption by effectively utilizing the characteristics and restrictions of the grid assets and coordinating the power intake of the asset groups. For the case study we look at thermal energy storage in terms of refrigerators. These refrigerators will by the objective of the control thus be timely coordinated to increase power consumption, and thereby store ”coldness”, before power peaks caused by the inflexible loads and then coordinated to safely turn off during that peak, and thereby use the previously stored ”coldness”. For refrigerators, the challenges are that these are restricted in terms of operating temperatures, which from time to time disallow operation of them as desired.

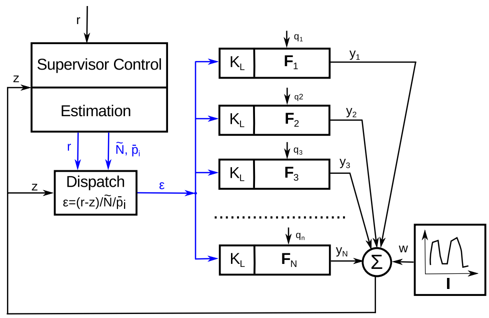



**Fig. D.4:** Control of thermostatic loads [12].

The operation of the control algorithm can be sketched as follows:

**for** *each minute* **do**
  Estimate Refrigerators States, $\tilde{N}^{off}$,$\tilde{N}^{on}$;
  Obtain Power Consumptions, $z$, $\bar{p}$;
  Compare to reference signal, $r$;
  Calculate Control signal, $\epsilon$;
  Dispatch signal to refrigerators;
  **for** *Each Refrigerator* **do**
    React to received signal, $\epsilon$;
  **end**
**end**

$\tilde{N}^{on}$, and $\tilde{N}^{off}$ are the estimated number of refrigerators that are in the *on* or $off$ state. $z$ and $\bar{p}$ is the measured power consumption of the LV grid, and the average power consumption of a refrigerator respectively. $\bar{p}$ is assumed known by the controller, but in practicality would be provided by each refrigerator when registering, and then averaged.

The LV grid controller then coordinates with the MV grid controller, and calculates a power consumption reference $r$ for a given time horizon, e.g. a day. In this paper we delimit ourselves from deriving the reference signal $r$, and the used reference is therefore generated artificially based on the used behaviour of the inflexible loads in the considered LV grid. The exact behaviour of the reference signal is not in focus in this paper, as we want to analyze the concept of adaptations based on imperfect communication networks and not include MV grid control behaviour.

$$\tilde{\epsilon}_k = \begin{cases} \frac{(r_k - z_k)}{\tilde{N}_k^{off} \cdot \bar{p}}, & \text{for } r_k > z_k \\ \frac{(r_k - z_k)}{\tilde{N}_k^{on} \cdot \bar{p}}, & \text{for } r_k < z_k \end{cases} \tag{D.1}$$

$$\epsilon_k = \begin{cases} \tilde{\epsilon}_k, & \text{for } -1 \le \tilde{\epsilon}_k \le 1 \\ sign(\tilde{\epsilon}_k), & \text{otherwise} \end{cases} \tag{D.2}$$

The control signal, $\epsilon$, is calculated according to Equation (D.1) and (D.2) and sent to the refrigerators. The control signal indicates the fraction of refrigerators in the right state that should change their state; if $\epsilon < 0$ then a fraction of $\epsilon$ refrigerators should switch to the *ON* state, if $\epsilon > 0$ in the reverse direction. For example, an $\epsilon$ value of 0.2 indicates a request of the supervisor center for 20% of the refrigerators currently in the $off$ state to switch to the *on* state, which as being executed gives an overall increase in power consumption, but at the same time adds to the thermal energy storage in the form of "coldness". Similarly, an $\epsilon$ value of $-0.2$ indicates a request of the the supervisor center for 20% of the refrigerators currently in the *on* state to switch to the $off$ state, which translates into an overall power consumption decrease, and a loss of "coldness".

The change of state at the i'th refrigerator leads to a power consumption, $y_i$, in which the total consumption of the refrigerators, $z$, are simply the sum of all these. Since there are many other loads on the electrical grid, noises and disturbances, $w$ (from inflexible loads in this specific scenario), is added to the measured load signal. The controller would need to estimate the controllable load of the refrigerators $y$, and thus the average power consumption of a single refrigerator, $\bar{p}$, is multiplied by the estimate of how many refrigerators are *on* or how many are $off$. All this happens at discrete points in time, notated by the index $k$ for the $k$'th step of the control.

In order to ensure that refrigerator temperature (and other potential constraints) are not broken at the individual unit, the actual power consumption decision is taken by the local refrigerator controller $K_i$. Every refrigerator in the group receives the same message, in case of perfect communication, and the dispatch is carried out by a local randomization in the following way. If the refrigerator unit is capable, given its local conditions, to safely turn *on* or $off$ respectively, it will do so with a $|\epsilon|$ probability.

In this way, a simple control signal is dispatched to the assets which also respects local constraints of the assets.

### Assumptions of the control for thermostatic loads

The control approach in [12] is based on several assumptions and we briefly review these here. First, the refrigerators have similar, although not identical parameters, and the average parameter values are known by the controller. This resembles a situation where the LV controller has been preconfigured, e.g. via registration of grid assets, such that a set of flexiable assets are able to be controlled.

Secondly, the group of refrigerators together with the inflexible consumption are within the same electrical LV grid domain, making it possible to measure the cumulated power consumption $z$, for example as illustrated in Figure D.2 at a MV/LV transformer station. Furthermore, it is assumed that the states $\tilde{N}_k^{off}$ and $\tilde{N}_k^{on}$ can be estimated accurately by the controller.

The communication in [12] is considered to be done over an ideal link, i.e. the assumption is that all refrigerators receive the control signal and in due time for the control operation.

### Modifications of the controller

In order to fit the controller to the scenario, it has been modified in certain areas.

- The first one regards a change of the $\epsilon$ calculation. In the original controller described in [12], the control signal $\epsilon$ is modified such that the controller performs a more aggressive control when the power consumed is above the reference. This aggression has been removed such that the controller attempts to reach the reference exactly, instead of trying to be just below it. The reference signal $r$ has been lowered accordingly.

- $\tilde{N}^{off}$ and $\tilde{N}^{on}$ are estimated via feedback from the refrigerators to the controller. After receiving the control command, all refrigerators reply with their (potentially updated) binary state. This feedback was not included in the original controller in Ref [12].

Since the controller has been modified, and a new performance metric is used, the controller can no longer be considered optimal as defined in [12]. This means that control adaptations can cause increased control performance by pure luck. For this reason the controller is recalibrated by changing the $\epsilon$ calculations to:

$$\epsilon_k = \begin{cases} \frac{(r_k - z_k)}{\tilde{N}_k^{off} \cdot \bar{p}} \cdot C, & \text{for } r_k > z_k \\ \frac{(r_k - z_k)}{\tilde{N}_k^{on} \cdot \bar{p}} \cdot C, & \text{for } r_k < z_k. \end{cases} \tag{D.3}$$

Where $C$ is a constant. The value of $C$ is found by simulating the controller under perfect network conditions using different values for $C$, and thereby determining its optimal value. This simple recalibration is done as controller design is out of scope of this paper, however, in a practical implementation, a recalculation of control parameters should be done instead.

## 4.2 Communication network

In this scenario we use narrowband PLC for communication between the LV grid controller and each refrigerator. Because we have 60 seconds to dispatch the control signal $\epsilon$, communication delays in the ranges of few seconds only have minor impact. However, packet losses are a focus which impact differently for UDP and TCP based protocols, especially for high packet loss communication technologies like PLC.

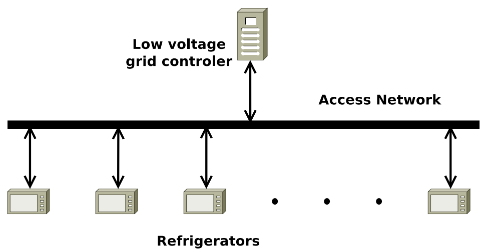



**Fig. D.5:** Overview of the considered communication network.

The parameters of the network scenario that is used in Section 6 for evaluation are listed in Table D.1. We assume a broadcast network with a star topology, for which we can broadcast or multicast the $\epsilon$ value directly to the assets, and each asset unicast their state back to the controller.

| **Parameter** | **Value** |
|---|---|
| PLC Packet loss | 70[%] |
| PLC throughput | $3000[\frac{b}{s}]$ |
| Refrigerators | 100 |
| Control interval | 60 [s] |
| Simulation Time | 15 [days] |
| Correction Factor | 1,2 |
| Avg. refrigerator power consumption | 100 [W] |
| Refrigerator temperature range | 2-5 [°C] |
| Min. refrigerator on/off time | 5 [min] |

**Table D.1:** Parameters used for the simulations.

A high packet loss probability of 70% was chosen to illustrate the impact of imperfect network conditions on the controller. The controller was simulated with different values of packet loss, but proved highly resistant to packet losses due to the fact that it was designed with stochastic deviations in mind. As delays does not have a significant impact on the controller in this scenario, the value for the throughput was chosen sufficiently high to support the scenario. Ref. [6] showed that PLC can have BER's as high as $10^{-2}$, which for the packet sizes in our scenario translate into 92% packet loss probability. In practice, the impact of bit errors can be reduced by deploying error correcting codes.

## 4.3 QoS-based Adaptive Control

The performance decrease is mainly caused by dropped packets on the communication network. One method to counteract this is to use a reliable transport protocol such as TCP. This does, however, come at the cost of increased delay and overhead, especially compared to UDP, since the control algorithm allows for the use of multi casting. Furthermore TCP becomes very slow in scenarios with high packet loss rates due to its congestion control, or it may even drop connections entirely and then has to perform connection re-establishment. If the overall packet-loss probability can be measured or estimated, the hypothesis is that the control signal can be modified to take into account the loss of recipients by counter adjusting the control signal to the amount of refrigerators expected to receive the signal. The fraction of refrigerators that should change their state then needs to take into account the predicted message loss to achieve the level of control expected by the controller.

In the considered control system, the control parameter $\epsilon$ is a measure of the amount of refrigerators which should change state and is calculated based on the amount of available refrigerators in the system, $N$. The effective number of reached devices will be smaller than $N$, as some refrigerators will not receive the control signal. These become an inflexible unit in practice. To counteract the effect of the packet loss, a new control parameter $\epsilon'$ is defined, as shown in eq.(D.4). This is calculated from the reduced value of number of nodes that receive the downstream adaptation request, $N \cdot (1 - P_l)$

$$\epsilon \cdot N = \epsilon' \cdot N \cdot (1 - P_l) \Rightarrow \epsilon' = \frac{\epsilon}{1 - P_l}. \tag{D.4}$$

In this way, $\epsilon$ is being scaled according to the expected loss of recipients, however, requiring that the packet loss is known.

When a refrigerator receives a control signal from the supervisor center, it will respond with its state information. In scenarios of packet loss, the correct reception of this response message at the controller requires that both the broadcast message and the response message are successfully transmitted. Assuming independent packet loss with probability $p_l$ in both directions, an estimator of packet loss is obtained from the the number of received response messages as follows:

$$\frac{\hat{N}_{received}}{N} = (1 - P_l)(1 - P_l) \Rightarrow P_l = 1 - \sqrt{\frac{\hat{N}_{received}}{N}}. \tag{D.5}$$

The controller uses the amount of refrigerators in a certain state when calculating epsilon, however the measured number of refrigerators in this state and the actual number might not be the same due to packet loss. Thus the $\tilde{N}_k^{on}$ or $\tilde{N}_k^{off}$ should be modified to account for packet loss. This leads to a new calculation of $\epsilon$, being:

$$\epsilon_k = \begin{cases} \frac{(r_k - z_k)}{\tilde{N}_k^{off} \cdot \frac{N}{\hat{N}_{received}} \cdot \bar{p}}, & \text{for } r_k > z_k \\ \frac{(r_k - z_k)}{\tilde{N}_k^{on} \cdot \frac{N}{\hat{N}_{received}} \cdot \bar{p}}, & \text{for } r_k < z_k. \end{cases} \tag{D.6}$$

This means that there are two kinds of adaptation of the controller, one which scales $\epsilon$ with the estimated packet loss, and one which corrects the counted amount of *on* or $off$ state refrigerators based on estimation of missing responses. Later we refer to these two types of adaptations to a) upstream adaptation and b) downstream adaptation, representing the direction of information flow, respectively. Using both adaptations the $\epsilon$ formula becomes:

$$\epsilon'_k = \begin{cases} \frac{(r_k - z_k)}{\tilde{N}_k^{off} \cdot \frac{N}{\hat{N}_{received}} \cdot \bar{p}} \cdot \frac{1}{1 - P_l}, & \text{for } r_k > z_k \\ \frac{(r_k - z_k)}{\tilde{N}_k^{on} \cdot \frac{N}{\hat{N}_{received}} \cdot \bar{p}} \cdot \frac{1}{1 - P_l}, & \text{for } r_k < z_k. \end{cases} \tag{D.7}$$

# 5 Simulation methodology

In order to evaluate the adaptation approach described in section 3.2 an example scenario has been implemented in a simulation framework. A framework for combined simulation of the control method and of the network behavior is developed for that purpose. We used MATLAB and OMNeT++ in our work, but conceptually, the procedure would be similar using other tools, it would probably require extensive changes to the framework. MATLAB is used to simulate the LV grid controller, and will therefore be considered the control simulator for the purpose of this paper.

Since the two tools, MATLAB and OMNeT++, are working with different time concepts, we need to ensure that the two tools interoperate properly in order to produce useful results. Coupling the operation between MATLAB and OMNeT++ is nothing new, as for example illustrated in Ref. [13] where it is used to model indoor wireless networks, and for that reason we will not go into detail regarding this coupling. Figure D.6 shows the workflow of the two simulators for a single control loop.

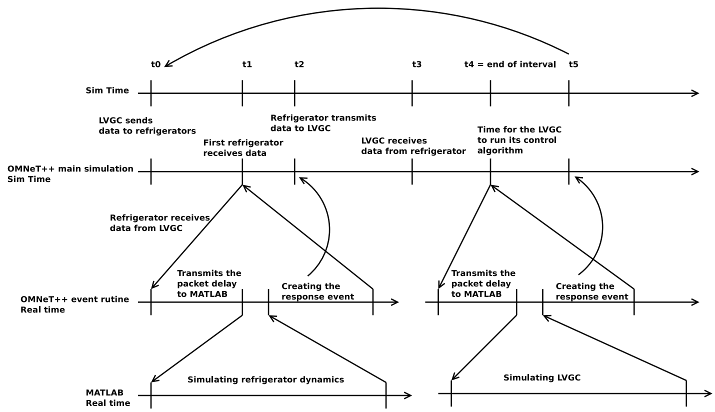



**Fig. D.6:** Message sequence diagram of an low voltage grid controller (LVGC) communicating with a single refrigerator.

OMNeT++ is handling the time of the simulation, and sends messages to MATLAB at the points where system dynamics and control actions need to be processed. To simulate the transport over the communication network, OMNeT++ requires information from MATLAB regarding the amount of data to send, but it does not require actual packet content, this content must therefore be handled internally in MATLAB. Once a packet reaches the destination in the OMNeT++ simulation, the simulation time is paused and MATLAB is given information on the specific receiver entity of the packet. The execution flow now moves into MATLAB, which performs the required processing and returns control to OMNeT++ when done. The simulation time in OMNeT++ is then updated with an added processing delay. The control program can be triggered from OMNeT++ by communication network events such as the arrival of a packet, or by clock events, as is the case of the main control loop.

It was chosen to have OMNeT++ initiate and handle the communication with MAT-

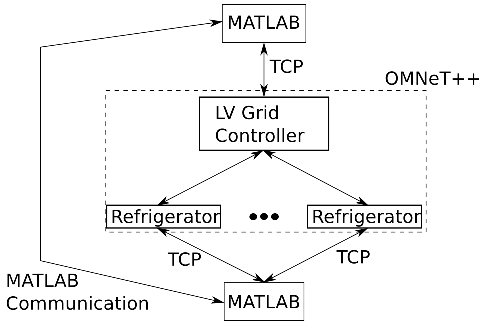



**Fig. D.7:** Interfaces between OMNeT++ and MATLAB.

LAB, and not the other way around, because it also handles simulation time. This means that OMNeT++ must know the time it takes for each control part to be processed. This process is illustrated in Fig. D.6. The structure of the communication between MATLAB and OMNeT++ is shown on Figure D.7.

# 6 Evaluation of the network aware control system

In this section we provide an overview of the results we obtained and discuss them in relation to the framework we setup.

## 6.1 Performance metric definition

The key performance metric of the control algorithm is defined as the error between the total consumed power and the power reference. This error to some degree reflects the effort, that the MV grid controller has to do in terms of extra effort of providing or distributing excess power over time. The parameter is calculated as

$$Err_p(k) = |r_k - z_k| \tag{D.8}$$

Summing up over time, and averaging, gives an indication of the energy demand for the particular LV grid domain over a time period.

$$Err_e(K) = \frac{1}{K}\sum_{k=0}^{K} |r_k - z_k| \tag{D.9}$$

## 6.2 Base scenario

In the base case scenario, we make a simulation run of the control simulation without any influence of the communication network.

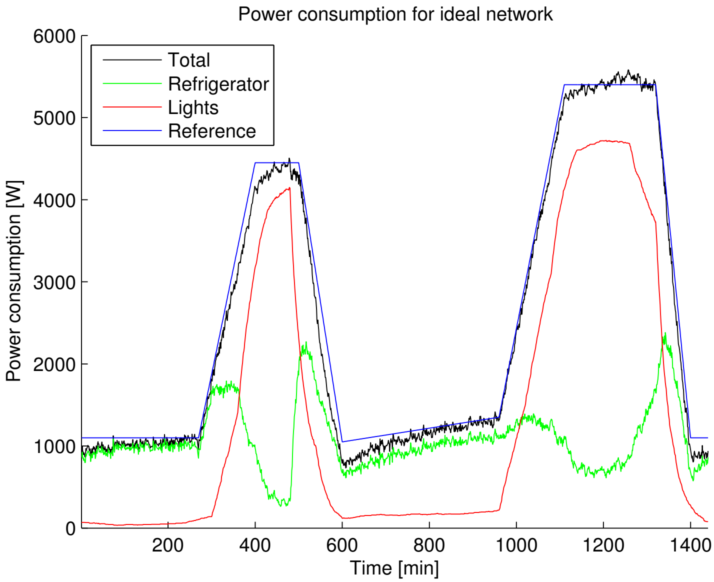



**Fig. D.8:** Base case simulation of the controller under ideal network conditions.

Figure D.8 illustrates clearly the role of the controller, namely to store energy in the refrigerators as before a power peak arises. It is seen by the initial rise of power consumption by the refrigerators, which during the peak load period is reduced. After the peak load period the refrigerator power consumption increases significantly to recover from the energy drain in the thermal energy 'bank'. The cycle repeats itself for the late afternoon peak, however, not without violating the setpoint for which the controller was supposed to keep. This is a result of the stochastic elements. Any deviation from the reference relates to cases where power must be either taken from or input to the LV grid with the interaction of the MV grid controller.

Investigating the control scenario with lossy and delayed communication is now used to analyze 1) the impact of imperfect communication networks on top of the control system, and 2) how well this can be remedied by our proposed solution of manipulating the control and feedback signaling without the knowledge of the controller to make up for packet losses.

## 6.3 Evaluation of the communication network influence and proposed solution

Five different simulations have been done, one with perfect communication network conditions, which will serve as the best achievable performance (lowest error), and the four combinations of with and without upstream and downstream adaptation. Where the one without any adaptation will show how the controller performs under imperfect communication network conditions, and the rest will show the effect of the adaptations. 15 days have been simulated for each simulation and the error is calculated for each minute. The average error was then calculated for each minute of the day. Due to significant fluctuations in this plot it was difficult to see any differences between the schemes. To show the trends in the error it was chosen to make a moving average of the error with a window size of 50 minutes, which can be seen in Figure D.9.

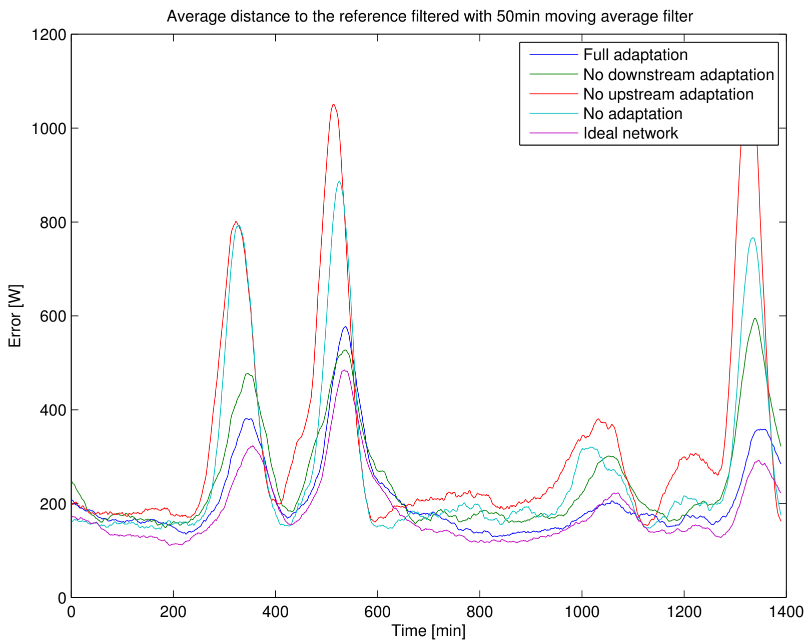



**Fig. D.9:** Moving average of the error with a window size of 50 minutes.

The results in Figure D.9 shows first a substancial impact due to the communication network imperfections (teal curve) and secondly that the proposed adaptations (blue curve) are effective. The error in terms of energy losses, becomes worse with the communication network degradation and if nothing is done, this may have impact on the overall strategy of which the MV controller needs to take if multiple LV grids should be considered within one MV control domain.

The average error over an entire day has been calculated for each scheme, by taking the average error for each day, and afterwards averaging this over the 15 simulated days. The standard deviations of the error have been calculated likewise.

| **Adaptation** | **Mean error** | **95 % Confidence interval** |
|---|---|---|
| Ideal network | 184,01 | 171,88 - 196,14 |
| No adaptation | 263,28 | 249,19 - 277,36 |
| No upstream adaptation | 324,19 | 314,91 - 334,12 |
| No downstream adaptation | 249,75 | 234,91 - 264,59 |
| Full adaptation | 211,42 | 200,91 - 221,92 |

**Table D.2:** Performance of the different schemes.

It is seen in Table D.2, that the imperfect communication network conditions have impact on the controller performance. It is also seen that only adapting for downstream imperfections will cause the controller to be over aggressive, and an increased error is experienced. It can also be seen that only adapting for downstream imperfections reduces the performance significantly. If we consider the calculation of $\epsilon$ as shown in equation (D.7), we see that if there is no modification of the received responses (no upstream adaptation), $\epsilon$ will increase, making the controller more aggressive. Furthermore if $\epsilon$ is adapted based on the packet loss it becomes even larger, thus even more aggressive. This leads to the controller becoming overly aggressive leading to poor performance. The performance can be increased significantly by adapting for upstream imperfections, and even more by adapting for both upstream and downstream imperfections, leading to a total improvement of around 20 %.

# 7 Conclusion and future work

We introduce an approach to take measured communication network performance into account via parameter adaptation in middleware functions and show the effectiveness in simulation experiments. We also show that including communication network considerations when designing systems, like smart grids, where faults can be very expensive, is important. We propose a solution for this inclusion, where control signals are adapted according to communication network QoS measurements, and evaluate this in a LV smart grid control scenario using simulations. This can in principle be realized via a middleware solution as sketched here; however the detailed design in order to allow the middleware to act transparently to the control algorithm is not given in this paper.

We show that by adapting the controller to current communication network QoS estimations, the controller performance can be increased significantly (as seen when using full adaptation). It can, however, also be concluded that adapting the controller to the communication network can in some cases decrease control performance, if the controller becomes over aggressive (as seen when only considering downstream adaptations).

This paper shows how packet loss probabilities can be effectively included to increase control performance, however, further explorations into the inclusion of other QoS parameters such as latency and throughput can prove useful.

## Acknowledgment

The authors would like to thank Tatiana Kozlova Madsen for help with the development of a suitable simulation platform and Kartheepan Balachandran for help with data processing and useful discussions. The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement N°318023 for the SmartC$^2$Net Project and from the Danish Research Council for the project EDGE. Further information regarding SmartC$^2$Net is available at www.SmartC2Net.eu.

The Telecommunications Research Center Vienna (FTW) is supported by the Austrian Government and by the City of Vienna within the competence center program COMET.

## References

[1] E. styrelsen, "Energistatistik 2010," 2010, iSBN: 978-87-7844-908-5.

[2] Y. Yan, Y. Qian, H. Sharif, and D. Tipper, "A survey on smart grid communication infrastructures: Motivations, requirements and challenges," *Communications Surveys Tutorials, IEEE*, vol. 15, no. 1, pp. 5–20, 2013.

[3] F. T. Wien, A. Universitet, T. U. Dortmund, Resiltech, R. S. S. Energetico, V. Omnitel, and E. E. E. Sistemas, "Fp7 smart control of energy distribution grids over heterogeneous communication networks project," 2013, http://smartc2net.eu/.

[4] V. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. Hancke, "Smart grid technologies: Communication technologies and standards," *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 4, pp. 529–539, 2011.

[5] F. Aalamifar, H. Hassanein, and G. Takahara, "Viability of powerline communication for the smart grid," in *Communications (QBSC), 2012 26th Biennial Symposium on*, 2012, pp. 19–23.

[6] M. Korki, N. Hosseinzadeh, and T. Moazzeni, "Performance evaluation of a narrowband power line communication for smart grid with noise reduction technique," *Consumer Electronics, IEEE Transactions on*, vol. 57, no. 4, pp. 1598–1606, 2011.

[7] J. Colandairaj, G. Irwin, and W. G. Scanlon, “Wireless networked control systems with qos-based sampling,” *Control Theory Applications, IET*, vol. 1, no. 1, pp. 430–438, 2007.

[8] M. Pohjola, “Adaptive control speed based on network quality of service,” in *Control and Automation, 2009. MED ’09. 17th Mediterranean Conference on*, 2009, pp. 784–789.

[9] D. Niyato, P. Wang, Z. Han, and E. Hossain, “Impact of packet loss on power demand estimation and power supply cost in smart grid,” in *Wireless Communications and Networking Conference (WCNC), 2011 IEEE*, March 2011, pp. 2024–2029.

[10] G. Dondossola(eds.) and R. Terruggia(eds.), “Smartc2net deliverable d1.1: Smartc2net use cases, preliminary architecture and business drivers,” 2013, http://smartc2net.eu/.

[11] M. Petersen, K. Edlund, L. Hansen, J. Bendtsen, and J. Stoustrup, “A taxonomy for modeling flexibility and a computationally efficient algorithm for dispatch in smart grids,” in *American Control Conference (ACC), 2013*, June 2013, pp. 1150–1156.

[12] L. C. Totu, J. Leth, and R. Wisniewski, “Control for large scale demand response of thermostatic loads*,” pp. 5023–5028, June 2013.

[13] Z. Zhang, Z. Lu, Q. Chen, X. Yan, and L.-R. Zheng, “Cosmo: Co-simulation with matlab and omnet++ for indoor wireless networks,” in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, 2010, pp. 1–6.

# Paper E

## Resilience of Urban Smart Grids Involving Multiple Control Loops

Jacob T. Madsen, Jayakrishnan Radhakrishna Pillai, Hans-Peter Schwefel

The paper has been published in the
*Proceedings of IEEE Second International Smart Cities Conference (ISC2 2016)*, In press, 2016.

*The layout has been revised.*

# Abstract

*Intelligent control of energy distribution grids is implemented via a hierarchy of control loops with different input values and different control targets, which also work on different time-scales. This control is enabled by a bi-directional communication flow, which can be interrupted due to ICT attacks. It is therefore necessary to analyze and understand the emergent behavior resulting from the interplay of the different control loops and how this behavior may change under different communication scenarios. The simulation scenario considered in this paper is a medium and low voltage grid in island mode with a limited grid buffer capacity subjected to ICT attacks. First the interplay of four different control loops that all react to time-varying prices is analyzed. A co-simulation framework is applied to specifically investigate the sensitivity of the emergent grid behavior to extreme conditions on the the price signal communication which could be the result of ICT attacks. In a second step, we introduce a modified pricing scheme, where the energy price is locally influenced by the distribution grid conditions. We then analyze the behavior of this modified scheme when both up-stream and downstream communication are subject to communication delays and losses resulting from ICT attacks.*


# 1 Introduction

The Smart Grid is an evolution of the power grid that utilizes a communication infrastructure and the ability to have two-way communication between entities in the power grid. Intelligent adaptation of grid behavior can be done communicating grid status upstream and set-points downstream; indirect control via energy prices can also be applied. We consider a scenario where a micro-grid with a limited capacity buffer attempts to balance the power in the grid while ensuring frequency stability. However, as there is a communication infrastructure, it is vulnerable to ICT attacks, either through malicious targeting of specific Smart Grid entities, or as collateral damage if the infrastructure is shared with other services. TLS can assure authentication and integrity protection, however, denial of service type of attacks like the flooding and TCP reset attack [1] cannot easily be prevented by TLS. That is why we focus here on timing impact of attacks, resulting in message delays and losses, as opposed to malicious modifications of grid status or set-point messages.

In a second step, we investigate a scheme for improved balancing of the micro-grid by local modifications of energy prices, i.e. a price server receives information on the current power balance in the grid and modifies the electricity price based on this power balance. This up and down regulation of the price is intended to further ensure that the entities in the local distribution grid attempt to balance the grid by increasing the price incentive. However, by adding this local market server we may also increase the susceptibility of the grid to communication disruptions, as this will add an upstream

communication flow.
There are several methods for studying interplays of controllers in the Smart Grid. A simulation framework which studies the MV and LV grid is the DiSC tool found in Reference [2]. This tool simulates distribution system voltage control using MATLAB. Reference [3] has built a testbed which allows greater detail when investigating the Smart Grid. However such a testbed is very costly, and it takes a long time to run emulations of the grid. When considering communication networks in power grids it is important to consider how attacks on the ICT infrastructure affects both the communication and the controller, such as done in Reference [4], which considers flooding and reset attacks in an MV grid. These attacks on the ICT infrastructure can cause increased delays, which are investigated in Reference [5] in the context of medium voltage control.
In this paper we simulate different Smart Grid and power grid controllers together and study the interplay between each controller, while also simulating a disrupted communication flow. Section 2 details the general Smart Grid scenario we consider, Section 3 introduces the simulation framework used, and Section 4 describes the specific simulation scenario. In Section 5 we show validation of the framework and results with a nation-wide market, while in Section 6 we show the performance of locally modified prices based on local grid balance, and we conclude in Section 7.

# 2 Control and ICT Architecture

The scenario we consider in this paper is the load frequency control (LFC) of a modern wind power dominated system, as seen on Figure E.1. The LFC consists of medium and low voltage grid units participating in power balancing control. This MV grid contains a distributed wind farm, supermarkets, distributed CHPs, and a low voltage grid. In this scenario the HV grid is seen as a limited size energy buffer. The grid entities are connected to the electricity market, from which it receives prices of electricity. The green lines denote communication flows, which are sometimes bi-directional and others uni-directional.

This scenario covers part of an urban power grid, in which the LFC will send the prices it receives from the market to the MV grid controller which will pass it on to the LV grid controller.

## 2.1 Market

The Market is assumed to be a single nation-wide entity which dictates the electricity price every hour based on the current power production and consumption in the whole nation-wide grid, and predictions of future power production and consumption. This price is communicated to each flexible entity in the grid. By setting the price it will influence the controllers of the Smart Grid, as they produce or consume based on, among other things, the price of electricity.

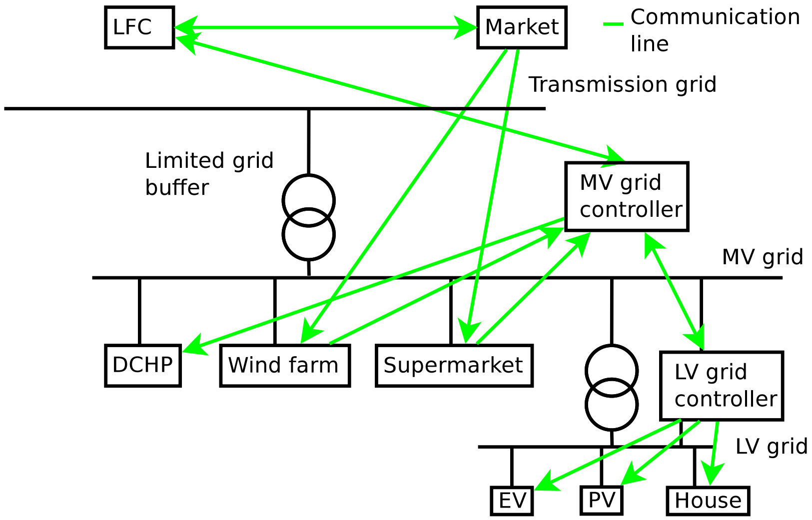



**Fig. E.1:** Overview of the scenario

## 2.2 Wind farm

The controller aims to reduce damage a wind farm sustains during normal operation. The controller estimates the damage to the wind farm based on both wind speed, and variations in wind speed. It is a trade-off between the cost of producing energy and the price of the electricity. A closer look into the effects of communication networks within a wind farm controller have been done in Reference [6] and [7].

## 2.3 Supermarket

The supermarket controller aims to regulate the temperature within a supermarket between an upper and lower bound. The amount of power required to keep the temperature within these bounds is dependent on the outdoor temperature at the time, as well as other aspects (number of humans in the supermarket, opening or closing freezer doors etc.). The controller is set to increase or decrease consumption/production based on the price of electricity, if at all possible. This is based on the work done in Reference [8]. Another method for modelling a supermarket controller can be found in Reference [9].

## 2.4 LFC

The first target for the LFC is to stabilise the frequency, and as a secondary priority attempts to balance the power consumption based on system flexibility to the lowest cost of the system. We assume it achieves this by calculating set-points for the aggregated generation units, medium voltage grid controller (MVGC) and low voltage grid controller (LVGC). It takes in measurements from the power grid on frequency and voltage, as well as information regarding state of the distribution grid from the MVGC/LVGC, and uses this when generating set-points. The LFC is based on the work done in Reference [10] and [11].

### 2.5 MVGC/LVGC

The coordinated MVGC/LVGC must follow the set-points from the LFC, and uses these to regulate the consumption and generation of power at the consumers. The MV grid involves distributed wind turbines and small/medium CHP units and flexible consumption loads such as supermarkets. The LV grid contains prosumers with different flexibilities, among these are electric vehicles (EV), photo voltaic units (PV) and households. A more detailed description of this can be found in Reference [12] and [13].

### 2.6 Communication network

In order for all these controllers to communicate a communication network is required. The communication network handles all information flows between the controllers, however, as this is a study on the interplay between the controllers we do not investigate the effects of communication networks that the individual controllers use to interact with their sensors and actuators, see Reference [14] for such and example. The evaluation scenario for the communication between the different controllers in Figure E.1 considers extreme communication delays and losses as may result from denial of service flooding and TCP reset attacks, which may result in periods of high packet loss, or total loss of connectivity for the system, as in Reference [1].
The information flow in the system can be seen on Figure E.2. The market calculates a price, which it sends to the wind farm, supermarkets and LFC (which sends it to the MV/LVGC). The wind farm and supermarkets send their production/consumption to the LFC through the MVGC, which then calculates and sends a set-point to the flexible regulation units in the MV and LV grid. In section 6 we will describe the information sent from the LFC to the market, as indicated by the dotted line in the figure.

## 3 Simulation tool

In order to evaluate the interaction between the controllers in the scenario, a modular simulation tool has been created that contains the following modules:

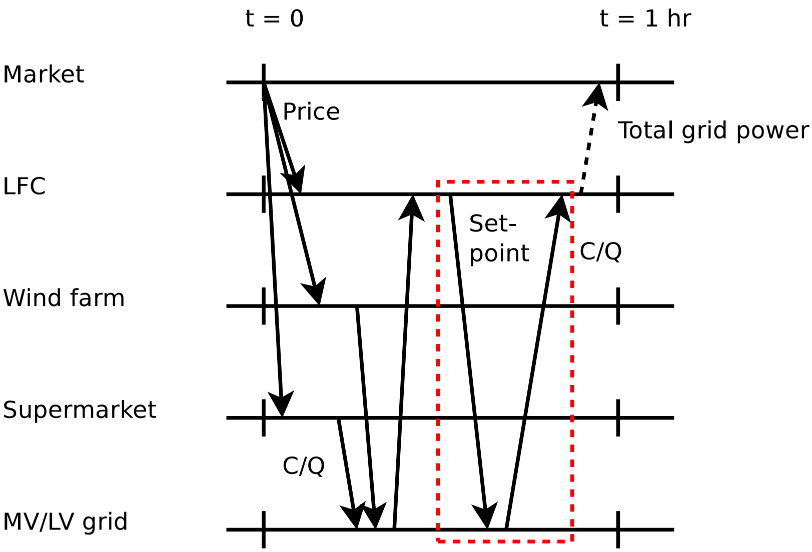



**Fig. E.2:** Information flow of the simulation

## 3.1 Wind farm

The simulation model is an aggregation of several wind turbines, abstracted based on the behaviour of a single wind turbine. The module uses both the current wind speed and the variation in wind speed over the control time. It uses these to estimate the damage wind turbines would sustain while operating under these conditions, and uses the price to gauge whether it is worthwhile to attempt to maximize production at the cost of life time on the wind turbine. The wind farm module assumes the layout of the wind farm to be wind turbines standing in lines, and thus each subsequent wind turbine in the line is affected by the wind turbines in front of it.

## 3.2 Supermarket

This controller calculates power consumption/production based on the previous indoor temperature, and the current outdoor temperature which is used as a disturbance. When the electricity price is below a threshold of 80 DKK/MWh the supermarket consumes more power to store up thermal energy, and when the price rises above the threshold it uses less power and uses the stored thermal capacity instead. The supermarket module has a model for the disturbances in the supermarket relating to the opening or closing of refrigeration units or the requirements for power due to people in the supermarket.

## 3.3 LFC and MVGC/LVGC

The frequency deviation in power systems is an indicator of power imbalances between the generation and load. The LFC uses a proportional-integral controller to correct these system deviations thereby creating new power set-points to those generation (CHP, DCHP) and flexible loads (EVs, thermal/indirect storages like supermarkets) units participating in frequency regulation. Transfer function models of these generators and flexible components are used for their design and analysis in the LFC [10] [11]. The MVGC and LVGC pre-calculates the estimation of flexibility from those aggregated models of components from the distribution grids (DCHP, EVs etc.) participating in the LFC regulation. Optimal schedules of these flexible distribution units are determined based on the market prices and component characteristics under steady state conditions [12] [13]. Their regulation capacities are finally computed by using load flow analysis considering the voltage and thermal constraints of the distribution grids.
The MVGC and the LVGC are simulated in the same module in the simulation framework, as there are interdependencies between the modules would greatly increase the simulation complexity if they where simulated separately. The module determines the number of EVs connected to the grid, the number of solar panels connected, and the production of the CHP. This is based on the current solar radiation, the outdoor temperature and a movement model of the EVs based on typical driving patterns and battery state of charge.
The grid model uses the line impedances and the current limits to calculate the voltage and currents through load flow analysis. The MV grid model is represented by a simple aggregated model of the grid components, whereas the LV grid model takes into account the physical location of each node.

## 3.4 Communication network

The communication network simulates an IP network, with a transport protocol on top of it. We can model different communication scenarios with this, where we can change both the delay and the packet loss of the communication network. The network simulator is made to allow changes in the network conditions that the modules operate under, and thus allows changing both delay distributions and packet loss probability. In this paper we assume delays and losses to be independent of previous losses or delays. Furthermore, due to the implementation of the MVGC and LVGC in the simulation framework, the communication network between these two is not simulated. Similarly the communication within each module is not simulated, as explained earlier in this paper.

# 4 Simulation scenario

In this section we detail the specifics of the simulation scenario for each module.
The wind farm module simulates a wind farm consisting of 90 wind turbines, in the formation of 6 lines containing 15 wind turbines each, with each wind turbine producing a maximum of 1.1 MW. This is considered to be a wind farm of average size for a distribution grid. The wind farm module takes the electricity price and the wind as input, and the variation in wind speed over the time period. It outputs its power production.
The supermarket module simulates 4 homogeneous supermarkets of size 60x30x7 meters, meaning they share the same temperature profile and use the same models. Each supermarket attempts to regulate the indoor temperature in the supermarket between 18 and 20 degrees Celsius. It uses the price and the current outdoor temperature as input, and the consumption as output.
The limited grid buffer has a size of 150 MW, which is used in case the power in the MV grid is not balanced sufficiently, in which case the buffer will attempt to supply the difference. The LV grid consists of 45 households, spread out on 15 power lines, each with a PV panel (1 MW), and 25 of the households own a EV (200 kW each), this is based on an average LV grid in Denmark, extrapolated for future penetration and technology development of EV charging and PV deployment. The MVGC/LVGC takes the output of the wind farm and the supermarket, as well as the current price. It gives as output the current power balance in the grid. It uses the solar radiation to model the production by the PVs.
The price data used by the market module is real world price data provided by [15]. This database of prices also contains the wind speed, temperature and solar radiation at the given time, which are also used by the modules in this simulation framework. All this data is correlated.

The communication network is simulated in OMNeT++ and we simulate scenarios where we increase the delays from 0 to up to four hours, and packet losses between 0 % and 100 % (100 % corresponding to a complete loss of control of the system). The packet loss scenarios correlate to different severity attacks on the ICT infrastructure. The packet loss is added to the IP layer in the simulation. Regarding delays we consider delays, which may be caused by attacks on the ICT infrastructure, that are rather large, up to several control cycles.
The information flow in the simulation is shown in Figure E.2 in Section 2.6. The specific timing in this communication flow is as follows: every hour the market will calculate a new price at the start of the price cycle and send this to the other modules. 5 minutes after the start of the price cycle the wind farm and supermarkets will determine their consumption and production, and send this to the LFC which forwards it to the MVGC. 10 minutes after the start of the price cycle the MVGC and LVGC determine set-points for the modules regulated by them. In the case of the extended scenario of a local

market in Section 6, the MVGC measures the current power balance in the grid and sends this to the market module 5 minutes before the next price cycle. When we in this paper discuss delays or packet losses, the delays or loss percentages are applied to all links descriped in the information flow. We are using both TCP and UDP for transport protocols, and it was chosen to use OMNeT++ for the network simulation as it has built in TCP and UDP.

# 5 Evaluation of the interplay of Smart Grid controllers

In this section we start by validating the behaviour of the simulation framework. Next we study the interplay of the controllers under a nation-wide market, and we investigate the effect of extreme communication network conditions on this interplay. In Figures E.3, E.4 and E.5 we show the price generated by the market module (upper left corner), and the consumption and production of the supermarkets and wind farm modules (bottom of figures). Negative numbers mean consumption, while positive production. We also show the power balance in the specific MV grid (upper right corner). This power balance is the sum of all production, minus the sum of all consumption in the grid. Thus if the number is positive, there is an excess of power in the grid, while a negative number indicates a lack of power in the MV grid.

## 5.1 Verification of simulation framework

We start by evaluating the behaviour of the grid and the flexible prosumers under fixed boundary conditions. To do so we run a simulation where we set the input price of the system to follow a step function. We also fix all temperature, wind and solar radiation inputs. This shows how the different modules in the simulation react to changes in price.

As we see from Figure E.3 all three modules that are intended to react to prices (Wind farm, supermarkets and LV grid), do indeed change consumption or production based on the price.

## 5.2 Behaviour of control loops under nation-wide market conditions

As we have now determined that the controllers and market behave in a manner which we expected, we consider how the simulation behaves when we introduce nation-wide market dynamics. This can be seen in Figure E.4 which shows the system behaviour with an ideal communication network, i.e. zero delay and no loss. The following figures (in this section and the next) use the same input data, in terms of nation-wide price, wind, temperature and solar radiation.

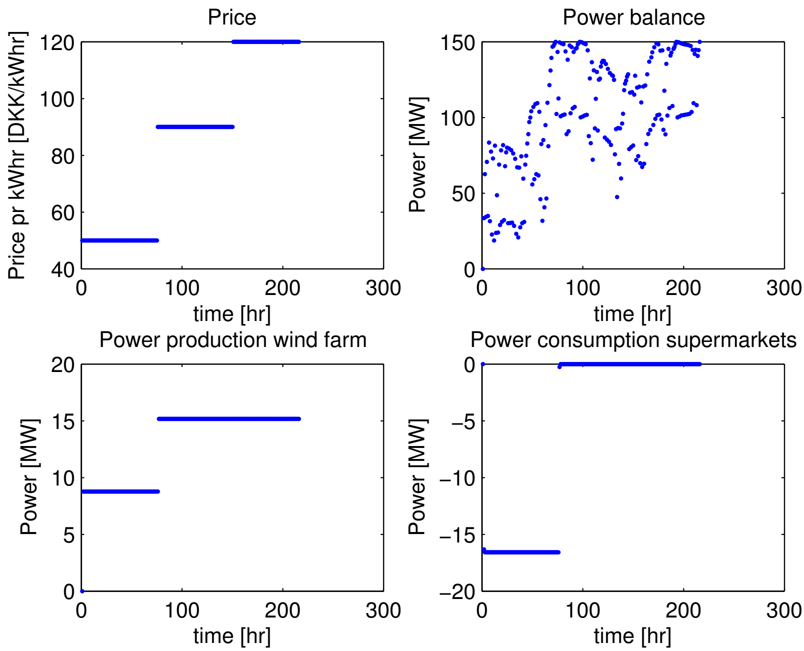



**Fig. E.3:** Simulation results with a price value that steps three times (50 to 90 to 120 DKK/KWh) and all other variability parameters set at fixed values, under ideal network conditions

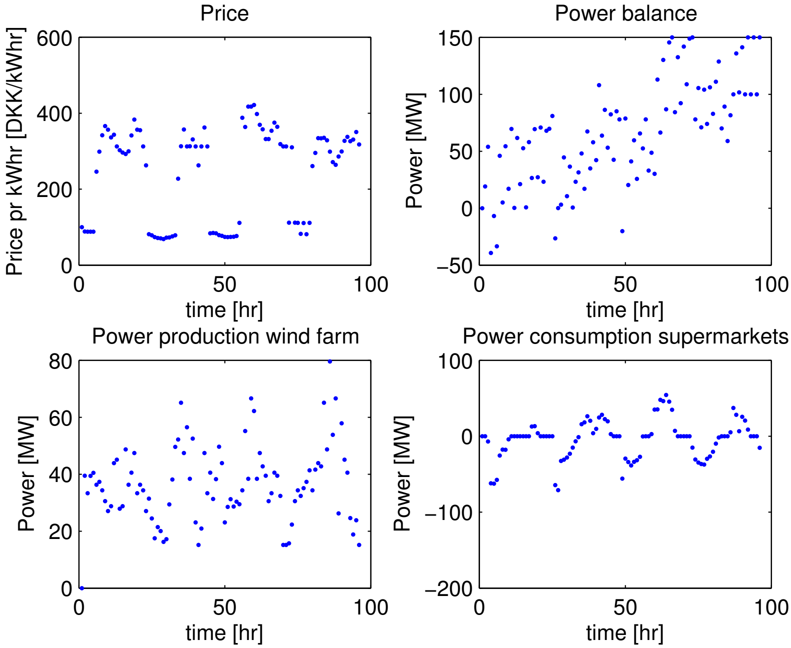



**Fig. E.4:** Simulation results using nation-wide market prices

Due to space limitations we only show Figure E.4 in this section. Other results show that the system is highly resilient to packet loss and delays. However, when investigating the system consumption during high and low price periods (described in further detail in Section 6) we saw a higher consumption during high price periods, and lower consumption during low price periods, compared to the total average consumption. This result shows that the price as taken from the real-world data is not effective to achieve consumption adaptation for the investigated system setting. Therefore we look at a modified system with stronger impact of the price, and hence higher relevance of the communication of prices next.

# 6 Extended Scenario with modified prices

In this section, we investigate how using locally modified prices based on the power balance in the grid can affect the ability to balance the grid, and ensure that we have the highest consumption when the price is lowest (and vice versa). We call the module implementing this the local market server. In this section, the results considering delays are done using TCP as transport protocol, and results featuring packet loss are done using UDP.

## 6.1 Extended scenario overview

The inclusion of the local market server adds an extra communication flow to the system, namely an upstream flow, illustrated by the dotted line in Figure E.2. This flow consists of the LFC sending the current power balance in the micro-grid. This information is the used by the local market server to up or down regulate the price, to encourage the controllers in the grid to change their production or consumption. By adding this upstream flow we do, however, expect the system to be more susceptible to ICT attacks and non-ideal communication networks. This is mainly due to the fact that if the information regarding power balance is lost the local market server will use outdated or possibly wrong information to modify the prices, which will potentially lead to regulating up when it should regulate down.

## 6.2 Market implementation

We implement the local market server by having the nation-wide price regulated up or down as detailed by Equation E.1. This equation will allow the local market to either decrease or increase the price of electricity based on the price from the nation-wide market and the current power balance in the local power grid.

$$P_{reg}(ti) = P(ti) - P(ti) \cdot \frac{1}{\pi} \cdot atan(Diff(ti)/10MW) \tag{E.1}$$

Where $P(ti)$ is the real world price found from available data at time $ti$, $P_{reg}(ti)$ is the regulated price, and $Diff(ti)$ is the current power balance in the MV grid. Atan was used in this model as it allows a steep curve at low power differences, and becomes less steep as the power difference increases, and we scale it such that the regulated price can be a maximum of 1.5 times higher or 0.5 times lower than the input price.

## 6.3 Results with modified prices

The results can be seen in Figure E.5 which shows the system behaviour with an ideal communication network.

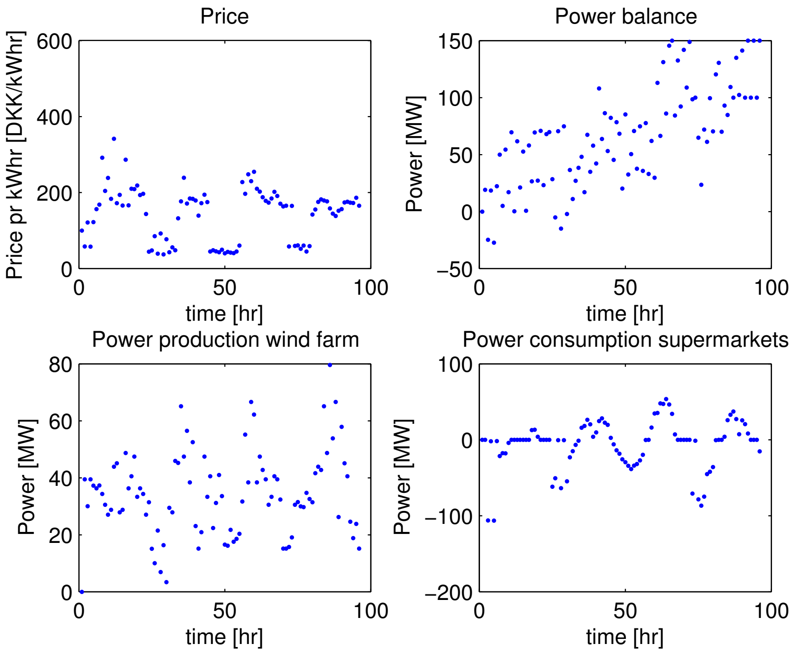



**Fig. E.5:** Simulation results using locally modified market prices based on MV grid balance

Figure E.5 show that when the power balance in the grid is not zero the market changes the prices to attempt to drive it toward zero. This is of course not always possible, as there are outside disturbances such as wind and temperature that also change. As the market does not take these disturbances into account when calculating a new price we find that it is not always possible to drive the power balance to zero.
Based on a histogram of the price in the ideal communication network scenario, a high

and low price threshold is set. These thresholds are set based on values where approximately 30 % of prices lie within the threshold, and we found $65 DKK/KWh$ as the low price threshold, and $165 DKK/KWh$ as the high price threshold. The consumption in the time period when each of these thresholds are exceeded is then averaged. We would hypothesize that when there is packet loss or substantial communication delays, the system's ability to regulate the consumption based on price will decrease. If this happens the average consumption in low price periods will decrease, as the entities in the system are not aware that they should increase consumption. Conversely, the average consumption in the high price periods will increase, as the system does not know that it should decrease consumption.

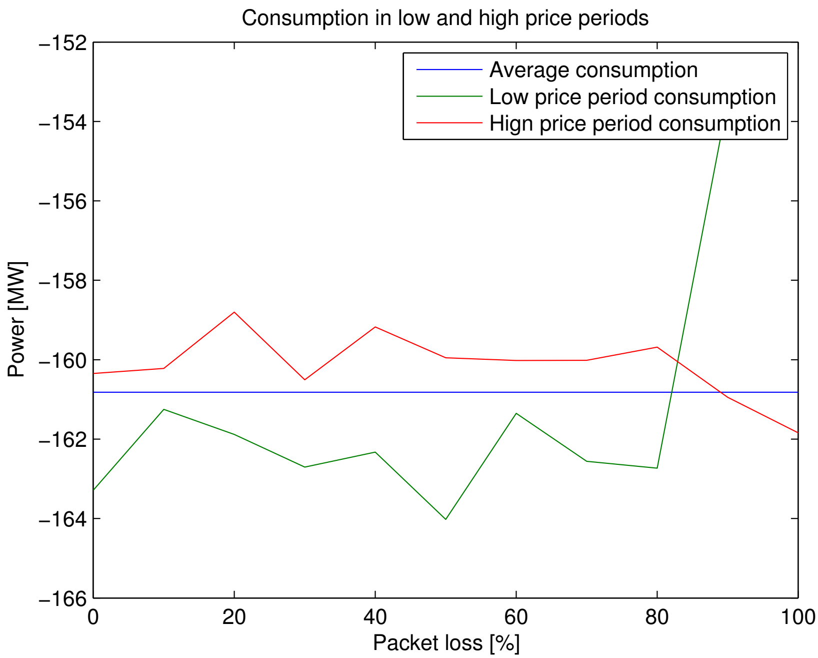



**Fig. E.6:** Simulation results showing the consumption on average and in low and high price period over increasing packet losses using UDP

Figure E.6 show that the system is rather resilient to packet loss, with needing more than 80 % packet loss to reduce performance of the system, and the results for delays show similar behaviour (not shown in this paper due to space delimitaions). The average consumption in the defined price periods does not change significantly until the packet loss exceeds 80%. We do see after 80% that the consumption in low price

periods decreases, and it increases in high price periods, as expected. Similar for delay the graphs cross at slightly over 2 hours delays.
We see from the results shown in Section 5 and in this section that when we deal with more extreme price conditions that the system is more susceptible to ICT attacks and disruptions of communication. The results using nation-wide market prices show little change in behaviour even under extreme conditions, where using a local market server seems significantly more susceptible to attacks. However, the local market server was able to ensure higher consumption during low price periods and lower consumption during high price periods.

## 7 Conclusion and outlook

In this paper we studied the effects of disrupted or delayed communication on an urban Smart Grid scenario. The scenario considered a micro-grid with a limited energy capacity buffer attached, and we studied the interplay of multiple controllers which are using variable prices as input. We found that by changing the price of electricity we change the behaviour of the controllers in the system.
We found that by introducing locally influenced price changes, which would up or down regulate the price based on the current power balance in the micro-grid, that the controllers where better able to regulate consumption. However the system was also more susceptible to ICT attacks. It should also be pointed out that the system was still resilient to communication disruptions, just not to the same degree as for the nation-wide market.
Another potential problem of the disrupted or delayed communication of price signals is that the controllers may act on price information that is old or incorrect. This may cause problems for billing, if a consumer changes behaviour due to receiving an old price signal while the power supplier is operating under a different price signal.
One of the primary actions for further work currently is the local market server implementation, which could be more realistic. This could be done by taking into account predictions of future meteorological data, or attempting to predict how the grid will react to different prices, to determine an optimal price strategy. Another avenue would be to change the time scale of the simulation modules, and controlling the system faster. This would put larger demands on the communication network, especially if the simulation is expanded to cover larger grid scenarios. This may lead to the system being more vulnerable to delays or loss in information.

## References

[1] G. Dondossola and R. Terruggia, "Cyber security of smart grid communications: Risk analysis and experimental testing," pp. 169–193, 2015.

[2] R. Pedersen, C. Sloth, G. B. Andresen, and R. Wisniewski, "Disc: a simulation framework for distribution system voltage control," in *Control Conference (ECC), 2015 European*. IEEE, 2015, pp. 1056–1063.

[3] C.-I. Ciontea, R. Pedersen, T. Le Fevre Kristensen, C. Eg Sloth, R. L. Olsen, and F. Iov, "Smart grid control and communication: the smartc2net real-time hil approach," in *PowerTech, 2015 IEEE Eindhoven*. IEEE, 2015, pp. 1–6.

[4] G. Dondossola and R. Terruggia, "Security of communications in voltage control for grids connecting der: impact analysis and anomalous behaviours," in *CIGRE Science and Engineering Journal, June 2015*, 2016.

[5] M. Findrik, R. Pedersen, C. Sloth, H. P. Schwefel, and E. Hasenleither, "Qos assessment of communication technologies for a smart grid power-balancing controller," in *IEEE International Energy Conference EnergyCon, 2016 IEEE Leuven*. IEEE, 2016.

[6] J. T. Madsen, J. de Jesus Barradas-Berglind, T. K. Madsen, and H.-P. Schwefel, "On the impact of information access delays on remote control of a wind turbine," in *PowerTech, 2015 IEEE Eindhoven*. IEEE, 2015, pp. 1–6.

[7] J. T. Madsen, M. Findrik, D. Drenjanac, and H.-P. Schwefel, "Investigating wind farm control over different communication network technologies," in *Energy Informatics*. Springer, 2015, pp. 129–140.

[8] T. Minko, R. Wisniewski, J. Bendtsen, and R. Izadi-Zamanabadi, *Cost Efficient Optimization Based Supervisory Controller for Supermarket Subsystems with Heat Recovery*. IEEE Press, 7 2015, pp. 2321 – 2327.

[9] R. Pedersen, J. Schwensen, B. Biegel, J. Stoustrup, and T. Green, "Aggregation and control of supermarket refrigeration systems in a smart grid," in *IFAC World Congress*, 2014, pp. 9942–9949.

[10] R. Kordheili, B. Bak-Jensen, J. Pillai, M. Savaghebi, and J. Guerrero, *Managing High Penetration of Renewable Energy in MV Grid By Grid Flexible Loads*. IEEE Press, 9 2015, pp. 127 – 132.

[11] J. Pillai and B. Bak-Jensen, "Integration of vehicle-to-grid in western danish power system," *I E E E Transactions on Sustainable Energy*, vol. 2, no. 1, pp. 12–19, 1 2011.

[12] R. Kordheili, B. Bak-Jensen, J. Pillai, and P. Mahat, *Determining Maximum Photovoltaic Penetration in a Distribution Grid considering Grid Operation Limits*. IEEE Press, 7 2014, pp. 1–5.

[13] R. Kordheili, B. Bak-Jensen, J. Pillai, and B. Bhattarai, *Improving and Handling Electric Vehicle Penetration Level by Different Smart Charging Algorithms in Distribution Grids.* IEEE Press, 7 2015, pp. 1–6.

[14] J. Madsen, M. Findrik, T. Madsen, and H.-P. Schwefel, "Optimizing data access for wind farm control over hierarchical communication networks," in *International Journal of Distributed Sensor Networks, Special issue on Smart Grid: ICT Control for Distributed Energy Resources.* Hindawi, May 2016.

[15] "Data regarding temperature, wind, solar radiation and prices were supplied by the Energy Department at Aalborg University, from 01/07/2011 starting at 12.00."

ISSN (online): 2246-1248
ISBN (online): 978-87-7112-824-6

AALBORG **UNIVERSITY PRESS**