TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

SANNA TOIVANEN

IMPLEMENTATION OF PERFORMANCE ASSESSMENT TOOL FOR MULTIVARIABLE, MODEL-PREDICTIVE CONTROLLER

Master of Science Thesis

Examiner: Professor Matti Vilkko
Examiner and subject approved by the Faculty Council of the Engineering Sciences on 9th March 2016.

ABSTRACT

SANNA TOIVANEN: Implementation of performance assessment tool for multivariable, model-predictive controller
Tampere University of Technology
Master of Science Thesis, 83 pages
July 2016
Master's Degree Programme in Automation Technology
Major: Process Automation
Examiner: Professor Matti Vilkko

Keywords: multivariable control, model-predictive controller, control performance assessment

The aim of the thesis was to implement a tool for performance assessment of a multivariable, model-predictive controller which was in this work NAPCON Controller developed by Neste Jacobs. The aim of control performance assessment and monitoring is to ensure that control systems operate as required. In practice, control performance assessment techniques are usually based on a comparison of the current controller performance and a benchmark value defined by some criteria. The result of this comparison is called the control performance index. In this thesis, the technological performance of the controller was measured with two techniques using different criteria for calculating the benchmark value. These selected methods were historical and design-case benchmarks. In addition, the economic performance of the controller was assessed.

In this work, an OPC UA database was used for storing the calculated performance indices as well as the related configuration parameters. This required the definition of new OPC UA based information models. The implemented performance assessment tool included a performance calculation application as a Windows service and a graphical user interface for configuring the performance assessment calculations.

The functionality of the implemented controller performance assessment tool was tested with a simulator of a distillation unit and against an actual MPC controller. The results of the different simulation cases showed that the calculated performance indices responded as expected when the process conditions or the control objectives changed. The tool requires some additional testing and development before it can be deployed to a real process environment as a part of the controller software, although the created performance assessment tool worked well according to the simulation results.

TIIVISTELMÄ

SANNA TOIVANEN: Monimuuttujaisen ja mallipreditiivisen säätimen suorituskyvyn arviointityökalun toteutus
Tampereen teknillinen yliopisto
Diplomityö, 83 sivua
Heinäkuu 2016
Automaatiotekniikan diplomi-insinöörin tutkinto-ohjelma
Pääaine: Prosessien hallinta
Tarkastaja: Professori Matti Vilkko

Avainsanat: monimuuttujasäätö, mallipreditiivinen säädin, säädön suorituskyvyn arviointi

Diplomityön tavoitteena oli toteuttaa työkalu, jota voidaan käyttää monimuuttujaisen ja mallipreditiivisen säätimen suorituskyvyn arviointiin. Työssä käytettävä säädin oli Neste Jacobsin kehittämä NAPCON Controller. Säädön suorituskyvyn arvioinnin ja mittaamisen tavoitteena on varmistaa, että säätöjärjestelmät toimivat vaatimusten mukaisesti. Usein käytännössä säädön suorituskyvyn arviointimenetelmät perustuvat tutkittavan säätimen suorituskyvyn ja joillakin kriteereillä määritetyn vertailuarvon suhteeseen. Tätä suhdelukua voidaan kutsua säädön suorituskykyindeksiksi. Tässä työssä säätimen teknisen suorituskyvyn mittaamiseen käytettiin kahta menetelmää, joissa vertailuarvojen laskeminen perustuu eri periaatteisiin. Valitut menetelmät olivat historialliseen vertailuarvoon (historical benchmark) ja suunnittelukriteereihin (design-case benchmark) perustuvat tekniikat. Lisäksi säätimen taloudellista suorituskykyä arvioitiin.

Laskettavat suorituskykyindeksit ja suorituskyvyn laskentaan tarvittavat konfigurointiparametrit tallennettiin OPC UA tietokantaan, mikä vaati ensin uusien OPC UA -pohjaisten informaatiomallien määrittelyn. Suorituskykytyökalu sisälsi Windows palveluna (Windows Services) toteutetun suorituskykylaskentaohjelman ja graafisen käyttöliittymän suorituskykylaskentojen konfigurointia varten.

Työssä toteutetun työkalun toimivuutta testattiin erään tislausyksikön simulaattorin ja oikean MPC-säätimen avulla. Eri simulointitapausten tulokset osoittivat, että lasketut suorituskykyindeksien arvot reagoivat prosessiolosuhteiden tai säätötavoitteiden muutoksiin, kuten saattoi olettaa. Vaikka toteutettu työkalu toimi simulointitulosten perusteella, tarvitaan kuitenkin lisätestejä ja -kehitystä, jotta työkalu olisi mahdollista ottaa käyttöön oikeassa prosessiympäristössä osana säätimen ohjelmistoa.

PREFACE

This master's thesis was done for the department of Technology and Product Development of Neste Jacobs during the time between the January 4th and the July 3rd 2016. The work was related to the development of NAPCON Controller, proprietary of Neste Jacobs.

First of all, I would like to give special thanks to M.Sc. Samuli Bergman and M.Sc. Jyri Lindholm for making this master's thesis possible. I also want to thank my instructors M.Sc. Stefan Tötterman and M.Sc. Markus Sintonen for their important help and comments throughout the thesis process. In addition, I would like to thank the examiner of this thesis, Professor Matti Vilkko from Tampere University of Technology, for his beneficial advice and guidance in the master's thesis project.

I want to thank my friends and colleagues for their support and inspiring company. Especially, I am thankful to Tuuli for the peer support in the university studies and now in the master's thesis process.

I am grateful to my dear family for their support and love throughout my life. It is indeed possible that even little girls can graduate as a Master of Science nowadays! In addition, I would like to thank my second family for their beneficial help by supporting my working in Turku.

Finally, I want to thank my boyfriend Lauri for his love for all these years. You brighten my life and with you, I'll get through everything.

Turku, 26th of July 2016

Sanna Toivanen

CONTENTS

LIST OF SYMBOLS

$A(q^{-1})$	Monic polynomial
$c_u^{(j)}$	Economic coefficient of jth input
$c_y^{(i)}$	Economic coefficient of ith output
$e(k)$	Model residual at time k
$e^o(k)$	Disturbance innovation at time k
$E[\cdot]$	Expectation operator
$E(t)$	Vector of output errors
$\hat{E}(t)$	Vector of predicted output errors
$f(x)$	Probability density function
$J_{ach,des}$	Achieved value of the cost function for design-case performance index
$J_{ach,hist}$	Achieved value of the cost function for historical performance index
J_{act}	Value of the cost function for current controller
J_{des}	Value of the cost function for design-case benchmark
J_{hist}	Value of the cost function for historical benchmark
J_{LQG}	Value of the cost function for LQG benchmark
J_{opt}	Value of the cost function for optimal controller
M	Control horizon
N	Time when prediction should start to follow the reference trajectory
P	Prediction horizon
P_C	Moving horizon of past data
P_{eco}	Expected economic performance
P_{MQI}	Data length of MQI assessment

q^{-k}	Backshift operation
Q	Weight matrix for controlled variables
R	Weight matrix for manipulated variables
$\Delta U(t)$	Vector of the changes in input variables
$\Delta\widehat{U}(t)$	Vector of the changes in predicted input variables
$\Delta U^{*}(t)$	Vector of optimal control moves
W_t	Vector of reference trajectory at time t
x	Location of the quality variables
$Y(t)$	Vector of outputs
$\hat{Y}(t)$	Vector of output prediction

Greek symbols

ε(t)	Uncorrelated, zero-mean, Gaussian noise signal
$\vartheta(x)$	Economic performance function
η_D	Performance index of the controller using a dissimilarity benchmark
η_{des}	Performance index of the controller using a design-case benchmark
η_{hist}	Performance index of the controller using a historical benchmark
η_{MQI}	Model quality performance index
λ	Coefficient of LQG cost function
λ_j^{ach}	Eigenvalues of transformed, achieved covariance matrix
λ_j^{mv}	Eigenvalues of transformed minimum variance covariance matrix

LIST OF ABBREVIATIONS

APC	Advanced process control
COM	Component object model
CPA	Control performance assessment
CPI	Control performance index
DCS	Distributed control system
DCOM	Distributed component object model
EPD	Economic performance design
ERP	Enterprise resource planning
GMVC	Generalized minimum variance control
GUI	Graphical user interface
ILC	Iterative learning control
LQG	Linear quadratic Gaussian
MES	Manufacturing execution system
MPC	Model predictive control
MQI	Model quality index
MVC	Minimum variance control
OPC	Open Platform Communications
OPC UA	OPC Unified Architecture
RPC	Remote procedure calls

1. INTRODUCTION

There are various challenges that the process industries need to face in order to be competitive. For example, price pressure of raw materials and the focus on environmental protection have forced the plants to consider their control operations. Advanced process control (APC) technologies have been applied to various processes to improve product quality and product yield, to ensure consistent process safety and to reduce energy consumption along with environmental emissions. [1] In many process industries, especially in the oil refining and petrochemical fields, the most popular advanced process control strategy is the model predictive control (MPC). When compared to other control strategies, the most important feature of model predictive control is the ability to handle constrained control problems. [2; 3, p. 97]

Degradation in the performance of advanced process control systems is common in the process industries. After the implementation, the control system operates at its nominal efficiency. Nevertheless, over the time the performance of the control system usually degrades due to various causes, such as a change in process conditions or process equipment, lack of maintenance and poor controller tuning. [4; 5] When the controller is functioning as designed, it delivers various benefits including the process efficiency and safety along with reduced environmental impact. Thus, it can be stated that poor control performance leads to poor plant performance. [3, pp. 4-6] The aim of controller performance assessment (CPA) and monitoring is to make sure that control systems perform according to their specifications. Therefore, control performance assessment is an important technique for ensuring the effectiveness of process control and safe and profitable plant operation. [5; 6]

In the process industry, the current trend at plants is a thin organization. This means that there is not necessarily expert knowledge of the process and automation at the plant itself. The goal of this work is to develop a performance assessment tool for a multivariable, model-predictive controller and then apply it to a process environment to verify that the tool is operating as required. The performance indicator tool aims to provide beneficial information of the controller's operation and so to ensure effective process control.

1.1 Background

Most of the theories and applications of control performance monitoring and assessment were developed during the 1990's. The first control performance assessment technique was established in 1989 by Harris [7] who demonstrated that the minimum variance benchmark can be estimated from routine operating data. Minimum variance control (MVC) is based on a comparison of this theoretical lower bound benchmark and the variance obtained. However, the minimum variance benchmark is not so easy to apply to multivariable processes where an interactor matrix is needed [8]. Other benchmarking techniques were developed, such as linear quadratic Gaussian (LQG) benchmark which was proposed as an alternative to the minimum variance benchmark by Huang and Shah in 1999 in their textbook "Performance Assessment of Control Loops" [3, see Huang&Shah 1999]. The LQG method is a suitable technique for assessing the controller performance because it minimizes the process output variance and takes the input variability also into account [1]. Control performance assessment techniques based on user-specified benchmarks and model-based benchmarks have also been developed. In addition, multivariate MVC techniques have been proposed. [6]

Besides researchers, also plants are interested in the assessment of controllers. Because the performance of the control system and the nominal efficiency usually degrade after the implementation, the investment of a new control system does not pay itself back as quickly as expected. In 2008, Bauer and Craig published a survey focused on the economic performance assessment of APC and both control system suppliers and users were interviewed to get a view of the role of the control performance assessment in general. Also several APC solution providers and the usage and recognition levels of different APC solutions were rated. [4] Many of the major APC suppliers provide a tool for performance monitoring and assessment of the control system. Some of the tools are based on web-based monitoring and some suppliers provide regular diagnostics and auditing services. Some of the available tools are not designed for assessing the performance of the APC systems but the plant's lower control loops in the distributed control systems (DCS). The more detailed operational principles of these performance assessment tools are not available. [9-11]

Control performance assessment techniques have been studied in 2012 by Janne Oksanen in his master's thesis "Performance Assessment of the Multivariable MPC Controller" [12]. The aim of his thesis was to develop a reliable and accurate application for computing the technological and economic performance of the MPC controller. He has done a wide research of different performance assessment techniques in order to select suitable methods for the application designed and applied to the experimental part of his work. The first selected method was based on an analysis of historical process data and the second was a model-based design-case approach. The methods were applied for computing the technological and economic performance indices and various simulation cases were used to examine the operation of the developed application. Several applications, proprietary of Neste Jacobs, were used in the work. The software included NAPCON Controller, ProsDS simulator, NAPCON calculation frame, Display Viewer, and Database Server and Explorer.

1.2 Known challenges

Since the model predictive control is the selected multivariable control method in many process industries, there are many researches about assessing the performance of MPC. Many performance assessment techniques can easily be applied when monitoring a univariate process. These techniques are not applicable to multivariable processes because of the relatively complicated process delay structure which is needed for the creation of the complete interactor matrix that the minimum variable control techniques require. In addition to the complexity of a multivariable controller performance assessment, MPC involves model errors, constraints, optimal target settings, disturbance changes and controller tuning. The performance monitoring of MPC is mostly an unsolved problem because of the MPC complexity. [3, p. 22; 8]

Though this particular subject was recently in 2012 studied by Oksanen [12] and the created application worked as designed, there are a couple of deficiencies in his work that need to be solved. One problem was related to the historical benchmarking. When the simulations included an unknown disturbance, the historical benchmark did not react as supposed and became biased. Other improvement to earlier application is related to the generalization of the implementation and deployment of the performance assessment tool. In particular, the economic performance is usually very case-specific. A simple solution to generalize the attainment of economic performance is required.

1.3 Aim of this work

The aim of this thesis is to implement a tool for assessing the technological and economic performance of a multivariable, model-predictive controller which is in this case NAPCON Controller developed by Neste Jacobs. This thesis focuses on the performance assessment of the controller and it does not cover the diagnosis phase of the control performance monitoring.

This work is a continuation to the master's thesis of Janne Oksanen done in 2012. He has done a research of control performance assessment techniques and selected the most suitable for the application developed in his work. In this work, a research of the recent control performance assessment methods is carried out and the recently published control performance assessment techniques are introduced. They are compared to the methods selected in Oksanen's work and considered, if they would assess the performance of the controller more suitably. In addition, solutions to problems such as the biasing of the historical benchmark are tried to be found.

In addition, this work needs to consider aspects related to the automation system, the industrial network and the information technology in order to create a software implementation of the performance assessment tool for multivariable, model-predictive controller. Measurement data from the field is involved for assessing the controller's performance and therefore discussion about automation communication protocols as well as information modelling aspects is required. User interfaces and their basic design principles are studied for delivering the received information clearly.

Testing of the tool is carried out with various simulations and assessment of an actual MPC controller's performance. The function of the performance assessment tool is examined with simulated process data and thereby the operation of the tool is verified.

1.4 Content of the work

The literature part of this work includes Chapters 2 and 3. First in Chapter 2, the model predictive control is presented. The basic principle is explained and the subjects affecting the performance of MPC are presented. In Chapter 3, the performance assessment of MPC is discussed. The performance assessment technologies are introduced generally and recent publications and their results are presented. Control performance assessment methods based on historical and design-case benchmarking are introduced more thoroughly.

Chapters 4 and 5 introduce the aspects that are to be considered before the implementation part of this work, when designing the performance assessment tool. Chapter 4 presents the automation information technology aspect of the work which is required in order to implement the performance assessment tool. Communication protocols in automation and OPC UA information modelling are introduced besides the different automation levels and data transfer from the plant to the actual controller. In order to present the received performance information clearly, graphical user interfaces (GUI) and their basic design methods are studied. In Chapter 5, aspects related to the general functionality of the program are introduced. The current status of the subject is presented along with the required development. In addition, the selected performance assessment techniques and their advantages and disadvantages are shortly discussed.

The experimental part of this work consists of Chapters 6, 7 and 8. Chapter 6 contains the implementation part of the work. The software environment related to the thesis is introduced along with the new object types, the performance calculation application and the graphical user interface. The test environment, the test arrangements and the obtained test results are presented in Chapter 7. In Chapter 8, the feasibility of the performance assessment tool and the reliability of the test results are concluded. Finally, the future aspects are discussed in Chapter 9 and the essential content and results of the work are summarized in Chapter 10.

2. MODEL PREDICTIVE CONTROL

Advanced process control (APC) has been adopted by the operating plants in order to meet a variety of challenges, such as the focus on environmental protection and the price pressure of raw materials, which the process industries have to meet nowadays in order to be competitive [1]. The development of model predictive control (MPC) in the 1970s has been a major innovation and since then MPC has been regarded as the most popular advanced multivariable control strategy [2].

Generally, MPC can be referred to as a class of control algorithms which predict the future process outputs based on the process model and compute a sequence of future inputs attempting to optimize the future process behaviour. At each control interval, the calculation is repeated and only the first input of the computed control sequence is applied to the system. [13] MPC executes control actions that often enable an improved process performance that even an experienced operator cannot achieve [14].

A model predictive controller is part of a multi-level control hierarchy in modern processing plants [13]. A general plant control hierarchy and approximate control intervals are presented in Figure 2.1.

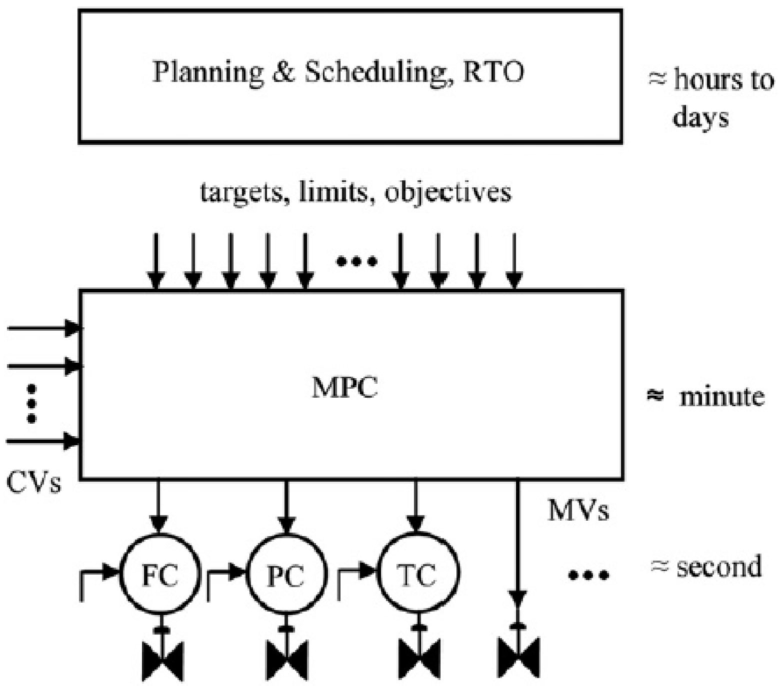

Figure 2.1. *General control hierarchy and approximate control intervals [15].*

At the top of the control structure, there is an optimization level which controls the plant-wide planning and scheduling and determines optimal steady-state settings, such as target and limit values. The optimization interval varies from hours to days. The information is delivered from the optimization level to a model predictive controller. The decisions from upper level functions to targets and limits in MPC are not constant, and they will change over time based on priorities and economic objectives. The control interval of the MPC can be generally expressed in minutes. Below the MPC is the regulatory control level in a cascade arrangement. Typically, the manipulated variables of the MPC are the set points of lower level basic controllers, such as PID controllers, which are part of the plant's distributed control system (DCS). The sampling period of the DCS is typically expressed in seconds and thus it executes at a shorter sampling interval than the MPC. [13; 15]

The way that MPC solves the process control problems has brought MPC wide popularity, especially in oil refining and petrochemical industries. MPC is the only control methodology that is developed in industry and accordingly, it has affected the industrial control engineering significantly. The methodology of MPC has various advantages when compared to other control techniques. For example, it is possible to be used to control a great variety of processes and its basic formulation allows extension to multivariable processes with almost no major modifications. Also operational and economic criteria can be taken into account in the formulation of objective function. [3, pp. 97-99] The basic principle of the control methodology of MPC is presented in Section 2.1. There are different control objectives that can be considered for determining that the MPC is performing well. Since the controller performance can deteriorate due to various causes the subjects associated with the performance of MPC are discussed in Section 2.2 along with the control objectives.

2.1 Basic principle

The basic methodology of a model predictive controller is characterized by a strategy referred to as a receding or moving horizon control, where the control action is attained by solving a finite horizon optimal control problem at each sampling instant. The optimization is carried out based on a process model. As a result of the optimization, an optimal control sequence is obtained and the first control action is executed. [3, p. 97] The basic principle of the model predictive control strategy is illustrated in Figure 2.2.

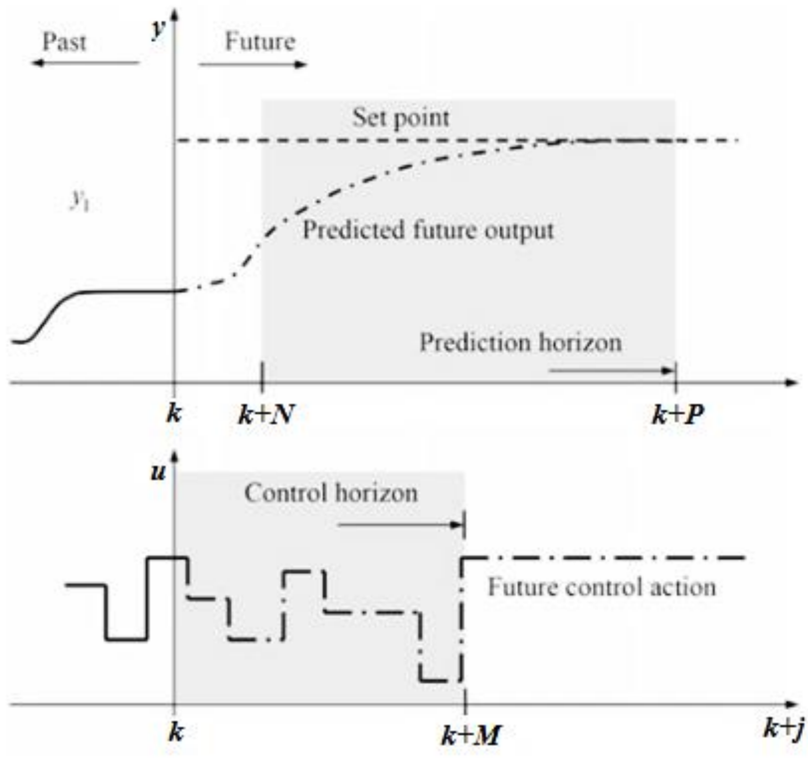

Figure 2.2. *The principle of model predictive control strategy, adapted from [3, p. 98].*

The future outputs of the process are calculated using the process model and they are predicted over a prediction horizon *P*. The time interval when it is desired that the process variable starts to follow the reference is defined by *N*. The predicted output values depend on the past inputs and outputs besides the future control actions. The future control actions are calculated over a control horizon *M*. A set of future inputs is obtained by solving the optimization problem. By minimizing the determined criterion, the future outputs are tried to keep as close as possible to the reference trajectory which defines the desired process outputs. The outputs aim to follow the reference trajectory during the time that is determined by *N* and *P*. If there are constraints involved, the model is nonlinear or the criterion is not quadratic, an iterative optimization method needs to be used. Otherwise an explicit solution can be attained. [3, pp. 97-100]

After the determination of the future control actions, the first input of the set is applied to the system while the rest of the obtained control sequence is rejected. This is because at the next sampling instant the next value of the output is actually known and the calculation of the future outputs can then be repeated with knowledge of this new value. In addition, it is possible that the reference trajectory has changed over the following sampling instants. The new control sequence is calculated with the new available information, using the receding horizon strategy. [3, p. 98]

Usually the optimization problem to be solved is an objective function which is minimized. Generally, the objective function *J* for a multivariate model predictive controller can be expressed as

$$J = \sum_{j=N}^{P} \left[\hat{Y}(t+j) - W_t(t+j)\right]^T Q \left[\hat{Y}(t+j) - W_t(t+j)\right]$$

$$+ \sum_{j=1}^{M} \left[\Delta\hat{U}(t+j-1)\right]^T R \left[\Delta\hat{U}(t+j-1)\right], \tag{1}$$

where $\hat{Y}(t)$ is the vector of the predicted output values, $W_t(t)$ is the vector containing the reference trajectory and $\Delta\hat{U}(t)$ is the vector of the changes in future input values. The weight matrices Q and R represent the relative importance of each output and input. [6]

Usually there are constraints in the actual process that restrict the behaviour of controlled or manipulated variables. Therefore the solution to the objective function in equation (1) is also restricted. In almost every actual model predictive controller the constraints are considered and the objective function in equation (1) is minimized subject to

$$Y_{min}(t) \leq Y(t) \leq Y_{max}(t), \tag{2}$$

$$U_{min}(t) \leq U(t) \leq U_{max}(t), \tag{3}$$

$$\Delta U_{min}(t) \leq \Delta U(t) \leq \Delta U_{max}(t), \tag{4}$$

where the input changes are computed as

$$\Delta U(t) = U(t) - U(t-1). \tag{5}$$

Ideally the output constraints are direct to the actual output. However, the controller does not directly constrain the actual output due to the unpredictable disturbances. Models are therefore essential in MPC because only the predicted output can actually be constrained. [16, p. 108]

2.2 MPC performance

The main goal of the control systems is to maximize profits by transforming raw materials into products while the requirements, such as product quality, safety and environmental specifications and operational constraints, are met. After a thorough implementation and tuning, the controller is usually performing as designed and required. Though the controller performs well initially, the performance can gradually or even abruptly decrease over time due to various causes, such as changed process conditions or disturbances that limit the achievable performance. [3, p. 1; 6] The performance of MPC may also be limited due to model uncertainty as well as possible conservative tuning [2].

Although MPC usually automates some tasks that were previously handled by the operator, it is however required that the operator has confidence to the MPC system in order to allow the MPC to perform as specified. It is possible that the MPC is turned off if the control actions are not understood and then not switched back on anymore. Thus, human factor may have an effect on achieving the complete performance from the MPC. An operator, who trusts the MPC and knows how it is supposed to act and what the control objectives of the MPC are, is likely to keep the MPC running. Correspondingly, the operator is able to identify the situations where the MPC is underperforming. [14]

2.2.1 Control objectives

Determining the process variables and their interaction with each other is a major part of the MPC design and implementation [15]. Process inputs include manipulated variables which are the variables that the controller adjusts and disturbance variables that are not controllable. [13] To implement MPC successfully, the configuration of DCS regulatory controls is to be done thoroughly. As presented in Figure 2.1, MPC is in close cooperation with the lower regulatory control level. The decisions related to the process input variables are made based on the expected constraints, qualitative knowledge of the expected disturbances and considerations of robustness. Additionally, the determination of the controlled variables of MPC is not straightforward and it cannot be done only based on the selection of the available measurements. The available measurements may not be sufficient and additional sensors would be required. In addition, every variable that would need to be controlled may not be available frequent enough. [15] In addition to the importance and priority of different variables, the variable types are to be determined. There may be ideal target values of manipulated or controlled variables which are tried to be reached if possible. Typically, there are also constrained variables in MPC that are not allowed to exceed specified limits. [13]

In order to retain feasible control with constraints, there are two basic types of constraints that are used in industrial MPC technologies. Constraints of manipulated variables often represent physical limitations, such as valve position, that cannot be violated. They are referred to as hard constraints that should never be violated. Constraints of controlled variables do not usually represent that kind of hard physical limits and they often represent desired operation ranges instead. These constraints are treated as soft constraints for which some violation may be allowed. Typically, the violation is minimized using a penalty in the objective function. Using a set point approximation is another way to handle soft constraints. The operating principle of a set point approximation of a constraint is that it penalizes deviations above and below the constraint. [13; 17]

The soft constraint formulation is not always sufficient because then all the constraints are violated at some scale based on the relative weights. Some constraints are more important than others and they should not be violated. Therefore, priorities are often determined for constraints in order to satisfy the ones with higher priority values. However, the optimal operating points are usually located near the constraints, and when possible the variables are operated at the presence of the constraints. The degrees of freedom of the process are an essential part of optimizing the process operations. The optimization is achieved by manipulating the degrees of freedom while satisfying the operating criteria. [13; 18] When the number between the controlled and manipulated variables is suitable and there are extra degrees of freedom available, optimization is possible and the process can be moved closer to an optimal operating point. When there are as many manipulated variables as controlled variables, a unique solution is attained. Sometimes some valves may become saturated or lower level control action is lost and there are more controlled variables than manipulated variables available. Then it is not possible to achieve all control objectives. [13]

2.2.2 Controller tuning

The performance of the controller may be limited due to inadequate controller tuning. It is possible that the controller tuning is left unchanged after the design and implementation, although there may be issues such as changes in the characteristics of the used input product or modifications of operating points that decrease the controller performance over time. [3, p. 9] Tuning of a model predictive controller is however not advised to be done on a day-to-day basis. To tune the MPC parameters, one needs to have good process knowledge besides a full understanding about the MPC control algorithm and the effect of the parameters to be tuned on the controller performance. Although, some of the variable constraint limits may be commonly changed in the daily operations, which can sometimes improve the controller performance. Nonetheless, understanding the interaction between constraints, process variability and objective functions is essential for constraint tuning. [2]

Generally, there are many tuning parameters for a basic model predictive controller. The fundamental parameters of the MPC are the prediction horizon, the control horizon and the cost weights for set point tracking and input changes. There are also additional tuning parameters that more advanced model predictive controllers have, for example parameters related to reference trajectories and output funnelling. The appropriate parameter values for achieving the desired controller performance are not easy to find, although there usually is a clear description of how each parameter is meant to affect the MPC formulation. [14] In order to achieve a good control performance, some guidelines for appropriate values of MPC tuning parameters performance have been obtained based on experience of applying MPC to different processes. The guidelines provide only a basis for the tuning of a model predictive controller, and in practice the parameters are often determined by trial and error. Because the controller parameters depend on each other, this can be really time-consuming. [3, p. 100]

Selecting the weight factors of each input and output variable and determining the weighting matrices Q and R affect the operation of the controller. An aggressive control is received when the weighting of input changes is decreased and greater changes in input variables are allowed. Thus, a more aggressive control makes the control response faster. On the other hand, increasing the weights of inputs in weight matrix R makes the control more damped and conservative. It is possible to weight different controlled variable errors with separate coefficients of each controlled variable in weight matrix Q. This enables that more important controlled variables, such as quality variables, can be emphasized with a greater weight factor. [3, p. 100]

The time interval in which it is desired that the process output follows the reference trajectory is defined with N and P in equation (1). Typically value N is chosen to be zero. However, if the system contains any time delays there is no reason to determine the value of N smaller than the time delay. The prediction horizon P is usually chosen approximately the same as the output settling time. The control actions will be more aggressive with a smaller value of P. The length of control horizon M should be less than prediction horizon. Typically, the value is a one-half or one-third of the prediction horizon in a process with large time constants, such as chemical processes. The control is more aggressive with a large control horizon. In order to assure the stability of infinite horizon MPC, the control horizon is supposed to be greater or equal when compared to the number of unstable poles in the process. [3, p. 100] However, using the values of the prediction horizon and the control horizon for the controller tuning is not highly recommended because the system behaviour is quite insensitive to changes in both parameters over a wide range of values. Consequently, the weight matrices Q and R are preferred in the controller tuning to affect control performance. [19]

2.2.3 Process modelling

The fundamental part of MPC is the model, which generally consists of two parts: process model and disturbance model. The success of using MPC depends greatly on having reasonably adequate process models. Typically, a process model represents relationship between input and output and a disturbance model is used to present either disturbance or simply to approximate model-plant mismatch. [14; 16, pp. 103-104] The parameters of the model are practically never exactly known and a model-plant mismatch exists. For example, the time delay of the process may be time variant due to changing flow rates. Thus, the presence of a model-plant mismatch causes a difference between the designed and achieved performance. [20] There is awareness of the imperfection of models and it is possible to overcome some effects of poor models with feedback, but the feedback may be late to be truly effective [17].

Most industrial processes are inherently nonlinear. However, most industrial MPC applications use linear process models. The nonlinear processes can be approximated by fixed linear models but only near the operating point. The mismatch does not however often cause much degradation in the performance of MPC. [17; 20] Besides changing operational objectives of the plant, there are various disturbances in the process and consequently the process variables do not always operate at the mean operating points [2]. A commonly used approach for applying a linear MPC to nonlinear processes is a gain scheduling technique which means that the process operations are divided into different operating regions. The process model parameters are determined for each operating region separately and as the process operations change, the MPC is updated to use the model parameters of the new region. [14]

There are some difficulties related to the identification of the models besides types of used models. Usually in order to identify the process models, well designed experiments are run for collecting the data needed for the identification. However, a common problem with this approach is that the experiments may cause perturbation in the process. Sometimes some disturbance in the process is acceptable when the performance of the MPC is supposedly increased. In order to maintain a desired performance over time, a re-identification of models may also be required and therefore simple identification methods would promote the maintenance of the MPC. Using complex identification software and models may lead to disregard of the model adequacy. Although the use of simple model form does not provide the optimum process model, it reduces the risk of large model errors because of the possible user unfamiliarity with higher order models. [14] Usually, more complex models are more expensive to develop but they provide a more specific prediction of the process behaviour [13].

3. MPC PERFORMANCE ASSESSMENT

The main objective of control systems is to produce maximum returns while meeting the requirements, for example quality specifications of the product or safety and environmental regulations. The control system is carefully designed and tuned, and after a thorough implementation the system usually performs at its nominal efficiency. However, the performance of the control system usually decreases over the time although the control system performs well initially. The degradation can be a result of many different causes, such as changing process conditions, a change in the plant equipment or disturbance characteristics, or lack of maintenance. [3, p. 1; 5; 6]

Recently, there has been both academic and industrial interest in the development and application of techniques for analysing the performance of control systems [21]. The aim of control performance monitoring is to give procedures for evaluating the performance of the control system and to provide information to the plant personnel of how the process is operating. The term monitoring refers to the action of detecting changes in a statistic that presents the control performance over time whereas the term assessment means the action of evaluating the statistic at a certain point. However, both of the terms are used in a rather similar meaning in the literature. Control performance monitoring and assessment are essential to make sure that the process control is effective and that the specified performance targets and response characteristics are achieved by the controlled process variables. In addition, profitable and safe operation of the plant is a significant and desirable objective. [3, p. 11; 6] Along with other control systems, also an increasing interest in the performance monitoring and assessment of MPC has appeared because of the popularity of the MPC in the processes. Although the monitoring of MPC has developed since the early applications, there is no consistent and standard solution to the MPC performance monitoring and benchmarking. [14]

The performance of MPC can be evaluated from various aspects. First in this chapter, two different approaches for assessing the MPC performance are introduced. After that, the techniques for the performance assessment of MPC are presented. In Section 3.2.1, the ideas of basic control performance assessment methods are outlined generally. In Section 3.2.2, recent publications related to the control performance assessment techniques are studied and some new techniques are introduced. Finally, the CPA techniques selected and applied earlier in Oksanen's work [12] are presented.

3.1 MPC performance approaches

The performance of model predictive control can be approached from different aspects. In order to get a complete presentation of both process and controller state and therefore more thorough results on control performance assessment, both technological and economic performance of MPC needs to be considered.

The technological performance of MPC concentrates on the functionality of the controller, how well the controller meets the requirements that are given present, in history and in future. The economic performance presents the financial profit of the process based on the raw material and product prices. It can be considered that the economic performance of the controller is at the desired level if the technological performance is as specified and the control objectives are met. [12]

3.1.1 Technological performance

The technological performance can be considered as how the controller itself is performing according to the given requirements and control objectives as well as to conventional control performance aspects, such as control robustness and process variability. The technological approach is closely related to the appropriate controller tuning and adequate process models which determine the technological operation of the model predictive controller. Subjects that can be considered to be related to the technological performance of MPC are presented more closely in Section 2.2 where the MPC performance is discussed.

Robustness related difficulty is connected to the open-loop nature of the MPC optimal control problem and the indirect feedback that the receding horizon structure brings [17]. Control system is generally defined to be robust when the process stability is maintained and the determined performance specifications are achieved over an uncertainty range. This means that a robust control system meets these requirements and specified performance criteria while process conditions change. [19]

In the most industrial processes, uncertainty is a basic characteristic. Significant process disturbances may be due to several causes such as flow rate and temperature variations or feed quality fluctuations. The essential goal of the process control is to reduce the variability and thereby increase the operational performance. [22] Stochastic performance criteria generally contain the variance of the controlled variable or control error and criteria that are directly related to process performance, product quality and energy or material consumption. [3, p. 4; 21] Figure 3.1 illustrates the reduction of the variance compared to the base case with the original variance.

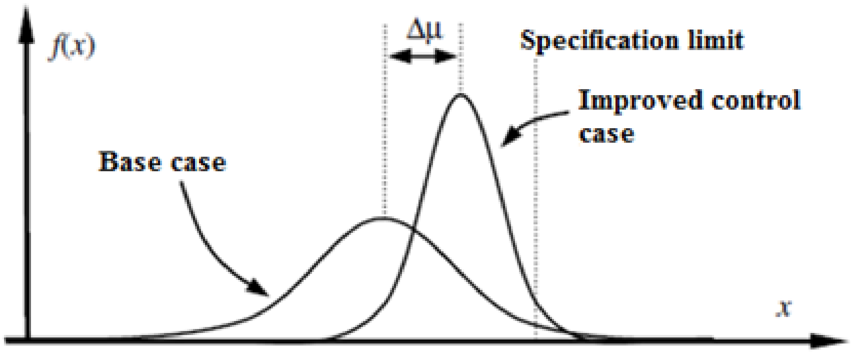

Figure 3.1. *The improvement when the variance is reduced and the process can be operated closer to the specification limit, adapted from [4].*

Improved control actions aim to reduce the variance of the quality variables or key process variables. The decreased variance allows the mean operating value to be moved closer to the limit, such as a quality specification or an operation constraint, without an increase in the frequency of limit violation. [22]

3.1.2 Economic performance

The reduction of the process variability is essential also for the economic performance of MPC. For major industrial processes, the most valuable operating strategy is to operate as close to the limits as possible in order to improve the process profitability. When the operating point of the process can be shifted closer to the limit and the process can be operated in the new operating point, the economic benefit and the profit increase are realized by an increase in the throughput and production. If the quality of the product remains unchanged, the additional quantities can be sold at the same price. [4; 22] Figure 3.2 presents an example of the profit increase when the variance is reduced.

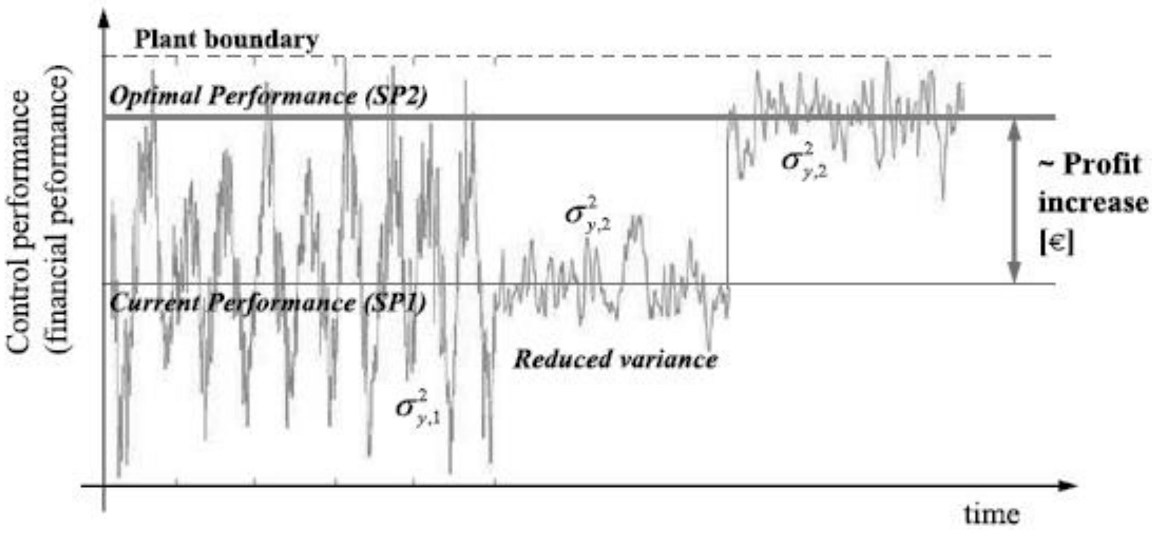

Figure 3.2. *The profit increase with reduced variance, adapted from [3, p. 5].*

Zhao et al. [22] have proposed that the expected economic performance is possible to calculate with the performance function $\vartheta(x)$ and the probability density function $f(x)$ of the quality variables, if the functions are available. The relationship between the steady-state economic performance and mean operating point is determined through this performance function. The expected performance can then be expressed as

$$P_{eco} = E[\vartheta] = \int_x \vartheta(x) f(x) dx \, , \tag{6}$$

where P_{eco} is the expected performance, $E[\cdot]$ is the expectation operator and x is the location of a quality variable. An economic performance objective function includes some key or quality variables which are functions of the operating conditions. For example, consumption rate of raw materials and product rate can be considered as quality variables and the economic performance function can be developed based on the quality variables of the process which affect the economic performance. In general, the economic performance function is expressed as

$$\vartheta = \sum_{i=1}^{P} c_y^{(i)} y_i - \sum_{j=1}^{M} c_u^{(j)} u_j \, , \tag{7}$$

where $c_y^{(i)}$ and $c_u^{(j)}$ are respectively the economic performance coefficients of *i*th output and *j*th input variables. The economic performance coefficients can be for example the market prices or demands of products. [1; 22]

Besides an increase in more valuable products, reduced variance of the controlled variables increases the quality consistency of the products and therefore the quality giveaways are also reduced. In addition, energy consumption is a major part of the economic performance of the process control. The energy consumption can significantly be reduced when the performance of the controller is as required and the appropriate process variables are adjusted and controlled. Reduced variability also improves the process stability and therefore ensures that the specifications of the product are met along with a safer process operation. [4]

3.2 Performance assessment techniques

The control performance can be specified by different criteria and an approach is to divide the criteria to deterministic and stochastic performance categories. Deterministic performance criteria contain the traditional performance measures of controller in the case of deterministic disturbances, such as set point changes. The measures include for example rise time, settling time, control error, overshoot and offset from set point. [3, p. 3] However, the inputs of a control system mostly vary at random and therefore the outputs of the performance measure will also be stochastic. Thus, statistical analysis techniques should be used for detecting changes in the controller performance. [6] The variance of the output error is the most widespread stochastic criterion considered for control performance assessment (CPA) because it usually is directly related to process performance, product quality and profit. In many applications, it is efficient to use both deterministic and stochastic criteria, which is typical in the optimal and model predictive control. [3, p. 4; 21]

CPA techniques are generally based on the comparison of the current controller performance and the ideal controller performance as a benchmark value, meaning that certain performance metrics are set in relation to what can be achieved by an optimal controller or a controller which has the desired properties. Consequently, this comparison is called the control performance index (CPI) which measures the performance of the controller. CPI is generally defined as

$$\eta = \frac{J_{opt}}{J_{act}}, \tag{8}$$

where J_{opt} is the ideal, optimal or desired value for a given performance criterion, which typically is the variance, to be minimized and J_{act} is the actual value of the criterion determined from the actual process data. When using perfect or optimal control as a benchmark, the control performance index is in the scale 0…1. Values close to 0 indicate that the control performance is poor, whereas values close to 1 mean better control performance. When a more realistic or a less severe user-specified benchmark is used, it is possible that the performance index reaches values higher than 1, which indicates that the current controller performs better than required and J_{act} is smaller than J_{opt} with given performance criterion. [3, pp. 13-14; 21]

3.2.1 CPA techniques in general

Control performance assessment (CPA) is a relatively young field of research as major of the theory and applications were developed in the 1990's. Various CPA techniques have been proposed and many of them are based on the benchmarking technique presented earlier in equation (8) in order to deliver a metric for assessing the control performance. Based on the type of the selected benchmarks, the stochastic CPA approaches can typically be classified into minimum variance control (MVC)-based and user-specified benchmarking. In addition, advanced and model-based approaches of CPA methods, including linear quadratic Gaussian (LQG)-based and model predictive control (MPC)-based benchmarking, have been proposed. [3, p. 19; 21]

The most commonly used metric for assessing the performance of a control system is MVC benchmark proposed by Harris [7] in 1989 and the MVC benchmark is therefore also known as the Harris index. Technique is based on the comparison of the performances of the actual control and the minimum variance control. Thus, the performance of the actual controller indicates how far the output variance is of the minimum variance which determines the theoretical lower bound benchmark. The MVC-based benchmarking is easy to apply for assessing the control performance in a univariate case and the main reason for the popularity of the Harris index is its relative simplicity. Though, the application of the minimum variance benchmark in a multivariable process is not a simple task because of the interactor matrix which is required in the calculation of the MVC benchmark. The knowledge of the interaction between all variables including delay terms mean that the requirements and the computational burden of the technique increase considerably. Methods for calculating the interactor matrix based on the Markov matrices and then using the MVC benchmark for multivariable controllers have been proposed along with other performance assessment techniques based on the minimum variance, which are known as generalized minimum variance control (GMVC)-based methods. For example, Huang et al. [8] proposed a method without the knowledge of the interactor matrix or Markov matrices. [3, p. 19, 163; 21]

An alternative to the MVC benchmark was proposed by Huang and Shah in 1999 in their textbook "Performance Assessment of Control Loops" [3, see Huang&Shah 1999]. They proposed a LQG-based benchmark which is based on the LQG trade-off curve. The trade-off curve, also referred to as the performance curve, displays the controlled variable's minimal achievable variance against the variance of the manipulated variables. This achievable performance limit can be used as a CPA benchmark. When the controller operates near to this limit, it can be considered to be working close to the optimal performance. [6] LQG benchmark is equal to the MVC benchmark when the weights of the manipulated variables are set as zeros. The problem of using LQG benchmark is also similar to the MVC benchmark because the knowledge of the process and disturbance models or their relative information as Markov parameters is required. [21]

Especially the performance assessment of model predictive controllers is a field of interest because of the major role that MPC has as a multivariable controller [3, p. 21]. In the case of model predictive controllers, the drawback of LQG method related to the need of models for computing the performance curve is not a problem because the process models are available. However, when the LQG method is applied to MPC environment, it has been demonstrated by Julien et al. [23] that the performance curve of MPC is significantly above the LQG benchmark. In addition, the results for a univariate model predictive controller and the issues related to the interactor matrices in multivariate cases remain unresolved. Another model-based approach for performance assessment is a method proposed and recommended amongst other by Shah et al. [24]. The technique is called design-case benchmarking and it evaluates the controller performance based on the criterion similar to the actual design objective. This performance value is compared to the actual performance. [25] The design-case benchmark is introduced more detailed in Section 3.2.4. Also a performance measure similar to design-case benchmark has been proposed by Zhang and Henson [26]. Their technique compares the expected and actual process performance. When controller actions are implemented on the process model, the expected performance is attained.

The benchmarking methods formerly presented compare generally the performance of the current controller with the benchmarks that are theoretically achieved. Then the actual control performance may seem low although the controller is working as specified. These more realistic and achievable performance indices are referred to as user-specified benchmarks. It is often suitable to compare the current control performance to performance considered to be acceptable. This kind of performance assessment approach is referred to as historical benchmarking which is in the category of user-specified benchmarks. [3, p. 20; 21] The historical benchmark technique is presented more detailed in Section 3.2.3.

3.2.2 Recent CPA techniques

Over the last few years since the work of Oksanen [12], some research has been done on control performance monitoring and assessment. Many of the recently proposed methods are based on the main techniques presented earlier. Methods based on the minimum variance approach have been proposed and a deficiency of the conventional methods has been presented. According to Li et al. [27] and Yan et al. [28], the existing methods do not provide a complete knowledge of the control performance when focusing on the comparison of traces or determinants of the output covariance matrices, and the results may be misleading. A method based on the dissimilarity analysis was proposed by Li et al. The technique is based on the analysis of the dissimilarity among the hyper-ellipsoids which are defined by different covariance matrices. They proved that the eigenvalues of the transferred covariance matrices determined the similarity of the original covariance matrices. A new control performance index η_D that they proposed can be expressed as

$$\eta_D = 1 - \frac{\sum_{j=1}^{K}\left|\lambda_j^{act} - \lambda_j^{mv}\right|}{K}, \tag{9}$$

where the jth eigenvalues of transformed covariance matrices are λ_j^{act} and λ_j^{mv} and K is the number of variables. If the actual control performance is close to the performance achieved with minimum variance control, the value of index η_D is close to 1. A similar control performance index was also proposed for historical benchmarking, which uses the difference of the eigenvalues of transformed data covariance matrix of reference period and monitoring period. [27] Yan et al. [28] have proposed a technique for multivariate CPA and control system monitoring which is based on a hypothesis test on output covariance matrices. They proposed that the equality of the output covariance matrices derived from the outputs of control system should be assessed with a hypothesis test. They also proposed a new index which is more sensitive to the changes in the multivariate covariance structure of the control system. The statistic of the control performance can be calculated through the proposed test.

Techniques which indicate the potential improvement on the economic performance of the controller have also been represented. Cai et al. [1] and Duan et al. [29] have proposed techniques to solve the problems related to the conventional LQG approach which takes both controlled and manipulated variables into account and is therefore a suitable basis for economic assessment. Cai et al. have proposed a design scheme where the idea is the integration of the economic performance design (EPD) using iterative learning control (ILC) and online MPC. The method is however designed for updating the desired operating condition and the tuning parameter for optimal MPC performance and not directly for assessing the control performance. An alternate, modified LQG benchmark for economic performance assessment of MPC has been proposed by Duan et al. They have presented that the achievable region of the control performance lies above the LQG performance curve and therefore there exists economic potential. According to Duan et al., the problem related to the conventional LQG benchmarking is related to the unbalanced distribution of the discrete points in performance curve, where points of lower part, representing the output variance, lie far away from each other leading to a less accurate benchmark. They proposed changing the LQG objective function into a new form

$$J_{LQG} = E[Y^T QY] + e^{\lambda} E[U^T RU]\,, \tag{10}$$

where the weighting parameter λ is assumed to be a function of *t*. The proposed method rebuilds the discrete points and is assumed to give a more reliable assessment result than the conventional LQG benchmark. [29]

Using the controller design objective as a benchmark assumes that the existing process model is valid. This is considered as a drawback by Sun et al. [30] who have proposed a method based on the model residual monitoring. The existing control performance assessment methods do not provide direct information of the reasons behind the performance degradation. The internal model is an important part of the MPC and it can therefore greatly affect the control performance. Sun et al. have proposed a model quality index (MQI) which is a minimum variance benchmark for the model residuals and is attainable from closed-loop data. They have stated that it would be beneficial to have an assessment method to provide model mismatch information using only closed-loop data. It is pointed that the model quality is directly related to the model residual and that a residual sequence resembling white noise usually indicates a good model. Sun et al. have demonstrated that the disturbance innovations $e^o(k)$ can be estimated from the residuals $e(k)$ and can be obtained from the closed-loop data. A model quality index with respect to the MPC control objective can be expressed as

$$\eta_{MQI} = \frac{\sum_{k=1}^{P_{QMI}} e^o(k)^T Q e^o(k)}{\sum_{k=1}^{K} e(k)^T Q e(k)}, \tag{11}$$

where Q is the weight matrix of the outputs and P_{QMI} is the data length of performance index assessment. If the model quality index η_{MQI} is close to 1, it means that the disturbance innovations are close to the residuals and that the model is near perfect. The advantage of using a separate MQI is that it indicates how adequate the model is and therefore enables a more efficient control performance monitoring. [30]

3.2.3 Historical benchmark

When assessing the performance of the controller, the techniques based on the theoretical minimum benchmarks may seem unachievable and sometimes unrealistic. Therefore it is often reasonable to compare the current performance of the controller to a reference value achieved when the controller was performing as required. The actual MPC performance can be measured using plant data when calculating the cost function expressed as

$$J_{act} = E^T(t)QE(t) + \Delta U^T(t)R\Delta U(t), \tag{12}$$

where the controlled variable errors are defined as $E(t) = Y(t) - W_t(t)$, control moves as $\Delta U(t)$, the weight matrix of control errors as Q and the weight matrix of control moves as R. Generally, the value of the cost function is a random variable due to the effect of disturbances and measurement noise. Accordingly, it is more suitable to measure the achieved performance of the controller as an average or expected value of the cost function, as presented in equation (13)

$$J_{ach} = E[J_{act}(t)] = E[E^T(t)QE(t) + \Delta U^T(t)R\Delta U(t)], \tag{13}$$

where $E[\cdot]$ is the expectation operator. Controlled variable errors $E(t)$ and control moves $\Delta U(t)$ are computed from the plant data. [6]

The value of historical benchmark J_{hist} is calculated similarly as in equation (13). The value is computed for the selected input and output data of the plant for a certain time frame when the controller is evaluated to be performing well and having a good response according to the given criterion. [3, p. 76; 31] When monitoring the controller performance online, the performance index η_{hist} is computed at each sampling time. In equation (14), a moving horizon P_c is used for calculating the value of the achieved cost function $J_{ach,hist}$.

$$J_{ach,hist} = \frac{1}{P_c}\left[\sum_{j=1}^{P_c}\Big(E^T(t+j-P_c)QE(t+j-P_c) + \Delta U^T(t+j-P_c)R\Delta U(t+j-P_c)\Big)\right], \tag{14}$$

where $E(t)$ is the vector of controlled variable errors at time t calculated from the plant data. Like in equation (8), historical performance index can be expressed as [6]

$$\eta_{hist} = \frac{J_{hist}}{J_{ach,hist}} \ . \tag{15}$$

It may be useful to use statistical monitoring for detecting statistically significant changes in the performance index. This can be done by monitoring the residuals between the values predicted by the model and the measured values. The historical performance index η_{hist} is possible to represent by using an autoregressive model:

$$A(q^{-1})\eta_{hist}(t) = \varepsilon(t) \ , \tag{16}$$

where $A(q^{-1})$ is monic polynomial and $\varepsilon(t)$ is zero-mean, uncorrelated, Gaussian noise signal. For estimating $\eta_{hist}(t)$, the equation (16) can be expanded as

$$\eta_{hist}(t) = (a_1q^{-1} + a_2q^{-2} + \cdots + a_{na}q^{-na})\eta_{hist}(t) + \varepsilon(t) \ , \tag{17}$$

where the estimates of coefficients a_i are attained from the process data analysis. [6]

In order to use a historical benchmarking technique, plant data needs to be collected during a time period when the controller is determined to perform as required. After the benchmark value is formed, the plant data is used for calculating the performance index at each sampling time. Therefore, historical benchmarks are suitable for assessing time-varying and nonlinear processes because they do not require a process model or knowledge of process delays. Though, the historical benchmark approach requires expert a priori knowledge that the controller performance is at the desired level during the selected time period. When selecting the time period when the benchmark value is collected, the selection may be too subjective and depend too much on the current performance conditions. It is possible that the controller seems to work satisfactorily although the performance is inadequate when compared to other benchmarks. [3, p. 76; 6]

3.2.4 Design-case benchmark

The design-case benchmarking technique is a model-based method. The idea is to compare the achieved control performance to the benchmark value which uses a criterion comparable to the actual design objectives of the model predictive controller that are presented in the equation (1). The optimal control moves are calculated by the MPC controller by minimizing the equation (1). The value of the design objective function can be expressed as

$$J_{des} = \sum_{j=N}^{P} \left[\hat{E}(t+j)\right]^T Q \left[\hat{E}(t+j)\right]$$

$$+ \sum_{j=1}^{M-1} [\Delta U^*(t+j-1)]^T R [\Delta U^*(t+j-1)]\,, \tag{18}$$

where $\Delta U^*(t)$ denotes the optimal control moves and $\hat{E}(t)$ the predicted errors of controlled variables. Respectively, the achieved value of the objective function is calculated using plant data and given by

$$J_{ach,des} = \sum_{j=N}^{P} [E(t+j)]^T Q [E(t+j)]$$

$$+ \sum_{j=1}^{M-1} [\Delta U(t+j-1)]^T R [\Delta U(t+j-1)]\,, \tag{19}$$

where the control errors $E(t)$ and the control moves $U(t)$ are the measurement values of the output and inputs. The actual output may differ substantially from the predicted output due to various reasons, such as model structure's insufficiency, nonlinearities and uncertainty of modelling. The achieved values of inputs will deviate from the design values because of the receding horizon nature of the MPC law. The performance index η_{des} is given as a ratio of the design and achieved objective function, expressed as [24]

$$\eta_{des} = \frac{J_{des}}{J_{ach,des}}\,. \tag{20}$$

If the achieved control performance is exactly the same as design requirements and the outputs and inputs are corresponding to the values given by the model, the performance index η_{des} is equal to 1 [3, p. 101]. Generally, the performance index is however smaller than one due to imperfect models, measurement noise or other uncertainties [6]. The main benefit of the design-case benchmarking approach is that it measures the difference of the actual controller performance from the designed performance. If the actual values deviate greatly from the values given by the model, a low performance index is received and therefore a low performance index is actually an indication of changes in the process or presence of disturbances. The controller calculates the design objective and only the plant data of measured inputs and outputs is required for computing the performance index, which makes the design-case benchmark a rather convenient technique to use for assessing a constrained multivariable MPC. [6; 24; 31] On the other hand, when the design objective of the controller is used as a benchmark, it is assumed that the existing process model is adequate when the poor controller performance can be due to an invalid model itself [30].

For online monitoring, the design-case performance index can be computed using equation (19) with predicted control errors $\hat{E}(t)$ and optimal control moves $\Delta U^*(t)$. Statistical monitoring of the design-case performance index η_{des} is similar to the statistical monitoring of η_{hist} presented in Section 3.2.2. [6]

4. AUTOMATION INFORMATION TECHNOLOGY

Process automation is nowadays applied to various plants to maximize production while meeting the product quality and safety requirements and making the process more economical. Although the industries and the technologies of production vary, the main principles of automatic control are generic and can be applied to different environments. Process automation has changed significantly since it was first introduced in process industry. [32, p. 529] As the initial mechanical technologies were replaced with electronic systems, the influence could also be seen in the industrial control systems and networks. The industrial network connects the different automation levels and enables the data transferring both in machine-machine and human-machine interactions. Industrial networking contains the implementation of communication protocols between field equipment, controllers, various software suites and also external systems. [33]

It is important to understand the environment in which the solution is applied when developing software. In order to be able to assess the controller performance, plant data is required for the computations. This chapter gives an overview of subjects that are related to the industrial networking and how data is transferred further from the field to be presented for example in a control room and at supervisory level for assessing the process and control operations. First, the different automation levels of a typical process automation system are presented. After that, the communication protocols in automation are introduced generally to provide information of the communication layers between different automation levels, focusing on the communication between control and supervisory level. The standardized fieldbus protocols are discussed along with OPC and OPC Unified Architecture standards. Next, the term information modelling in OPC UA is presented. Lastly, the aspects related to the presentation of information and human-machine interaction is discussed and the basic design principles of a graphical user interface are presented.

4.1 Automation levels

The development of automation technology has led to increased functions which are performed by automated equipment as a part of technical systems. As a result of the development of various automation functions, modern automation systems have a hierarchical structure where each hierarchical level has different automation functions. The hierarchical structure is referred to as an automation pyramid. [34] Figure 4.1 illustrates an automation pyramid with different automation function levels and communication layers. Usually, different physical media and protocols are used in different communication layers to connect the different automation levels [33].

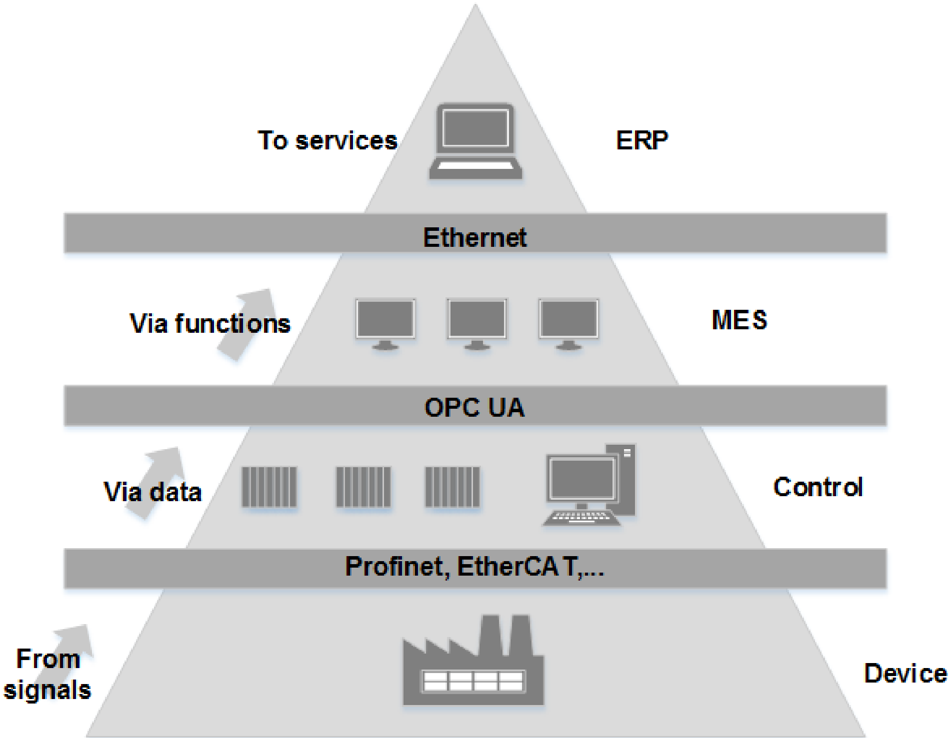

Figure 4.1. *Automation and communication levels, adapted from [35].*

The first level of the automation pyramid is the device level which includes the measurement devices and actuation equipment. Measurement devices, such as sensors and transmitters, are used to measure the process variables. Actuation equipment, for example control valves, is used for implementing the calculated control actions. Evidently, the device level containing the measurement and actuation functions is a necessary part of an industrial control system. [32]

Above the device level is the control level which consists of the machine controllers that compute the control actions that are delivered further to the lower level control actuators. The control actions are determined based on the calculations that take the measurement signals into account. Basic regulatory control is achieved with standard feedback and feedforward techniques. More advanced process control techniques are applied to solve difficult process control problems, such as complex process interactions and constraints, when the standard process control techniques may not be adequate. The APC technologies are positioned above the basic regulatory control. APC and basic regulatory control form together the control level of the automation hierarchy. [32]

The top two levels of the automation hierarchy pyramid include the plant and enterprise level control systems. Manufacturing execution system (MES) means the functions related to the operative control of the plant manufacturing which are integrated as one system. The MES level is a link between the plant's process control systems and upper enterprise level system, collecting for example information of the process condition and actions and then delivering it to upper level enterprise resource planning (ERP) systems. Conversely, the production orders and status enquiries are examples of the information transferring from enterprise-wide ERP level to production levels through plant-wide MES level. [36] As the main task of the MES level is to collect and store information of the production events, also the provided information of the plant's control performance could be taken into account in the MES level decisions.

4.2 Communication protocols in automation

In order to provide access to data in different levels of an enterprise information system, there is a need of using different communication systems within the plant, control and device levels. Each different level has its own requirements related to the communication determined by the nature and type of the transferred information. Some of the performance characteristics used to classify and to group specific network technologies are for example physical size of the network, number of supported devices, response time and sampling frequency. Real-time requirements depend on the type of exchanged information. Generally, process automation and data acquisition present soft real-time requirements, which means that no critical problems occur if deadlines are not met. [37, pp. 982-984] The higher levels of an automation network usually have lower time requirements than the lower levels [33].

The industrial networks have generally many hierarchy levels. Due to different requirements of each level, different communication protocols are used in different levels. The connection between instruments and controllers is for example at one level, interconnection of controllers at next one, human-machine interface above that and finally at the top level, the network for data collection and external communication is located. The development in process automation and movement toward digital systems generated a need of new communication protocols to the field and between controllers. Commonly, these communication protocols are referred to as fieldbus protocols. Typically, fieldbus systems are used at the field level for collecting and distributing process data between sensors or actuators and controllers. Controller networks are at the control level where data is transmitted between field devices and controllers as well as between controllers. [33; 37, p. 985]

4.2.1 Fieldbus protocols

Fieldbus as a term covers many different industrial control protocols and they are widely used in plant automation. Fieldbus systems are standardized and the digital data communication standards for the use of Fieldbus IEC 61158 and 61784 contain different Fieldbus concepts. One of them is the Foundation Fieldbus, which can be used versatilely in process automation applications. [37, p. 986; 38, p. 280] Foundation Fieldbus was developed by the American Fieldbus Foundation. The original Foundation Fieldbus is now referred to as Foundation Fieldbus H1 and it is used in lower level to usually connect field devices and host systems. Ethernet-based fieldbus Foundation Fieldbus HSE (High Speed Ethernet) was developed to address the need of Fieldbus Foundation's protocol suite for H2 level communications. It is completely compatible with the application level H1 protocol, and the Foundation Fieldbus model carries out the communication tasks by using these two bus systems. [33; 38, p. 283]

One of the most widely-used and largest fieldbus is the PROFIBUS (PROcess FIeld BUS), which is also included in the international standards IEC 61158 and IEC 61784. PROFIBUS can be used both in fast, time-critical applications and in complex communication tasks and it is in plant wide use both in factory and process automation sectors. There are different profiles that are defined for different applications within PROFIBUS, for example PROFIBUS Process Automation (PA) is designed especially to be used in hazardous environment. [33; 38, p. 289] An Ethernet-based adaption of PROFIBUS data models and objects is PROFINET, and it is defined in IEC 61158 and IEC 61784. PROFINET uses remote procedure calls (RPC) and the distributed component object model (DCOM) besides modified ether types for real-time communication. [33]

Another protocol that uses Ethernet as the transmission technology is MODBUS, which is an application layer messaging protocol. It is a protocol that is designed for client/server communication between devices that are connected with different types of buses or networks. [37, p. 988] In addition, there are new, real-time Ethernet-based fieldbus protocols that the IEC has ratified and added to the 61158 and 61784 standards [33].

4.2.2 OPC

Open Platform Communications (OPC) provides a standardized interface for communication of industrial data. The OPC standard is maintained by the OPC Foundation which has defined a standardized interface between different level automation systems in the automation pyramid. The COM (Component Object Model) and DCOM technologies from Microsoft Windows are the base of classic OPC interfaces. OPC uses a client-server model in the information exchange. An OPC server encases the source of process information like a device. The information is then available via its interface. The OPC server is connected to an OPC client that can access and consume the data. Client and server can be both applications that consume and provide the data. [33; 39] A typical use case of OPC clients and servers is presented in Figure 4.2.

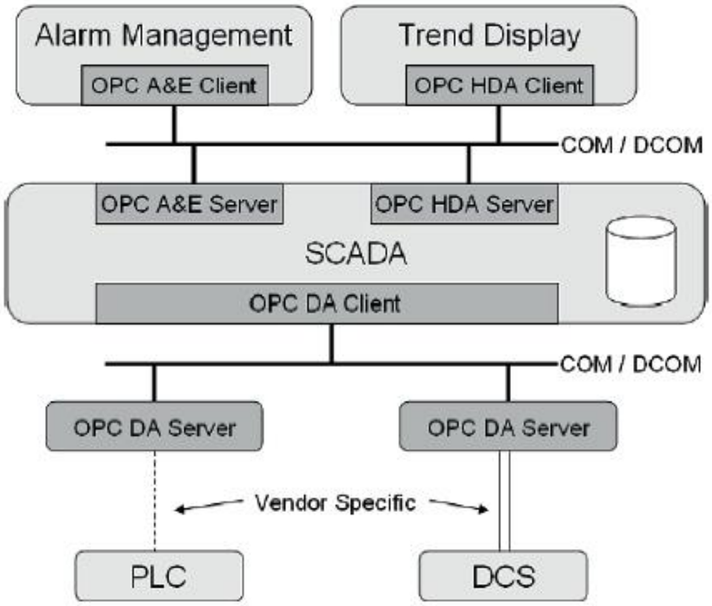

Figure 4.2 *Typical OPC client-server use case [39, p. 3].*

Variants of the classic OPC have been developed to meet the different requirements of industrial applications. Figure 4.2 shows the interaction of different OPC specifications. The Data Access (DA) specification enables reading, writing and monitoring of variables that contain current process data. The DA interface is mainly used for moving real-time data from PLCs, DCSs and other control devices to human-machine interfaces and other display clients. The Historical Data Access (HDA) provides functions to access stored data and retrieve historical archives. Besides read methods, OPC HDA defines methods for replacing, inserting and deleting data in the history database. The Alarm & Events (A&E) describes an interface that enables the reception of process alarm and event notifications to be transmitted from different event sources. [39, pp. 3-6]

4.2.3 OPC Unified Architecture

The OPC Unified Architecture (UA) is a communication protocol developed by the OPC Foundation and it was created to replace the COM dependent OPC specifications while keeping the desired communication features and performance. Additionally, the aim of OPC UA was to cover all requirements for platform-independent system interfaces with versatile modelling capabilities that enable the depiction of even complex systems. The most important requirement is the interoperability between systems from different vendors. It is also important to provide a reliable communication between distributed systems by robustness and fault-tolerance as well as redundancy. In addition, independence of platform and scalability are necessary in order to enable integrating of OPC interfaces directly into the systems that run on various platforms. An important requirement is high-performance in intranet environments, but also internet communication should be allowed through firewalls, and therefore the security and access control is another essential requirement. [39, pp. 8-9; 40]

Transport mechanisms and data modelling form the basis of the OPC UA. There are different transport mechanisms that are designed for different cases. The three main tasks for exchanging data between applications are data encoding, securing the communication and transporting the data. OPC UA data transport defines a binary TCP protocol for high performance intranet or internet communication along with mapping to accepted internet standards. The other fundamental component of OPC UA basis is data modelling, which defines the rules and base of information modelling in OPC UA. A consistent and integrated *AddressSpace* within an OPC UA Server is provided. OPC UA Server is allowed to integrate for example data and history into its *AddressSpace*. An example of OPC UA Server architecture is illustrated in Figure 4.3. The main objective of *AddressSpace* is to give a standard way for *Servers* to represent *Objects* to *Clients*. OPC UA Services provide Client an access to Server's *AddressSpace*. These *Service*s are divided into different *Service Sets*. Each of them defines a set of related *Services* used to access a specific aspect of *Server*. [39, p. 10, 191; 41; 42]

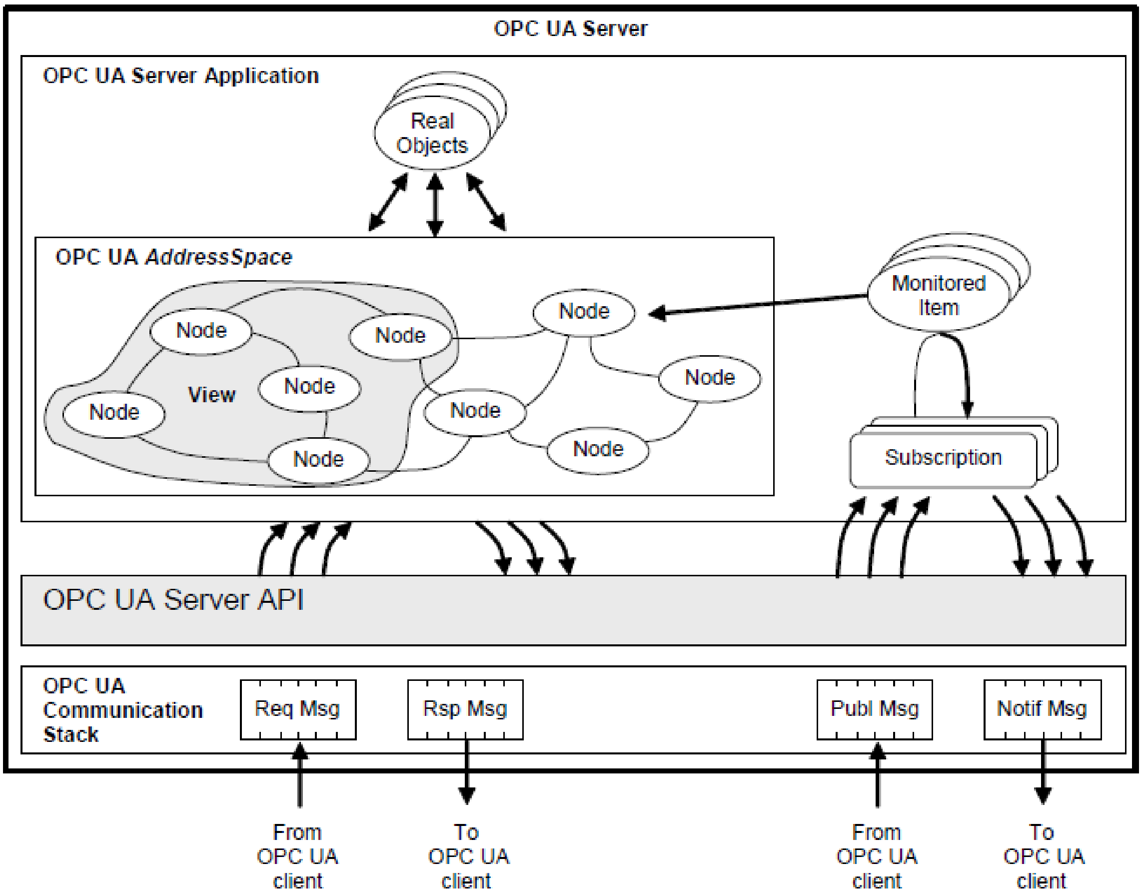

Figure 4.3. *OPC UA Server architecture [41].*

OPC UA uses a similar client-server approach that is used in Classic OPC. UA Server is the application that exposes information to other applications and UA Client is the application consuming information from other applications. However, when compared to Classic OPC there are more applications that will be both server and client in one application. The software layers of a typical OPC UA application are presented in Figure 4.4. [39, pp. 13-14]

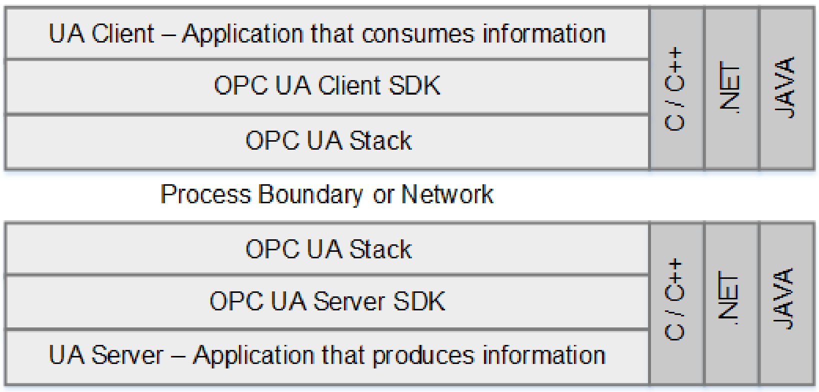

Figure 4.4. *Software layers of OPC UA, adapted from [39, p. 14].*

Generally, OPC UA application is a system that either exposes or consumes data via OPC UA. The software layers shown in Figure 4.4 can be implemented for example with C/C++, .NET or JAVA, which were first the only environments used for implementing the OPC Foundation UA Stack deliverables. The OPC UA Application uses OPC UA Stack and an OPC UA Software Development Kit (SDK) for mapping specific functionality to OPC UA. The UA Stacks implement the communication channels whereas the OPC UA Server or Client SDK implements the common OPC UA functionality part of the application layer. [39, p. 14]

4.3 Information modelling in OPC UA

In order to have a complete understanding of the significance of a simple measurement value, there may be a need for additional information about measurement time, engineering units and the measurement range along with information about the current state of the device. This can be done with information modelling which allows keeping the devices themselves simple while the complicated part is realized in the middleware. [43] Classic OPC has a limited method for modelling of data which needed to be enhanced to provide a common, object-oriented model for all OPC data. The model needs to include an extensible system to enable offering of meta information and description of complex systems. Although it is an important requirement to enhance the modelling capabilities, it is also important to enable simple models with simple concepts. Therefore the base model is kept simple and abstract but still extensible in order to enable models from simple to complex. OPC UA provides more powerful possibilities for exposing the more detailed semantics of the provided data. [39, p. 9, 19]

The basic idea of OPC UA enables the client to have an access to smallest pieces of data without understanding the entire model which is implemented with complex systems. The different layers of information models are illustrated in Figure 4.5. Only the infrastructure to model information is provided by the base OPC UA specifications. Examples of the provided base models are OPC UA for Devices (DI) and OPC UA for Analyzer Devices (ADI) which uses the OPC UA DI model as a basis. The OPC UA model defines for example concepts or the logical grouping of components, methods and parameters. Above the base specifications, information models for the domain of process information are defined. On top of them are specified information models of other organizations and additional vendor-specific information models. Vendors can use the base model and then extend it with specific information about their devices. [39, p. 11, 19; 44]

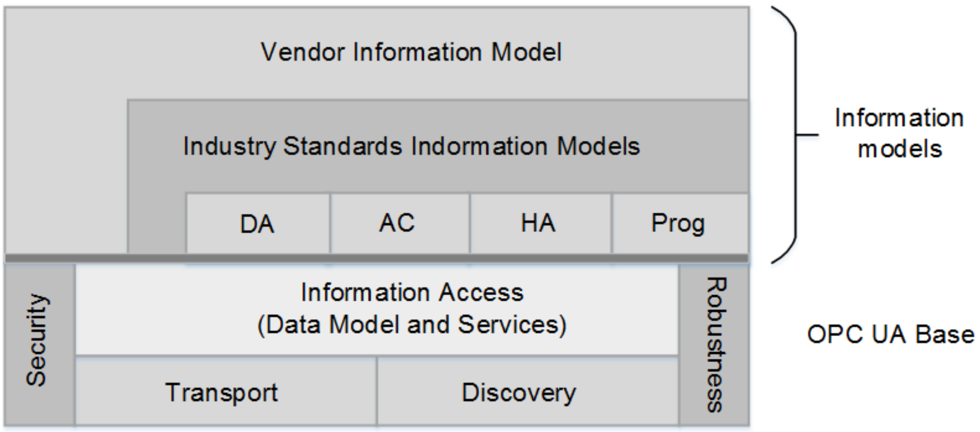

Figure 4.5. *Layered architecture of OPC UA, adapted from [40].*

There are some base principles of information modelling in OPC UA. OPC UA uses object-oriented techniques that include type hierarchies and inheritance so that the client can handle all instances of the same type in the same way. The type hierarchies enable the clients to work with base types and when necessary, to ignore more specialised information. OPC UA Server provides the type information which can be accessed with the same mechanisms that are used to access instances. The information models exist only on *Servers*. The information is possible to expose in various ways by providing different paths and ways of information organization in the full meshed network of nodes that are used to structure the *AddressSpace*. OPC UA allows supporting several hierarchies when exposing different semantics and references between nodes. In addition, OPC UA can be extended in different ways regarding the type hierarchies and reference types between nodes. [39, pp. 19-20, 30] In Figure 4.6, an example of OPC UA *AddressSpace* structured by *Nodes* is presented.

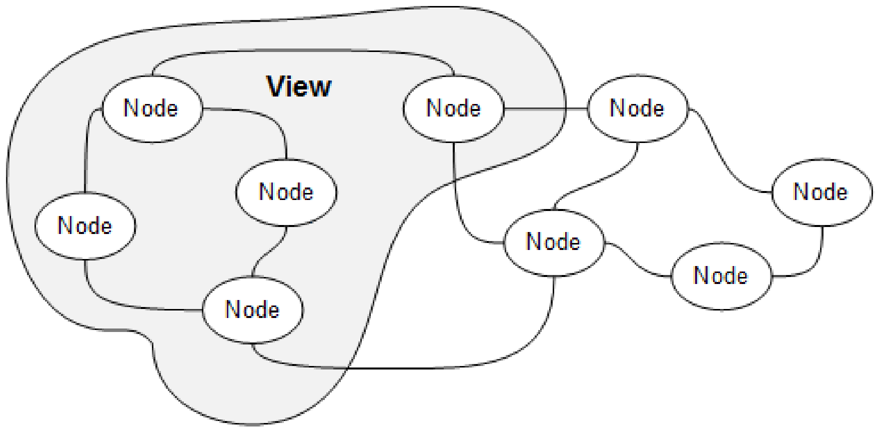

Figure 4.6. *OPC UA AddressSpace, adapted from [41].*

Real objects, such as physical devices or software objects, their definitions and their *References* to each other are represented in the *AddressSpace* as a set of *Nodes*. *Nodes* can be organized as the *Server* decides because *References* allow the hierarchical structure as well as a full mesh network or any possible organization. Often *Clients* are interested in only a specific subset of the available data and their burden is therefore tried to keep lighter by the *View* standard. A subset of *Nodes* and thus *AddressSpace* is defined by a *View*. Instances of object and variable types model real objects in the *AddressSpace*. [41; 45] Figure 4.7 provides an example of an *Instance* with a type definition. The graphical representation in OPC UA is also presented.

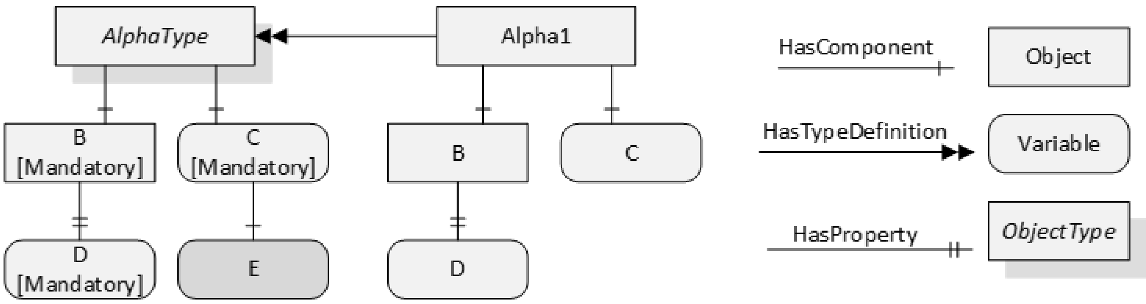

Figure 4.7. *Modelling in AddressSpace, adapted from [45].*

When a new *Instance* of a *TypeDefinitionNode* is created, OPC UA Server creates the same hierarchy of *Nodes* beneath the new *Object* or *Variable*. *Nodes* that are referenced by the *TypeDefinitionNode* without a *ModellingRule* do not appear in the instance. The information of *TypeDefinitionNodes* can be utilized by a *Client* for accessing *Nodes* that are in the instance hierarchy. [45]

4.4 User interface design

The main task of a user interface is to enable communicating information from the machine to the user as well as from the user to the machine. In any industrial control system, information can be delivered from machine to people and this allows people to control, monitor and record the system. Although the provided functionality and information are important, it is equally important to consider the way in which they are given. [38, p. 351, 368] A good graphical user interface (GUI) provides situation awareness which consists of three levels. First level is that the needed data is provided and presented. However, raw data does not necessarily provide the user with relevant information in order to understand how the system is actually operating, which is the second level of situation awareness. The third level is that the user is capable to predict what will happen in future and understands what the meaning of the information is in future. [46]

A great amount of data is available for the users nowadays, and the way to present the data needs to be defined so that the information can be effectively processed. There are decisions that need to be made in order to enable an effective human-machine interaction. When designing a GUI, the essential information and presentation way should be defined. In addition, how the information is organized and emphasized are issues that should be considered. [46] There are various principles that determine how the quality of GUI design can be improved. The structure principle means that the interface is organized in a meaningful and useful way. The things that are related are put together and unrelated things separated, so that there is a diversification between dissimilar things. In addition, the reuse principle defines that the need for users rethinking can be reduced by reusing internal and external components and behaviours. According to the simplicity principle, GUI design should allow clear communication and tasks to be done easily. The visibility principle defines that all information needed is visible. The confusion due to redundant and unneeded information should be avoided. The feedback principle outlines that GUI design should inform clearly about relevant information in which user is interested, such as changes of condition or state and errors. Additionally, the tolerance principle defines that GUI design should be tolerant and flexible so that undesired actions are not allowed and errors are prevented. [38, p. 369] There are also some main guidelines for graphical design. Presentation of raw data as numbers should be avoided and present values as information, such as trends and other graphic presentations. The presented information should be necessary. Use of flashing and spinning graphics is to be avoided along with bright and inconsistent colours. An effective user interface usually has a grey background and very limited use of colour which is generally used for alarming. [46]

5. DEVELOPMENT

The aim of this thesis is to provide a tool that is suitable for the performance assessment of NAPCON Controller in various application environments. The performance assessment tool is meant to be a new feature included in the existing controller software environment. The usability of the tool is required to be simple and the provided information should be easily assimilated. The deployment of the performance assessment tool is required to be as automatic and generic as possible. Additionally, the operational data that is needed for performance calculation is received automatically from the database.

The subject of the thesis has earlier been studied by Oksanen [12]. Although the application that was designed worked mainly as planned, there were a couple of problems in the work that should be solved. In addition, a tentative outline for the presentation of the controller performance was designed but a concrete realization of the user interface has not been carried out earlier. First in this chapter, the current state of the performance assessment tool is presented and the required development aspects are introduced. After that, the selected performance assessment techniques and their advantages and disadvantages compared to other methods are shortly discussed.

5.1 Current state

As a result of the master's thesis of Oksanen, an application for computing the technological and economic performance of the constrained, multivariable MPC controller was produced. Methods based on historical and design-case benchmarks were used for assessing the technological performance of the controller. The economic performance was calculated by utilizing the historical benchmarking method. The functionality of the application was tested with various simulation cases.

The performance assessment application and the required calculations were programmed and built in the NAPCON calculation frame with C#. The main module of the application contained the performance calculation class and it was the core element of the entire application. Necessary calculations for main function of performance calculation were computed by supportive modules. The definition of the performance calculation application was executed with a text file defining the name of the controller, lists of the names of controlled and manipulated variables, and manipulated variable constraints.

The functionality and the accuracy of the performance assessment application were accomplished by different simulation cases. Simulations were carried out with ProsDS simulator, proprietary of Neste Jacobs. Each simulation case used benchmark values that were achieved during a steady state during which the control objectives and the process conditions were unchanged. The simulation cases included set point changes, unknown process disturbances, tuning parameter changes, constraint violations and a model-plant mismatch. The simulations showed that the application worked mostly as designed and it provided valuable information about the prevailing conditions of the controller. In addition, it was generally demonstrated that the economic performance is usually at an acceptable level if the technological performance is high.

5.2 Selected performance assessment techniques

There are different methods that can be used for assessing the performance of the multi-variable MPC controller. Earlier in 2012 in his work, Oksanen decided on using histori-cal and design-case benchmarks to achieve performance indices for the technological and economic performance of the controller. The combination of these methods con-tained information of controller performance when history, present and future of the controller is considered. Also in this thesis, the historical and design-case techniques are utilized in the performance calculation application. They have earlier been studied with promising results and the methods are further studied with minor changes in this work.

In practice, it is common to use a benchmark value extracted from historical data during time period when the controller is determined to perform well. The historical bench-mark technique may however be too subjective and the controller may seem to be per-forming well as the value of the benchmark relies on the determination done by experts. [25] As earlier mentioned, the historical benchmarking method was applied to the per-formance assessment calculations in the work of Oksanen. The value of the historical benchmark value was continuously updated, which induced some problems in the per-formance assessment of the controller. Problems appeared in the case where the un-known disturbance caused an increase in the controller performance and therefore bias-ing in the benchmark value. However, the historical benchmark is determined to remain constant in the literature. When the benchmark is constantly updated, it is ensured that the current performance is always compared to the best achievable value and the benchmark value does not rely on the performance during the time period decided by the professionals. Nevertheless, when a controller is brought into use in a new process, a proper examination of the controller behaviour is anyway required and the benchmark could be determined on the side. The historical performance index may be subjective and the results depend on the process and the controller, and therefore the results are not directly comparable with other controllers. However, it can be considered reasonable to compare the current performance to a value that is achievable with the assessed control-ler.

The continuous update of the historical benchmark value adapted the benchmark to be suitable for controller configuration changes automatically. Now that the historical benchmark is once defined, the adequacy of the benchmark value can be considered if the controller configuration changes. In this work, the historical benchmark is decided to be updated automatically to a new value based on the individual variable benchmark values if at least one of the variable states changes. The individual variable benchmark values are collected during the time the historical benchmark search is on. As the benchmark is defined as a sum including the performance of the different variables, it is updated in a relation to the individual benchmark values. For example if one of the controlled variables is left out of the control, its individual benchmark value is subtracted from the total sum. However, the configuration alteration usually changes the control objectives and the updated historical benchmark may not provide the best possible information of the controller performance.

In order to provide a performance index that is more reliant on the controller definitions, the design-case performance index is computed alongside the historical performance index. The design-case benchmarking method evaluates the controller performance by comparing the achieved behaviour to the actual design objectives. The core element of a model predictive controller is naturally the process model. Thus, it is reasonable to utilize a model-based approach of controller performance assessment. Design-case performance index is suitable for indicating if something causes the process to behave not how the controller assumes. The design-case benchmarking method is slightly changed in this thesis when compared to the earlier work of Oksanen. Earlier, the benchmark value was compared to a value achieved by variance estimation over a future horizon and not by the value given by equation (19). Using this kind of approach provided an estimation of the future of controller's performance. In this work, the design-case benchmark method is applied as it is in literature proposed and its suitability is studied.

Control performance assessment techniques are continuously developing and new methods are proposed. In Section 3.2.2, recently proposed CPA methods are introduced generally. However, the introduced methods are not applied in this thesis. Many of the new methods are based on the main performance assessment techniques and therefore the modifications have mainly the same features as the basis methods. As stated earlier, the application of methods based on minimum variance benchmarking in multivariable processes is not simple since they require the entire interactor matrix. Typically, it can be concluded that the calculation of more sophisticated and realistic benchmarks require more prior knowledge and data and they require more computational burden [25]. The historical and design-case performance indices provide versatile information of the controller performance when utilized together. Both indices are based on the objective function which consists partly of control errors and partly of control moves. Basically, they describe how well the process variables stay at the target values and how much control effort is required to achieve that performance. Neither of the methods requires much additional calculations and the computational burden of the performance assessment program remains reasonable. The historical benchmarking technique is also applicable to the economic performance assessment. When assessing the economic performance of the controller, equation (7) can be applied to equation (14) and an economic performance index is achieved as a result.

6. IMPLEMENTATION

When implementing a new application, the software environment, to which the new software is applied, needs to be taken into account. The role of the new software needs to be defined along with the interconnection with other components. The main goal of this work was to produce a reliable and informative tool that is suitable for assessing the performance of a multivariate MPC controller. In addition, it was required that the operation of the performance assessment tool does not affect the actual controller functionality. The new tool was also required to be both adequate for the current database structure and easily applicable to various controller environments.

The software environment, to which the performance assessment tool was applied, is first introduced in this chapter. After that, the new information model object types are presented. Two new object types were defined with OPC UA information modelling in order to provide a new structure for the OPC UA address space and thus allow the new calculated performance variables to be stored to the database. Lastly, the software implementation of the performance assessment tool is presented, including the performance calculation application and the graphical user interface.

6.1 Software environment

The performance assessment tool was applied to a software environment which includes various components. The general structure of the software environment related to the new performance assessment tool is presented in Figure 6.1.

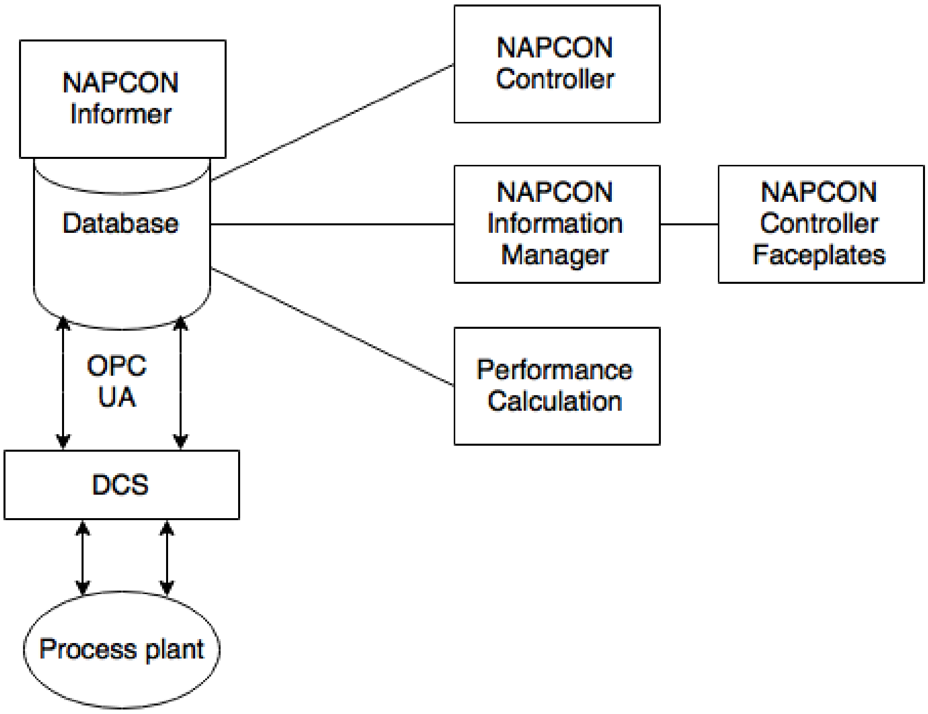

Figure 6.1. *Software environment structure, adapted from [47].*

The NAPCON Suite software package includes several integrated modules that operate together, including NAPCON Controller and NAPCON Informer. NAPCON Controller is a multivariable, optimizing, multistep, model predictive controller with embedded constraint handling. The controller software is developed and designed by Neste Jacobs Oy. It normally operates on top of the basic regulatory control in the automation hierarchy. The controller can be used in Windows environment and there is a product family that supports the use of the controller software, including NAPCON Informer, a real-time database and history database, and OPC UA interface to connect the controller with external systems, such as automation systems (DCS). The controller reads the process measurements, control objectives and online tuning parameters from the database. After calculating the optimal manipulated variable values and predictions of controlled and predicted variables, controller writes the results to database. [47] Figure 6.2 illustrates the operation principle of NAPCON Controller. Besides different variable types, other significant aspects of the controller are represented.

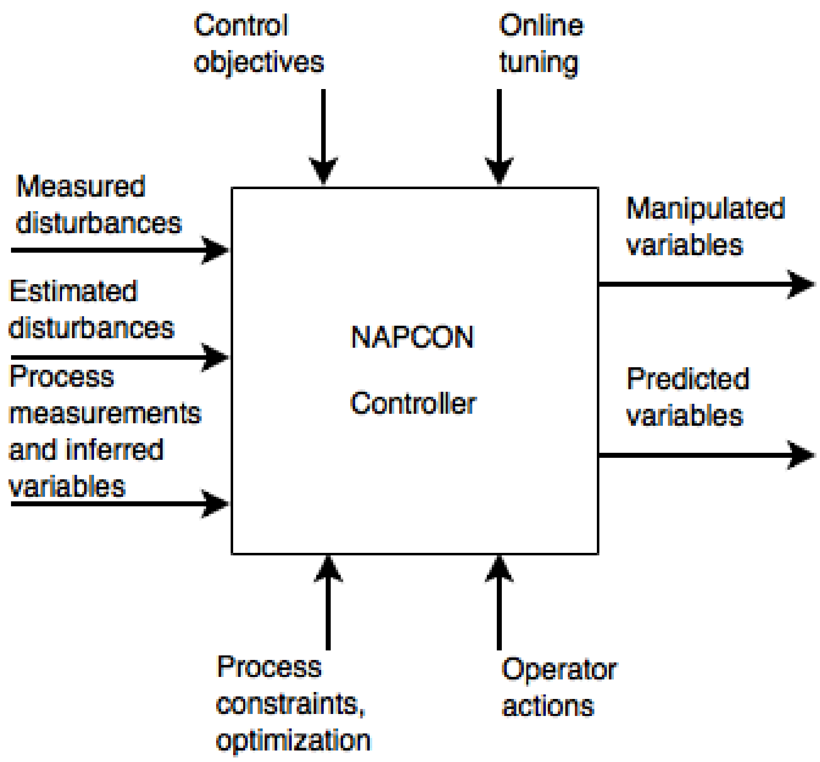

Figure 6.2. *Logical structure of NAPCON Controller, adapted from [47].*

The logical structure of the controller includes control objectives, variables, various tuning parameters and different operator actions. Four different types of variables are used: manipulated (MV), controlled (CV), predicted (PV) and disturbance variables (DV). The nature of the control problem depends on the degrees of freedom. The controller configures itself automatically if the numbers of CVs or MVs and thus the control problem characteristics change. The controller can be defined to have various objectives, such as relative importance between different variables, which enables the controller to perform adaptably in different control situations. [47] The priorities between the variables define the relative importance if all the control objectives cannot be satisfied. The MV constraints (MVC) have always higher priority than CVs and thus they are the first to be satisfied. All target CVs are controlled to their defined set points and constraint CVs between the defined limits if the degrees of freedom allow this. [48]

The engineering interface of the controller is provided by NAPCON Information Manager which enables operations including online tuning, model updating and controller reconfiguration, for instance. It also allows viewing the contents of the real-time process database and history database. [47] NAPCON Controller Faceplates are a part of the engineering interface of the controller and they give such a view to database variables that is based on the functionality of the controller rather than on the actual database structure. Their main purpose is to make the monitoring, tuning and troubleshooting of multivariable controller applications easier.

6.2 Created object types

To provide a generic way for the performance calculation of every NAPCON Controller, OPC UA address space structure needed to be extended and two new object types were defined in this thesis. A new performance calculation child object (*NAPCON_PerfCalc*) was defined for the object type of NAPCON Controller. The children of *NAPCON_PerfCalc* include properties which the performance calculation application computes and utilizes in these computations, along with the economic variable folders. The folders contain the economic variable objects (*NAPCON_Eco*) that define the economic inputs and outputs of the process unit. Figure 6.3 illustrates the created object types of *NAPCON_PerfCalc* and *NAPCON_Eco*.

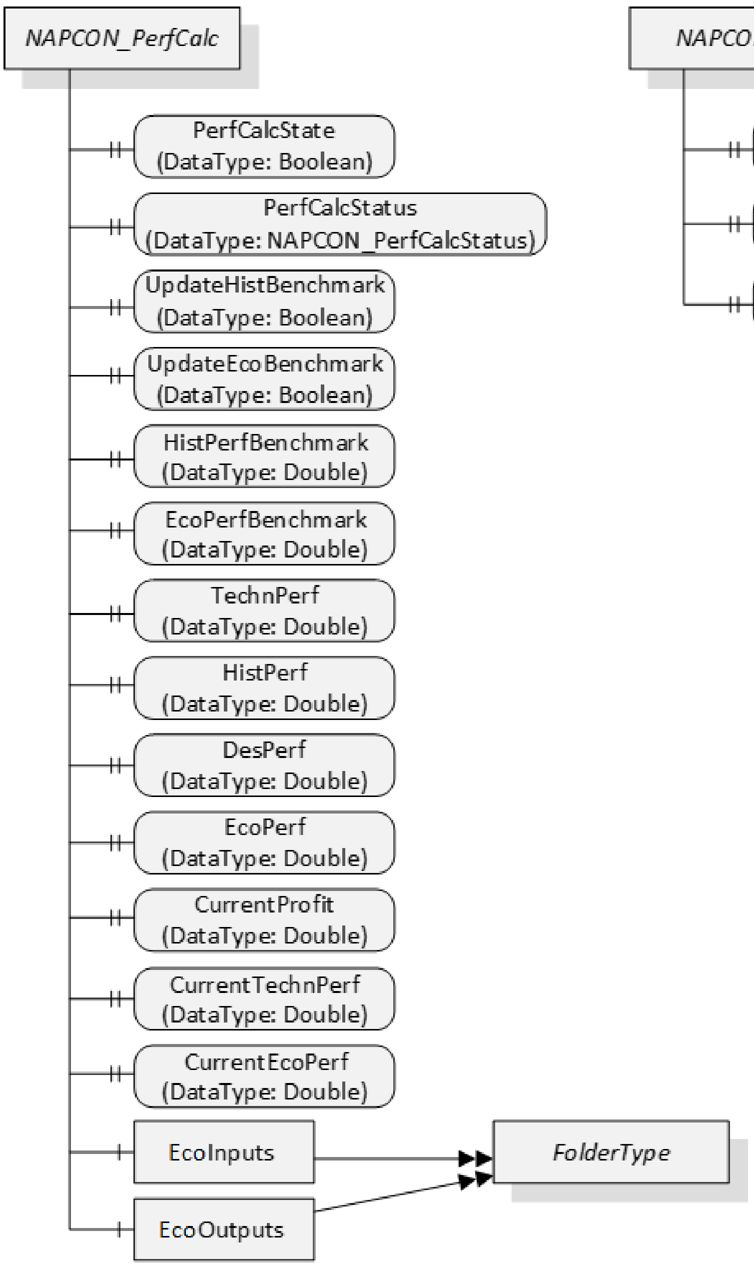

Figure 6.3. *Created object types of NAPCON_PerfCalc and NAPCON_Eco.*

The new *NAPCON_PerfCalc* object type organizes different variables as its children. Most of the values of the variables are calculated in the performance calculation program and derived from there to the OPC UA Server. *PerfCalcState* Boolean variable determines the switch for performance calculation whereas *PerfCalcStatus* enumerable variable indicates the status of the performance calculation. The status is defined to be *OK*, *Initializing*, *Incomplete* or *Invalid*. Switches for benchmark value searches are defined with Boolean variables *UpdateHistBenchmark* and *UpdateEcoBenchmark*. Double variables, including performance indices, momentary values of performance indices, current profit and benchmark values, are computed by the performance calculation program. The folders *EcoInputs* and *EcoOutputs* contain the economic variables that define the economic inputs and outputs of the process unit, such as steam usage, feed flow and output products.

NAPCON_Eco object type has child variables that are used for defining the type, price and amount of the real process object. *Type* is an enumerable variable that indicates whether the object is an economic input or output of the process unit. *Price* variable defines the actual price of the economic object or it can alternatively be used as a relative factor if prices of the variables are not defined or available. *Meas* variable is defined to have a non-hierarchical *HasInput* reference pointing to an actual DCS measurement that indicates the amount of an economic object, such as feed's flow rate.

6.3 Performance assessment tool

The software related to the performance assessment tool was implemented with C# in .NET framework. The performance calculation application was implemented as a Windows service which runs automatically in the background after installation and starting from the Windows Services. Functionality and structure of the performance calculation application as well as the user interface were designed to be generic so that the performance assessment tool could be deployed as automatically as possible to a new control environment. The structure of the implemented performance calculation program was also designed so that the calculation is executed in small, logical functions, which would allow applying additional performance assessment techniques as part of the calculation program subsequently.

The presentation of controller performance was implemented in NAPCON Controller Faceplates, part of NAPCON Information Manager. A new faceplate type was determined for the *NAPCON_PerfCalc* object type to operate as the graphical user interface of the performance calculation program.

6.3.1 Performance calculation application

The program that carries out the calculation required for performance assessment was implemented as a Windows service. The program solution includes a service class that creates a new instance of performance calculation handler class (*PerfCalcHandler*) when the service is started. The performance calculation handler checks the connection configuration and creates the connection to UA server based on the defined server's URI in the application's configuration file. OPC UA supports transferring data over network, which enables that the UA server is not necessarily on the same computer as the service. As a result of the query to the server, the handler receives nodes whose type matches the controller type. A new instance of performance calculation class is constructed per each controller node so that there are as many performance calculation (*PerfCalc*) instances as controllers defined on the UA server. The performance calculation service is required to be restarted if the number of the controllers on the server or the structure of the database changes. The functionality of the program's start is illustrated in Figure 6.4.

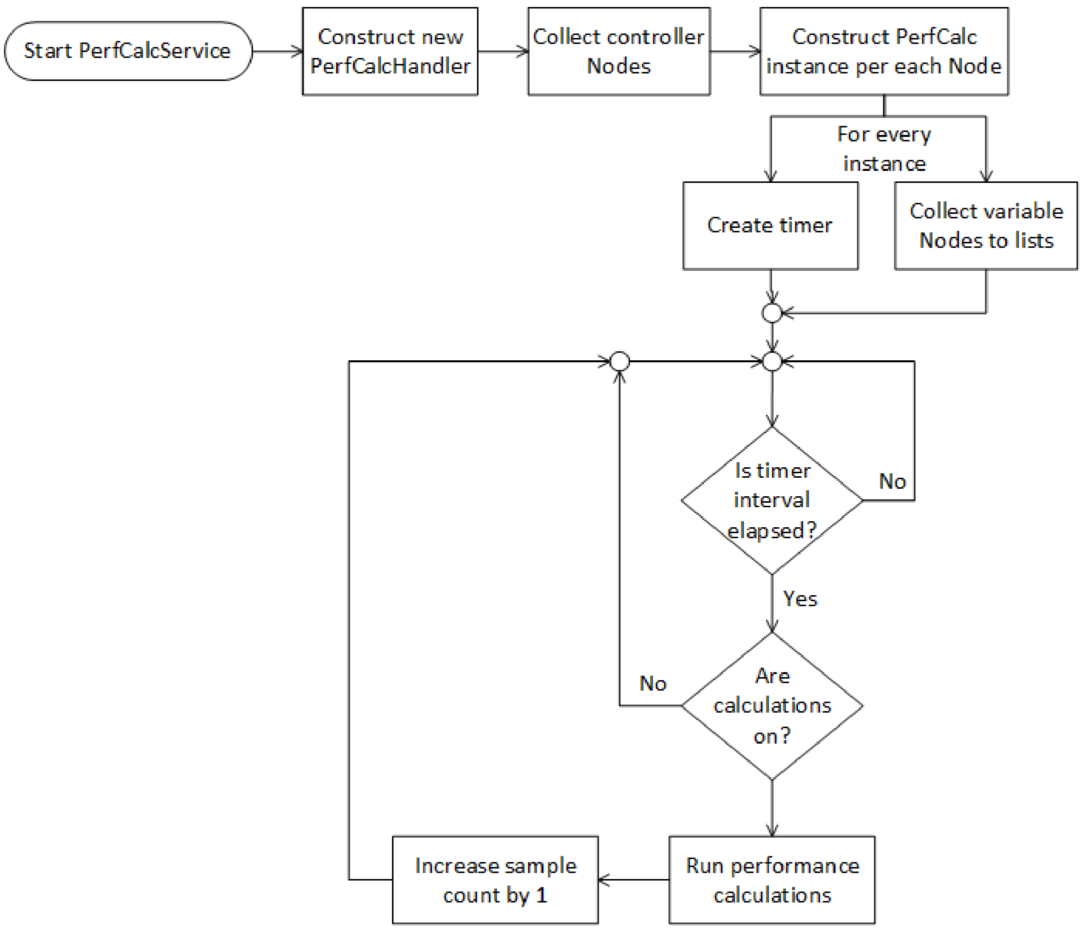

Figure 6.4. *Start of the performance calculation application.*

Variable nodes and related UA variables are resolved from the server when a new *PerfCalc* instance is constructed. The UA variables are connected to *PerfCalc* properties so that when the value of the property is changed, the UA variable in the database is also updated. In addition, child CV, MV, MVC and economic variable nodes of the current main controller node are collected to separate lists and related objects are constructed when the instance is initialized.

After constructing the *PerfCalc* instances, the performance calculation handler calls a method of *PerfCalc* that creates a timer and starts the performance calculation. Using separate timers for each *PerfCalc* instance allows the execution of performance calculation of separate controllers in parallel if there is more than one controller on the server. The timer calls the *PerformanceCalculation* function each time the timer interval has elapsed. The timer interval is set equal to controller's control cycle. The performance calculation is executed if the controller's control calculations and the performance calculation are on. The sampling instants are counted and each time the performance calculation is carried out, the sample count is increased by one. If all the calculation states are not true, the performance calculation is not performed and the states are checked again the next time the timer interval has elapsed. The timer along with the performance calculation is stopped when the service is stopped.

The general performance calculation functionality is illustrated in Figure 6.5. Each time the performance calculation is carried out, the current values of weighted and scaled control errors and control moves of CVs and MVs are computed for each variable. Additionally, lists containing the history of these values are updated. These lists are utilized for computing the historical benchmark value and the achieved values for historical and design-case indices. For constraint CVs, the control error is handled as a limit, and it is defined to be zero if the limit is not violated. Additionally, the sums of predicted values for control errors and control moves over prediction and control horizons are computed. The lists containing the predicted sum values are utilized for calculation of the design-case benchmark value. The current value of each economic variable is also calculated based on the price and the measurement. After that, the current profit of the process unit can be computed based on the input and output variable values, and the history list holding the profit values is updated. In addition, the list containing CV and MV states is updated for comparing the current states and the states of the previous calculation instant. The comparison is performed in order to register if there has been a change in the controller configuration.

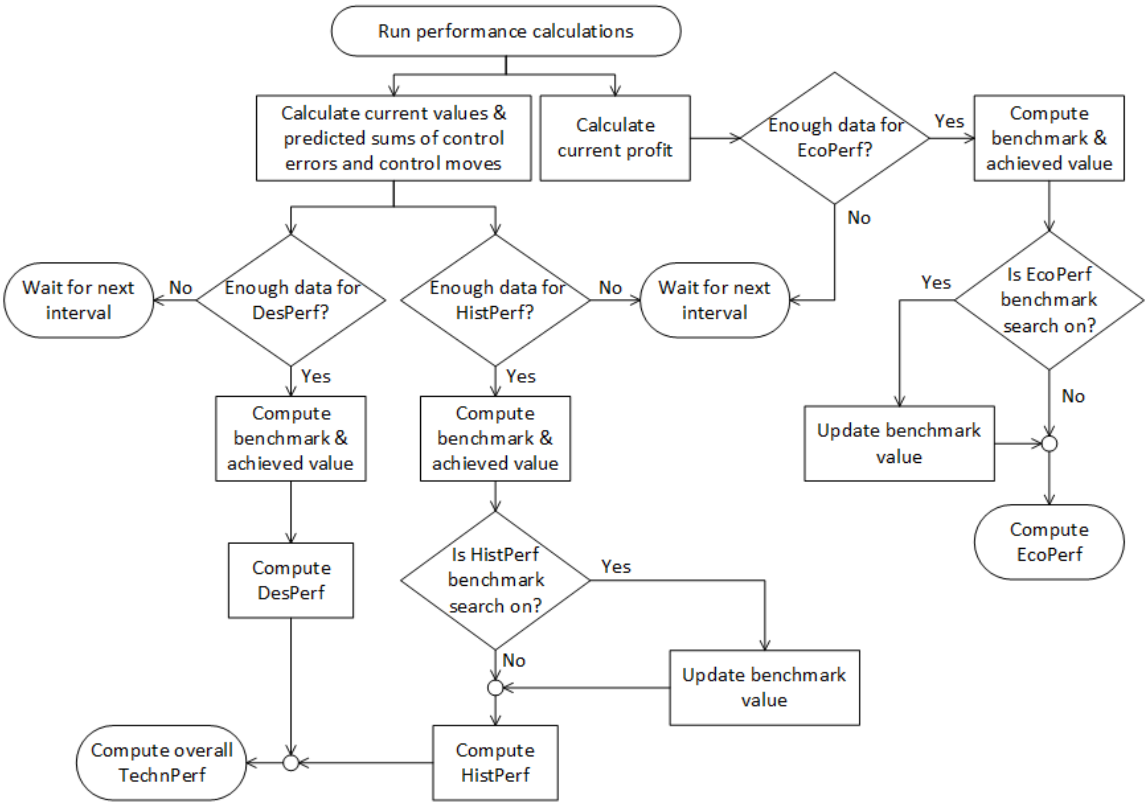

Figure 6.5. *General performance calculation functionality.*

Technological performance index is achieved by computing the historical and design-case indices separately and then using the average value of these two indices for assessing the overall technological performance of the controller. Both indices are limited so that they do not exceed the value of 1, and therefore also the overall technological index remains in the range of 0 to 1. If either of the indices would be allowed to exceed the value of 1, it would result as a misleading overall technological performance index. It is possible that the historical and design-case indices require different number of sampling instants before they can be computed normally. Technological performance index is set equal to the individual performance index that is first achieved until both of the indices can be calculated. As presented in equation (8), a performance index is composed of a benchmark value and an achieved value. Both values are computed as an average value over defined horizons, which in this work depend on the controller dynamics and parameters so that the performance indices of different controllers could be as comparable and generic as possible. The achieved value for historical and economic performance indices is computed over a moving horizon with a length of half of the control horizon. It has been earlier discovered that the horizon for historical performance benchmark value should be twice the horizon over which the achieved value is computed [12]. In this work, the historical benchmark is defined to require sampling instants the same number as the length of the control horizon is, whereas the design-case benchmark depends on the lengths of prediction and control horizons as in equations (18) and (19) is stated. The economic performance index is computed in a similar way as the historical performance index and the horizons of the economic benchmark and achieved values are equal as in the historical benchmark and achieved values. Besides the average values over the defined horizons, momentary values are calculated both for technological and economic indices. The achieved historical and economic values at current sampling instant are compared to the benchmarks and the momentary performance indices are achieved.

In the performance calculation program, the benchmark values for historical and economic performance indices are not updated automatically to the best achievable value. These benchmark values are updated during the time period, when the process unit is performing well and the controller performance is at desired level according to the requirements defined by the user. There are switches for benchmark searches that are switched on for the update of the benchmark values. When the benchmark values are suitable according to the professional experience, the switches are turned back off. Individual benchmark values for each CV and MV are also collected when the historical benchmark search is on. They are utilized when at least one of the variable states changes and thus the controller configuration changes. The historical benchmark value is redefined with relation to the variables included in the current controller configuration.

The performance assessment tool includes a status variable that indicates the current status of the performance calculation program. The status is defined to *Initializing* when a *PerfCalc* instance is constructed after the service starting but there is not enough data to perform the actual performance calculations. If some performance index can be computed but there is not enough sampling instants for all indices, the status is set to *Incomplete*. When the program has run long enough to compute all the performance indices normally, the status is set to *OK*. If the calculation cannot be computed normally for some reason, such as a connection problem or situations where the main controller calculation state is false or the performance calculation service is stopped, the status is set to *Invalid*.

6.3.2 Graphical user interface

An outline for the graphical user interface of the performance assessment tool had been designed earlier. However, nothing concrete had been realized. In this thesis, the former outline was revised notably so that the user interface is suitable to be a part of NAPCON Controller Faceplates in the existing software environment.

The implemented graphical user interface of the performance assessment tool is connected to the database. Variable and parameter values are read from database and the user is also able to write new values to the database through the user interface. The GUI includes the indication of the performance indices as well as the performance calculation status. The user can determine with a switch whether the performance calculation is on or off. The user is also able to define with switches the time period when the benchmark values of the historical and economic performance indices are updated. The change of parameter values, which define the prices of the economic variables, is possible through the user interface. The performance calculation GUI consists of two tabs which each share a header section including the object name and the state and status of the performance calculation. In Figure 6.6, the first tab of the performance calculation faceplate is presented.

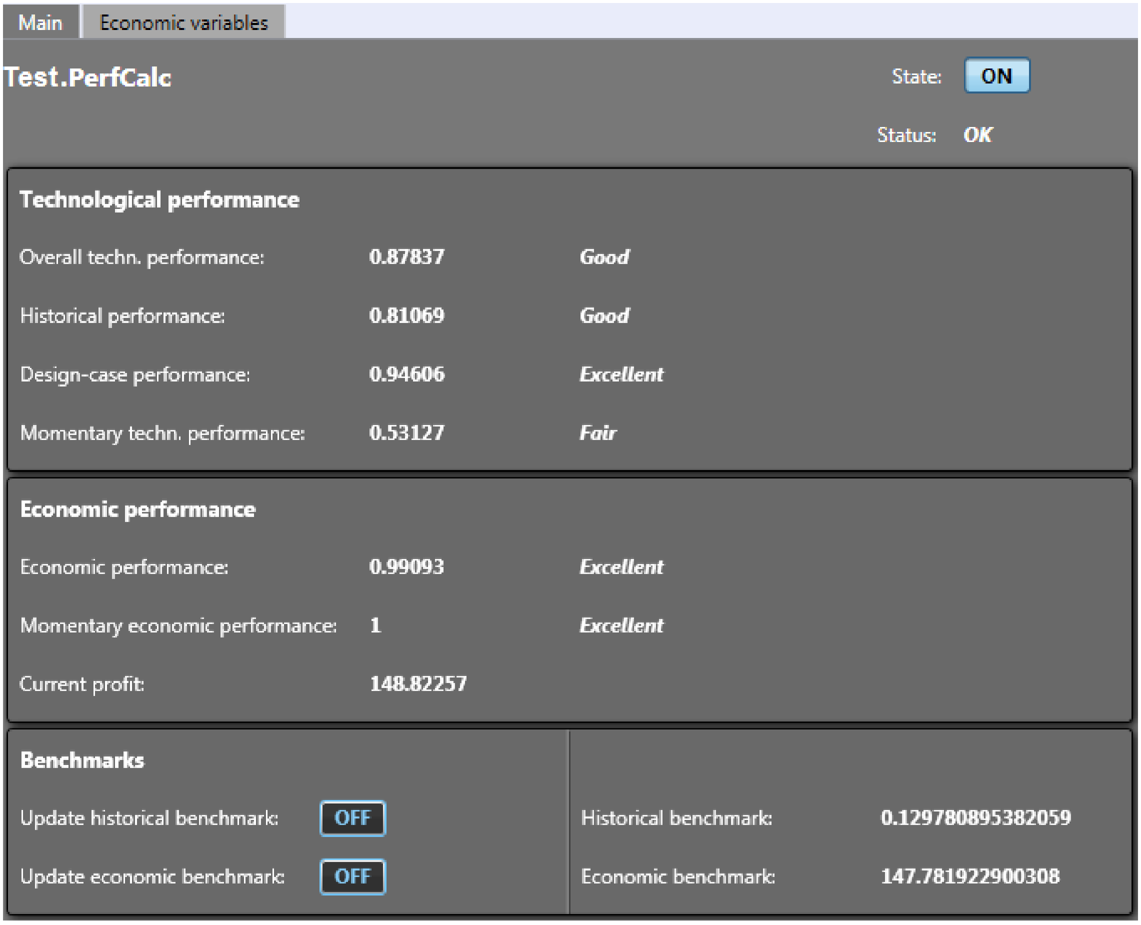

Figure 6.6. *The main tab of the performance calculation user interface.*

At the main tab, the different performance index values are indicated besides the benchmark switches and the current benchmark values. Variable values are grouped based on their relations. The level of the performance index is indicated verbally next to the index value. The different index levels are presented in Table 1.

Table 1. *Performance index levels.*

Value	Level
> 0.9	Excellent
> 0.6	Good
> 0.3	Fair
≤ 0.3	Poor

The second tab of the performance calculation user interface includes lists of economic variables, separated to economic inputs and outputs. The tab allows the user to see the economic variables of the unit at one glance. The outline of the tab is presented in Figure 6.7.

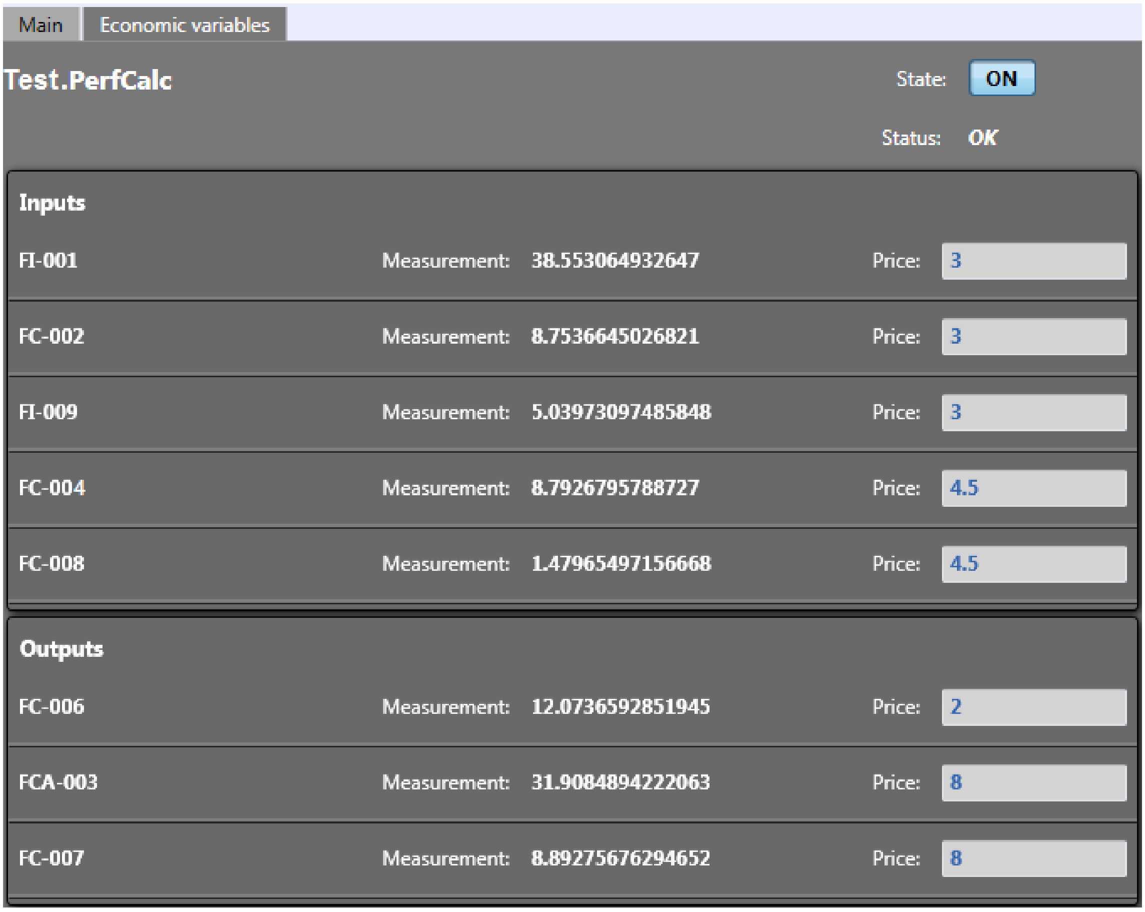

Figure 6.7. *The second tab of the performance calculation user interface.*

Variable names, current measurement values and price coefficients are presented. A more detailed description of the variable is available as a tooltip when the user holds the cursor on the variable name.

7. TESTING

After the implementation, it is necessary to test the program and verify that it operates properly. Testing of the performance assessment tool can be done for example with real or simulated process data. In this thesis, the testing of the performance assessment tool was carried out in a simulation environment by assessing the performance of an actual MPC controller. The operation of the tool was examined with simulated process data to verify that the performance calculation program and the graphical user interface work as required.

First in this chapter, the test process is introduced. The process unit, with which the performance assessment tool was tested in this thesis, was a distillation unit included in the gasoline production of an actual oil refinery. After that, the test arrangements including various simulation cases are described. Finally, the test results achieved from simulations are presented.

7.1 Description of test process

The test process unit of this thesis includes two distillation columns: a main column and a side stripper column. The main aim of the unit is to remove lighter hydrocarbon from the feed as a distillate so that the bottom product's flow rate is maximized while meeting the bottom product's quality requirements and minimizing the use of reboiler steam and the distillate flow rates. The feed of the unit comes from various units that are earlier in the production line. Due to various feed sources, there are a lot of feed fluctuations in the unit. [48] A piping and instrumentation diagram of the process unit is presented in Figure 7.1. Only the components included in the main APC strategy of the unit are presented in the diagram.

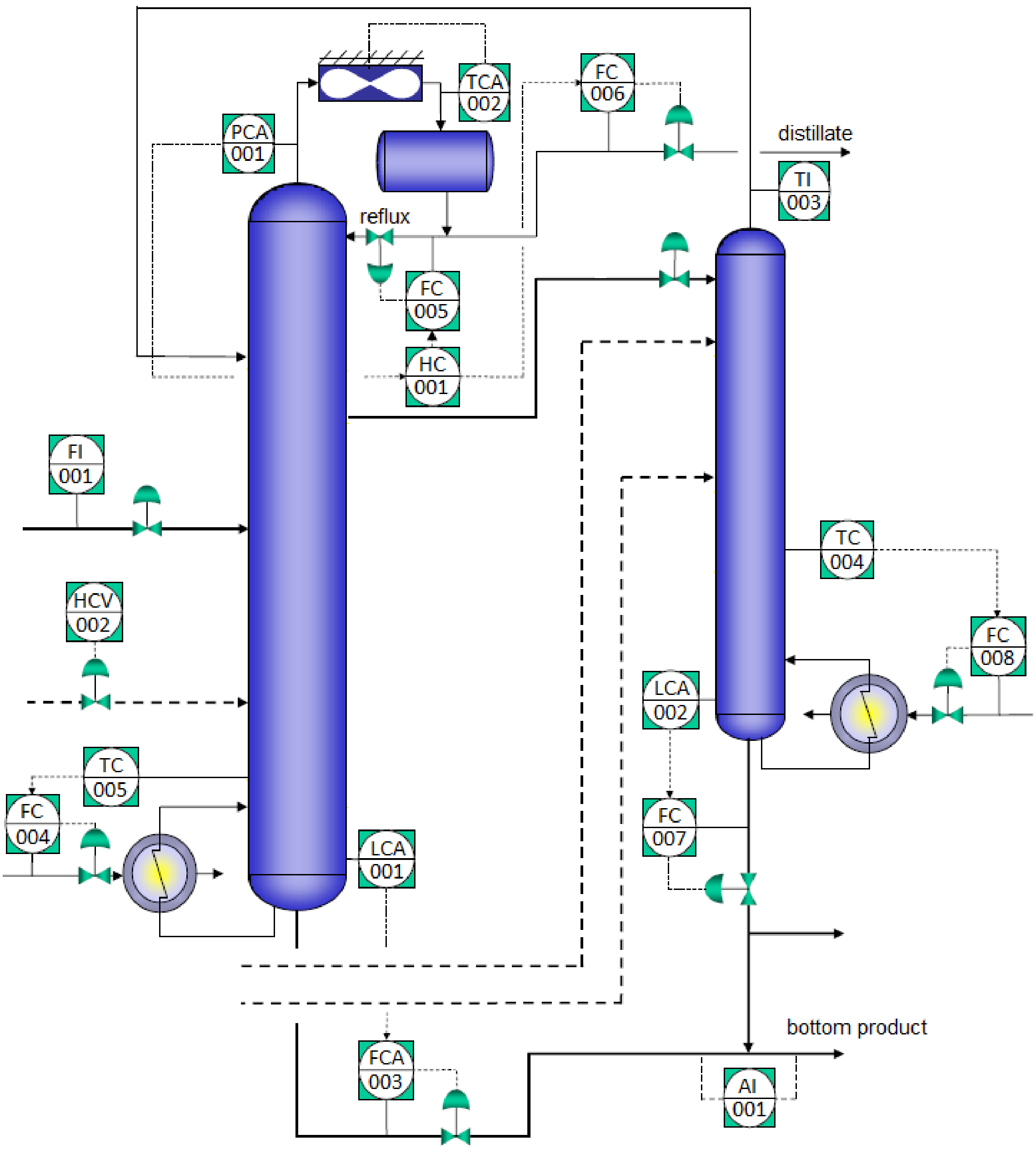

Figure 7.1. *The test process, adapted from [48].*

The aim of the APC is to stabilize the unit operations sufficiently so that the process can be operated closer to the limits. The APC strategy of the unit consists of upper and lower level controls. The controller aims to keep the separation of the main column stable and at the right level based on the analyser measurement and the temperature measurement correlating the main column's dissolution potential. The APC also aims to keep the composition of the bottom product near to the target value and to minimize the overhead flow rate and reboiler steam usage. In addition, the controller attempts to keep the overhead condenser operation, unit's pressure control and operation of columns in good operating ranges. [48] The controlled and manipulated variables (CVs and MVs) of the unit are presented in Tables 2 and 3.

Table 2. Controlled variables.

Tag	Unit	Type	Description
AI-001	vol-%	Target	Composition of bottom product
TC-005	°C	Target	Temperature of main column
TC-004	°C	Target	Temperature of side stripper column
CX_CALC	w-%	Maximum	Calculated composition of distillate
TCA-002VP	rpm	Maximum	Condenser's speed of rotation
FC-005	t/h	Maximum	Reflux flow
TI-003	°C	Minimum	Overhead temperature of stripper column

Table 3. Manipulated variables.

Tag	Unit	Description
HC-001	%	Reflux ratio of main column
FC-004	t/h	Reboiler steam of main column
FC-008	t/h	Reboiler steam of side stripper column
PCA-001	kPa	Unit pressure
TCA-002	°C	Condensation temperature of overhead flow
HCV-002	%	Side feed valve's position
T-005_T	°C	Temperature of main column (cascade)

Besides controlled and manipulated variables, some disturbance variables of the unit have been determined for the control at the lower level of the controller functionality. The predicted impact of DVs can be compensated prospectively by MVs. The flow rate and temperature of the main feed along with the main column's side feed are handled as disturbance variables. In addition, the flow rate and temperature of the side stripper column's feed and the overhead temperature of the stripper column are determined as disturbance variables. [48]

7.2 Test arrangements

The test environment of this work was a Windows virtual machine which included a simplified and dynamic simulator model of a process unit used in an earlier control related project, and an actual MPC controller. The operation of the implemented performance assessment tool, including the performance calculation program and the graphical user interface, was tested with various simulation cases in order to verify that the application works as required.

The test simulator includes models between the CVs and MVs or DVs which have earlier been determined. In this thesis, some fluctuation was added to the feed stream so that the simulations would model real process more suitably. Standard deviation was added to the process input variables. Standard deviation of the main feed was 0.2 t/h as the value for extra feed was 0.1 t/h. In addition, temperature of main feed had a standard deviation of 0.05 °C. The economic variables of the unit were defined and new *NAPCON_Eco* instances were created in the OPC UA address space. The values for measurement variables were obtained with OPC UA references to APC related variables or measurements in DCS. The economic inputs of the unit included the main feed flow rate FI-001 and extra feed flow rates FI-009 and FC-002. In addition, the reboiler steam flow rates of both columns, FC-004 and FC-008, were defined as economic inputs. The economic outputs of the unit included the distillate flow rate FC-006 and the bottom product flow rates of both columns, FCA-003 and FC-007. The price coefficients of the economic variables were determined to be in a sensible relation to each other, but they did not present actual prices in this thesis and thus the computed profit and benchmark values were not presented in any currency.

The purpose of the simulations was to study how the controller's performance reacts in different cases. The first case to be simulated was a steady state case where all the variables were included in the control calculations and the targets and limits of the CVs were left unchanged. The steady state case allowed examining how the controller performance indices change in time when the control objectives were kept the same. The expectation was that the indices would change slightly due to the fluctuations on the process unit's inputs.

The next simulated cases were changes in the set point of a target CV and in the limit of a constraint CV. The aim was to examine how the values of performance indices and economic profit react when the control objectives are changed. The set point of the composition of the bottom product was first decreased by 10 %. After the process response had settled, the set point was then increased so that it was 5 % greater than the original set point value. Secondly, a case with a limit change of a constraint CV was simulated separately. This was carried out by decreasing the maximum value of the condenser's speed of rotation by 10 %.

The last simulation case included an unknown disturbance which in this content means that the controller does not have an internal model of the disturbance variable in question. Thus, the controller is not able to include the effect of the disturbance variable in the control computations. The simulation was carried out by switching off a disturbance variable in the process unit's controller and then manipulating the DV. The main column's feed flow rate was selected to be the disturbance variable to be switched off. After the DV was switched off, a decrease of 10 % was done in the level of the main feed flow rate. After the process had stabilized, the flow rate was increased to a level 5 % higher than the initial feed flow rate. The aim of this simulation case was to study how an unknown disturbance affects the performance indices and whether it is possible to detect presence of the disturbance with the performance application.

7.3 Results

The purpose of the simulations was to find out how well the MPC controller is working according to the implemented performance assessment tool. Different simulation cases were used to examine the reliability and accuracy of the calculated performance indices. All of the simulation cases were carried out so that the controller parameters were unchanged and thus the controller tuning remained constant. The control horizon of the examined controller was 200 cycles and so the benchmark values for historical and economic performance indices were computed as an average over a defined 200 cycle time period whereas the achieved values were obtained as an average over moving horizons of 100 cycles. The momentary performance indices were computed for the latest control cycle for indicating the direction to which the performance is likely heading.

The start of each simulation case was kept as similar as possible and let the process stabilize to a steady state before starting the simulation. All the CVs, MVs and DVs of the test process unit were included in the controller configuration. The benchmark values for the historical and economic performance indices were calculated over a steady state, during which the process was determined to be working as required. These achieved benchmark values were used in every simulation case of this thesis so that the results would be comparable with each other. The presented results of each case were also scaled and therefore the axes do not include the variable units. The scaling was done so that different simulation results would remain comparable. However, the scales of axes in the figures vary between different cases and the figures cannot be compared directly with each other.

7.3.1 Steady state

In the steady state simulation, the control objectives were not changed and all available variables were included in the control calculations. The main input feed flow rate was kept at a constant level. The standard deviations of input feeds were the only variations in the process. Figure 7.2 presents the scaled measurement and set point values of bottom composition AI-001 and the main column's temperature TC-005.

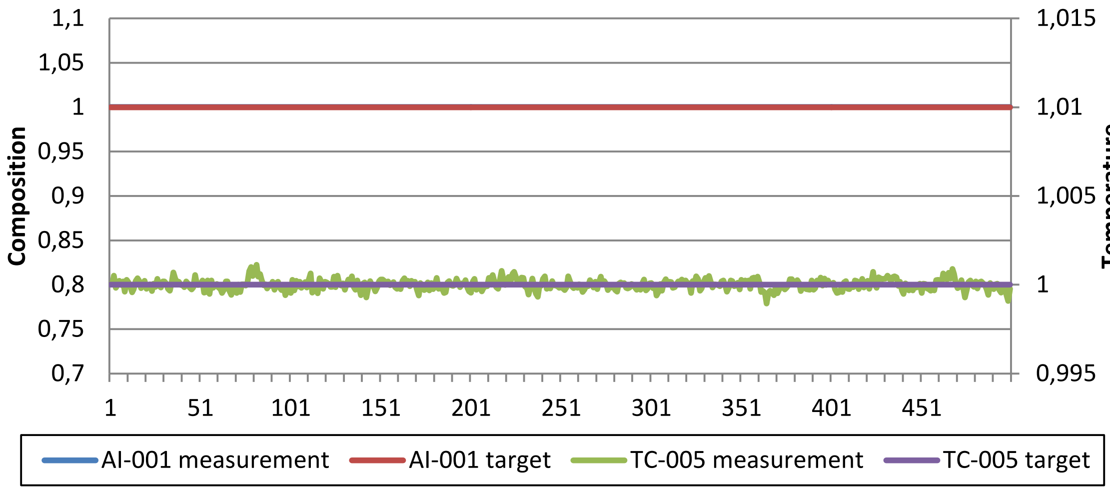

Figure 7.2. *The composition of the bottom product and the main column temperature in the steady state simulation.*

At the upper control level of the APC strategy, the composition of the bottom product AI-001 is handled as a target CV. The measurement is achieved from an analyser. The analyser cycle is equal to 40 control cycles of the current MPC. The main column's temperature is handled as a target CV at the lower level. It also operates in the process unit as an upper level MV having an effect on the bottom composition. As in Figure 7.2 can be seen, there were no changes in the measurement AI-001 and it remained close to the set point value. There was a slight fluctuation in the main column's temperature measurement TC-005 due to standard deviations of process inputs.

The APC strategy of the process unit includes the control of the overhead temperature as condenser's speed of rotation TCA-002VP, operated as a constraint CV. In Figure 7.3, the measurement and limit of the condenser's speed of rotation are presented. The maximum value is achieved from the condenser's physical limit and its maximum speed of rotation. When the process is at a steady state, the condenser's operation point is shifted close to its maximum limit if possible.

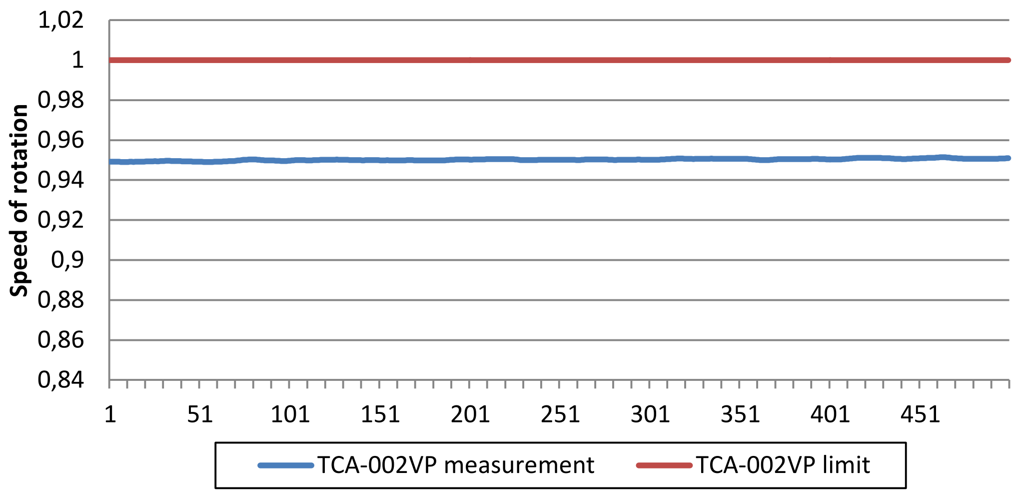

Figure 7.3. *The condenser's speed of rotation in the steady state simulation.*

The lower control level of the APC strategy includes the main column's reflux ratio HC-001 and the reboiler steam flow rate FC-004, which both affect the main column's temperature TC-005. Figure 7.4 shows how the reflux ratio and the reboiler steam of the main column react when the process is in a steady state.

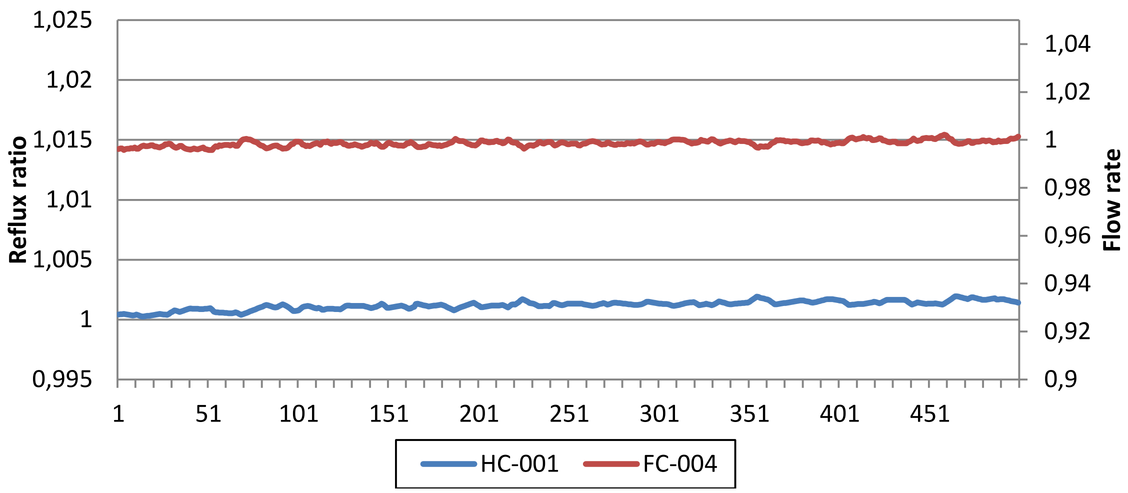

Figure 7.4. *The reflux ratio and the main column's reboiler steam in the steady state simulation.*

The control of the process unit aims to keep the composition AI-001 close to the target value while minimizing the usage of the reboiler steam. When the process was in a steady state, the reboiler steam flow rate slowly moved closer to a level at which it stayed. As the reboiler steam flow rate stabilized, also the reflux ratio remained at a constant level. The steady state, during which the both examined MVs had settled and stayed close to a specific level, can be seen in Figure 7.4.

Figure 7.5 shows the scaled flow rate values of the main input feed FI-001, the distillate FC-006 and the sum of the bottom products FCA-003 and FC-007. The distillate and bottom product flow rates are presented in a relation to the main input feed level. The total bottom product amount is higher than the main input feed since there are two extra feeds in the process unit, which remained at a constant level in all simulation cases of the thesis and therefore are not studied more closely in the results of this thesis. From Figure 7.5 can be noticed that both distillate and bottom product flow rates remained at a constant level as there were no level changes in the main input feed.

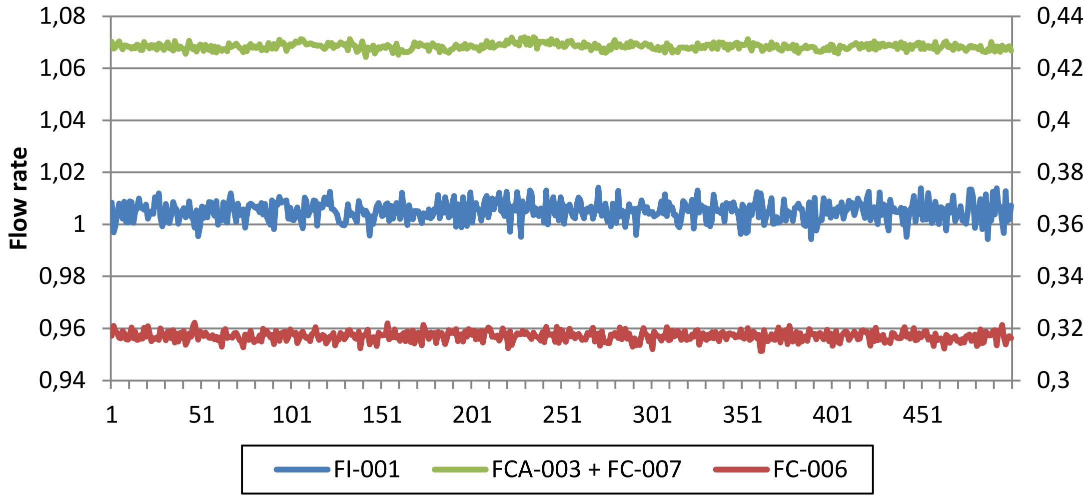

Figure 7.5. *The main feed, distillate and bottom flow rates in the steady state simulation.*

The performance indices related to the technological performance of the MPC controller are presented in Figure 7.6. As it was expected, there were not any notable changes in the performance indices and the overall technological performance remained close to 1. There was a slight fluctuation in the momentary technological performance index due to small temperature and level variations in the input feeds.

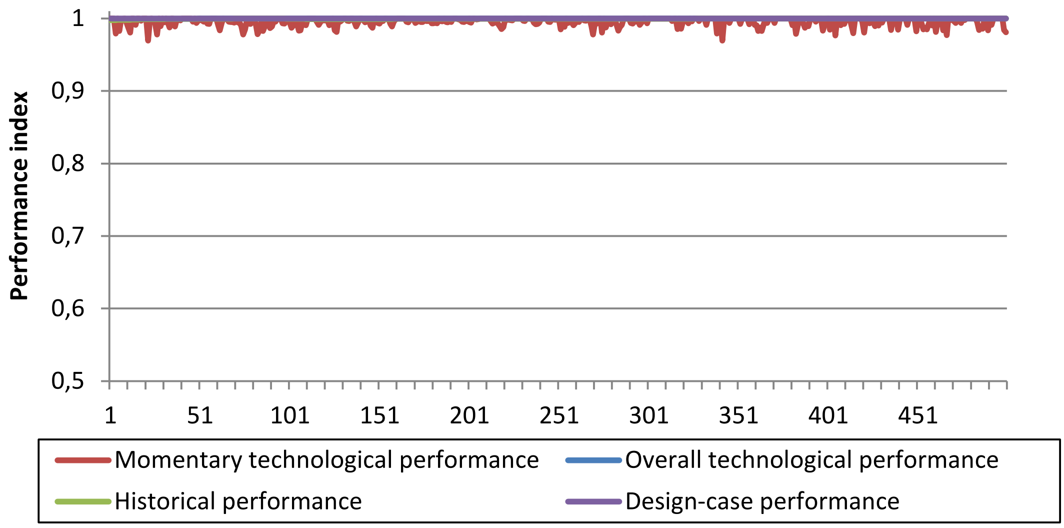

Figure 7.6. *The technological performance in the steady state simulation.*

The economic performance of the controller was also at an excellent level during the steady state simulation. Figure 7.7 illustrates the economic performance index along with the momentary profit and the momentary economic performance index that is calculated based on the profit.

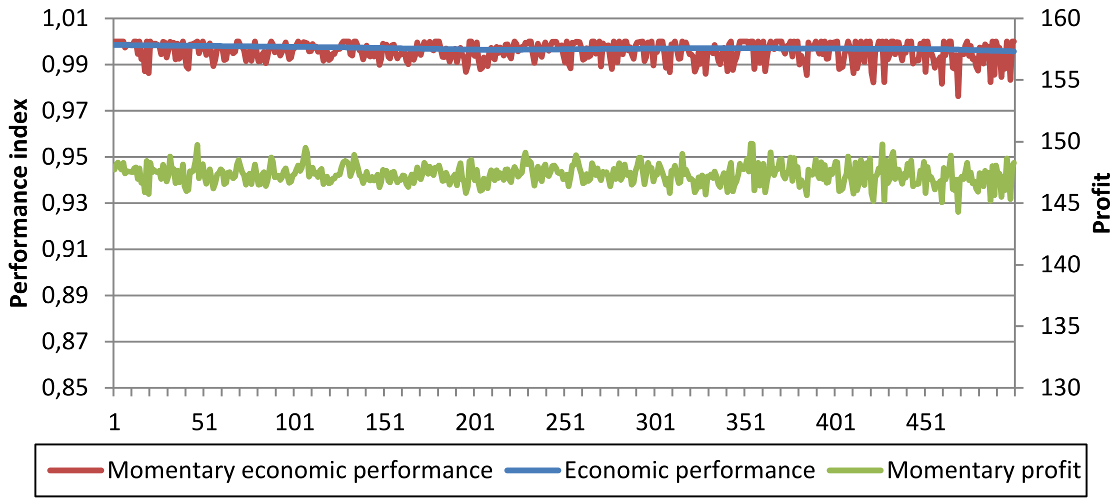

Figure 7.7. *The economic performance in the steady state simulation.*

The momentary profit and thus the momentary economic performance varied slightly due to the variation in the process unit's input and output flow rates. The achieved value for economic performance index was calculated as an average over a 100 cycle moving horizon and as it remained close to the economic benchmark, the economic performance index stayed close to 1.

7.3.2 Set point change

The second simulation case was a set point change. The set point of the bottom product's composition was first decreased by 10 % at sampling instant 90. After the process had stabilized and the AI-001 measurement had reached the new target value, the set point was then moved to a point 5 % higher than the original target at sampling instant 360.

Figure 7.8 illustrates the measurement and target values of the bottom product's composition and the main column's temperature. It can be seen that after the AI-001 set point was decreased, the target value of TC-005 increased. The measurement of TC-005 reached the new set point with a small delay. The AI-001 measurement responded more slowly to the set point change. At sampling instant 360, the set point of the AI-001 was increased. Similarly as in the first set point change at sampling instant 90, the target value and consequently the measurement value of TC-005 reacted and they decreased to a lower level. The measurement of AI-001 responded more slowly but gradually reached the new set point value.

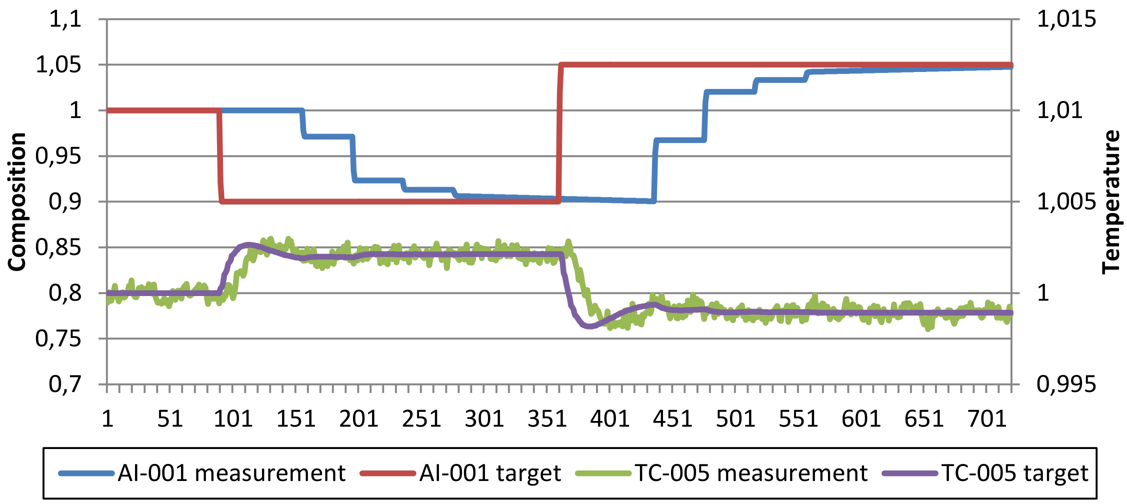

Figure 7.8. *The composition of the bottom product and the main column temperature in the set point change simulation.*

In Figure 7.9, the condenser's speed of rotation during the set point change simulation is presented. There were not any notable changes in the level of TCA-002VP measurement and it did not exceed the maximum limit. As the set point of AI-001 was decreased at sampling instant 90, TCA-002VP measurement slightly rose. After the set point of AI-001 was increased, TCA-002VP dropped to a lower level.

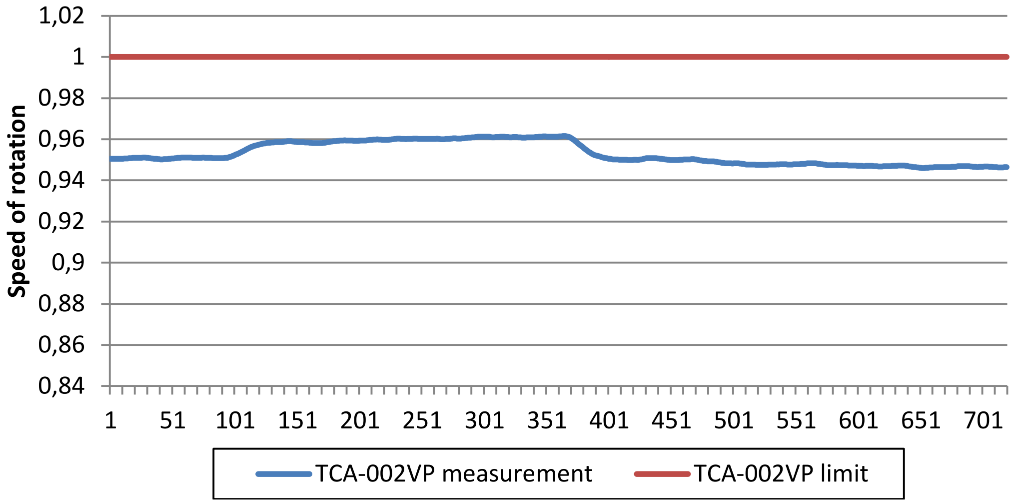

Figure 7.9. *The condenser's speed of rotation in the set point change simulation.*

Figure 7.10 presents how the reflux ratio HC-001 and the main column's reboiler steam flow rate FC-004 reacted to the set point change of the composition AI-001. As the set point of AI-001 was decreased, the flow rate of the reboiler steam increased almost by 2 %. Additionally, the reflux ratio increased slightly. After the set point of AI-001 was increased at sampling instant 360, both FC-004 and HC-001 dropped to levels lower than at the start of the simulation case.

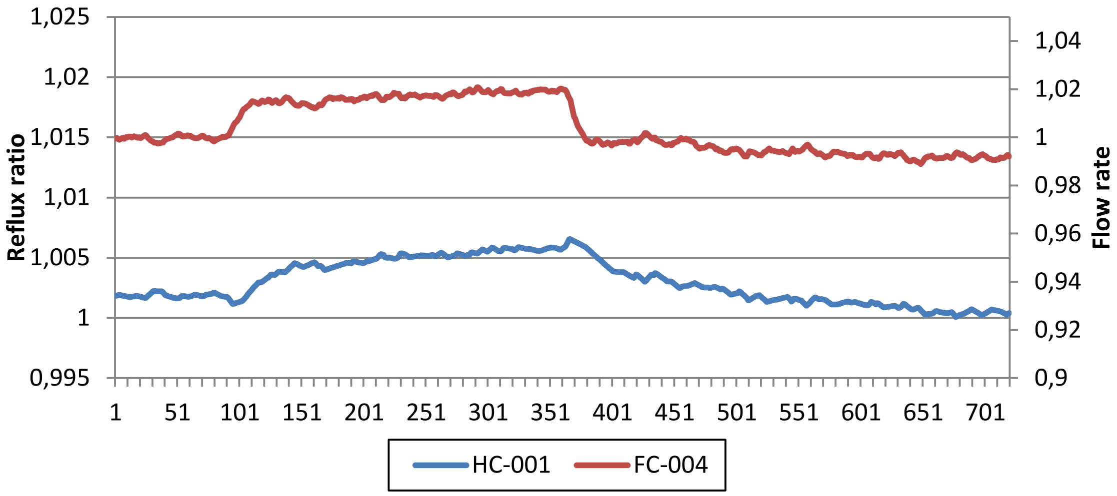

Figure 7.10. *The reflux ratio and the main column's reboiler steam in the set point change simulation.*

The main input feed, distillate and bottom product flow rates of the set point change simulation are presented in Figure 7.11. There were no significant changes in the flow rates when compared to the steady state simulation. A slight decrease can be seen in the distillate flow rate FC-006 during the time period that the reflux ratio HC-001 was at a higher level.

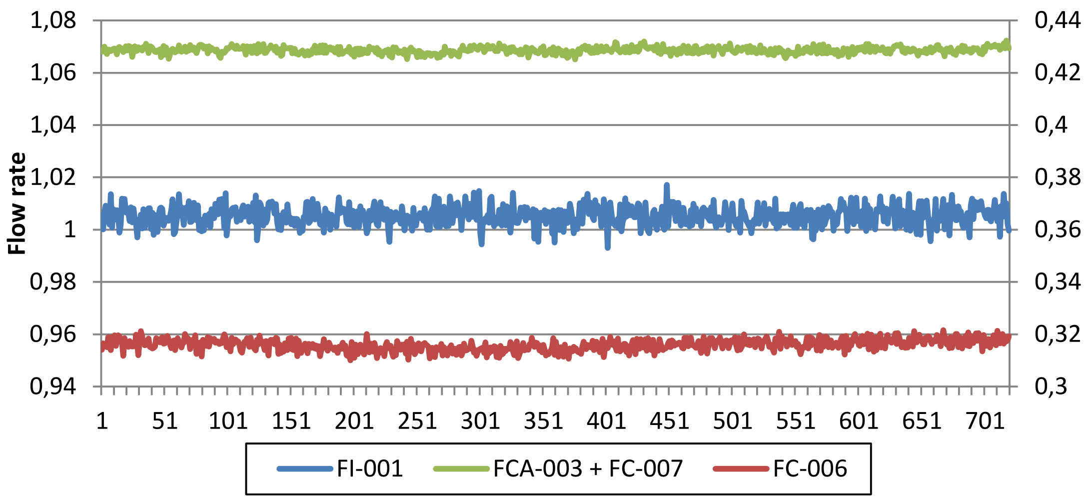

Figure 7.11. *The main feed, distillate and bottom product flow rates in the set point change simulation.*

The technological performance of the controller is presented in Figure 7.12. The deviation between the set point and measurement values of target CVs had the greatest impact on the technological performance of the controller. The level changes of the MVs also required more control moves than staying at a constant level, which also affected the performance indices.

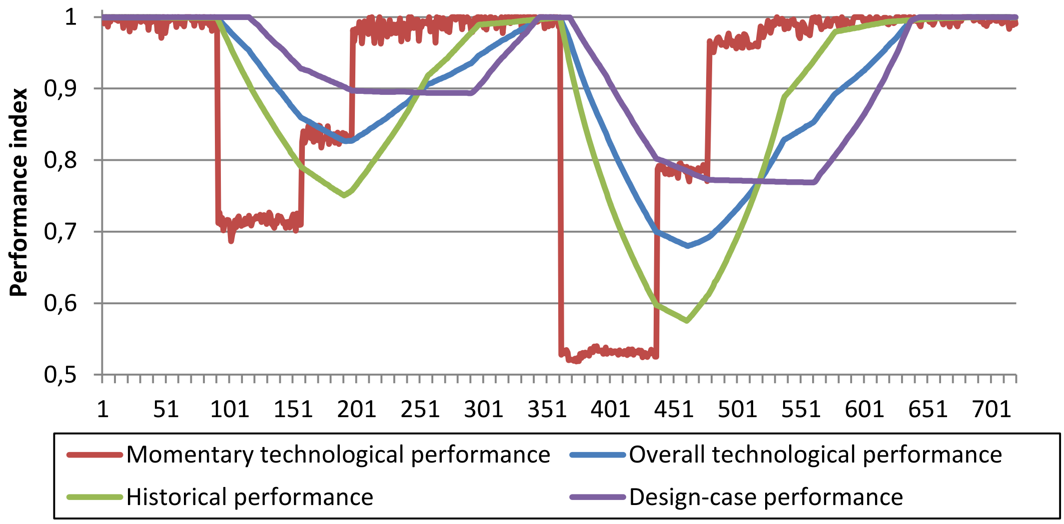

Figure 7.12. *The technological performance in the set point change simulation.*

The momentary technological performance reacted instantly to the set point changes and consequently to the deviation of composition AI-001. Figure 7.12 shows that there were gradual changes in the momentary performance like in the measurement of AI-001. The historical performance index reacted more quickly to the set point changes than the design-case index which has the achieved value calculated as an average over a longer time period than the historical performance index. The overall technological performance remained however in the first 10 % set point change at a good level, and after the bigger set point change it fell to a level around 0.7 before returning close or equal to 1.

The economic performance of the controller remained at an excellent level during the set point change simulation. Figure 7.13 illustrates the economic performance index besides the momentary economic performance index and profit.

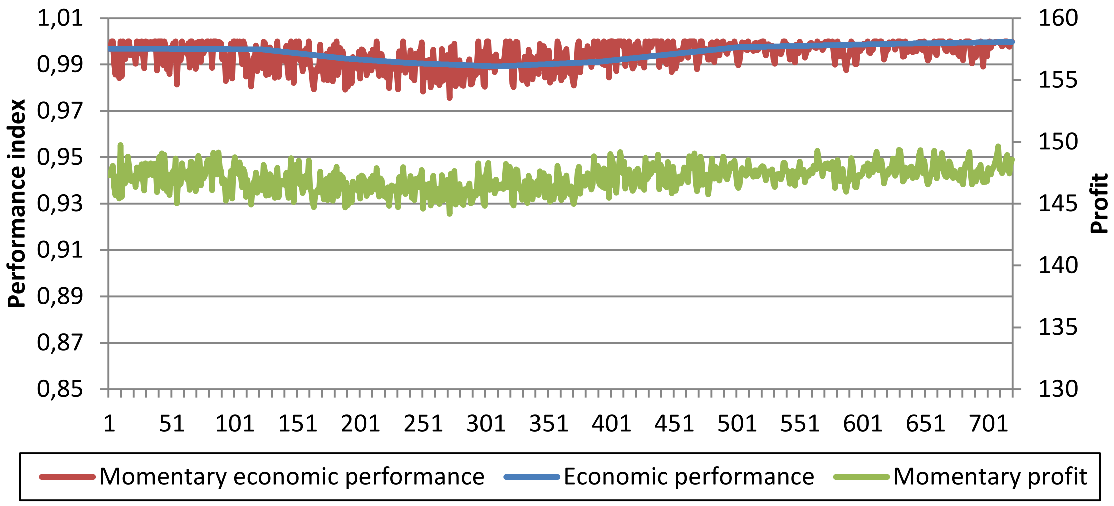

Figure 7.13. *The economic performance in the set point change simulation.*

From Figure 7.13 can be seen that the momentary profit dropped after the sampling instant 90 when the AI-001 set point was decreased. This set point change caused the main column's reboiler steam flow rate FC-004 to increase. Also the reflux ratio HC-001 increased, which caused a minor reduction in the distillate flow rate FC-006. As FC-004 increased and FC-006 decreased, the relation of the economic input and output flows changed. The decrease of the momentary profit caused the momentary economic performance and moments later the economic performance to drop. After the sampling instant 360 when the AI-001 set point was moved to a higher value than the initial, the momentary profit moved also to a slightly higher level, due to which the economic performance reached some sampling instants later a level that was higher than at the beginning of the simulation case.

7.3.3 Limit change

The effect of a limiting constraint CV on the controller performance was tested. The test was done by decreasing the maximum limit of the overhead condenser's speed of rotation TCA-002VP by 10 % at the sampling instant 150. The main input feed was kept at the initial level and other control objectives were also not changed. Figure 7.14 shows how the TCA-002VP measurement reacted to the limit change. The measurement started to go down quickly after the limit change. It reached the new limit in approximately 30 cycles and remained after that near the maximum limit.

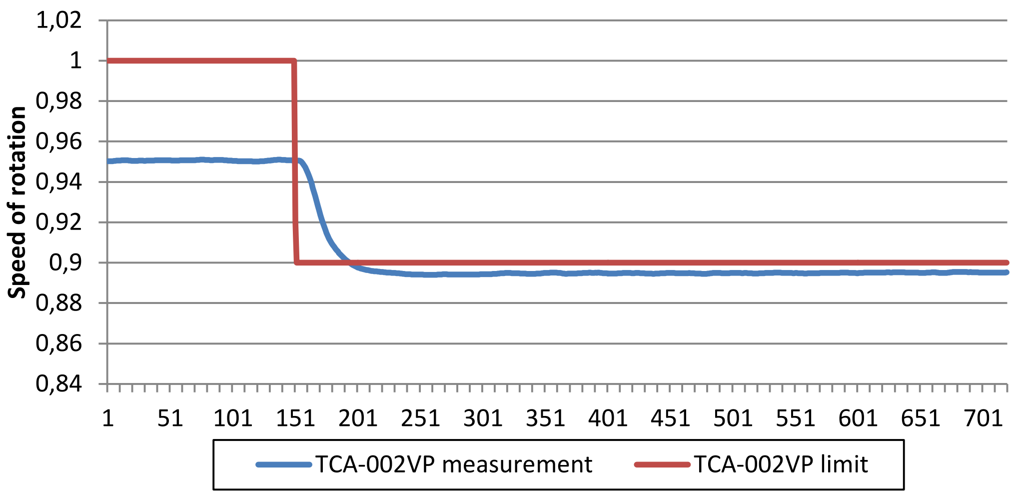

Figure 7.14. *The condenser's speed of rotation in the limit change simulation.*

Figure 7.15 presents how the examined target CVs reacted to the limit decrease of TCA-002VP. The set point of the main column's temperature TC-005 decreased after the limit was changed at the sampling instant 150. The measurement also dropped but it was too low when compared to the set point, which produced a constant deviation between the set point and measurement of the temperature. The measurement of the composition AI-001 started to rise after a delay. At highest, there was a difference of approximately 35 % between the composition measurement and set point. After that AI-001 measurement began to gradually decrease. However, the decreasing happened extremely slowly and the composition measurement did not reach the set point and thus a constant deviation remained during the end of the simulation.

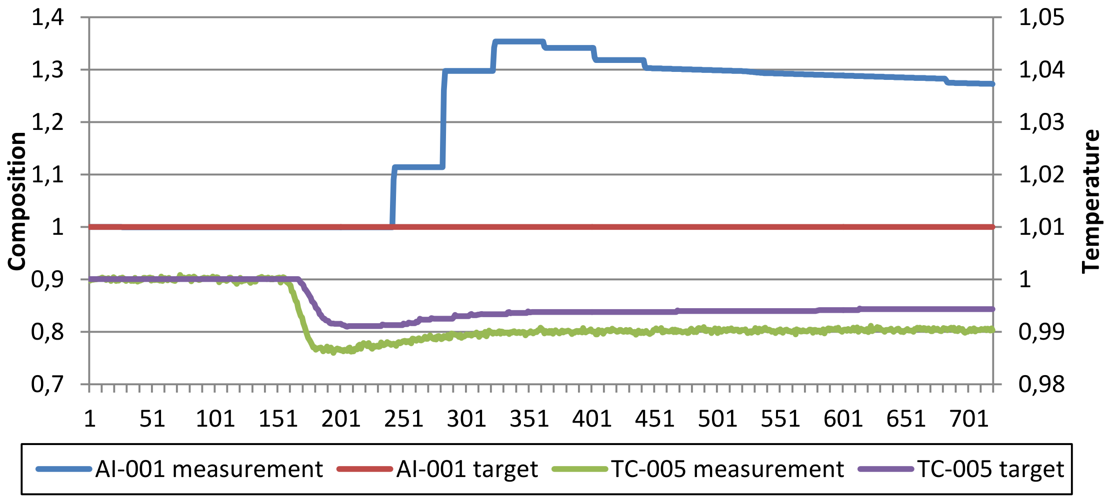

Figure 7.15. *The composition of the bottom product and the main column temperature in the limit change simulation.*

Reactions of the reflux ratio HC-001 and the reboiler steam flow rate FC-004 to the limit change at sampling instant 150 are presented in Figure 7.16. Both reflux ratio and reboiler steam flow rate decreased quickly after the limit change. HC-001 dropped slightly under 3 % whereas the decrease of FC-004 was approximately 10 %. By cutting down the reboiler steam flow rate, the overhead condenser's speed of rotation could be dropped under the new maximum limit. However, this impacted also on the main column's temperature TC-005, which in turn added the composition of the bottom product.

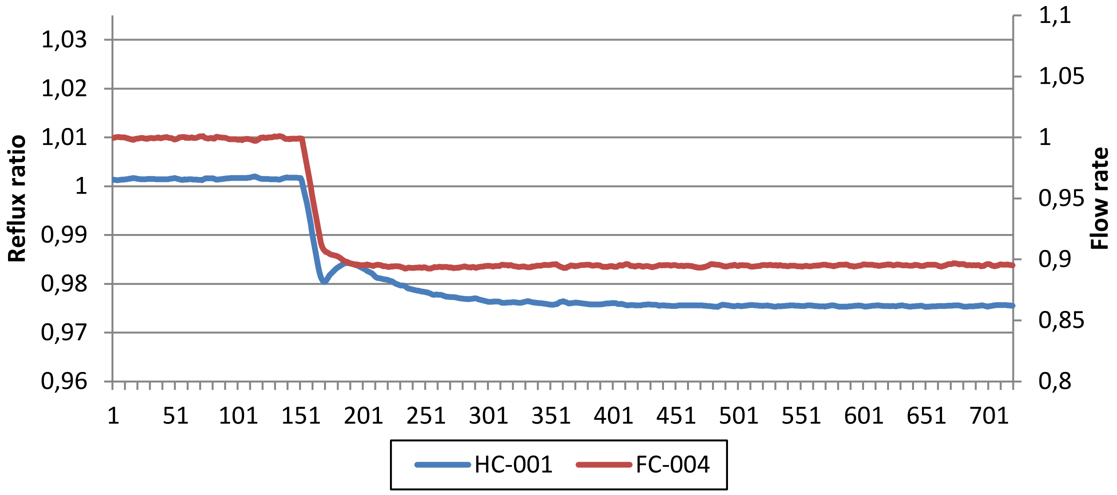

Figure 7.16. *The reflux ratio and the main column reboiler steam flow rate in the limit change simulation.*

The main input remained at a constant level during the limit change simulation. The main feed flow rate is presented in Figure 7.17 along with the distillate and bottom product flow rates. Figure 7.17 shows that the distillate flow rate FC-006 went up after the limit change at the sampling instant 150 as the reflux ratio of the main column decreased.

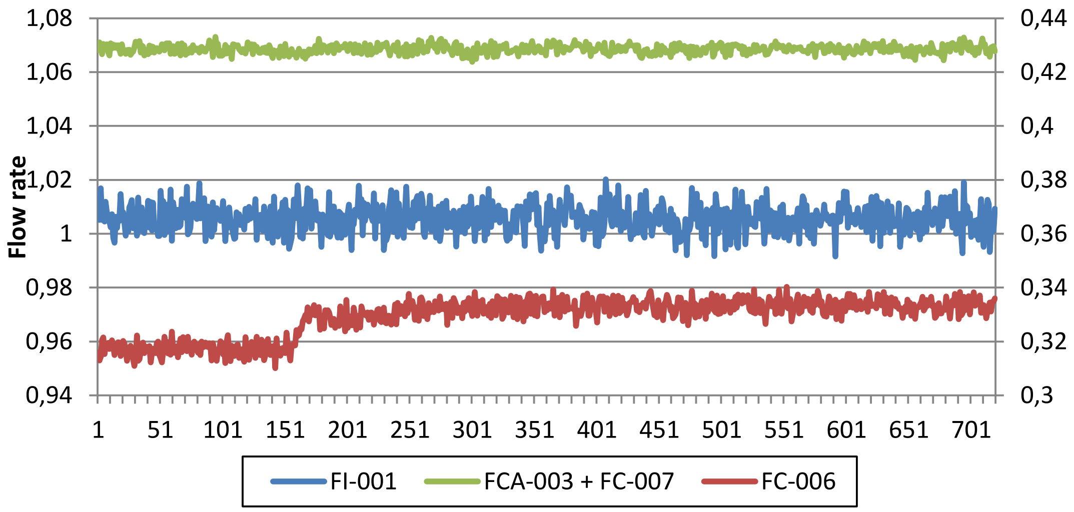

Figure 7.17. *The main feed, distillate and bottom product flow rates in the limit change simulation.*

The technological performance of the controller during the limit change simulation is presented in Figure 7.18. It can be seen that the momentary technological performance depended mostly on the deviations between the measurement and set point values of the target CVs besides the maximum limit's exceeding of the constraint CV. The momentary technological performance dropped close to 0 as the maximum limit of TCA-002VP was changed and the limit was exceeded. In addition, the change of the maximum limit caused the MVs to change their level which induced the achieved value of performance indices to deteriorate. However, the momentary technological performance went up quite quickly as the TCA-002VP measurement reached the new limit. The difference between the measurement and set point of the composition AI-001 started to grow gradually after the sampling instant 230, which induced gradual decreasing of the momentary technological performance. As the historical performance index is based on the average over a moving horizon, its behaviour was strongly related to the momentary technological performance. At first, the historical performance index dropped significantly but it rose back to a good level after the sampling instant 250. Since the control error of AI-001 increased, the momentary technological performance index and thus the historical performance index went down. As presented in Figure 7.15, neither of the target CVs reached the set point over the end of the simulation, which induced the historical performance to remain at a poor level.

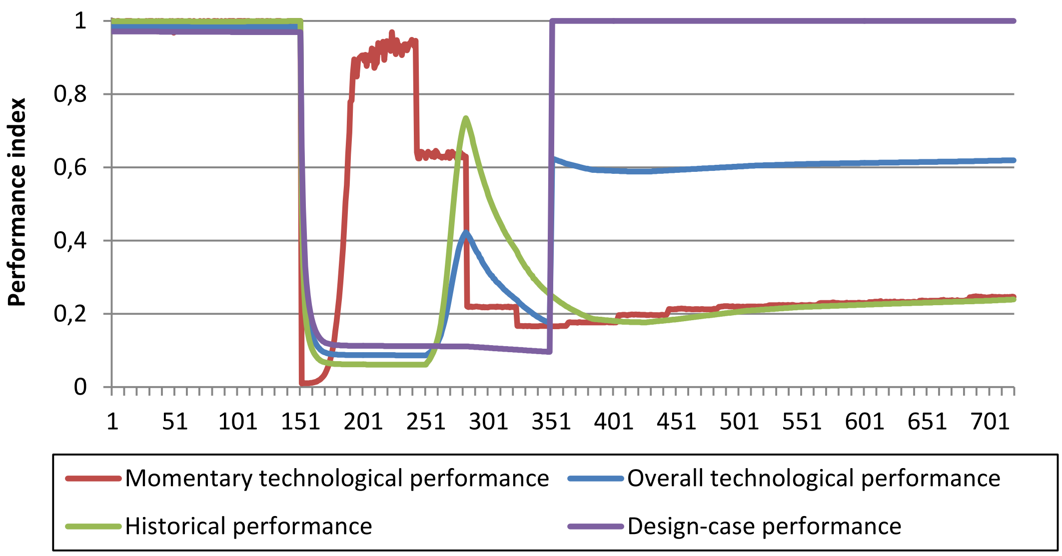

Figure 7.18. *The technological performance in the limit change simulation.*

When the maximum limit of TCA-002VP was decreased, the design-case index dropped also instantly. The design-case performance stayed close to value 0.1 for 200 cycles. After that, the design-case index was computed using a benchmark value that included the changed control objectives. Although the examined target CVs did not reach their set points, the design-case performance index ran to 1 and stayed at the excellent level for the end of the simulation. As the overall technological performance is calculated as the average of the historical and design-case indices, the controller's technological performance remained good in spite of the poor historical performance index.

The economic performance of the controller during the limit change simulation is presented in Figure 7.19. In addition, the momentary profit and momentary economic performance are presented.

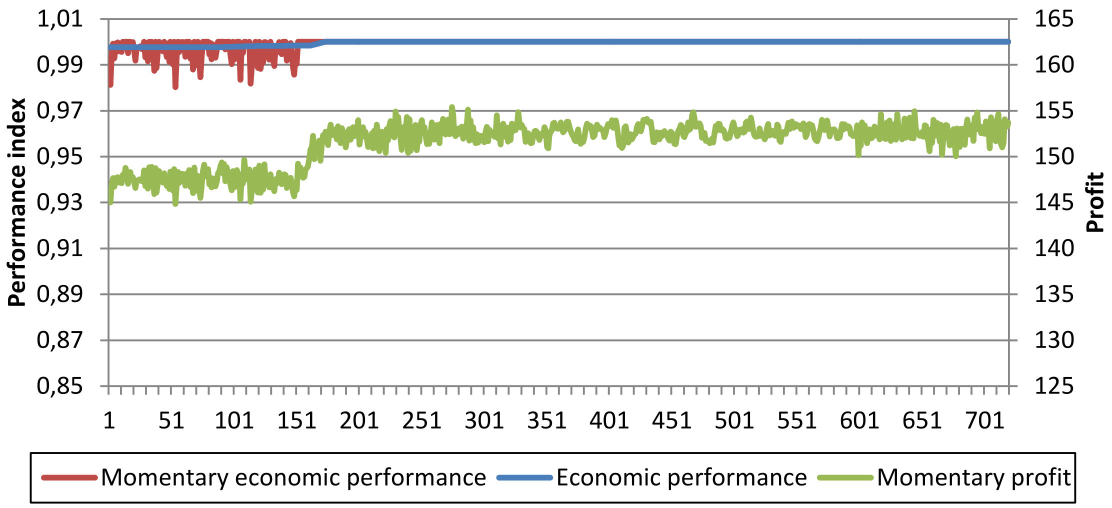

Figure 7.19. *The economic performance in the limit change simulation.*

After the limit was changed, the distillate flow rate FC-006 increased whereas the main column's reboiler flow rate FC-004 decreased, which changed the relation of economic inputs and outputs so that the momentary profit rose to a higher level. This caused the momentary economic performance and eventually the economic performance to be limited to value 1.

7.3.4 Unknown process disturbance

The last case to be simulated was an unknown process disturbance. When a DV is switched off, the controller is not able to include its effect in the control computations as it otherwise would be included as feed forward control. The simulation was carried out by switching the main input feed flow rate FI-001 off and then manipulating its level. First at sampling instant 90, FI-001 was decreased by approximately 10 %. After the process had stabilized, the main feed was raised to a level 5 % higher than initially. The main feed, distillate and bottom product flow rates are presented in Figure 7.20.

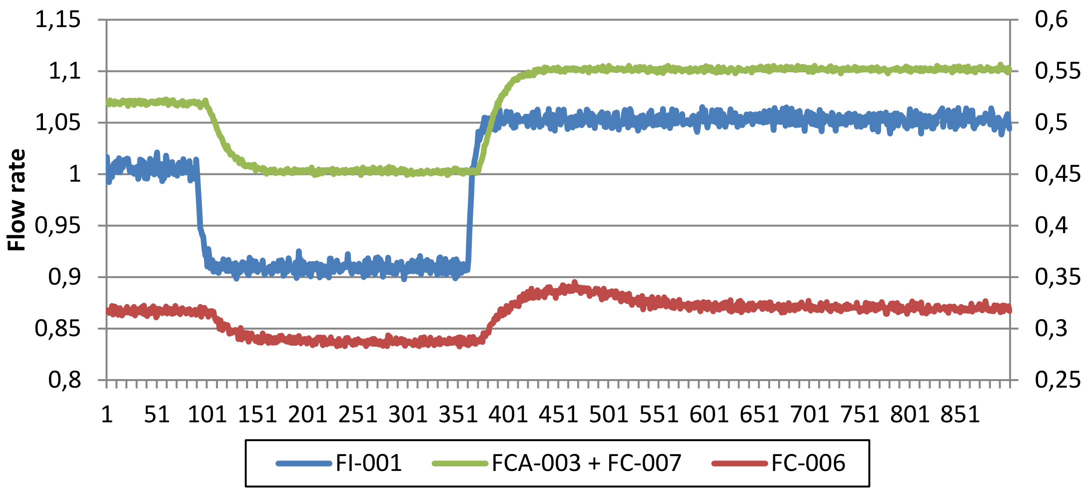

Figure 7.20. *The main feed, distillate and bottom product flow rates in the unknown disturbance simulation.*

Figure 7.20 shows that the distillate flow rate FC-006 and the total bottom product flow rate as the sum of FCA-003 and FC-007 changed as the main feed level decreased or increased. There was a small delay from the main feed to the output flow rates. After FI-001 was increased at sampling instant 350, the distillate flow rate FC-006 also rose. However, FC-006 slightly decreased after a while and then remained for the end of the simulation close to the initial level with unchanged main feed FI-001.

The measurement and set point values of the composition AI-001 and the main column's temperature TC-005 are presented in Figure 7.21. After the decrease of FI-001, the main column's temperature TC-005 started to increase. After a while, TC-005 measurement returned to the set point. As a result, the measurement of the composition AI-001 differed from its set point by 5 %. After the process had settled and the level of FI-001 was increased at sampling instant 350, the temperature TC-005 dropped. Additionally, the set point of the main column's temperature decreased but it returned to the initial level after approximately 200 cycles. The measurement of TC-005 reached the set point for a while but then decreased to a 0.5 % lower level than the set point and a constant control error remained over the end of the simulation. The composition AI-001 increased gradually and was at maximum 30 % higher than the set point after the increase in the main feed. After that, the AI-001 measurement eventually fell and returned close to the set point.

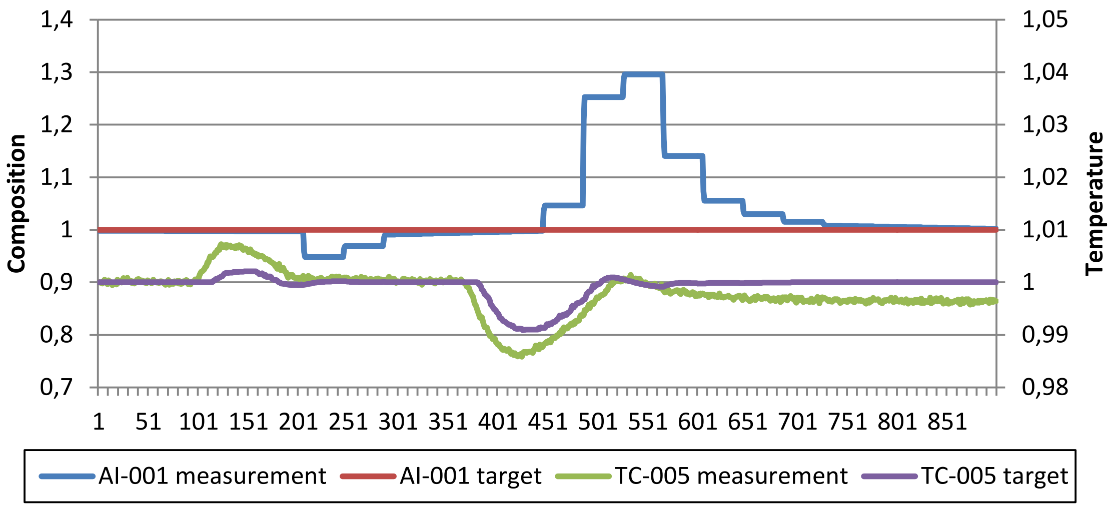

Figure 7.21. *The composition of the bottom product and the main column temperature in the unknown disturbance simulation.*

Figure 7.22 presents how the condenser's speed of rotation reacted to the changes of the main feed level as the unknown process disturbance. During the simulation case, the maximum limit was not exceeded and TCA-002VP measurement remained under it. TCA-002VP was related to the level of FC-001 and it decreased as the main feed of the process unit dropped and rose as the main feed was increased. When FC-001 was at the higher level during the end of the simulation, the condenser's speed of rotation stayed very close to the maximum limit. Should the main feed have been higher, TCA-002VP maximum limit would likely have been exceeded and the control of the process would have been weakened.

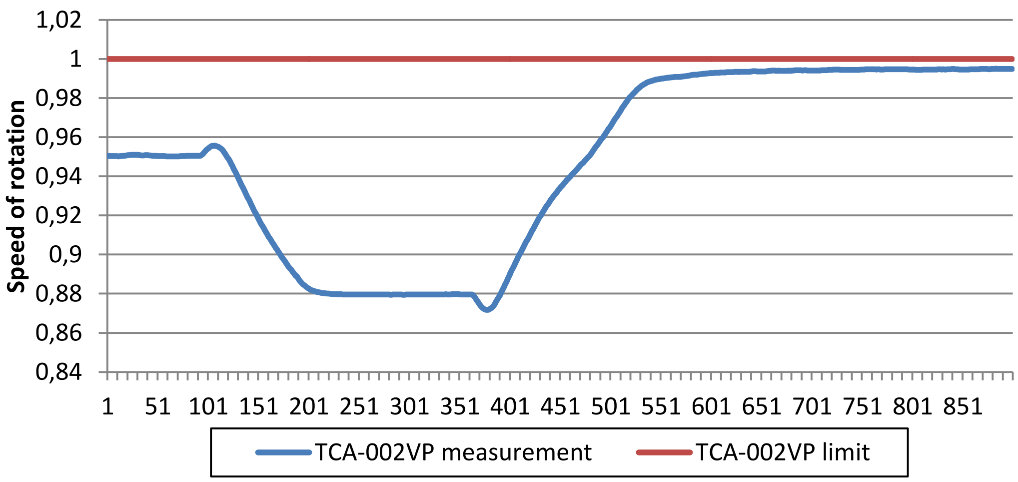

Figure 7.22. *The condenser's speed of rotation in the unknown disturbance simulation.*

The reflux ratio HC-001 and the main column's reboiler steam flow rate FC-004 during the unknown disturbance simulation are presented in Figure 7.23. FC-004 started to decrease after the level of main feed FI-001 was lowered. From Figure 7.23 can be seen that FC-004 reached a minimum defined by the physical limit of the steam valve position and stayed after that at a constant level. The reflux ratio HC-001 stayed quite close to the start level when FI-001 level was decreased. After the level was increased at sampling instant 350, the reboiler steam flow rate rose steadily. HC-001 decreased first for 100 cycles but then started to rise to a higher level.

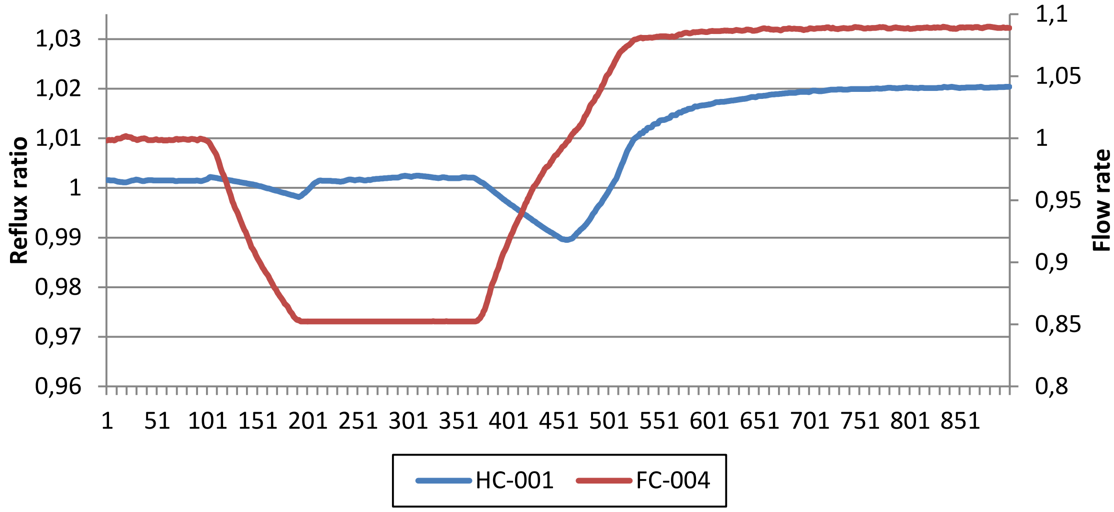

Figure 7.23. *The reflux ratio and main column's reboiler steam in the unknown disturbance simulation.*

The technological performance of the controller during the unknown disturbance simulation is illustrated in Figure 7.24. When the main feed was increased at sampling instant 90, a control error of the main column's temperature TC-005 appeared. In addition, decrease of the reboiler steam flow rate FC-004 required more control moves than staying at a steady state, which increased the value of cost function determining the achieved value of computed performance indices.

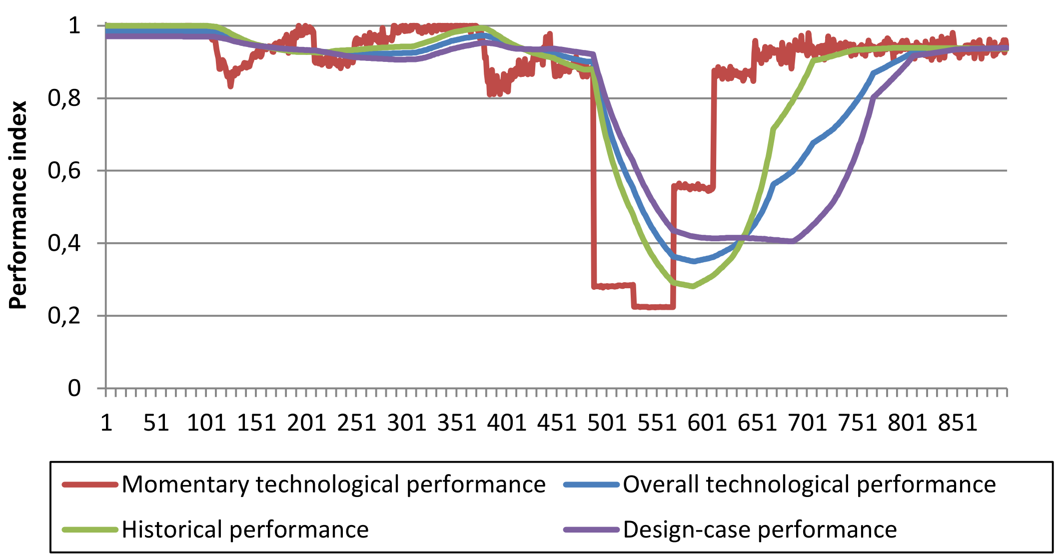

Figure 7.24. *The technological performance in the unknown disturbance simulation.*

As the control error of the main column's temperature TC-005 decreased, the momentary technological performance index rose back close to value 1. Around sampling instant 200, AI-001 measurement deviated from the set point causing the momentary technological performance to drop. The momentary technological performance rose as AI-001 measurement returned to the set point. The historical and design-case performance indices decreased to about 0.9 after the first level change of FI-001. After sampling instant 350, the momentary technological performance reacted again to the control error of TC-005. As the control error of bottom composition AI-001 increased, the momentary technological performance dropped to the poor level. The historical performance was at lowest approximately 0.3 whereas the design-case index dropped to 0.4. After the measurement of AI-001 started to decrease and returned to the set point value, the momentary technological performance rose to excellent level. Also the historical and design-case performance indices and thus the overall technological performance index returned to a value slightly over 0.9. The performance indices did not reach the value 1 because a deviation between TCA-005 measurement and set point value remained.

The economic performance of the controller is presented in Figure 7.25 along with the momentary profit and the momentary economic performance index. After the main feed was decreased at the sampling instant 90, a significant increase in momentary profit can be seen in Figure 7.25. This is due to that although the main feed FI-001 was decreased, there was a small delay before the decrease could be seen in the process unit's outputs, which caused a sudden improvement in the momentary profit and thus in the momentary economic performance. An opposite event appeared after the level of FI-001 was increased and the output flow rates stayed at the old level before rising higher, which caused the momentary profit and the momentary economic performance to drop momentarily.

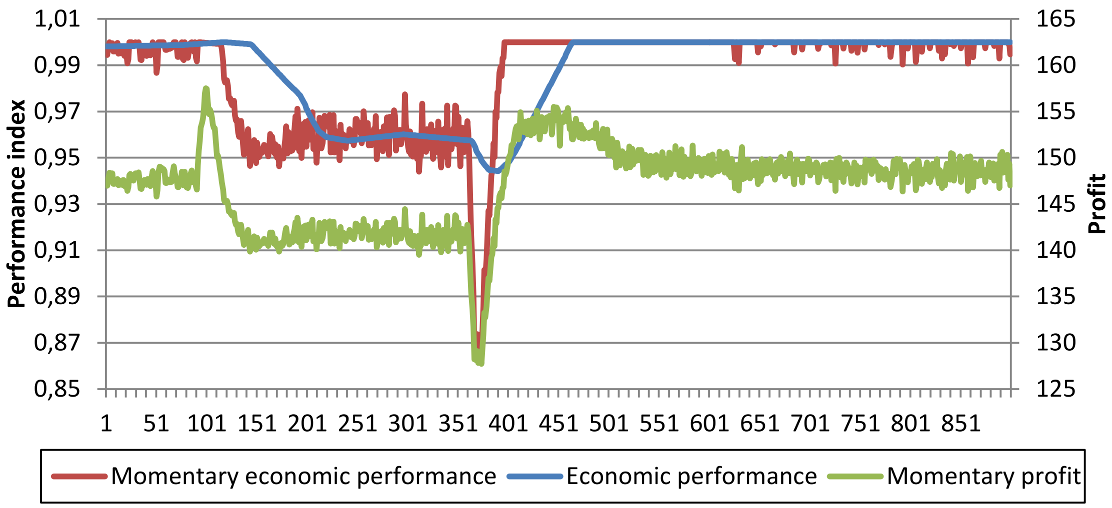

Figure 7.25. *The economic performance in the unknown disturbance simulation.*

The economic performance of the controller followed the momentary economic performance index in outline as it was computed as an average over a moving horizon. As the momentary profit and the momentary economic performance decreased, the economic performance also dropped close to 0.96 after a while. Similarly after the increase of FI-001 level, the economic performance went up to 1. The momentary profit reached such a high level that the momentary economic performance was limited to 1 after sampling instant 400 for over 200 cycles. This caused also the economic performance to be limited to 1. After this, the momentary profit returned to a lower level close to the benchmark value so that the momentary economic performance fluctuated around value 1. The economic performance remained at value 1 over the end of the simulation.

8. DISCUSSION ON RESULTS

The implemented performance assessment tool was designed so that the deployment could be done as automatically as possible. The function of the performance calculation application as well as the structure of the user interface was designed to be general and suitable for assessing the performance of multivariable and model predictive NAPCON Controller. The deployment of the performance assessment tool required an update of the software environment including NAPCON OPC UA Server, so that the new object types could be created. Before the economic performance of the controller could be assessed, the economic inputs and outputs of the test process needed to be defined along with their prices and references to available measurements. The structure of the updated OPC UA address space allowed the new economic variables to be added easily for the economic performance assessment of the controller.

As the performance calculation was started and the calculations were on, the values of different variables related to the controller's performance were computed and updated to the database of the test environment. The start of the different simulation cases varied slightly since each simulation was carried out separately and the process stabilization point varied. However, the simulation results were scaled and same historical and economic benchmark values were used so that the results of the different cases would be comparable with each other. Generally, the simulations showed that the achieved performance indices responded as expected when there were changes in the process or control objectives, and correspondingly they remained at a constant level when the process conditions and the control objectives were kept unchanged. The different simulation cases showed that the technological performance of the controller decreased as the control errors of target CVs increased or the maximum limit of the examined CV constraint was exceeded. Additionally, the operating points of the examined MVs changed at the same time and more control moves were required than the historical benchmark defined or the controller prediction calculations presumed. This increased the achieved values defined by the cost functions of equations (14) and (19) and thus decreased the indices related to the technological performance of the controller.

The momentary technological performance indicated the changes in the controller performance instantly whereas the historical and design-case performance indices and thus the overall technological performance index reacted more slowly. This was due to that they were computed as an average over defined moving horizons. The horizon of the design-case index was determined based on the controller parameters and the achieved value was compared to the benchmark value that was earlier calculated of the controller's variable predictions. The achieved and benchmark values of the historical performance index were computed over horizons dependent on controller parameters. The historical performance index would have reacted more quickly if the horizon, over which the achieved value was calculated, was defined shorter. In the limit change simulation, the design-case performance index rose to 1 whereas the historical performance index remained at a poor level. This indicated that although there were control errors, which the historical benchmark value did not allow, the controller prediction calculations had adjusted to the new conditions. The controller calculations had taken into account that every control objective could not be satisfied and the design-case benchmark was also in accordance with that. Conversely, if a constant deviation between measurement and set point values remained against the controller predictions, the design-case index would not reach value of 1.

The results on the momentary profit, the momentary economic performance and the actual economic performance were also collected by the different simulation cases. The results showed that the selected methods provided a method to assess the economic performance of a multivariable and model predictive controller. The simulations showed that the values of economic related variables depended on the measurements of the defined economic inputs and outputs of the process. The effect could be seen on the controller's economic performance if the relation between the process inputs and outputs changed. The changes were not so notable and the economic performance stayed at the excellent level in the different simulation cases. Although the economic performance remained excellent in the obtained results, the current economic performance assessment method does not take quality product specifications into account. The composition of the bottom product deviated greatly in some simulation cases from the target value. At the worst, the output product could be worthless which would cause the economic performance of the controller naturally to decrease.

Based on the achieved results, the selected control performance assessment methods and techniques provided a way to assess performance of a multivariable MPC both from technological and economic aspects. The basic ideas of the selected methods were generally quite simple and the performance calculations did not add the computational burden too much. However, the simplicity of the selected control assessment techniques could also be seen as a disadvantage as they do not take everything into account. Although the problem related to the possible biasing of the historical benchmark was avoided by the definition of the historical benchmark value from a user-defined time period, a new challenge appeared. A problem with the historical benchmark is related to the situation where the controller configuration changes. Then the earlier defined benchmark value would not necessarily be suitable for the new configuration as the control objectives could be different than initially.

The examined methods are based on a comparison between achieved and benchmark values which are obtained from a cost function. The cost function is computed as a sum of control errors and control moves. Using a total sum does not separate the different process variables and their individual performances. This could allow some variables to perform poorly while some others are outperforming the requirements, and the controller would seem to be performing well according to the performance indices.

9. FURTHER STUDY

The results of the different simulation cases showed that the implemented performance assessment tool worked properly and the controller's performance could be assessed based on the selected methods. Although the results were promising, further study on the application's functionality is required before it can be released as a finished product. The performance assessment application worked well at the tested simulation environment, but it should be also tested in a real process environment with actual process data. Additionally, deployment to various processes would be required to verify that the implemented tool is general and suitable to assess performance of various controllers.

Besides testing in different process environments, the results of performance calculations could also be collected from further test simulations. Especially the suitability of the historical benchmark in different controller configuration cases should be studied more. Now the historical benchmark is updated automatically to a new value based on the individual variable benchmark values which are collected during the time the historical benchmark search is on. Normally, the aim is that controller configuration includes all available variables in the calculations. However, it is possible that there are for example two alternative controlled variables defined in the process unit and only either of them is included in the control calculations at a time. It is also possible that lower level control action is lost and therefore there are less manipulated variables available, which would cause the controller configuration and thus the control problem formulation change.

The importance of separate variables was not specifically studied in this thesis and the weighting for performance assessment cost functions was done based on the available controller parameters. However, for example the relative importance of quality variables could be emphasized more from the performance aspect and extra variable coefficients could be included in the performance calculations. Nevertheless, this would require additional work as the importance of the different variables on control performance was defined when the performance assessment tool was applied to a new environment. Defining the performance weights of different variables would require tailoring for different process units and controllers. Besides the variable weighting, the individual performance indices of variables could be studied for further examination and identification of controller's performance.

NAPCON Controller structure contains different types of CVs and MVs. Currently, only the control moves of the manipulated variables are included in the performance calculations. Besides the required control moves, the state of an optimized MV could be examined. Some optimized MVs are comparable to target CVs in a way and an error part of optimized MVs could be included in the objective functions forming both the achieved and benchmark values of performance indices. Unfortunately, the controller does not provide a direct way to implement this at the moment, and the application would not be that simple.

The economic performance depends on the price coefficients and the amount of the economic inputs and outputs in the implemented performance assessment tool. In order to provide a more valid presentation of the controller's economic performance, also the specifications of the quality products should be taken into account. Additional coefficients and limits for quality variables could be defined. If a quality variable violated a defined limit, the economic performance would decrease according to the coefficients and at worst be zero if the provided output product was worthless.

The implemented GUI in this thesis contained switches for performance calculation and benchmark update states. In addition, the calculated performance indices were presented as a number and the performance level was indicated verbally. In future, the presentation of the performance index as well as the level could be visualized so that the user is able to see more easily how the controller is performing. An alternative is using graphical measurement bars that would have a colour code for the level indication. In addition, including trend views to the GUI of the performance assessment tool would allow the user to see at a glance how the controller performance has been recently. Currently, the history of the performance variables is stored to the history database and it can be studied with the trend tool of NAPCON Information Manager. The trend views could contain information of the MV constraints. If some MVC is limiting, this can affect the controller's performance. In this thesis, the performance indices were not manipulated based on active MVC limits but they could be used for flagging in the trend views, which would allow the user to see whether the controller performance has decreased due to a limiting MVC. Additionally, separate variable performance indices could be used in the control performance diagnosis phase after their further study. The information could be included in the user interface for example as an alarm list containing variables which are performing poorly when compared to their benchmarks. A second alternative could be using the separate variable performance indices in trend views for flagging of the overall technological performance decrease.

The functionality of the performance assessment tool does not currently take into account if the number of the controllers on the server or the database structure changes. At present, the performance calculation service needs to be stopped and then restarted if these kinds of alterations are carried out on the server. The restart of the application enables the re-initialization of the performance calculation object. However, the stopping of the service is not highly preferred and an alternative solution would be beneficial. A potential way to handle the change of the database structure is adding a switch for re-initialization of the performance calculation object. The switch would allow reinitializing the related performance calculation object of the controller. This requires the redefinition of the historical and economic benchmark values by the user, which would be a reasonable task while the database structure is altered. When the structure of the controller is changed, a proper examination of the controller behaviour is likely to be done and the benchmark values could then be determined on the side. Monitoring and handling of the number of the controllers could be implemented as a task of the performance calculation handler object.

After the functionality of the performance assessment tool is verified by different tests, the tool can be deployed to new environments as a part of the controller software to indicate the controller performance. In this work, the controller tuning was kept unchanged as the controller tuning parameters remained constant. The tool could also be used in the implementation phase of a new controller as the controller is tuned and suitable controller parameters are studied, after the operation of the performance assessment tool is proved to be as required.

10. SUMMARY

APC technologies have been applied to various process environments in order to improve for example product quality and yield. In many process industries, the most popular APC strategy is the model predictive control. MPC's ability to handle constrained problems can be considered as the most important feature when compared to other APC strategies. A control system normally operates at its nominal efficiency after design and implementation. However, the performance of the control system usually decreases over the time due to various causes. The aim of control performance assessment and monitoring is to ensure that control systems operate as required in order to secure effective process control as well as safe and profitable plant operations.

The aim of this thesis was to design and implement a tool for performance assessment of a multivariable, model predictive controller. It was required that the structure of the tool was generic so that it would be suitable for different controllers and control environments. It was also required that the operation of the performance assessment tool does not affect the actual controller functionality. In addition, the deployment of the performance assessment tool to a new environment was to be kept as simple and automatic as possible.

As the thesis was a continuation to the work of Janne Oksanen, who had done a wide research on various CPA methods, different CPA technologies alongside recently proposed methods were introduced generally in this thesis. Historical and design-case benchmarking methods were studied more closely as they were selected to be utilized in the implemented performance assessment tool for the assessment of the controller's technological performance. Both of the selected methods are based on a comparison of an achieved value and a benchmark value, which are both calculated based on the cost function included normally in the MPC's optimization problem. The historical benchmarking method is a user-specified method and the benchmark value is calculated over a time during which the controller is defined to be performing well by the user. The design-case benchmark provides a model-based approach for assessing control performance, as the benchmark value is calculated of the controller predictions. In addition, the economic performance of the controller was assessed.

After the selection of CPA methods, aspects related to the automation information technology were studied before the performance assessment tool could be implemented. As the automation hierarchy of a plant contains several different levels, also various communication protocols are required in order to provide access to data between different automation levels. OPC UA is a communication protocol developed by the OPC Foundation and it provides a more effective way for information modelling than its precursor. OPC UA based information modelling was required before the performance index values and the configuration parameters could be stored to the database. The software environment including UA server needed to be updated before the deployment of the performance assessment tool. In addition, the economic inputs and outputs of the test process were defined and related objects were created to the database. The implemented performance assessment tool included a performance calculation application as a Windows service and a graphical user interface for the configuration of the performance calculations.

The functionality of the implemented control performance assessment tool was tested with a simulator of a distillation unit and an actual MPC controller. Different simulation cases were used to verify the operation of performance calculation application. The results showed that the calculated performance indices responded as expected when the process conditions or control objectives changed. Correspondingly, the performance indices remained at a constant level when there were no changes in the process or control objectives.

Improvement to both performance calculation application and graphical user interface is needed although the created performance assessment tool worked well. Additional simulation tests are also required to verify that the functionality of the tool is as desired in situations that were not included in the test cases of this thesis. In order to verify the generality of the performance assessment tool, testing with various processes is also required besides different simulation cases. After the tool is verified to work properly, it can be deployed to a real process environment as a part of the controller software for the assessment of the controller's performance.

REFERENCES

[1] X. Cai, P. Sun, J. Chen, L. Xie, ILC strategy for progress improvement of economic performance in industrial model predictive control systems, Journal of Process Control, Vol. 24, No. 12, 2014, pp. 107-118.

[2] N. Agarwal, B. Huang, E.C. Tamayo, Assessing Model Prediction Control (MPC) performance. 2. Bayesian approach for constraint tuning, Industrial and Engineering Chemistry Research, Vol. 46, No. 24, 2007, pp. 8112-8119.

[3] M. Jelali, Control Performance Management in Industrial Automation - Assess ment, Diagnosis and Imrovement of Control Loop Performance, Springer London, London, 2013, 478 p.

[4] M. Bauer, I.K. Craig, Economic assessment of advanced process control - A survey and framework, Journal of Process Control, Vol. 18, No. 1, 2008, pp. 2-18.

[5] J. Yu, S.J. Qin, Statistical MIMO controller performance monitoring. Part II: Performance diagnosis, Journal of Process Control, Vol. 18, No. 3-4, 2008, pp. 297-319.

[6] J. Schäfer, A. Cinar, Multivariable MPC system performance assessment, monitoring, and diagnosis, Journal of Process Control, Vol. 14, No. 2, 2004, pp. 113-129.

[7] T.J. Harris, Assessment of control loop performance, Canadian Journal of Che mical Engineering, Vol. 67, No. 5, 1989, pp. 856-861.

[8] B. Huang, S.X. Ding, N. Thornhill, Alternative solutions to multi-variate control performance assessment problems, Journal of Process Control, Vol. 16, No. 5, 2006, pp. 457-471.

[9] Control Performance Monitor, Honeywell, web page. Available (accessed 18.01.2016): https://www.honeywellprocess.com/en-US/explore/products/advanced-applications/profit/Pages/control-performance-monitor.aspx.

[10] Aspen Watch Performance Monitor, Aspen Technology, web page. Available (accessed 18.01.2016): http://www.aspentech.com/products/advanced-process-control/ aspen-watch-performance-monitor/.

[11] Loop Performance Monitoring Service, ABB, web page. Available (accessed 18.01.2016): http://new.abb.com/process-automation/process-automation-service/advanced-services/control-loop-services/loop-performance-service.

[12] J. Oksanen, Performance Assessment of the Multivariable MPC Controller, Aalto University, Otaniemi, 2012.

[13] S.J. Qin, T.A. Badgwell, A survey of industrial model predictive control technology, Control Engineering Practice, Vol. 11, No. 7, 2003, pp. 733-764.

[14] M.G. Forbes, R.S. Patwardhan, H. Hamadah, R.B. Gopaluni, Model predictive control in industry: Challenges and opportunities, 9th IFAC Symposium on Advanced Control of Chemical Processes, ADCHEM 2015, 7 June 2015 through 10 June 2015, IFAC Secretariat, pp. 531-538.

[15] M.L. Darby, M. Nikolaou, MPC: Current practice and challenges, Control Engineering Practice, Vol. 20, No. 4, 2012, pp. 328-342.

[16] B. Huang, R. Kadali, Dynamic Modeling, Predictive Control and Performance Monitoring, Springer London, London, 2008, 240 p.

[17] J.B. Rawlings, Tutorial: Model predictive control technology, Proceedings of the 1999 American Control Conference (99ACC), 2 June 1999 through 4 June 1999, IEEE, Piscataway, NJ, United States, pp. 662-676.

[18] F. Xu, B. Huang, S. Akande, Performance assessment of model pedictive control for variability and constraint tuning, Industrial and Engineering Chemistry Research, Vol. 46, No. 4, 2007, pp. 1208-1219.

[19] M. Morari, J. H. Lee, Model predictive control: Past, present and future, Computers and Chemical Engineering, Vol. 23, No. 4-5, 1999, pp. 667-682.

[20] R.S. Patwardhan, S.L. Shah, Issues in performance diagnostics of model-based controllers, Journal of Process Control, Vol. 12, No. 3, 2002, pp. 413-427.

[21] Q. Yuan, B. Lennox, M. McEwan, Analysis of multivariable control performance assessment techniques, Journal of Process Control, Vol. 19, No. 5, 2009, pp. 751-760.

[22] C. Zhao, Y. Zhao, H. Su, B. Huang, Economic performance assessment of advanced process control with LQG benchmarking, Journal of Process Control, Vol. 19, No. 4, 2009, pp. 557-569.

[23] R.H. Julien, M.W. Foley, W.R. Cluett, Performance assessment using a model predictive control benchmark, Journal of Process Control, Vol. 14, No. 4, 2004, pp. 441-456.

[24] S.L. Shah, R. Patwardhan, B. Huang, Multivariate controller performance analysis: Methods, applications and challenges, Chemical Process Control - VI, Assessment and New Directions for Research: Proceedings of the 6th International Conference on Chemical Process Control, 7 January 2001 through 12 January 2001, pp. 190-207.

[25] M. Jelali, An overview of control performance assessment technology and industrial applications, Control Engineering Practice, Vol. 14, No. 5, 2006, pp. 441-466.

[26] Y. Zhang, M.A. Henson, A performance measure for constrained model predictive controllers, 1999 European Control Conference, ECC 1999, 31 August 1999

through 3 September 1999, Institute of Electrical and Electronics Engineers Inc., pp. 918-923.

[27] C. Li, B. Huang, D. Zheng, F. Qian, Multi-input-multi-output (MIMO) control system performance monitoring based on dissimilarity analysis, Industrial and Engineering Chemistry Research, Vol. 53, No. 47, 2014, pp. 18226-18235.

[28] Z. Yan, C. Chan, Y. Yao, Multivariate control performance assessment and control system monitoring via hypothesis test on output covariance matrices, Industrial and Engineering Chemistry Research, Vol. 54, No. 19, 2015, pp. 5261-5272.

[29] M. Duan, N. Li, S. Li, A modified LQG benchmark for economic performance assessment of model predictive control, Proceedings of the 33rd Chinese Control Conference, CCC 2014, 28 July 2014 through 30 July 2014, IEEE Computer Society, pp. 7811-7816.

[30] Z. Sun, S.J. Qin, A. Singhal, L. Megan, Performance monitoring of model-predictive controllers via model residual assessment, Journal of Process Control, Vol. 23, No. 4, 2013, pp. 473-482.

[31] J. Gao, R. Patwardhan, K. Akamatsu, Y. Hashimoto, G. Emoto, S.L. Shah, B. Huang, Performance evaluation of two industrial MPC controllers, Control Engineering Practice, Vol. 11, No. 12, 2003, pp. 1371-1387.

[32] T.F. Edgar, J. Hahn, Process Automation, in: S.Y. Nof (ed.), Handbook of Automation, Springer-Verlag, Berlin Heidelberg, 2009, pp. 529-543.

[33] B. Galloway, G.P. Hancke, Introduction to industrial control networks, IEEE Communications Surveys and Tutorials, Vol. 15, No. 2, 2013, pp. 860-880.

[34] T. Wagner, An Agent-Oriented Approach to Industrial Automation Systems, in: J.G. Carbonell, J. Siekmann, R. Kowalczyk, J.P. Müller, H. Tianfield, R. Unland (ed.), Agent Technologies, Infrastructures, Tools, and Applications for E-Services, Springer Berlin Heidelberg, Berlin, 2003, pp. 314-328.

[35] TC4 - Communication, PLCOpen, web page. Available (accessed 10.02.2016): http://www.plcopen.org/pages/tc4_communication/.

[36] O. Rask, Informaatiojärjestelmät, course material of ASE-2710 Automaatiojärjestelmät ja -suunnittelu, Tampere University of Technology, Tampere, 2012.

[37] C.E. Pereira, P. Neumann, Industrial Communication Protocols, in: S.Y. Nof (ed.), Springer Handbook of Automation, Springer Berlin Heidelberg, Dordrecht, 2009, pp. 981-999.

[38] P. Zhang, Industrial Control Technology, William Andrew Inc., Norwich, New York, 2008, 865 p.

[39] W. Mahnke, S. Leitner, M. Damm, OPC unified architecture, Springer Berlin Heidelberg, Berlin, 2009, 339 p.

[40] Unified Architecture, OPC Foundation, web page. Available (accessed 10.03.2016): https://opcfoundation.org/about/opc-technologies/opc-ua/.

[41] OPC Unified Architecture Specification, Part 1: Overview and Concepts, Release 1.03, OPC Foundation, 2015.

[42] OPC Unified Architecture Specification, Part 4: Services, Release 1.03, OPC Foundation, 2015.

[43] W. Mahnke, A. Gössling, M. Graube, L. Urbas, Information modeling for middleware in automation, 2011 IEEE 16th Conference on Emerging Technologies and Factory Automation, ETFA 2011, 5 September 2011 through 9 September 2011.

[44] OPC Unified Architecture - The universal communication platform for standardised information models, OPC Foundation, web page. Available (accessed 11.04.2016): https://opcfoundation.org/wp-content/uploads/2014/05/OPC-UA_CollaborationOverview_EN.pdf.

[45] OPC Unified Architecture Specification, Part 3: Address Space Model, Release 1.03, OPC Foundation, 2015.

[46] P. Gruhn, Human Machine Interface (HMI) design: The good, the bad, and the ugly (and what makes them so), 66th Annual Instrumentation Symposium for the Process Industries, 25 January 2011 through 27 January 2011.

[47] NAPCON Controller 8.0 Reference Manual, Neste Jacobs Oy, Porvoo, 2015, 303 p.

[48] A. Pelkola, APC strategy description of a distillation process unit, Neste Jacobs Oy, Porvoo, 2014.