On the correlation between Turbidity, Conductivity and COD

ing. G. Tommassen

30 July 2014

Delft University of Technology

Challenge the future

On the correlation between Turbidity, Conductivity and COD

Gerben Tommassen

for the degree of:
Master of Science in Civil Engineering

Date of submission: 30 July 2014
Date of defense: 7 August 2014

Committee:

prof.dr.ir. F.H.L.R. Clemens	Delft University of Technology Sanitary Engineering Section
dr.ir. J.L. Korving	Delft University of Technology Mathematical Physics Section
ir. W.M.J. Luxemburg	Delft University of Technology Water Resource Management Section

Sanitary Engineering Section, Department of Water Management
Faculty of Civil Engineering and Geosciences
Delft University of Technology, Delft

Preface

This report is an MSc thesis of the Delft University of Technology, faculty of Civil Engineering and Geosciences, department of water management, section of sanitary engineering. It is the result of a research on the correlation between turbidity, conductivity and COD in the sewer system of Ulvenhout, The Netherlands. The objective of the research is to find out whether turbidity and conductivity measurements can be used as a proxy for COD measuerements. The research was conducted in the framework of the Msc thesis at TU Delft and with help from the Water Board Brabantse Delta.

Many people earn my gratitude for their help on this research. I will mention the ones that I would like to thank the most. The laboratory staff from the Faculty of Civil Engineering and Geosciences who helped me to get the measuring equipment operational, especially Tonny Schuit, Sander de Vree and Paul Vermeulen. Although the equipment didn't work in the end, I still want to thank them for their efforts.

From the Water Board Brabantse Delta, I would like to thank Frans Schouwenaars and Etteke Wypkema for supporting the project allocating a part of the budget to finance the samples that are the foundation of my research and Kees Smeekens for giving me access the Ulvenhout pumping station. Dio van de Ruit from Aquon institute for water research for the organization of the analysis of the taken samples. The amateur meteorologist Eric Kramers, who has kindly allowed me to use his measurement data. People like Eric invest a lot of their time, funds and hard work to collect good data without expecting anything in return.

My supervisors François Clemens, Hans Korving and Wim Luxemburg have been very important to me. Not only have they given me guidance and assistance, but they also motivated and supported me when I needed it. Finally I would like to thank my friends and family who have supported me during the research for this thesis.

Gerben Tommassen
Breda, 30 July 2014

Summary

COD is one of the most important parameters for wastewater quality. Measuring COD is an expensive and time consuming process. There is a need for an inexpensive, accurate, reliable and continuous technique to monitor COD. Turbidity can be used as a proxy for COD, but is not accurate enough for some applications. Some research suggests that conductivity measurements can be used to improve the turbidity prediction of the COD. This has led to the research question: "Can combined turbidity and conductivity measurements be used to accurately estimate the COD of wastewater in a Dutch sewer?"

To research this fifty samples were taken from the sewer system of Ulvenhout, The Netherlands during three storm events in 2012. The samples are analyzed in a laboratory for COD concentration, turbidity and specific EC. The results are analyzed with MATLAB to find a correlation between turbidity, conductivity and COD. Twelve different fitting formulas were tested using a Levenberg-Marquart algorithm to find the best relation.

It is concluded that measurements of conductivity and turbidity can be used to estimate the COD of wastewater to some extent. The r^2 of the correlation found during this research is only about 0.65-0.69. The turbidity is dominant in the estimation of the COD and the added value of the conductivity measurements is limited. Although the conductivity could be used to make the fit a bit better, the improvement is normally not worth the effort of the additional measurements and may even be insignificant. The highest correlation is found with quadratic fits, but the r^2 of linear fits is only slightly lower then the r^2 of quadratic fits. Since the linear fit of turbidity to COD (without conductivity) has a similar r^2, but is simpler, the use of the linear fit is preferred in many cases. The results of the study are only partly in line with literature.

Samenvatting

CZV is een van de belangrijkste parameters voor afvalwaterkwaliteit. Het meten van CZV is een duur en tijdrovend proces. Er is behoefte aan een techniek om CZV goedkoop, nauwkeurig, betrouwbar en continu te meten. Troebelheid kan gebruikt worden als proxy voor CZV, maar dit is niet nauwkeurig genoeg voor sommige toepassingen. Er is onderzoek dat aangeeft dat geleidbaarheidsmetingen gebruikt kunnen worden om de voorspelling van troebelheid voor CZV te verbeteren. Dit heeft geleid tot de onderzoeksvraag: "Kunnen gecombineerde troebelheids- en geleidbaarheidsmetingen gebruikt worden om het CZV van afvalwater in een Nederlands riool nauwkeurig te voorspellen?"

Om dit te onderzoeken zijn tijdens drie neerslaggebeurtenissen in 2012 vijftig monsters genomen uit het rioolsysteem van Ulvenhout. De monsters zijn in een laboratorium geanalyseerd op CZV-concentratie, troebelheid en soortelijk elektrische geleidbaarheid. De resultaten zijn geanalyseerd met MATLAB om een correlatie te vinden tussen troebelheid, geleidbaarheid en CZV. Twaalf verschillende formules zijn getest met een Levenberg-Marquart algoritme om het beste verband te vinden.

Er is geconcludeerd dat troebelheid en geleidbaarheid tot op zekere hoogte gebruikt kunnen worden om het CZV van afvalwater te schatten. De r^2 van de tijdens het onderzoek gevonden correlatie is slechts 0,65-0,69. De troebelheid is dominant in de schatting van het CZV en de toegevoegde waarde van de geleidbaarheidsmetingen is beperkt. Hoewel de geleidbaarheid gebruikt kan worden om het verband iets beter te maken, weegt de verbetering normaal gesproken niet op tegen de inzet benodigd voor de extra metingen en is deze mogelijk zelfs niet significant.

De hoogste correlatie is gevonden bij kwadratische verbanden, maar de r^2 van lineaire verbanden is slechts iets lager dan de r^2 van kwadratische verbanden. Omdat het lineaire verband tussen troebelheid en CZV (zonder geleidbaarheid) een vergelijkbare r^2 heeft, maar veel simpeler is, heeft het lineaire verband in veel gevallen de voorkeur. De resultaten van het onderzoek zijn slechts deels in lijn met de literatuur.

Content

List of Illustrations

List of Tables

Abbreviations

A	Ampere
BOD	Biochemical Oxygen Demand
BOD_5	Biochemical Oxygen Demand measured over 5 days
BOD_7	Biochemical Oxygen Demand measured over 7 days
C	specific EC
C_{cal}	specific electrical conductivity at the calibration temperature
c_{clay}	concentration of Goudse Clay
$c_{DO:i}$	initial dissolved oxygen concentration
$c_{DO;f}$	final dissolved oxygen concentration
c_{NaCl}	concentration of sodium chloride
COD	Chemical Oxygen Demand
COD_{eq}	calculated COD equivalent concentration
$COD_{eq;f}$	calculated fictive COD equivalent concentration
COD_f	fictive COD
CSO	Combined Sewer Overflow
DO	Dissolved Oxygen
DUT	Delft University of Technology
EC	Electrical Conductivity
FAU	Formazin Attenuation Unit
FNU	Formazin Nephelometric Unit
FTU	Formazin Turbidity Unit
G	conductivity
H_2S	Hydrogen sulfide
I	current
JTU	Jackson Turbidity Unit
NAP	Dutch Datum Level (Dutch: Nieuw Amsterdams Peil)
NTU	Nephelometric Turbidity Unit
p	dilution factor
PAH	Polycyclic Armomatic Hydrocarbons
ppm	parts per million
PS	Pumping station
R	electrical resistance or resistivity
RMSE	Root Mean Squared Error
RTC	Real Time Control
S	Siemens
SAC	Spectral Absorption Coefficient
SSE	Sum Squared Error
STT	Small Tube Test
SUDS	Sustainable Urban Drainage Systems
T	Turbidity
T_{meas}	measured temperature
T_{cal}	calibration temperature
TDS	Total Dissolved Solids
TSS	Total Suspended Solids
UV/VIS	Ultraviolet visible spectrometry
V	volume of clay suspension in the sample
V_{clay}	potential difference
V_{naCl}	volume of clay suspension in the sample
V_{sample}	volume of salt solution in the sample
WWTP	volume of sample
α	temperature compensation parameter

1. Introduction

In this thesis the use of turbidity and specific electrical conductivity (EC) measurements as a substitute for the sampling and analyzing of the Chemical Oxygen Demand (COD) of wastewater are discussed.

In section 1.1 an introduction to urban drainage is given. Section 1.2 is on the information needed for sewer system management. The objective of this thesis is described in section 1.3. In section 1.4 the organization of this research project is briefly shown. Section 1.5 is a reading guide for this thesis.

1.1. Introduction to urban drainage

1.1.1. History of urban drainage

Urban drainage was already used by the Mesopotamians 3.500-2.500 years BC (Mudde, 2012). Ancient civilizations such as the Minoans and Greeks also used urban drainage. The Romans built impressive systems, like the Cloaca Maxima (built around 616 B.C.), to drain there large cities effectively, but most of this knowledge was lost in the Dark Ages. When Dr. John Snow discovered the link between water and cholera in 1854 in London the importance for sanitation became clear to the modern world (Mudde, 2012). As a result the construction of sewer systems began. First in the major cities and later in the smaller cities sewer systems were constructed. The first sewer systems just transported sewage out of the cities and discharged it into rivers. Later on, the sewage was treated before it was discharged on rivers. Environmental concerns and legislation have caused an increasingly large portion of the households to become connected to the sewer system. In the Netherlands the Law on Pollution of Surface Waters (Ministry of Infrastructure and Environment,1970) and later the Water Law (Ministry of Infrastructure and Environment, 2009) resulted in the connection to sewers and treatment of wastewater. The treatment of sewage is extended and sewer systems are improved to prevent extensive pollution of the environment.

1.1.2. Types and operation of urban drainage

Most sewer systems are combined sewer systems. Foul water and storm water are collected in the same conduits of the system. The main alternatives to the combined sewer system are the separate sewer system and the so called improved separate sewer system. In the separate sewer system foul and storm water are each collected in a separate system. In an improved separate sewer system foul and storm water are also collected in a separate system, but a portion of the storm water is pumped to the foul water system.

The conduits of a combined sewer system are placed under a slope towards the lowest point of the system, where a pumping station is constructed. In this way, the collected wastewater is transported towards a sewage pumping station under gravity flow. The pumping station transports the wastewater towards the wastewater treatment plant (WWTP). If the hydraulic load is too high due to a storm event, the pumping station is not able to transport all the water to the WWTP. As a result the water level in the sewer system rises, filling the storage of the system. If the storage capacity of the system is filled and the hydraulic load continues to exceed the pumping capacity of the system, the wastewater spills onto the surface water through combined sewer overflows (CSO's). If the hydraulic load becomes less then the pumping capacity, the storage of the sewer system is emptied by the pumping station. A schematic impression of the combined sewer system is shown in figure 1.1.

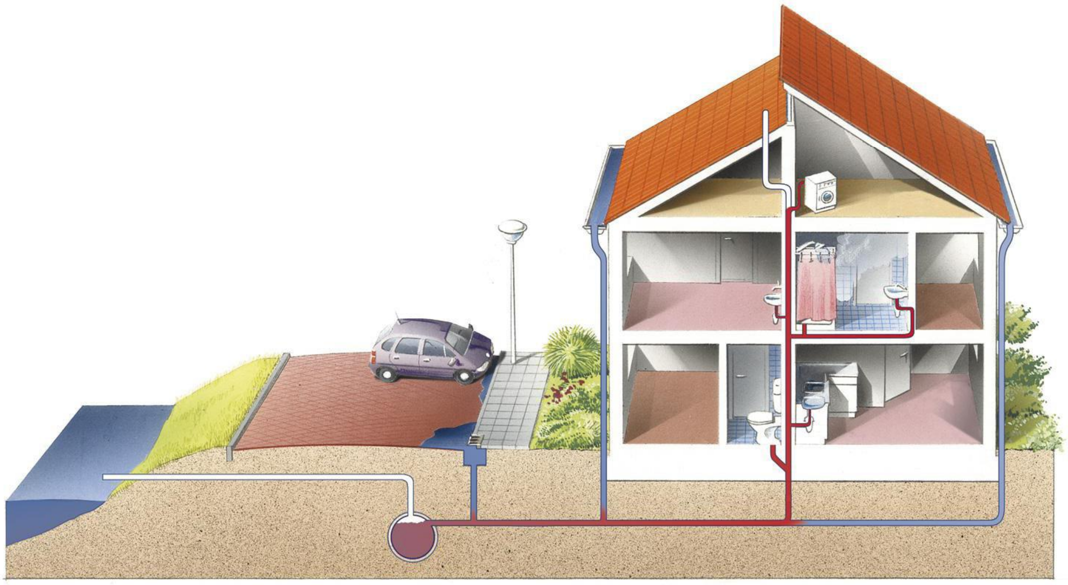


Figure 1.1 - Schematic impression of combined sewer system (source: Paul Maas Tilburg/ Rioned Foundation)

The foul water system of a sewer system. Foul water systems do not have CSO's, but may have emergency overflows to the surface water or the storm water system to discharge storm water from illicit connections. The collected storm water of a separate system is discharged onto the surface water or used to complement the local hydrology or reduce water use.

The slope of the conduits of a storm water system of a separate sewer system may be towards the discharge point, away from the discharge point or non existent. A disadvantage of separated systems is that illicit connections may result in foul water being directly discharged to the surface water. Figure 1.2 represents a schematic impression of a separate sewer system.

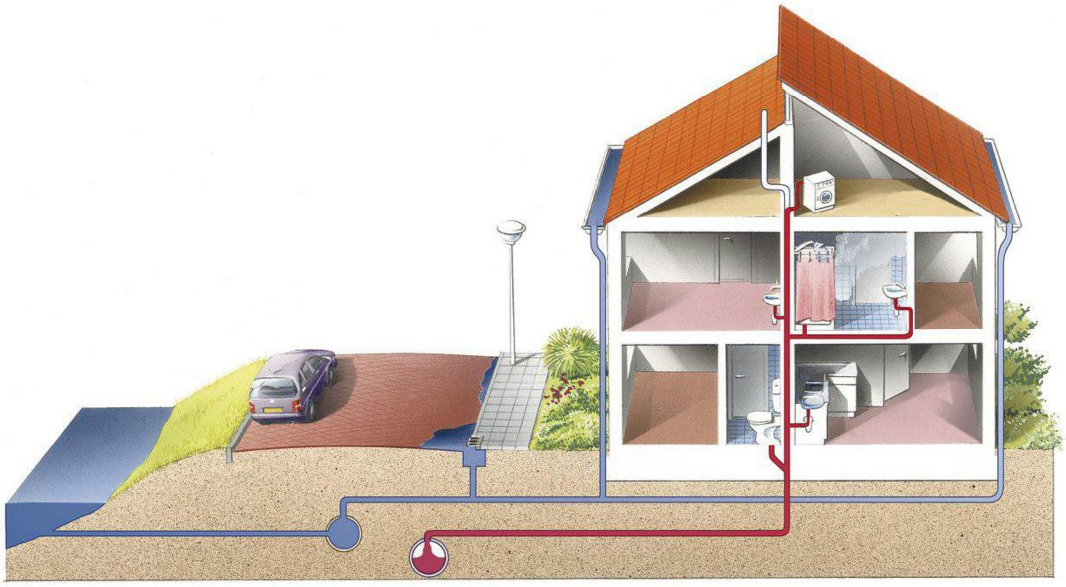


Figure 1.2 - Schematic impression of separate sewer system (source: Paul Maas Tilburg/ Rioned Foundation)

The storm water system of a so called improved separate sewer system functions like a combined sewer system. The improved separate sewer system discharges a portion of the water collected by the storm water system to the foul water system to prevent pollution by illicit connections and possibly waste carried in the disputed 'first flush' of storm water. The storage capacity and pumping capacity of an improved separate sewer system is often considerably smaller then the storage capacity and pumping capacity of a combined sewer system. Most of the storm water is drained to the WWTP, but peak loads at the WWTP are much smaller.

When wastewater is discharged through CSO's, wastewater pollutes the surface water in several ways. Apart from suspended solids, heavy metals, chemicals, nutrients and pathogens, the wastewater has an oxygen demand. Organic compounds in the wastewater are biologically degraded with the use of oxygen. If the oxygen demand is high, the amount of dissolved oxygen in the surface water drops and the surface water becomes hypoxic. A part of the aquatic life, such as most types of fish, cannot live in water that is hypoxic. Fish mortality starts when the oxygen concentration sinks below 2-3 mg/l, especially at high water temperatures. (Ven, van der, 2007). Discharges from storm water systems can also pollute the surface water with polluted runoff and water from illicit connections.

Several measures are taken to prevent these polluting discharges. The pumping capacity or storage capacity of combined and improved separate sewer systems can be increased to make discharges from occur less often. Reducing the drained paved area increases the relative pumping and storage capacity of the system. Sustainable Urban Drainage Systems (SUDS) is a collective name for a group of urban drainage techniques that are meant to drain developed areas in a more natural way (Butler and Davies, 2000). SUDS include techniques like infiltration trenches, pervious pavements and green roofs. Storage settling tanks and several types of filters are constructed at discharge points. Storage settling tanks allow additional storage of wastewater and settling of particles in the wastewater to reduce the frequency of discharges from CSO's and the concentration of pollutants in the discharged wastewater. Filters improve the quality of the discharged water.

1.1.3. Challenges in urban drainage

Urban drainage faces some major technical challenges. A large part of the sewer system in developed countries has to be replaced in the coming decade requiring huge investments.

The pollution of surface water through CSO's and other discharge points is already mentioned in section 1.1.2. This includes the discharge points of WWTP´s which pollute the water because WWTP´s normally do not treat the wastewater up to a level that is similar to the (desired) surface water quality.

Rainfall that is collected through paved surfaces and the sewer system is drained faster compared to a rural situation. This results in higher water levels in the sewer system and the receiving water bodies. The higher water levels can result in flooding and, combined with pollution, public health risks.

Addressing the challenges is traditionally done with investments such as the ones mentioned in section 1.1.2. The resources for these investments are limited and governments tend to reduce the resources even further. On the other hand, it is possible to reduce the required investments. Smarter management and cooperation increases the efficiency of the investments and operational costs. This could lead to a cost reduction of 13% in the Netherlands (Feitenonderzoek Commission, 2010).

1.2. Information for sewer system management

1.2.1. General

Like most man made complicated systems, sewer systems need management to function properly. Sewer systems in the Netherlands are managed by municipalities by law. In some cases municipalities have outsourced the sewer system management to separate companies or institutions such as the water cycle company Waternet in the city of Amsterdam and Aquario in the province of Friesland. The WWTP and in many cases the pressurized transportation from the free flow system to the WWTP is managed by a water board. The management of a sewer system consists of several tasks such as the cleaning, inspection, rehabilitation or replacement of damaged parts of the sewer system, improving the system to meet new demands and connecting new urban or rural areas to the system.

Efficient management of an urban drainage system requires decisions about subjects such as measures, replacements, maintenance and so on. These decisions are based on information about the urban drainage system. If only little information is available, management of the system may still be possible, but it is less efficient then management with decisions based on more information. Selecting measures based on measurements instead of just models is essential to keep investments at an acceptable level (Feitenonderzoek Commission, 2010, Langeveld, 2011 and NLingenieurs, 2010)

Municipalities in The Netherlands are obligated by law (Ministry of Infrastructure and Environment, 1979) to make a municipal sewer plan (Dutch: gemeentelijk rioleringsplan) for a set time period. The municipal sewer plan contains at least:

- an overview of the urban drainage network
- an overview of the planned measures and replacements for the set time period
- an overview of the management of the urban drainage
- the environmental consequences of the urban drainage
- an overview of the financial consequences of the described activities

The information required for efficient management can be distinguished into several classes:

- information on the physical aspects of objects
- information on the condition of objects
- information on the hydraulic functioning of a sewer system
- information on the water quality in the sewer system.

1.2.2. Information on the physical aspects of objects

The most basic information describes the physical properties of sewer system objects. For sewer conduits this may include for example the location, the shape and size, the insert level and the material of the conduit. When the required information is available for all objects of the sewer system, the sewer manager has an overview of the entire system. Commercial packages for sewer systems are available to store, change, manage and view the information. All Dutch municipalities use digital packages for sewerage management (RIONED Foundation, 2010b). The information is often originally collected from construction or as-built drawings. The object information can change over time and the actual construction is likely to differ a bit from the construction drawings. Constructions subside in weak soil types like peat, resulting in a change in for example the insert level of a conduit or weir. To keep the data up to date measurements to the sewer system can be performed.

Rehabilitation or replacement of a part of the sewer system, connecting new parts to the system or changing for example the weir height to improve the functioning of the system are done on a regular basis. For many municipalities it is difficult to manage the data in a sufficient way. Errors are common in the data and changes in the system are often recorded late or not at all in the information management system. In a 2005 survey about 41% of the Dutch municipalities stated that no revisions had been entered in the sewer management system in the past year (RIONED Foundation, 2005).

With the coming of separate sewer systems, having two or more urban drainage systems in the same street has become increasingly common. The increased use of SUDS has increased the types of objects, properties of objects and system types. Both these developments make it increasingly challenging to keep track of the basic system information.

Additional information, like the year of construction and the type of sewer (combined sewer, storm water sewer, foul water sewer and so on), can be stored together with the physical properties of a sewer object in a digital sewer management system.

1.2.3. Information on the condition of objects

For the management of a system which objects are managed and what these objects look like is not the only information of importance. The condition of these objects is also important. Traffic, subsidence, tree roots, H2S-gas, wear and tear and other influences can change the condition of objects of the sewer system. This can lead to infiltration of sewage into the subsurface, infiltration of groundwater into the sewer, lost storage, reduced flow, changed physical properties of objects and so on. In extreme cases it can even lead to the collapse of a sewer. The Dutch municipalities report 11.3 cases of clogging per 100 km of km of free flow sewer. In 2009 2.3 collapses per 100 km of free flow sewers were reported and pumping stations had about 3.6 failures on average (RIONED Foundation, 2010b).

When the condition of objects is reduced in such a way that the system cannot function properly rehabilitations or replacements of objects have to be executed. A widely used strategy for rehabilitation and replacement of objects is ad hoc. The condition of the system is allowed to reduce until the system fails. Failure is often detected through complaints of citizens. A planned rehabilitation and replacement strategy is less expensive and causes less inconvenience then an ad hoc strategy. In addition, it is possible to combine the sewer construction works with other works, like road works, reducing costs and inconvenience even further.

A planned rehabilitation and replacement strategy is most efficient when it is supported by measurements on the condition of the objects. Information on the condition of objects is gained through inspection of these objects. Between 2006 and 2010, the average inspection cycles was about 14 years in the Netherlands (RIONED Foundation, 2010b). To ensure uniform inspections, inspectors are trained and inspect according to standards like NEN-EN 13508-1/2 en (Nederlands Normalisatie Instituut, 2003a and b). A plan for rehabilitation and replacement can be made based on the inspection results and other considerations like road works. Although the inspections provide helpful information on the condition of the sewer system, it is still not easy to predict the lifetime of a sewer object. The reduction in the condition of an object cannot just be extrapolated to the future and the quality of inspections and the inspection methodology is not always sufficient. The problems with the quality of inspections and the inspection methodology are illustrated by reports of sewer objects increasing instead of reducing in condition between two inspections without rehabilitations or replacements being executed (Dirksen, 2007).

1.2.4. Information on the hydraulic functioning of a sewer system

Information on the functioning of the sewer system is important to determine whether the sewer system functions sufficiently or not. If alterations to the sewer system are required information on the functioning of the system is required to determine which alterations are effective and efficient. The hydraulic functioning of the sewer system is often analyzed with computer models. About 68% of the Dutch municipalities have an up-to-date sewer model (RIONED Foundation, 2010b). The models require a lot of input such as the basic physical object information mentioned in paragraph 1.2.2., connected subcatchment surfaces and many parameters. Errors and inaccuracies in the model input can result in inaccurate calculations.

Validating or calibrating the model with field measurements improves the quality of the model (Clemens, 2001). Water level measurements are most commonly used together with rain gauges because they are inexpensive, relatively easy to perform and accurate. The measurements are obtained in measurement campaigns that usually about one year to get enough data for a good validation or calibration. This period is needed because the water level time series should include rain events that fill the system as well as dry periods. Constant monitoring is also applied. About 88% of the Dutch municipalities have level measurements in their system, but 64% of them do not compare field level measurements to model output (RIONED Foundation, 2010b).

1.2.5. Information on the water quality in the sewer system

For investments aiming to reduce the environmental impact of the sewer system, knowing the hydraulic functioning of the sewer system is not enough. Information on the quality of the water is also required. In current practice most sewer managers use little information on the quality of sewage. Standard values for water quality parameters are used instead of real data. The reason for this approach is twofold. Firstly, the computer models for sewers are not yet capable of sufficiently accurate water quality calculations. Secondly, water quality measurements are difficult to perform and expensive. Therefore water quality measurements are executed seldom.

The most important sewage quality parameter, COD, is measured by taking a sample of the sewage and taking it to a laboratory for analysis. COD is broken down by bacteria over time, so the sample has to be chemically preserved and analyzed quickly or cooled to slow the degradation process down. The main advantage of the method is the accuracy. Disadvantages include the high cost, the resulting low frequency, the long time interval between the measurement and the result and the high demand on man power. UV/VIS-spectrometers correlate the absorbed spectrum to the COD of sewage. The advantage of UV/VIS-spectrometry is the ability to perform continuous real time measurements. The data are less accurate than analyzed samples, but still sufficiently accurate for most objectives. The disadvantage is that an UV/VIS-spectrometer needs a local calibration with analyzed samples to gain this accuracy. Changes in the matrix of the sewage result in the need for a new local calibration. The UV/VIS-spectrometry has high investment costs. Alternative measurement techniques that provide continuous real time water quality information and require lower investment costs would allow more water quality measurements. Turbidity measurements fit these requirements and can be correlated to COD. The disadvantage of turbidity is that the correlation to COD is not sufficiently accurate. This may be due to the part of the COD that is not in particles, but that is dissolved in the sewage. The specific EC is a measure for the amount of dissolve substances. In theory measuring the specific EC as well as the turbidity would increase the correlation to COD.

1.3. Objectives of this thesis

The objective of this thesis is to research if measuring turbidity and specific EC is a good way to collect information on the water quality of sewage by correlating it to COD.

The main research question of this graduation work is: "Can combined turbidity and conductivity measurements be used to accurately estimate the COD of wastewater in a Dutch sewer?"

1.4. Organization of this research project

This thesis is done for the section Sanitary Engineering of the department of Water Management within the faculty of Civil Engineering and Geosciences of the Delft University of Technology. This research project is part of research theme 3, insight in dynamic functioning of the knowledge program Urban Drainage.

The project is supervised by:

- prof.dr.ir. F.H.L.R. Clemens (François), TU Delft, faculty of Civil Engineering and Geosciences, department of water management, section of sanitary engineering
- dr.ir. J.L. Korving (Hans), TU Delft, faculty of Electrical Engineering, Mathematics and Computer Science, department of applied mathematics, section of mathematical physics
- ir. W.M.J. Luxemburg (Wim), TU Delft, faculty of Civil Engineering and Geosciences, department of water management, section of water resource management

1.5. Reading guide

Chapter two introduces the theory for this research based on a review of the available literature on turbidity, specific EC and COD. The chapter contains the properties and measurement of these parameters.

The third chapter describes the way this research project is executed. An extensive description of the laboratory test and the field test that have been executed is given. The chapter also describes the method used for correlating the turbidity and the specific EC to the COD.

In chapter four the results of the laboratory and field tests are given. The result of the analysis of the produced data is also given in this chapter. The results are discussed for better comprehension.

Chapter five presents conclusions of the research project and answers the research question of the thesis. Recommendations for further research and application are also given in this chapter.

2. Literature study

2.1. Introduction

This study focuses on turbidity, specific EC and COD. In this chapter these three measurement parameters are elaborated.

2.2. Water quality

Spills from combined sewer overflows have an influence on the receiving surface water. The spilled combined sewer overflow water contains a number of contaminants that can have a significant negative effect on the receiving surface water. A poor water quality has a negative effect on the aquatic ecology, is a source of nuisance and even a threat to public health. Contaminants that have a significant negative effect on the surface water are for example phosphate, nitrate, oxygen consuming compounds, pathogens, heavy metals, suspended solids and polycyclic aromatic hydrocarbons (PAH's) amongst others (RIONED, 2010a).

For some contaminants the surface water has a buffer capacity that allows the surface water quality to recover naturally after a CSO spill. This means that the CSO spill only has a significant effect when the pollutant load of a CSO spill exceeds the buffer capacity of the surface water. The surface water had a buffer capacity for contaminants like oxygen consuming compounds and pathogens. Other contaminants accumulate in the surface water or in the surface water bottom. These contaminants often do not have a large short term negative effect on the surface water quality.

2.3. Turbidity

2.3.1. General

Water often contains dispersed suspended particles. These particles are in the water, but not dissolved into the water. The particles can include silt, clay, algae and other microorganisms, organic matter and other particles. The dispersed suspended particles cause the water to be cloudy or hazy when looked through. This cloudiness or haziness is caused by the obstruction of light travelling through the water. Through the obstruction the light is partly absorbed and partly scattered in different directions. A part of the light is often allowed to travel through the water without being obstructed by the dispersed suspended particles in the water. The cloudiness or haziness of the water is called the turbidity of the water (Sadar, 1998). The water itself also causes a slight scatter and absorption of the light travelling through the liquid. The scattering and absorption of the water itself is only significant in the clearest of waters. This effect can be neglected in water containing many constituents such as water spilled through CSO's (Davies-Colley, 1993).

Turbidity is a measure for the scattering effects of light by dispersed suspended particles in water. In the international ISO standard (Aumond, 2005 and ISO, 1999) turbidity is defined as the reduction of transparency of a liquid caused by the presence of undissolved matter. Turbidity is often sed as an indication parameter for suspended solids. The turbidity in wastewater ranges from about 20 NTU to 500 NTU or even 3000 NTU (Aumond, 2005, Henckens, 2001, Veldkamp, 2003 and Lombard, 2010).

2.3.2. Turbidity measurements

The method for measuring turbidity depends on the level of the turbidity of the water. For this report only methods that are suitable for highly turbid water are treated, because the wastewater in combined sewer systems is regarded as water with a high turbidity.

The first turbidity measurements date back to 1900 when Whipple and Jackson developed the Jackson Candle Turbidimeter. The Jackson Candle Turbidimeter consisted of glass tube with a burning candle underneath. The glass tube had a flat bottom and an open top. The glass tube was filled with sample water until the light from the candle could no longer be seen. A suspension of 1000 ppm diatomaceous earth and distilled water was used as a reference fluid for the Jackson Candle Turbidimeter. The turbidity measured this way is referred to in terms of Jacksin Turbidity Units (JTU). Other reference fluids could also be created, but this results in problems with consistency. A disadvantage of the Jackson Candle Turbidimeter is the low practicality at low turbidity. With low turbidity, the Jackson Candle Turbidimeter needs to be very large to achieve extinction of the light from the candle. In addition the result of the test was dependent on the researcher to determine the extinction point, making the test less independent and reproducible. The candle flame is an unsuitable light source for some samples because the flame only emits light in a certain portion of the spectrum.

The development of advanced photoelectric detectors allowed better measurement techniques. Advanced photoelectric detectors are highly sensitive to even very small changes in light intensity. The technique that is nowadays most applied is nephelometry. With nephelometry a detector is used to measure the light that is scattered in a 90° angle instead of the light that is transmitted through the sample. The light scattered in that angle is related to the turbidity of the sample. The advantage of nephelometry is that it is not limited by low or high turbidity. When measuring transmitted light through a low turbidity sample, the change in light is too small to detect, since the signal is insignificant to that of electric noise. At high turbidity multiple scattering interferes with direct scattering when measuring transmitted light. Turbidity that is measured with the nephelometer is referred to in nephelometric turbidity unit (NTU) or formazin nephelometric unit (FNU). These units are similar to formazin turbidity units (FTU) and formazin attenuation units (FAU), which are also used to express turbidity, but not to JTU. The principle of nephelometric turbidity measurement is shown in figure 2.1.

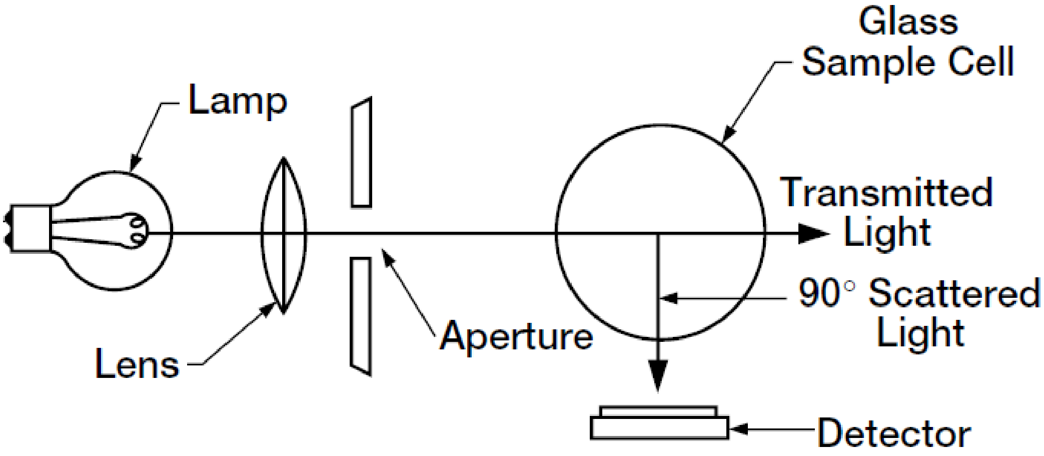


Figure 2.1 - Principle of nephelometric turbidity measurement (source: BIO-DESIGN for the REAL WORLD). Light from a lamp is send through a sample. Turbidity of the sample causes scattering of the light. At a 90°-angle a detector catches the scattered light, which is related to the nephelometric turbidity

The nephelometric turbidity is measured as a reflection coefficient, which is influenced by the properties of the suspended particles (shape, color, structure, size), the environmental (temperature, other light sources, hydraulic situation) and the measurement set up (reflection from objects, orientation regarding to flow direction). The turbidity can be seen as a measure for the amount of formazin that has to be suspended to achieve the same reflection coefficient (Henckens, 2002). Formazin is widely accepted as the primary reference standard for turbidity measurements. Formazin is preferred because it can be created anywhere from raw materials. When the polymer is created it randomly forms perfect white particles in various different shapes and sizes. The distribution in shapes and sizes is wide and irregular, but the distribution is statistically constant. Therefore formazin is a liquid with reproducible characteristics (Sadar, 1998).

The installation of turbidity sensors should meet certain requirements to reduce influences from measurement set up on the turbidity readings. The requirements are a free space around the sensor to reduce reflections from other objects, an orientation in the direction of flow and a location without excessive turbulence to reduce the formation of air bubbles in the water (Henckens, 2002). Measures to prevent pollution are important to reduce failure (Schellart, 2003). Effort is nonetheless important to bring the reliability over 70% (Schellart, 2003).

2.4. Conductivity

2.4.1. General

Dissolved solids in water increase the capability of water to conduct electrical current. Pure distilled water is able to conduct electrical current, but the resistivity of the water is so high that it is insignificant compared to water containing dissolved solids. The specific EC is often used as a parameter for the amount of dissolved solids in a liquid. The electrical conductivity often increases as the concentration of dissolved solids increases. For some high concentration substances (e.g. high concentrations of sulphuric acid) the electrical conductivity does not due to ionic interactions. The specific EC of wastewater ranges from 150 µS/cm to and 2000 µS/cm and even up to 500 to 4500 mS/cm (Levlin, 2007, Schilperoort, 2007a and Lombard 2010).

2.4.2. Conductivity measurements

Conductivity is measured by placing two electrodes in the sample liquid. A potential difference is created between these two electrodes resulting in a current from one electrode to the other. The working principle of electrical conductivity in an ion solution is shown in figure 2.2.

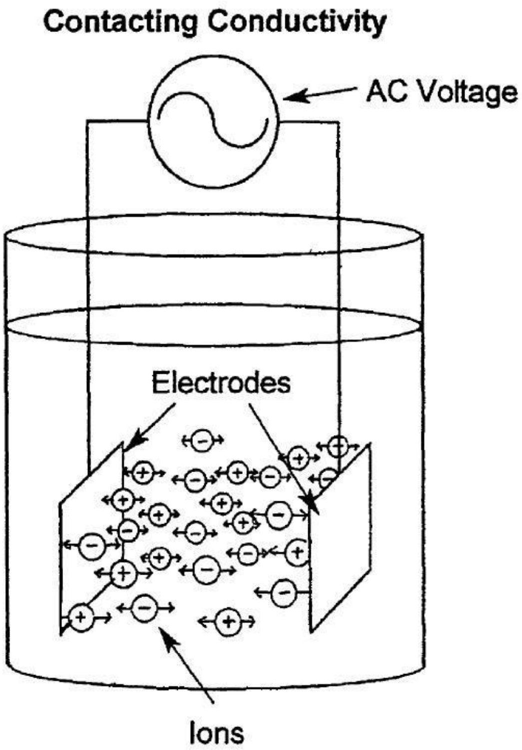


Figure 2.2 - Principle of electrical conductivity measurement (source: Rosemount Analytical). A potential difference is created between two electrodes in a sample. The resistivity can be calculated from the potential difference and the current between the electrodes. The conductivity is the inverse of the resistivity

The potential difference and the current can be measured. When the potential difference and current are known, the electrical resistance or resistivity can be calculated with Ohm's law:

$$R = \frac{V}{I}$$ *(equation 2.1)*

in which:
R = electrical resistance or resistivity (Ω)
V = potential difference (V)
I = current (A)

The conductivity is the inverse of the resistivity:

$$G = \frac{1}{R} = \frac{I}{V}$$ *(equation 2.2)*

in which:
G = conductivity (S)

The electrical conductivity is measured in Siemens (S). Dimensions of the measurement device influence the conductivity measurement. Therefore every conductivity meter has a cell constant. The cell constant is a measure for the influence of the conductivity meter dimensions and is defined as length parted by the electrode area. So if the electrodes are 1 cm apart and have an area of 1 cm by 1 cm, the cell constant is $\frac{1\text{cm}}{1\text{cm}^2} = 1\text{cm}^{-1}$. The cell constant is used to calculate the specific electrical conductivity: the electrical conductivity multiplied by the cell constant. Dimensions are Siemens per (centi-)meter (S/m or S/cm).

When the temperature rises, ions are able to move more freely in the liquid. This increases the ability of the ions to transport electrical charge through the liquid. As a result the measured specific electrical conductivity is higher then the measured specific electrical conductivity of a cooler sample. Since the specific electrical conductivity is used as a characteristic of the liquid, the parameter should be temperature independent for use. Usually the conductivity meter is calibrated at certain temperature (usually 25°C). Measurements are compensated to show the electrical conductivity at the reference temperature with:

$$C = C_{cal} \cdot (1 + \alpha \cdot (T_{meas} - T_{cal})) \qquad \text{(equation 2.3)}$$

In which:
C = measured specific electrical conductivity (μS/cm)
C_{cal} = specific electrical conductivity at the calibration temperature (μS/cm)
α = temperature compensation parameter (S)
T_{meas} = measured temperature (°C or K)
T_{cal} = calibration temperature (°C or K)

The temperature compensation parameter is dependent on the sample substance, concentration and temperature. This parameter can vary greatly for different substances. As a result problems may arise when the characteristics of the sample substance are not known exactly as with wastewater. Almost all electrical conductivity meters include a thermometer. The temperature compensation parameter is adjustable or fixed. Meters with a fixed temperature compensation parameter usually have it at 2%/°C, which is close to the temperature compensation parameter for NaCl solutions.

2.5. Oxygen Demand

2.5.1. General

Wastewater contains compounds that can be oxidized by aerobic biological organisms and chemical processes in the water. When water with a demand for oxygen is discharged to the surface water, the surface water becomes hypoxic. An important part of the aquatic life, such as most types of fish, cannot live in water that is hypoxic. Fish mortality starts when the oxygen concentration sinks below 2-3 mg/l, especially at high water temperatures. (Ven, van der, 2007) The oxygen that is needed to oxidize these compounds is called the oxygen demand of the water.

A good way to measure the oxygen demand is by measuring the biochemical oxygen demand (BOD) of water. The BOD is measured by storing a (diluted) sample of the water in fixed laboratory conditions (like a temperature of usually 20°C and darkness) for a number of days (e.g. 5 days for BOD5 and 7 days for BOD7). Microorganisms and chemical processes in the sample break down organic compounds in the sample consuming dissolved oxygen (DO). By measuring the DO concentration at the beginning and the end of the test the BOD can be calculated:

$$BOD = \frac{(c_{DO;I} - c_{DO;F})}{p} \qquad \text{(equation 2.4)}$$

in which:
$c_{DO;I}$ = the initial dissolved oxygen concentration (mg/l)
$c_{DO;F}$ = the final dissolved oxygen concentration (mg/l)
p = the dilution factor (-)

Seed microorganisms may be added if insufficient are available. One of the main disadvantages of determining the BOD is the long time required for the test (Butler, 2004). Alternatively the Chemical Oxygen Demand can be determined. With COD analysis the organic matter is oxidized using a strong chemical oxidizing agent. The oxidizing agent speeds up the oxidizing process allowing results to be presented in about 2,5 hours or even faster (Tchobanoglous, 2003). A disadvantage of COD is that compounds are oxidized that are not degraded under normal conditions. In the case of urban drainage, this disadvantage is outweighed by the advantage of quick results. This makes COD the most important guiding parameter for the water quality (CUWVO 1992, WRW 1996 and CIW, 2001).

The COD is partly held by the particles suspended in the water and partly dissolved in the water. During dry weather flow both the suspended and dissolved fraction hold an important part of the COD, but during storm water conditions the dissolved fraction is predominant (Schilperoort, 2007 and Flamink, 2004).

The parameter COD is not just a property of the water. It also implies the method used to measure the oxygen demand. In this report COD is regarded as a property parameter, which can also be measured indirectly.

2.5.2. Measuring COD

There are several techniques available for measuring COD. The measuring principle is very different from the original COD sampling method for some techniques. Upcoming techniques with good results are small tube tests (STT) and UV-visible spectrometry. The high costs of traditional as well as upcoming techniques prevents that COD measurements are widespread used.

2.5.2.1 Measuring frequency

Some measuring techniques allow higher measuring frequencies then others. When the measuring frequency is increased, the measurements are able to follow changes in the water quality more accurately. As a consequence, the estimated water quality over a period of time can be accurate, compared to a situation with more accurate, but less frequent measurements. Low frequency measurements can also miss peaks in the water quality or have peaks labeled as an erroneous reading because it is just a single reading that resembles an outlier. Research shows that a frequency of about one measurement per minute allows the capture of most processes in the sewer (Henckens, 2002, Veldkamp, 2003, Lombard, 2010 and Schilperoort, 2011).

2.5.2.2 Sampling

COD can be measured by a international standard technique. A sample of the water is taken, a known surplus amount of strong oxidizing agent and acidic medium is added. Typically potassium dichromate is used as oxidizing agent, but potassium permanganate, ceric sulphate and potassium iodate are also used. The agent is allowed to oxidize all organic compounds in about 2 or 3 hours at a fixed temperature (typically 20°C). After this time the remaining oxidizing agent is measured using titrimetric or spectrophotometric determination. The amount of used oxidizing agent is a measure for the oxygen demand.

An alternative to the traditional sampling and analyzing are small tube tests (STT). The small tube test is similar to the traditional method, with the exception that the sample is taken in a small tube which already contains the oxidizing agent and an acidic medium. Small tube tests still need analyzing but this is less time consuming then with traditional sampling.

The need for lab analysis of samples makes COD measurement a costly and time consuming process making it less desirable for measurement campaigns.

2.5.2.3 Indirect COD measurement using turbidity

The suspended particles in water contain a lot of organic matter, which is responsible for the COD of the water. As mentioned before, turbidity is strongly related to the concentration of dispersed suspended particles in water. As a result measuring the turbidity gives an indication of the COD of water.

The correlation that is found between turbidity and COD is not always high, always site specific and normally the inter-event variability is high (Lacour, 2009, Lombard, 2010 and RIONED, 2002). Research shows that there is a very low correlation during dry weather flow. Lorenz found an r^2 of 0,6 (Lorenz, 2001) and Moens states the correlation is insufficient (Moens, 2001). This restricts the use of turbidity as an indirect COD measurement to a limited number of purposes. During storm water conditions the correlation is higher. Moens found r^2 is 0,83 in one experiment (Moens, 2001) and 0,79 to 0,86 in another experiment together with Veldkamp with a reduced data set. Veldkamp and Moens also found that the results for a complete data set, without statistical reduction, were much lower: r^2 is 0,56. The same correlation is found for both linear and exponential regression and the power in the exponential regression function is close to unity. This suggests that the relation is linear. When four experiments equally divided over two different catchment areas are compared, the regression seems to be comparable within a catchment area, but not between catchment areas. This is shown in figure 2.3 (Veldkamp, 2002).

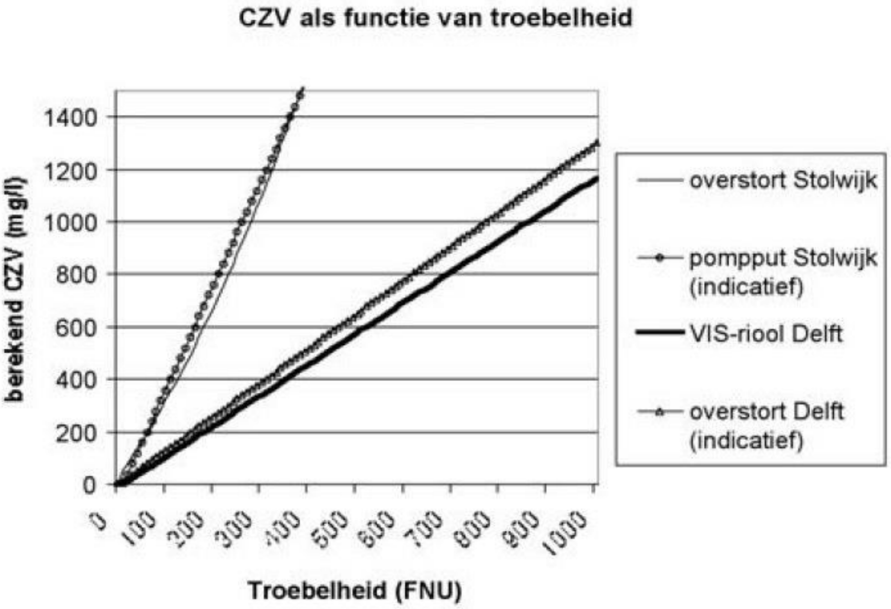


Figure 2.3 - Calculated COD as a function of turbidity at four locations. The regressions at the pump pit of Stolwijk and the CSO in Delft are only indicative. The figure nonetheless shows that the regression within a catchment area is similar, unlike the regression in different catchment areas. The figure shows calculated COD, but the shown fits are based on samples (Veldkamp, 2002)

It has been suggested that turbidity can be used as indicator of pollution load in general (Fletcher, 2007 and Irvine, 2005). Advantages of this method are the low investment cost and the continuous real time nature of the measurements.

2.5.2.4 UV-Probe

An UV-probe measures the absorption and reflectance of ultraviolet light. The light is emitted by a lamp inside the probe using a wavelength of by example 254 nm. The UV-absorption is correlated to the COD based on a calibration with samples. The measured parameter is called spectral absorption coefficient (SAC).

The accuracy of the UV-probe is limited. A correlation of r^2 is 0,5 during dry weather conditions (Lorenz, 2001) to r^2 is 0,76 (Grüning, 2001) may not be sufficient for most purposes. The accuracy can be increased if the SAC is combined with turbidity measurements. Combined measurements increase the correlation to r^2 is 0,83 (Häck, 2001) to r^2 is 0,9 (Grünning, 2001).

The UV-probe is less reliable then a turbidity sensor. Malfunctions are mainly caused by pollution of the sensor.

The advantage of the UV-probe is the ability to perform real time continuous measurements and the disadvantage is the low reliability and accuracy of the UV-probe and the need for local calibration. The accuracy may be increased by combining the UV-probe's measurements with turbidity measurements.

2.5.2.5 UV-visible spectrometry

UV-visible spectrometry (UV/VIS) is a technique which relies on the absorption and reflectance of ultraviolet and visible spectral region. The UV/VIS-spectrometer is similar but more extensive then the UV-probe. Compounds in water absorb certain wavelengths of UV and visible light. The UV and visible wavelength range from 200 nm to 750 nm. As a result the degree of extinction of a certain wavelength is a measure for the concentration of that compound in the sample. By analyzing the entire UV-visible spectrum the compounds and their concentration can be determined. The UV-visible spectrometer has a parallel reference beam travelling through a similar path, but not through the wastewater that has to be measured. The parallel reference beam is called the blank. The blank is necessary to compensate for the absorption and reflectance within the sensor itself. (Langergraber 2003 and Schilperoort, 2011). A figure showing the principle of UV-visible spectrometry is shown as figure 2.4.

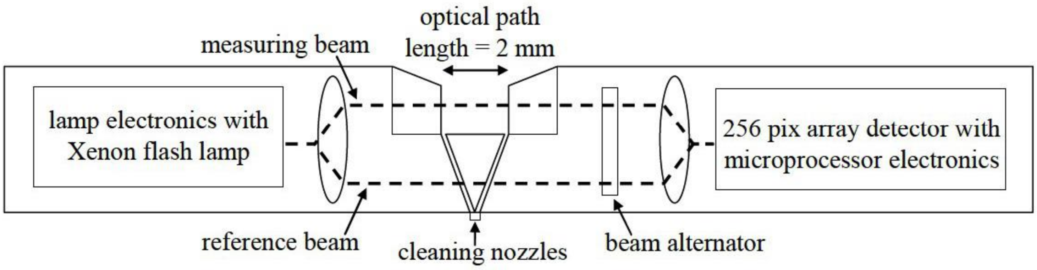


Figure 2.4 - Principle of UV/VIS spectrometry (Schilperoort, 2011). A lamp creates a beam in the UV and visible range. The beam is send through the sample, causing absorption and reflection, and caught by a detector. The concentrations of compounds in the sample are calculated from the detected spectrum. A second beam is send past the sample to serve as a blank

Manufacturers of UV-visible spectrometers supply their equipment with a factory or global calibration. The global calibration allows for use of the UV/VIS on purchase, but the matrix of the measured wastewater is probably different from the matrix of the wastewater used for the global calibration. Experiments with UV/VIS show that a good indirect measurement of COD is only possible when a local calibration is applied. In a local calibration samples are taken and analyzed to train the UV/VIS for the matrix of the measured water. When measurements are done in different water or the matrix of the water changes, a new calibration is needed (Bertrand-Krajewski, 2007, Gruber 2004, Langergraber, 2003 and Lombard, 2010).

UV/VIS-spectrometers are very susceptible to malfunctions due to pollution of the sensor. This effects the reliability of the spectrometer greatly. Great effort is needed to achieve a high percentage of reliable data (Schilperoort, 2012).

The advantage of UV-visible spectrometry is that real-time continuous measurements with reasonable accuracy become available. The disadvantages include the high price of the UV/VIS, the problems with reliability and the need for local calibration.

2.5.2.7 Indirect COD measurement using conductivity

The correlation between COD and conductivity is not strong enough to use conductivity as a proxy for COD. However, a relation between the two parameters has been found in studies. Measurements in the influent of the WWTP Eindhoven with a conductivity sensor and an UV/VIS-spectrometer show a clear correlation between specific EC and COD_f during dry weather flow (Daal van, 2013). This is illustrated in figure 2.5.

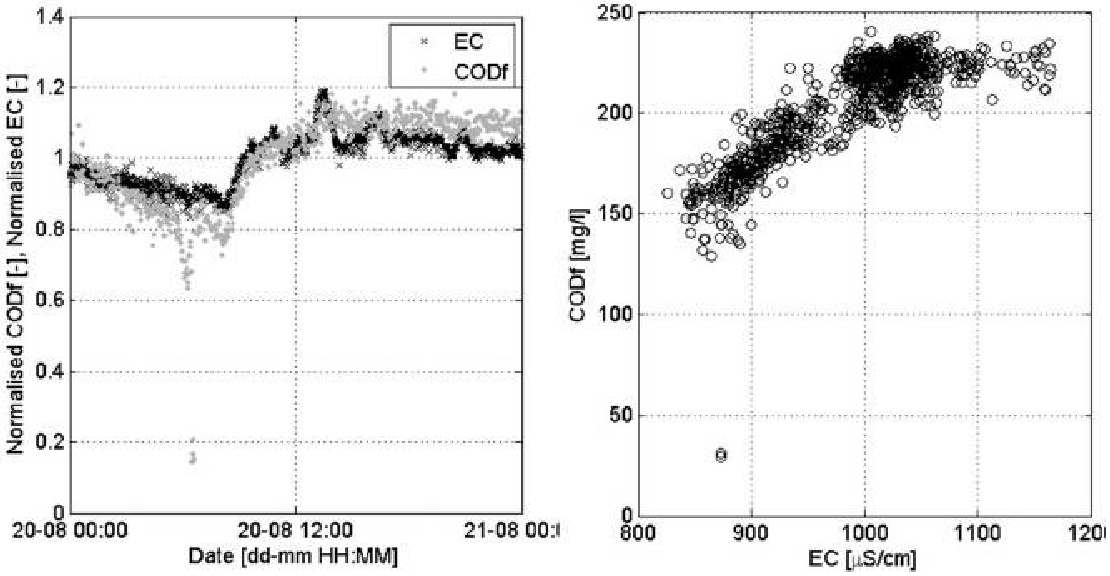


Figure 2.5 - EC and COD_f measurements in the influent of WWTP Eindhoven. A clear correlation can be seen during dry weather flow (Daal van, 2013)

2.5.2.7 Indirect COD measurement using turbidity and conductivity

A very limited amount of research had been done on the connection between COD and turbidity and conductivity. Lombard (Lombard, 2010) has performed measurements in the city of Romans, France to establish a relationship between turbidity and COD. Although a high correlation was found at first, systematic differences were found when the correlation was verified with additional data. This was consistent with a change of the waters matrix due to upcoming industrial discharges. Measuring just turbidity implicitly assumes a linear relationship between the COD fraction that is found in particles and the COD fraction dissolved in the water. A change in the matrix means a change in the relation between these two fractions so the assumed relation is no longer valid and a new calibration is necessary.

Lombard adds specific electrical conductivity as a parameter for the COD estimation to adjust for the change in the matrix of the wastewater. This helps to improve the estimation to a correlation of 98%. The found relation between conductivity and turbidity and COD reads as:

$$COD_{eq} = -6 \cdot 10^{-5}(T \cdot C)^2 + 0.847(T \cdot C) + 104.92$$

(equation 2.5)

in which:
COD_{eq} = the calculated COD equivalent concentration (mg/l)
T = the turbidity (NTU)
C = the specific electrical conductivity (mS/cm)

The parameters -6·10-5, 0.847 and 104.92 are found by calibration. The turbidity and specific electrical conductivity are related to the COD as a product: turbidity x specific electrical conductivity. This mathematical relationship gives the most satisfactory correlation to COD for the particular data set of Lombard. However, it has no theoretical background related to the composition of the wastewater. The correlation is shown in figure 2.6.

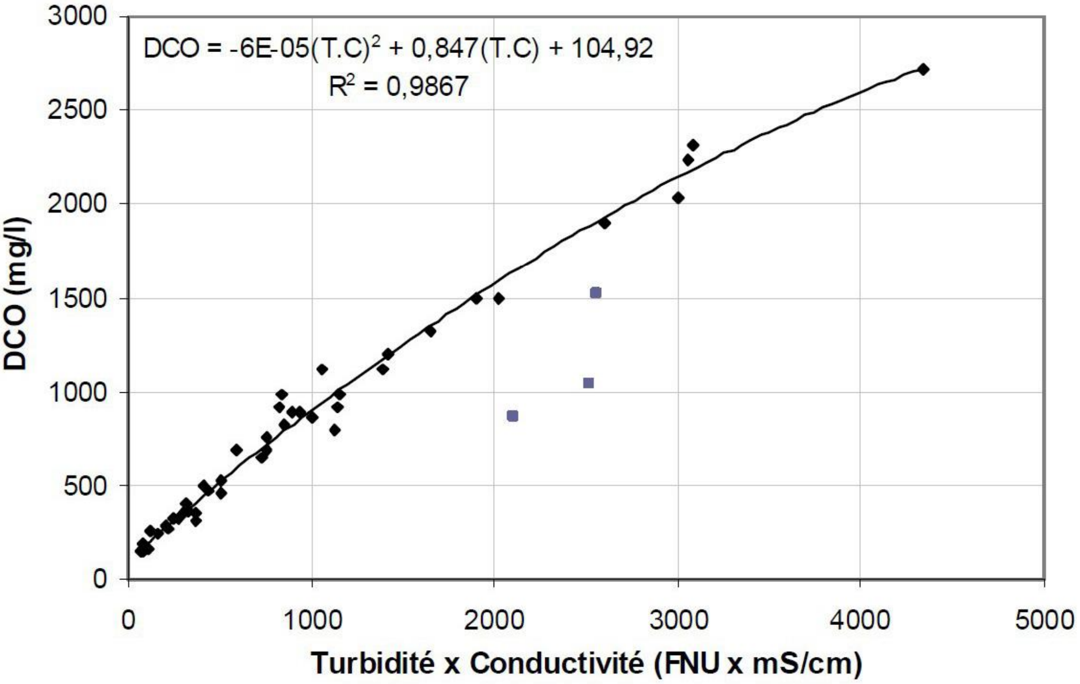


Figure 2.6 - Correlation between turbidity and specific EC and COD (Lombard, 2010). The COD (vertical axis) increases when the turbidity times specific EC (horizontal axis) increases. Data points and a fitted second degree relation are shown

Adding specific electrical conductivity to the correlation relation increases the flexibility to changes in the matrix of the wastewater. This does not mean that all changes in the matrix can be accounted for. Lombard et al indicates that three data points (grey in figure 2.6 above) with a very high specific electrical conductivity do not fit well to the found relation.

Research at a WWTP combined sewer inlet compared various surrogate sensors like turbidity sensors, UV-visible spectrophotometer, pH meter and conductivity meter to estimate the COD or pollution loads in general. The comparison showed that the combined use of turbidity and conductivity is a good surrogate for COD measurement and offers a lot of perspective for the future (Daal van, 2012 and 2013 and Lepot, 2012). Another advantage of using combined measurements is that the use of two parameters instead of just one increases the possibilities for validation of the data. The behavior of an indication parameter can be checked using the data from the other indication parameters sensor (Schilperoort, 2007b).

3. Research approach and test installation

3.1. Introduction

The literature study of chapter two has shown that there is a lack of knowledge on the relationship between COD and turbidity and conductivity in wastewater. In this chapter a research approach is presented to close a part of this gap.

First the objectives of the research is stated clearly. A general approach to answer the research questions is given. After that the set up of a preparatory laboratory test and a field test is elaborated in more detail. The last part of this chapter is dedicated to the analysis of the produced data.

3.2. Research objectives

The objective of this thesis is to research if measuring turbidity and specific electrical conductivity is a good way to collect information on the water quality of sewage by correlating it to COD.

The main research question of this graduation work is: "Can combined conductivity and turbidity measurements be used to reliably estimate the COD content of wastewater in a Dutch sewer?"

3.3. General approach

To answer the research question measurement data from Dutch sewers is necessary. A field test has been designed to acquire this data.

During the preparations of the installation of the field test a laboratory test is done. The objective of the laboratory test is to gain insight in the relation between turbidity, conductivity and COD and to obtain a data set to program a script for analysis.

3.4 Laboratory test

The laboratory test is performed in the Stevin III laboratory. This is the fluid mechanics laboratory of the faculty of civil engineering and geosciences of the Delft University of Technology. This lab is also used by sanitary engineering. The air temperature in the lab during the experiment was between 19.6° C and 20.0° C.

The results and analysis of the results of the laboratory test can be found in chapter 4.

3.4.1. Samples

To be able to determine the correlation function, a system of 100 data points has been created. The data points origin from 100 samples with a different turbidity and specific conductivity. The data points form a grid of 10 by 10 data points in the turbidity-specific conductivity field. The samples are created by creating a clay suspension with a certain turbidity and a very low conductivity and a salt solution with a certain conductivity, but very low turbidity.

The clay suspension was created by suspending of Goudse Clay into demineralized water. Goudse Clay is clay from the area of Gouda with a reasonably fixed composition. When suspended the Goudse Clay causes the water to be turbid, but barely influences the conductivity of the sample. An amount of 8.5 g of Goudse Clay is suspended into 1907 ml of demineralized water. This gives a clay concentration of 4.4·10-6 g/l. The clay suspension had a turbidity of 3636 NTU and a specific electrical conductivity of 17.5 µS/cm.

The salt solution was created by dissolving sodium chloride (NaCl) into demineralized water. When sodium chloride, or (table) salt, is dissolved it breaks into Na+ and Cl- ions. The ions allow the conductivity of electric current. When completely dissolved into the water, the sodium chloride has no effect on the turbidity. An amount of 11.4 g of sodium chloride is dissolved into 1691 ml of demineralized water. This gives a salt concentration 0f 6.7·10-6 g/l. The salt solution had a turbidity of 0.03 NTU and a specific electrical conductivity of 12,080 µS/cm.

Every sample is developed by the following procedure:
1) a flask is cleaned
2) the clean flask is weighted
3) a certain approximate volume between 0 and 50 ml from the clay suspension is added to the flask
4) the flask is weighted again
5) a certain approximate volume between 0 and 50 ml from the salt solution is added to the flask
6) the flask is weighted a third time
7) the flask is filled to approximately 100 ml
8) the flask is weighted for the last time
9) the turbidity and specific conductivity of the sample are determined

Because of the large quantity of required samples it was not practical to perform all 100 experiments in one batch. Instead, the experiment was done in 10 batches of each 10 samples. All samples in a batch had approximately the same volume of the clay suspension. An additional advantage of this strategy is that the risk of human errors is lowered.

The flask is first weighted dry. Since the same flasks are used for all 10 experiments after each other it is not necessary to dry the flask after each cleaning session. The difference between the clean wet and the clean dry flask is assumed to be demineralized water and added to the quantity added in step 7.

The approximate amounts of the clay suspension and the salt solution are shown in table 3.1. The samples are numbered 00 to 99 with the first figure of the identification number representing the amount of the clay suspension and the second figure of the identification number indicating the amount of the salt solution in the sample.

Table 3.1 - Amount of clay suspension and salt solution in a sample

	turbidity	nr. 0	nr. 1	nr. 2	nr. 3	nr. 4	nr. 5	nr. 6	nr. 7	nr. 8	nr. 9
EC	ml	0 ml	1 ml	2 ml	3 ml	4 ml	5 ml	10 ml	20 ml	30 ml	50 ml
nr. 0	0 ml	00	10	20	30	40	50	60	70	80	90
nr. 1	1 ml	01	11	21	31	41	51	61	71	81	91
nr. 2	2 ml	02	12	22	32	42	52	62	72	82	92
nr. 3	3 ml	03	13	23	33	43	53	63	73	83	93
nr. 4	4 ml	04	14	24	34	44	54	64	74	84	94
nr. 5	5 ml	05	15	25	35	45	55	65	75	85	95
nr. 6	10 ml	06	16	26	36	46	56	66	76	86	96
nr. 7	20 ml	07	17	27	37	47	57	67	77	87	97
nr. 8	30 ml	08	18	28	38	48	58	68	78	88	98
nr. 9	50 ml	09	19	29	39	49	59	69	79	89	99

The concentration of the clay suspension and the salt solution have been chosen together with the amounts of the mix for the samples. The combination of the concentration and the amounts in the mix makes the turbidity and the specific electrical conductivity of the main bulk of the samples fit within the range of the turbidity and the specific electrical conductivity found in wastewater. The range of turbidity and specific electrical conductivity approximately fits the turbidity and the specific electrical conductivity of the samples in the range between samples 11 and 77.

The grid is extended to lower and especially higher concentrations to accommodate for a larger range of values. The sampling grid is less dense in these regions to limit the number of samples.

3.4.2. Instruments

The turbidity is measured with a Hach 2100 Turbidimeter THM-05 (figure 3.1). The specific conductivity is measured with a WTW LF 325 microprocessor conductivity meter with a WTW TetraCon 325 standard conductivity cell. The turbidity meter is calibrated with formazin solutions before and after the laboratory test in accordance to the procedure supplied by the instrument manufacturer. The calibration of the conductivity meter is performed according to the ISO 7888 with a standard solution before and after the laboratory experiments. Calibration results of both instruments were within the specifications of the manufacturer. The automatic pipette used for the experiment showed a large deviation from the specified quantity. Therefore all pipetteted quantities have been weighted on a laboratory scale. The data from the laboratory scale is used for the analysis and not the pipetteted quantities. The laboratory scale is a Mettler AT261 Dettarange FACT. The laboratory scale is calibrated by the laboratory. The calibration was still valid at the time of the experiment.

Figure 3.1 - Turbidity meter (photo by: Gerben Tommassen)

3.4.3. CODf

The measurement of COD is expensive and time consuming and therefore undesirable. Therefore, the COD was not measured during this laboratory experiment. Instead an artificial substitute for COD is used: fictive COD. The fictive COD is linearly related to the concentration of suspended and dissolved compounds in the wastewater. A fictive COD can be made up based on this linear relation. The parameters in the linear relation to calculate the fictive COD are chosen in such a way that the fictive COD has the same order of magnitude as the turbidity and the specific electrical conductivity. This is just for convenience during analysis. The fictive COD is calculated with:

$$COD_f = 10^5\left(\frac{c_{clay} \cdot V_{clay}}{V_{sample}} + \frac{c_{NaCl} \cdot V_{NaCl}}{V_{sample}}\right) \qquad \text{(equation 3.1)}$$

In which:
COD_f = fictive COD (g CODf/l)
10^5 = dimensionless parameter to make the COD_f of the same order of magnitude
c_{clay} = concentration of Goudse Clay in the clay suspension (g/l)
V_{clay} = volume of clay suspension in the sample (l)
V_{sample} = volume of the sample (l)
c_{NaCl} = concentration of sodium chloride in the salt solution (g/l)
V_{NaCl} = volume of salt solution in the sample (l)

Sodium chloride does not contain organic compounds and has no COD, but for the purpose of the laboratory test this is not relevant. The sodium chloride is assumed to increase the fictive COD just like the Goudse Clay.

3.5. Field test

In the field test data is collected on the COD, the turbidity and the specific EC of water in a Dutch sewer. The EC and turbidity data were originally planned to be gathered by automatic sensors to obtain a large data set with continuous measurements. Because the equipment did not work appropriately, samples were taken instead. Information on the calibration of the equipment and errors in the measurements is found in appendices I and II.

A minimum of 25 samples with sufficient dispersion have to be gathered for regression analysis (Veldkamp, 2002).

The results and analysis of the results of the field test can be found in chapter 4.

3.6. Field test location

The field tests are performed in the system of the town of Ulvenhout, in the province of North Brabant.

3.6.1. Ulvenhout

The town of Ulvenhout is situated on the South East of the city of Breda. The town is situated in the municipalities of Breda and Alphen-Chaam. The urban area of Ulvenhout is part of the municipality of Breda and the rural area South of the highway A58 is part of the municipality of Alphen-Chaam. The location of Ulvenhout in the Netherlands is shown in figure 3.2 with a red circle.

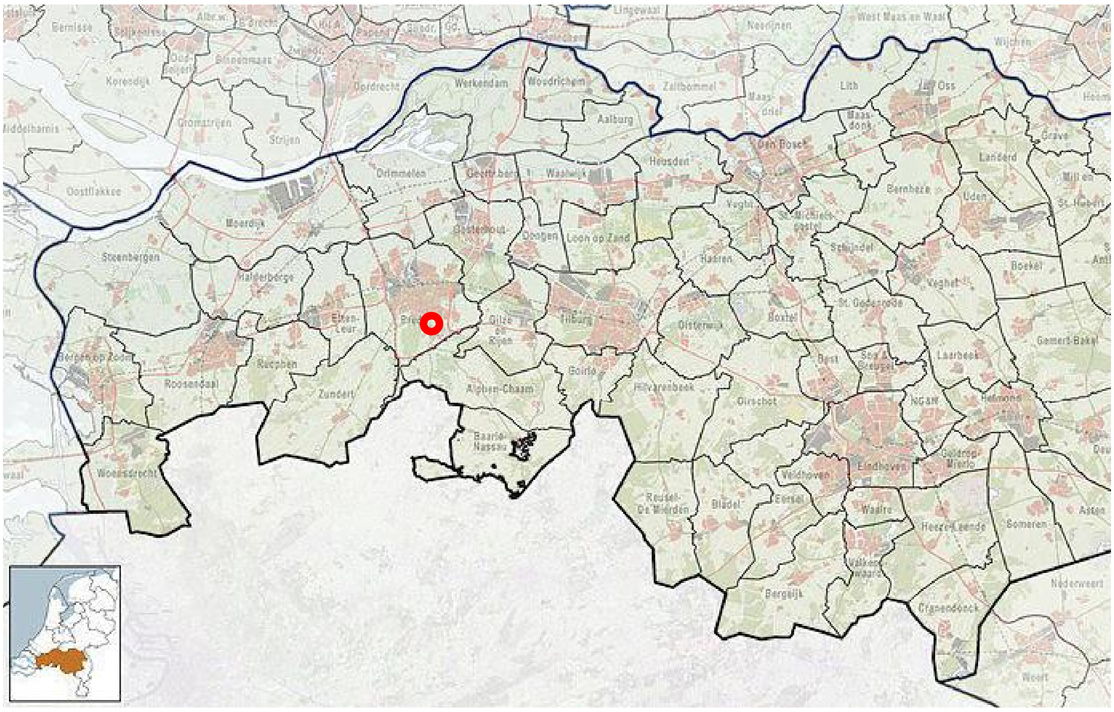


Figure 3.2 - Location of Ulvenhout (background source: Jan-Willem van Aalst, 15 January 2010). Ulvenhout is at the red circle near the Belgian border in the province of North Brabant. The small bottom left map shows the North Brabant in The Netherlands

The municipality of Breda is responsible for the management of almost the complete sewer system of Ulvenhout. Exception are the strings of pressurized sewer South of the highway A58 (Annevillelaan, Chaamseweg and Strijbeekseweg), which are in the municipality of Alphen-Chaam. These pressurized sewers are managed by the municipality of Alphen-Chaam. The water board Brabantse Delta is responsible for management of the main pumping station of Ulvenhout and treatment of the wastewater from Ulvenhout. Brabantse Delta is also responsible for the water quality of the main receiving surface water at Ulvenhout: the Mark.

An edited areal photograph map of Ulvenhout is shown in figure 3.3. The river Mark is visible on the left of Ulvenhout. The highway A58 between Breda and Alphen-Chaam is clearly visible South of Ulvenhout. The main pumping station of Ulvenhout is in the north corner of the roundabout indicated by the red circle.

Figure 3.3 - Edited areal photograph of Ulvenhout (background source: Google maps). A red circle indicates the location of the pumping station

There is no industrial area in Ulvenhout.

3.6.2. Structure of the sewer system of Ulvenhout

The sewer system of Ulvenhout is mainly of the combined type. There is also a small improved separated system in the east of Ulvenhout and some pressurized sewers south of the town of Ulvenhout and a small part of the paved area has been disconnected from the combined sewer system and is drained through storm water sewers. The pressurized sewer systems discharge in the combined sewer systems on the south side of Ulvenhout. All the combined sewer systems and the improved separated sewer system discharge in the largest catchment area of Ulvenhout, conveniently named Ulvenhout. One of the combined sewer systems, Beekhoek 2, has no external CSO's. This system is connected to the main combined sewer system of Ulvenhout through conduits with high invert levels. These conduits effectively function as internal weirs. (Witteveen+Bos, 2009)

The largest catchment area of Ulvenhout is divided into two parts: Ulvenhout High and Ulvenhout Low. The division between Ulvenhout High and Ulvenhout Low has been made to use the storage capacity of the system of Ulvenhout more effectively. Ulvenhout High and Ulvenhout Low are connected through two orifices and three internal weirs. Catchment area Beekdal 2 discharges to Ulvenhout High with a pump, but also has high level free flow connections with Ulvenhout High. During dry weather flow (DWF), the wastewater of Ulvenhout flows through the orifices from Ulvenhout High to Ulvenhout Low. During storms, the inflow of the Ulvenhout High exceeds the capacity of the orifices. The storage capacity of Ulvenhout High is used before the CSO's of Ulvenhout Low begin the discharge. If the event exceeds the capacity of the orifices and the storage capacity of Ulvenhout High, the system discharges through internal weirs to Ulvenhout Low and through external weirs on the river Mark. Ulvenhout Low has a CSO that discharges on the river Mark. This CSO, called Markdal, has a storm water retention and settlement tank of 480 m3. (Witteveen+Bos, 2009)

The main pumping station of Ulvenhout is situated in Ulvenhout Low and discharges to catchment area 0 of the city of Breda. The wastewater from Ulvenhout and the city of Breda amongst others is treated at the Nieuwveer wastewater treatment facility. (Witteveen+Bos, 2009) The Nieuwveer wastewater treatment facility is situated on the northwest side of the city of Breda. The structure of the sewer system of Ulvenhout is shown schematically in figure 3.4.

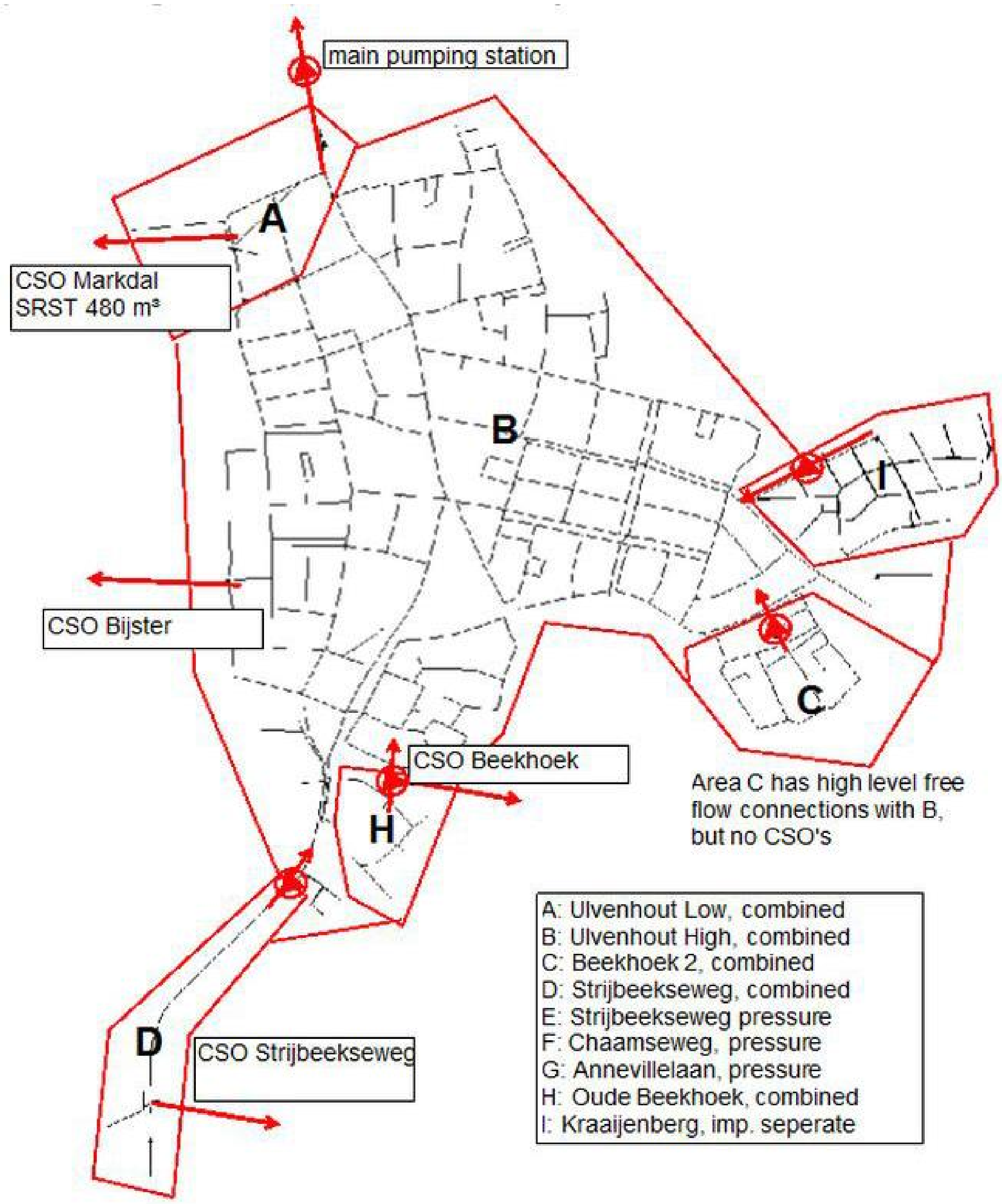


Figure 3.4 - Sewer system of Ulvenhout. (altered from: Witteveen+Bos, 2009) CSO's are indicated with an arrow and pumping stations are indicated with an arrow with pump sign (closed triangle in circle). Pressurized sewer systems are not shown

3.6.3. Characteristics of the sewer system of Ulvenhout

The main characteristics of the sewer system of Ulvenhout are shown in this section. Table 3.2 shows the main characteristics of the catchment areas of Ulvenhout.

Table 3.2 - Characteristics of catchment areas (translated from: Witteveen+Bos, 2009)

		Ulvenhout total	Ulvenhout A+B+C
type		-	combined.
inhabitants		5290	4555
DWF	m^3/h	57.6	50.30
paved area	ha	52.33	49.00
storage	m^3	2629	1979
	mm	5.00	4.04
injections	m^3/h	0	41.9
pump cap.	m^3/h	342	342
POC	m^3/h		250
	mm/h		0.51

3.6.4. Dry weather flow

During the calibration of the system of Ulvenhout calibrated dry weather flow curves have been created. Because of the differences during week days and weekend days, two separate curves were constructed. The calibrated dry weather flow curve is shown in figure 3.5.

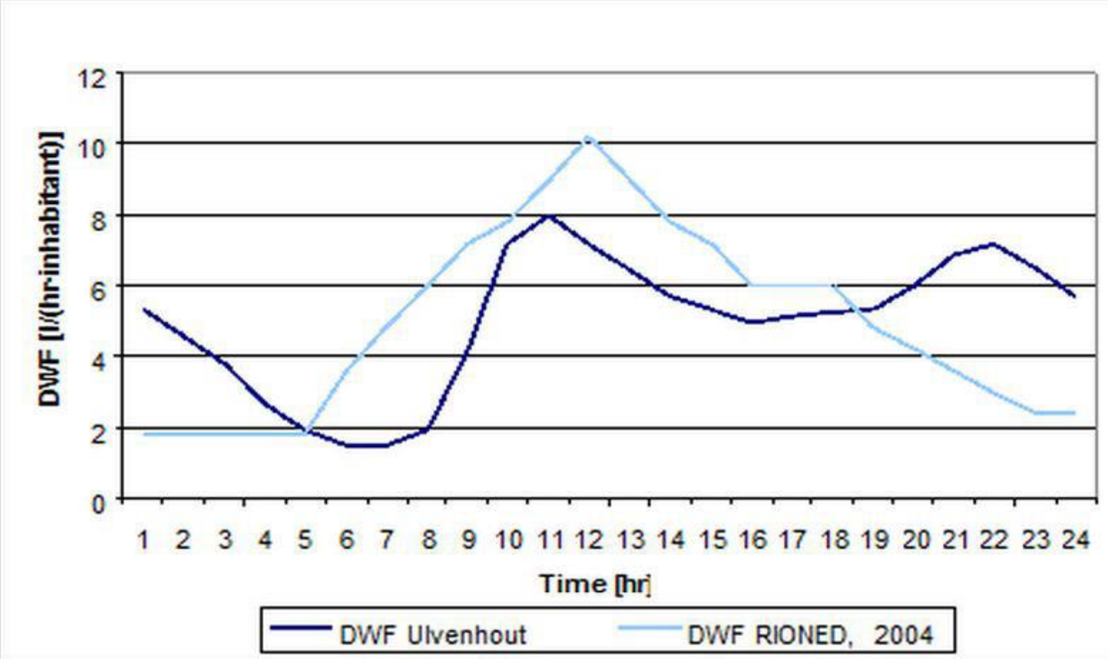


Figure 3.5 - Calibrated DWF curve (translated from: Witteveen+Bos, 2009)

The calibrated DWF curves are different from the standard DWF curve (RIONED Foundation, 2004). The DWF curve for weekdays, clearly shows two peaks. One in the morning and one in the evening. The weekend DWF curve shows just one peak.

3.6.5. Pumping Station Ulvenhout

The main pumping station (PS) of Ulvenhout is situated in Ulvenhout Low on the North side of Ulvenhout. PS Ulvenhout has a capacity of 342 m^3/h. A top view of the pumping station is shown in figure 3.6.

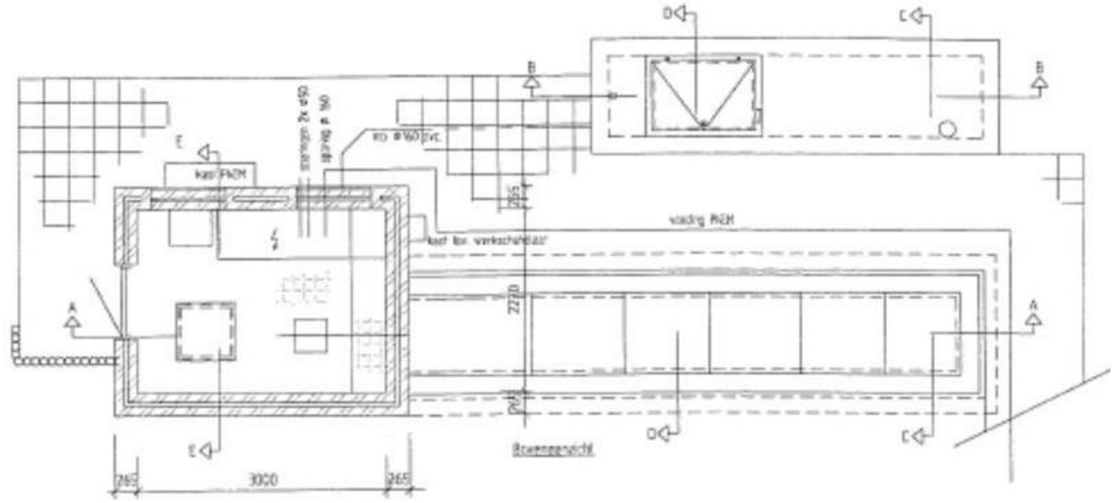


Figure 3.6 - Top view of main PS Ulvenhout (Waterschap Brabantse Delta, 1997)

The pumping curve of the pumps has been determined during the calibration of the sewer system of Ulvenhout. The measured pumping curve is shown in figure 3.7.

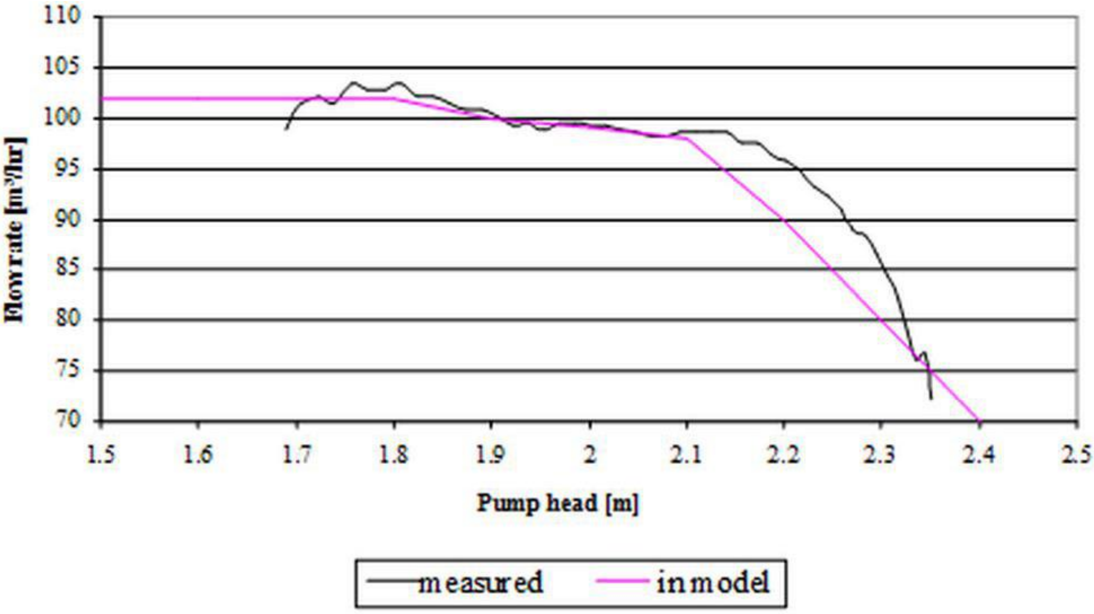


Figure 3.7 - Q-H relationship main PS Ulvenhout (translated from: Witteveen+Bos, 2009)

The bottom of the main pump pit has a level of 0.65- mNAP, the main pump off level is 0.05- mNAP and the main pump on level is 0.80+ mNAP. The surface level is 3.40+ mNAP. A section of the pump pit is shown in figure 3.8.

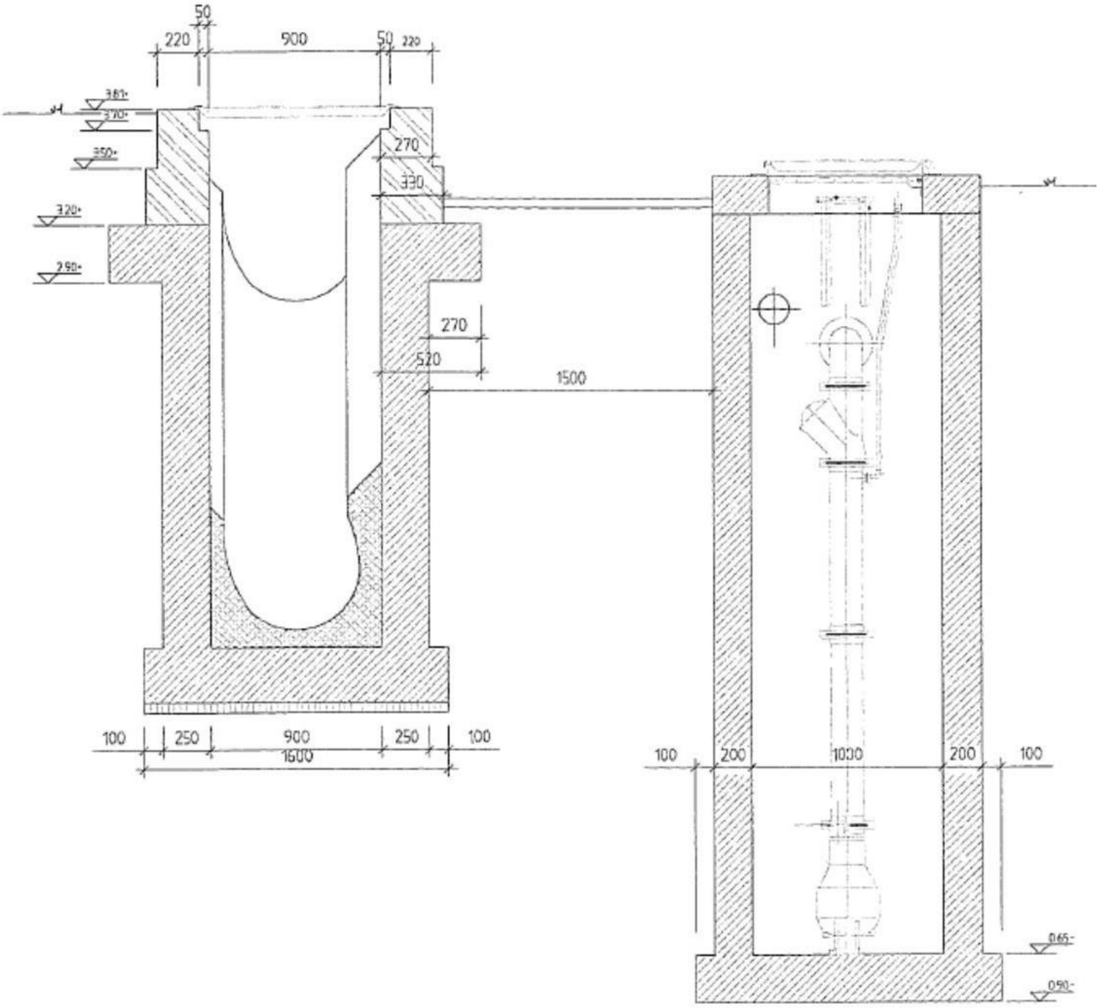


Figure 3.8 - Section D-D of pump pit (Waterschap Brabantse Delta, 1997)

An extensive drawing of the pumping station is attached in appendix IV.

3.6.6. Weir Markdal

The weir and storm water retention and settlement tank Markdal is situated in Ulvenhout Low on the North West side of the town. The CSO discharges upon the river Mark.

The internal weir of the tank has a level of 1.94+ mNAP and the external weir has a level of 1.84+ mNAP. The storage capacity of the tank is 480 m3. According to hydrodynamic calculations with the calibrated sewer model of Ulvenhout, the external weir discharges with an average frequency of 5.4 times per annum. The average discharged volume is 10.232 m3/a. The surface level at the external weir is 3.04+ mNAP and the bottom level in the internal weir manhole is 0.87+ mNAP.

3.7. Correlation and analysis

MATLAB is used to correlate the measurements of turbidity and specific EC to the COD. Measurements are scaled if this is necessary to increase the fitting algorithm performance. MATLAB can be used to find a good solution based on given relations using a Levenberg-Marquart algorithm (Marquart, 1963).

The Levenberg-Marquart algorithm is a gradient based algorithm that is a part of MATLAB's Curve Fitting Toolbox (which also fits surfaces). The algorithm finds a local minimum and not (per se) a global minimum. As a result the outcome of the calculation may be dependent on the starting points of the algorithm. To test the sensitivity of the calculation to the starting points, a sensitivity analysis is performed in which the analysis has been performed with 10 additional random starting points.

Earlier research (Lombard, 2010) used the product of the turbidity and specific EC as input variable for a second degree relation. The same product of turbidity and specific EC is used in a linear relation and an exponential relation. Next to the product of turbidity and specific EC these three relations are also tested with turbidity and specific EC separately. Table 3.3 shows the relations and the number of coefficients to be determined for the laboratory test analysis.

Table 3.3 - Tested relations between turbidity (T) and specific EC (C) and fictive COD ($COD_{eq;f}$)

fit	formula	number of coefficients	equation
1	$COD_{eq;f}=aT^2+bC^2+cT+dC+e$	5	3.2
2	$COD_{eq;f}=aT+bC+c$	3	3.3
3	$COD_{eq;f}=aT^b+cC^d+e$	5	3.4
4	$COD_{eq;f}=a(TC)^2+b(TC)+c$	3	3.5
5	$COD_{eq;f}=a(TC)+b$	2	3.6
6	$COD_{eq;f}=a(TC)^b+c$	3	3.7

For the field test, next to second degree, the linear relation and the exponential relation, relations with either the product of the turbidity and conductivity or the turbidity and conductivity separately, six more relations are tested.

Either the turbidity or the specific EC could have no or no significant influence on the COD, therefor relations with only one of the independent parameters are also tested. These fits are not meaningful in the laboratory test due to the nature of the COD_f. Table 3.4 shows the relations and the number of coefficients to be determined for the field test analysis.

Table 3.4 - Tested relations between logarithmic turbidity (x) and specific EC (y) and logarithmic COD (z)

fit	formula	number of coefficients	equation
1	$z=ax^2+by^2+cx+dy+e$	5	3.8
2	$z=ax+by+c$	3	3.9
3	$z=ax^b+cy^d+e$	5	3.1
4	$z=a(xy)^2+b(xy)+c$	3	3.11
5	$z=a(xy)+b$	2	3.12
6	$z=a(xy)^b+c$	3	3.13
7	$z=ax^2+bx+c$	3	3.14
8	$z=ax+b$	2	3.15
9	$z=ax^b+c$	3	3.16
10	$z=ay^2+by+c$	3	3.17
11	$z=ay+b$	2	3.18
12	$z=ay^b+c$	3	3.19

Overparameterization is a common pittfall in regression analysis. When a model is overparameterized a high number of parameters is used without adding substantially to the model outcome and possibly even deteriorating the result. After all, there is always a high power polynomial that (almost) perfectly fits the measured data, but this does not necessarily improve the prediction capabilities of the model. Especially when the data has relatively large random errors. This is illustrated in figure 3.9.

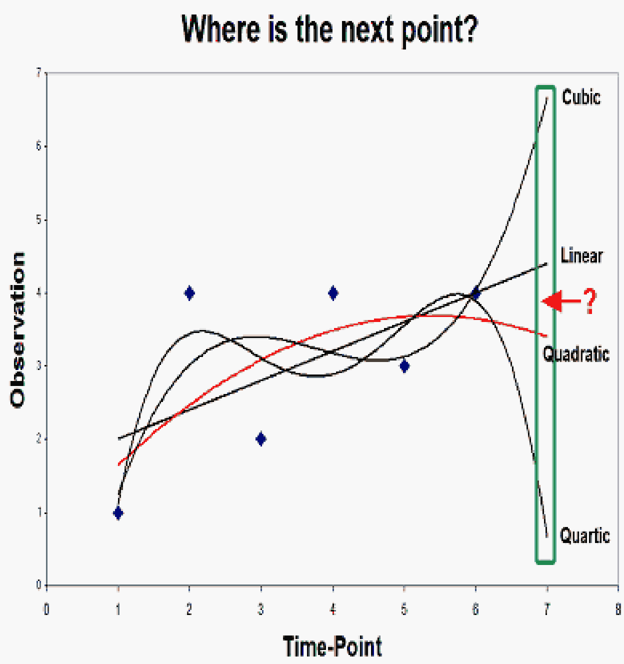


Figure 3.9 - Increasing the number of parameters does not increase the model quality (Walter, 2008)

To prevent overparameterization a model with less parameters that fits the measurements good is preferred over a model with more parameters that fits comparably or only slightly better. No mathematical check to prevent overparameterization is used.

Several goodness of fit parameters, of which the r^2 is the most notable, will be calculated to determine the best fit. Furthermore the fits will be plotted with the data points to further assess the results.

4. Results and discussion

4.1. Results laboratory test

The measurements in the laboratory test have been done at approximate locations in the Goudse Clay-salt field. Figure 4.1 shows how this translates to locations in the field of turbidity and specific EC as measured.

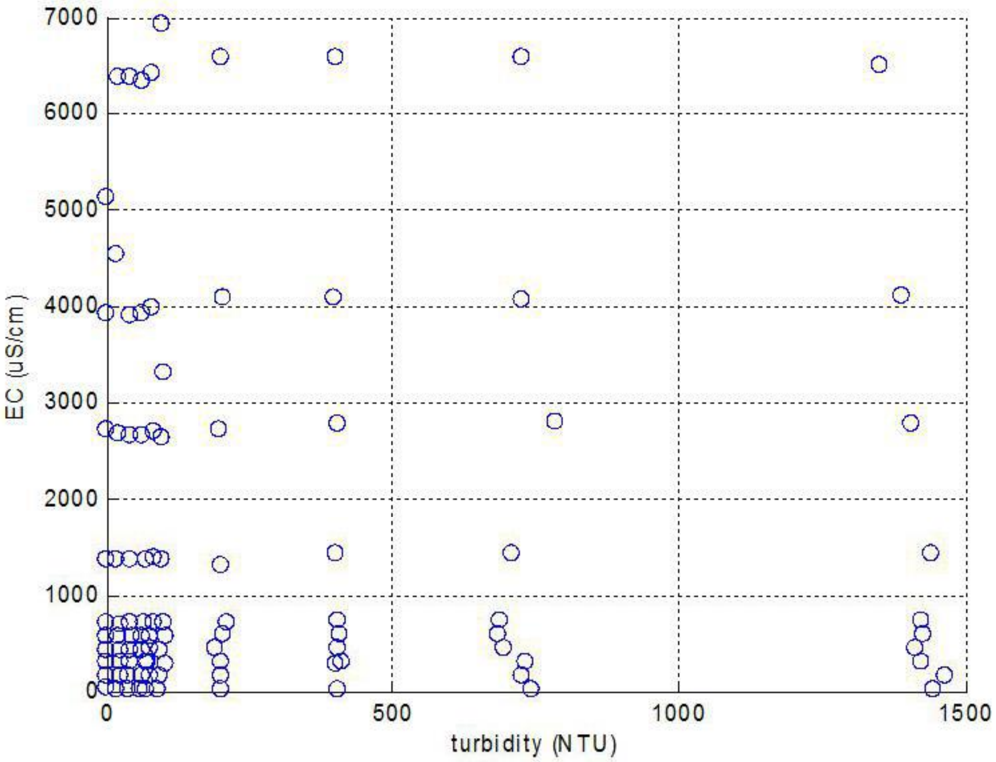


Figure 4.1 - Measured positions in turbidity-specific electric conductivity field

The results of the laboratory test are presented graphically as a 3D-scatterplot in figure 4.2. A table with the tests results is presented in appendix IV.

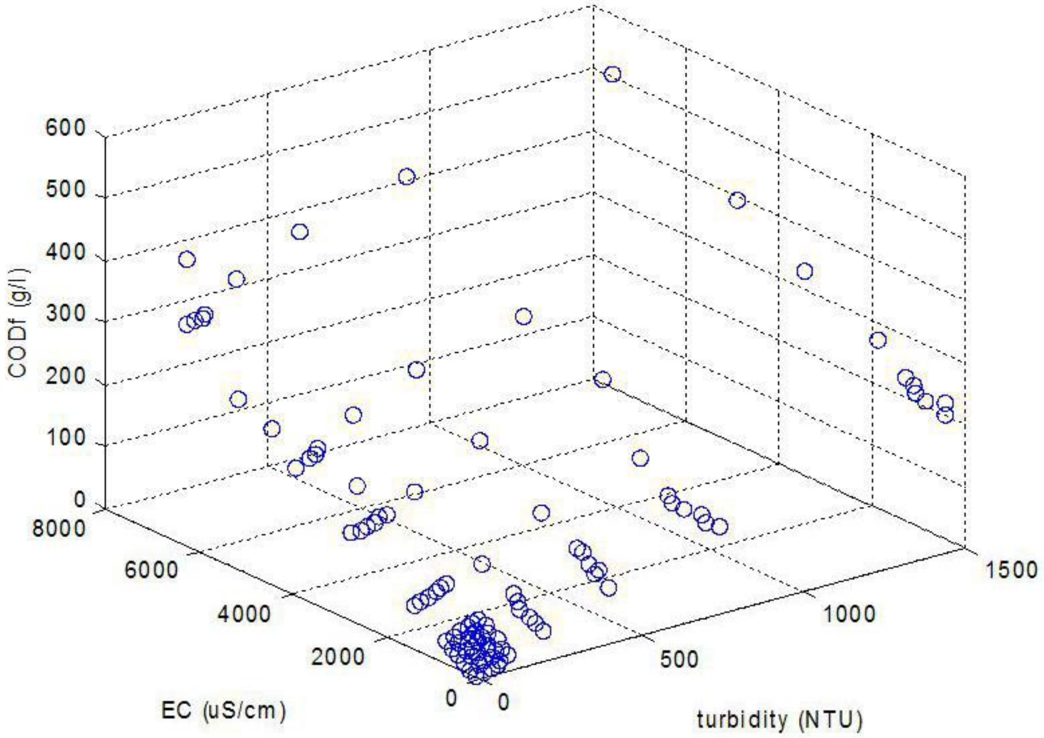


Figure 4.2 - Results laboratory test

The 3D-scatterplot shows the relation between turbidity, specific electrical conductivity and fictive COD.

4.2. Analysis of laboratory test

MATLAB has been used to correlate the measurements of turbidity and specific EC to the fictive COD according to the relations of equations 3.2 tot 3.7. The MATLAB script code has been attached as appendix VIII. MATLAB can be used to find a good solution based on given relations using a Levenberg-Marquart logarithm (Marquart, 1963).

The found coefficients and the r^2 are shown in table 4.1.

Table 4.1 - Coefficients and the r^2 of the laboratory test fits

Fit	formula	a	b	c	d	e	r^2
1	$COD_{eq;f}=aT^2+bC^2+cT+dC+e$	$-5.01\cdot10^{-5}$	$8.48\cdot10^{-7}$	0.226	0.048	0.711	>0.99
2	$COD_{eq;f}=aT+bC+c$	0.158	0.0533	4.49			0.99
3	$COD_{eq;f} = aT^b + cC^d + e$	0.90	0.76	0.030	1.06	-3.47	>0.99
4	$COD_{eq;f} = a(TC)^2+b(TC)+c$	$-1.18\cdot10^{-11}$	0.000147	87.49			0.61
5	$COD_{eq;f} = a(TC)+b$	$7.19\cdot10^{-5}$	104.37				0.51
6	$COD_{eq;f} = a(TC)^b + c$	0.91	0.40	29.29			0.72

The first fit formula gives the highest r^2. The surface of $COD_{eq;f} = -5.01\cdot10^{-5}T^2 + 8.48\cdot10^{-7}C^2 + 0.226T + 0.0480C + 0.711$ *(equation 4.1)* is shown in the scatter plot of the turbidity, the specific EC and the fictive COD in figure 4.3. The results of the other fits are attached as appendix V.

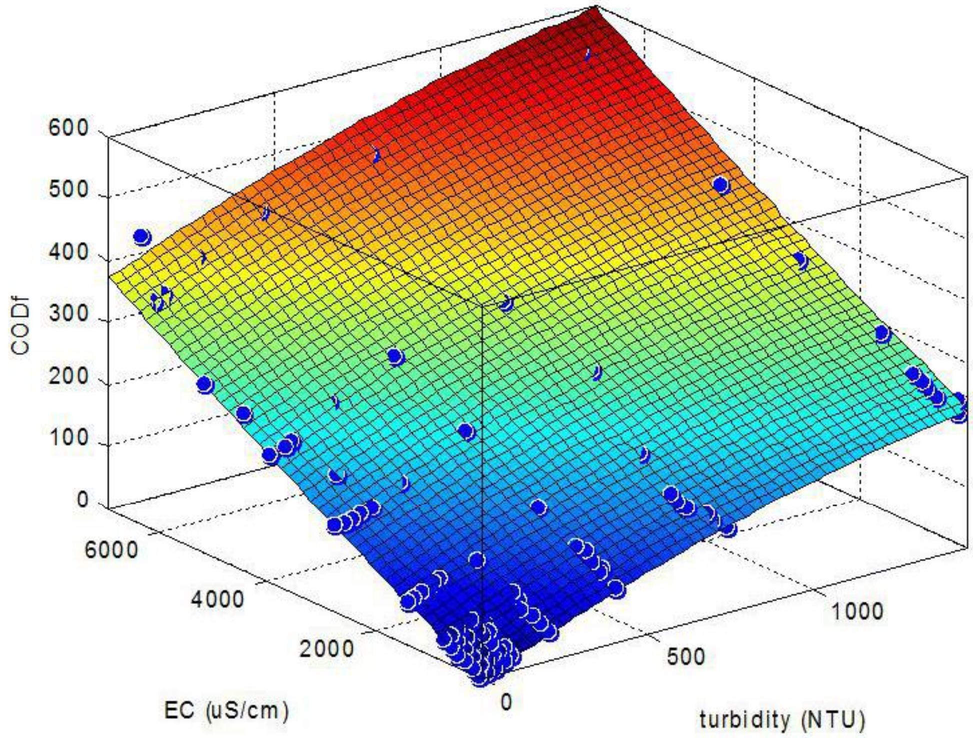


Figure 4.3 - Result of fit 1 of the laboratory test. The blue dots indicate actual measurements. The colored plane is the calculated fit

4.3. Discussion of the laboratory test results

The Levenberg-Marquart algorithm finds a local minimum and not (per se) a global minimum. This study copes with some uncertainties and with many variations in the sample water matrix. Taking these into account it becomes clear that a perfect correlation cannot be achieved when dealing with wastewater flows. A good solution will do and this is exactly what the Levenberg-Marquart algorithm provides.

The COD is not measured in the laboratory test, but a fictive COD is used instead. The fictive COD is constructed using a linear relation to the Goudse Clay and sodium chloride concentration. The relation between total suspended solids and turbidity is not linear although it is sometimes modeled as such with fairly good results (Fletcher, 2007, Lombard, 2010). The sodium chloride concentration is linearly related to the specific EC. As a result the fictive COD has an almost linear relation to the turbidity and the specific EC.

4.4. Results field test

The fifty samples taken during the field test are analyzed on COD, turbidity and specific EC. Tables with the data are attached in appendix VI. Figure 4.4 shows the COD, turbidity and specific EC over time for all three sampled events. It is noted that the events are not entirely sampled.

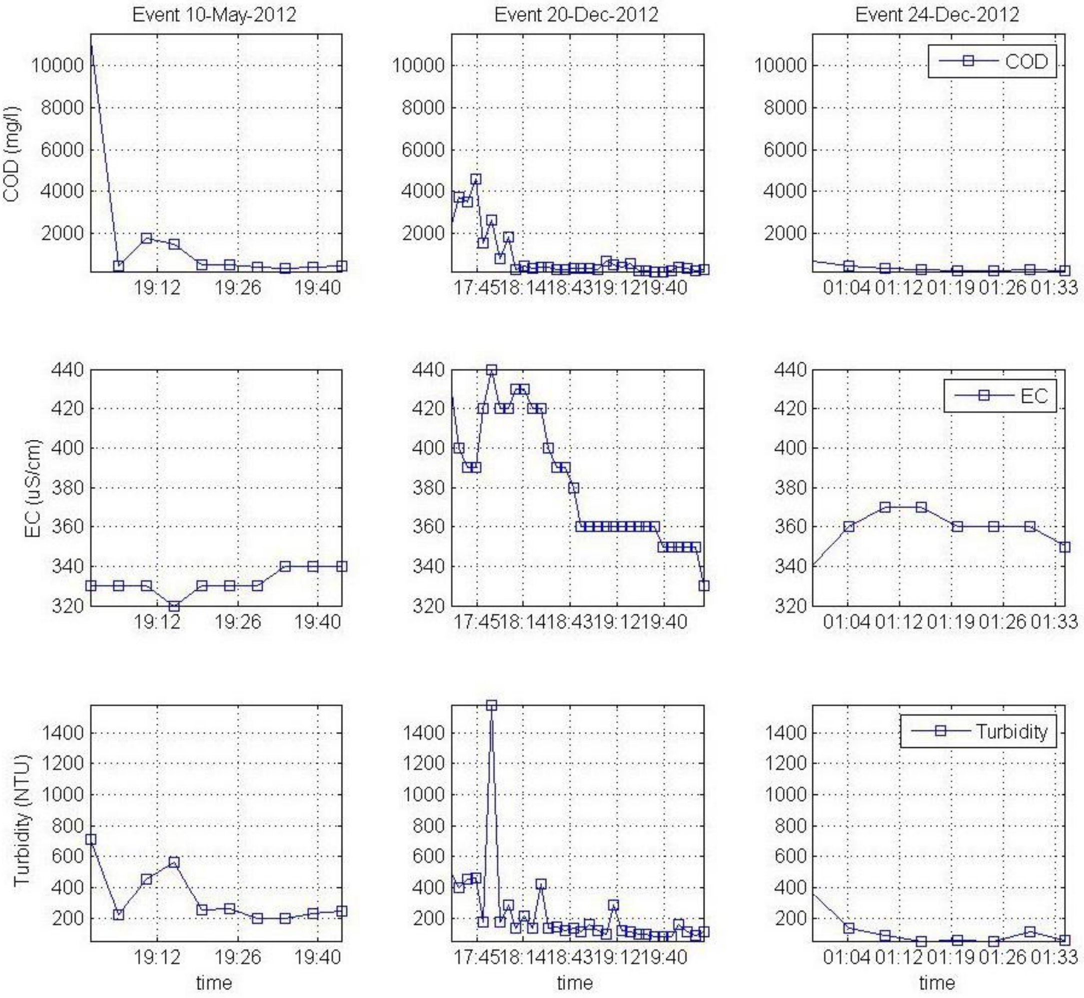


Figure 4.4 – Field test sample values over time. The scale of the x-axis is different for each event

The precipitation data from a nearby amateur meteorologist is compared to the sampled values to see if there is a relation. The weather station is about five km from the sampling location at Ulvenhout. Especially with heavy rain storm, the spatial distribution can be very large, but precipitation data from a closer location is not available. The precipitation data is not used for detailed analysis because of the spatial differences between the rain gauge and the sampling location. The precipitation data is shown in figure 4.5 to 4.7.

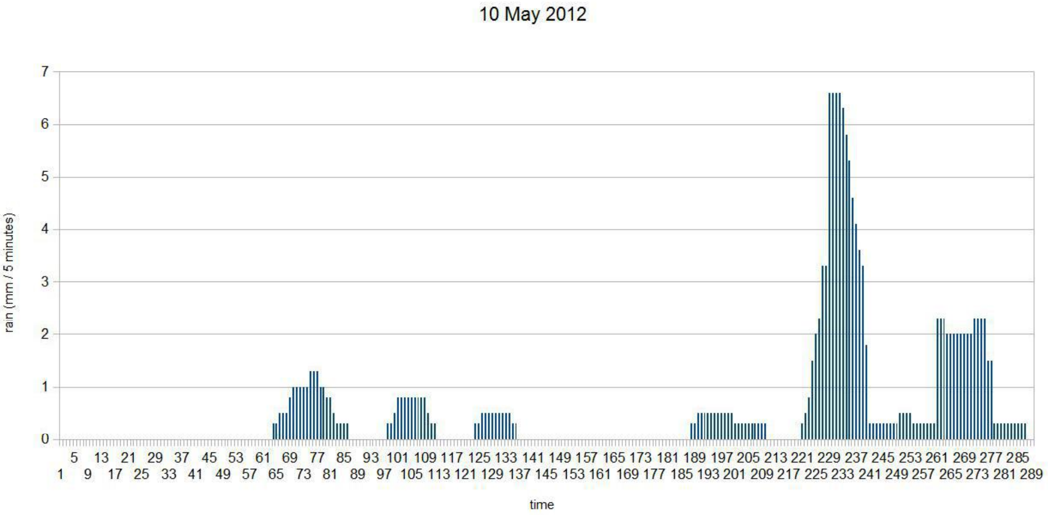


Figure 4.5 - Precipitation at 10 May 2012. Time stamp 224 corresponds to the first measurement at 19:00 and time stamp 233 corresponds to the end of the sampling at 19:45. Rain is the only form of precipitation on 10 May 2012.The sampled event starts just before the sampling and is characterized by very high intensities (data: Eric Kramers)

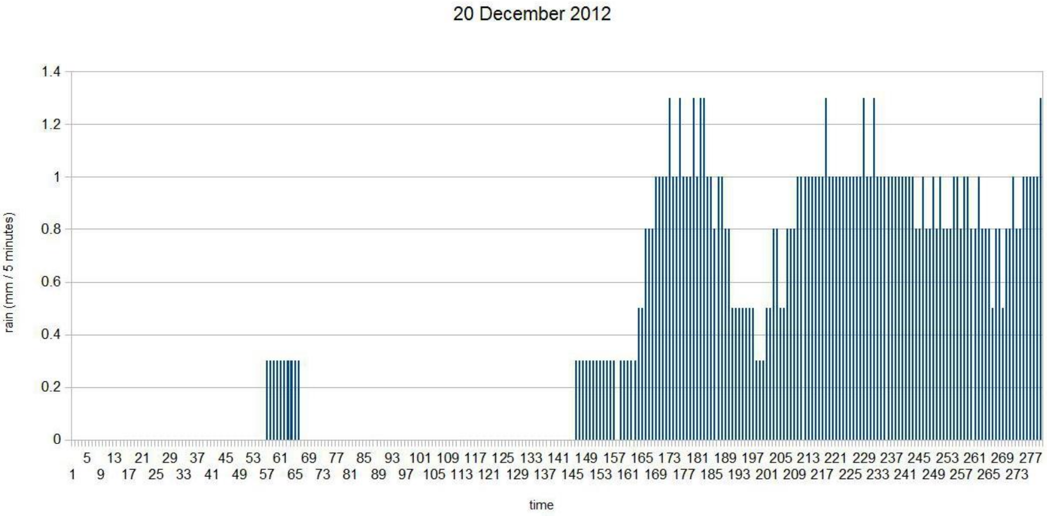


Figure 4.6 - Precipitation at 20 December 2012. Time stamp 207 corresponds to the first measurement at 17:30and time stamp 238 corresponds to the end of the sampling at 20:05 . Rain is the only form of precipitation on 20 December 2012.The sampling starts some time after the first rain has fallen and continues as the rain holds on (data: Eric Kramers)

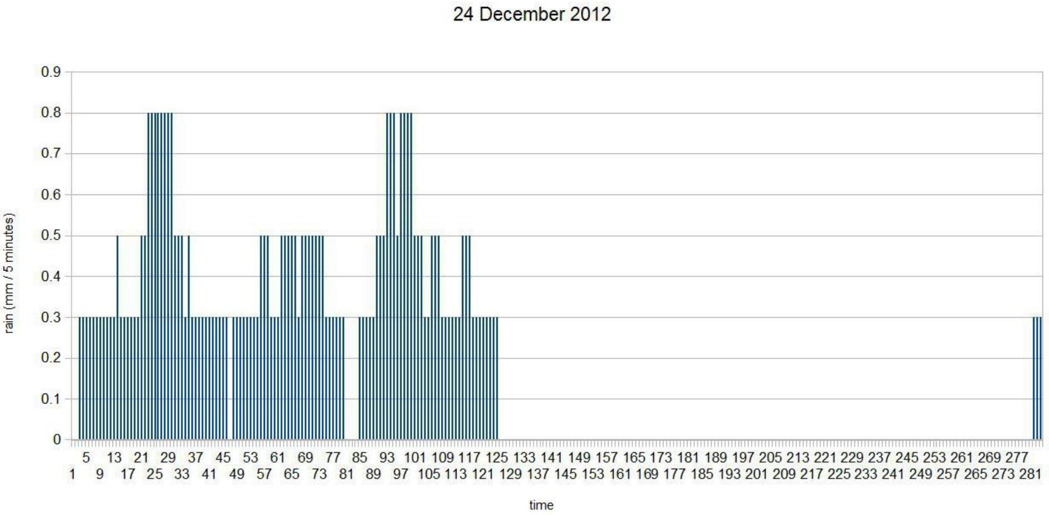


Figure 4.7 - Precipitation at 24 December 2012. Time stamp 12 corresponds to the first measurement at 01:00 and time stamp 19 corresponds to the end of the sampling at 01:35 . Rain is the only form of precipitation on 24 December 2012.In the week before 24 December significant amounts of rain have fallen. The sampled event started at midnight and is less severe then the other sampled events (data: Eric Kramers)

Some peaks in the measurements can be distinguished. In general values seem to peak at the beginning of the sampling period and decrease during the event. Next to discharges, the following processes influence turbidity peaks (Henckens, 2001):

1. flow velocity
2. flush
3. extra turbulence, increasing shear stress, but not flow velocity
4. the availability and characteristics of sediment

The conductivity is influenced by flush and discharges only and the COD is assumed to be influenced by a combination of the processes influencing turbidity and conductivity. No conclusions on the occurrence of a first flush or final flush or any other explanation for peaks can be drawn based on the available data.

Figure 4.8 shows the samples in scatter plots regardless of time.

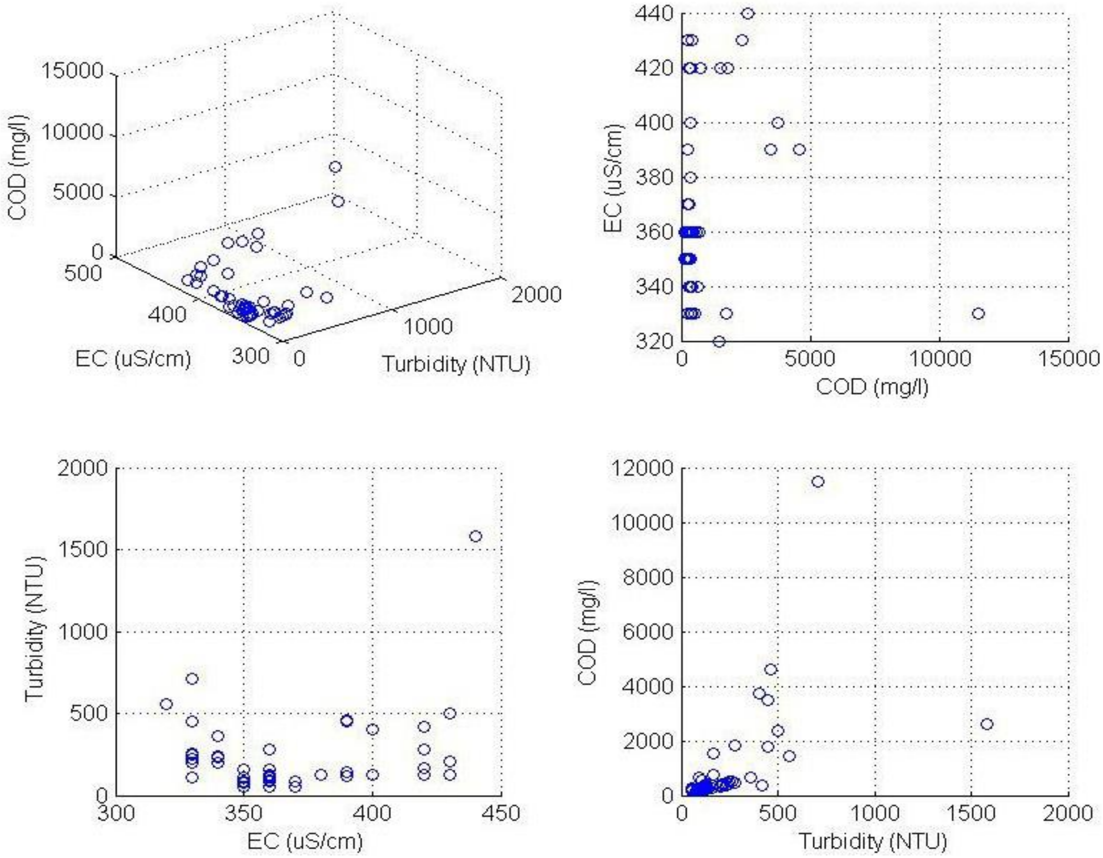


Figure 4.8 - Scatter plots of the data of the field test

It can be seen that the measured parameters differ greatly in scale and distribution. This is even more obvious when the data is represented in histograms (see Figure 4.9).

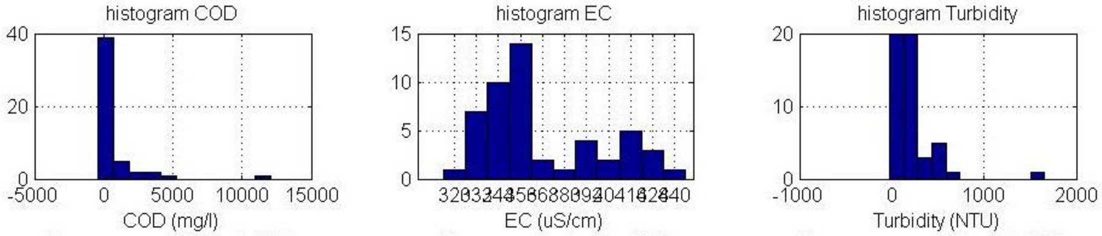


Figure 4.9 - Histograms of the data of the ifeld test

It is considered to use logarithmic or normalized equivalents of the values and to uniformize the data to improve stability of calculations and the quality of fits. Runs with adjusted input data showed that the best results were achieved with logarithmic COD, logarithmic turbidity and unchanged specific EC. Figure 4.10 shows the adjusted input data.

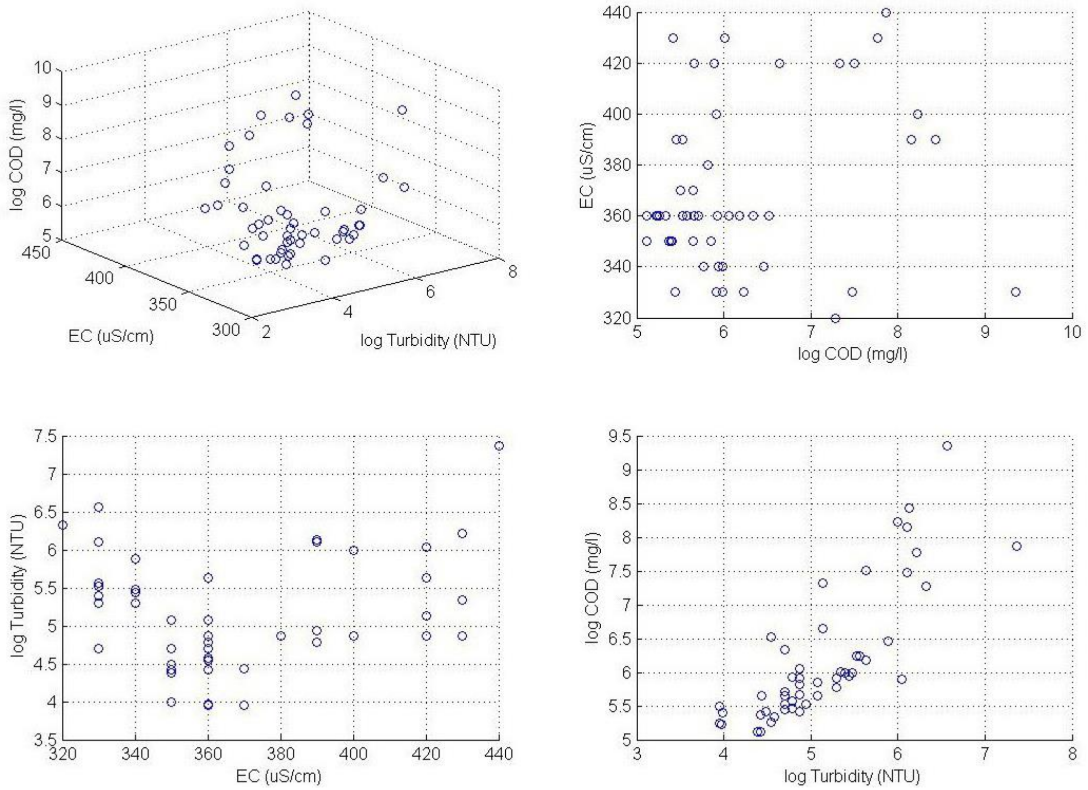


Figure 4.10 - Scatter plots of adjusted input data of the field test

4.5. Analysis of field test

MATLAB has been used to correlate the measurements of logarithmic turbidity and specific EC to the logarithmic COD according to the relations of equations 3.8 tot 3.19 . The MATLAB script code has been attached as appendix IX. MATLAB can be used to find a good solution based on given relations using a Levenberg-Marquart logarithm (Marquart, 1963).

The result of the sensitivity analysis on starting points for the Levenberg-Marquart algorithm is that each of the calculations had the same outcome. Based on the sensitivity analysis it can be concluded that the calculation is not sensitive to the chosen starting point.

The found coefficients and the r^2 are shown in table 4.2.

Table 4.2 - Coefficients and the r^2 of the field test fits. The log COD_{eq} is z, the log turbidity is x and the specific EC is y.

Fit	formula	a	b	c	d	e	r^2
1	$z=ax^2+by^2+cx+dy+e$	0.15	$-0.15\cdot10^{-3}$	-0.42	0.12	-18.09	0.69
2	$z=ax+by+c$	1.07	$2.13\cdot10^{-3}$	-0.08			0.66
3	$z=ax^b+cy^d+e$	2.62	0.69	5.65	0.14	-14.58	0.65
4	$z=a(xy)^2+b(xy)+c$	$-0.72\cdot10^{-6}$	0.01	-0.74			0.55
5	$z=a(xy)+b$	$1.99\cdot10^{-3}$	2.41				0.52
6	$z=a(xy)^b+c$	2.46	0.25	-9.99			0.54
7	$z=ax^2+bx+c$	0.13	-0.27	4.18			0.66
8	$z=ax+b$	1.09	0.61				0.65
9	$z=ax^b+c$	0.09	2.05	3.53			0.66
10	$z=ay^2+by+c$	$6.69\cdot10^{-3}$	-0.03	10.11			0.08
11	$z=ay+b$	0.01	3.72				0.05
12	$z=ay^b+c$	-40.24	-0.11	27.26			0.04

The first fit formula gives the highest r^2 (0.69) but some other fits have comparable results. Fit 8, by example has an r^2 just slightly lower (0.65) with only two instead of five coefficients. Moreover, fit 8 fits the only the turbidity to the COD and doesn't include the specific EC measurements.

The surface of $\log COD_{eq} = 0.15(\log T)^2 - 1.54\cdot10^{-4}C^2 - 0.42\log T + 0.12C - 18.09$ *(equation 4.2)* (fit 1) is shown in the scatter plot of the turbidity, the specific EC and the COD in Figure 4.11.

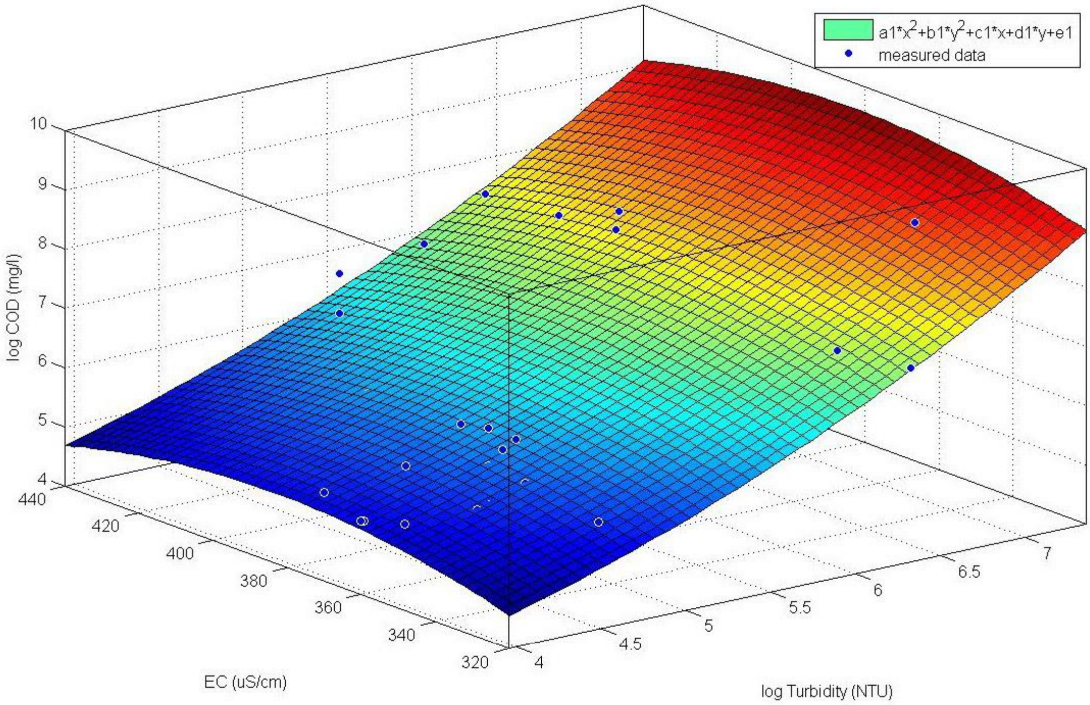


Figure 4.11 - Fit 1 of the field test

The line of $\log COD_{eq} = 1.09\log T + 0.61$ *(equation 4.3)* (fit 8) is shown in the scatter plot of the turbidity and the COD in Figure 4.12.

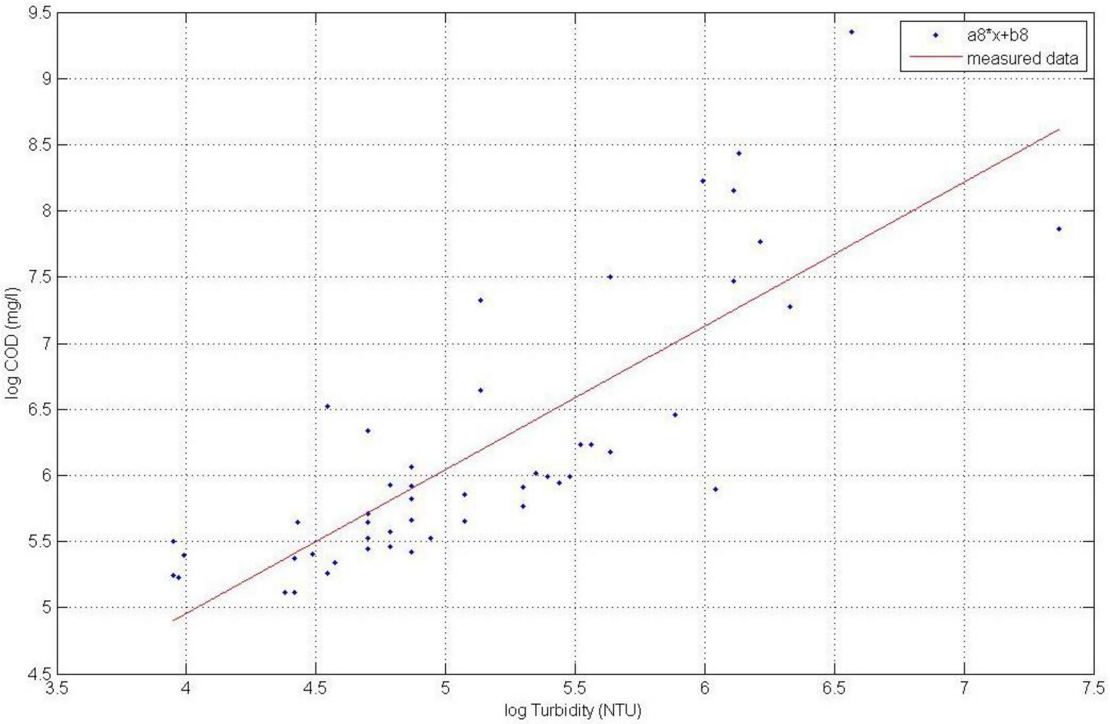


Figure 4.12 - Fit 8 of the field test

The results of all fits are attached as appendix VII. Fit 8 is preferred because of the lower number of parameters.

4.6. Discussion of the field test results

As mentioned earlier the Levenberg-Marquart algorithm finds a local minimum and not (per se) a global minimum. This study copes with some uncertainties and with many variations in the sample water matrix. Taking these into account it becomes clear that a perfect correlation cannot be achieved when dealing with wastewater flows. A good solution will do and this is exactly what the Levenberg-Marquart algorithm provides.

The results of the analysis show that the contribution of the specific EC to the COD is small. A fair prediction of the COD can be achieved without measuring the specific EC. This is in sharp contrast with the available literature (Lombard, 2010, Daal van, 2013). Location specific effects may be the cause of the differences, but this is not proven.

All the samples for the field test were taken on a single location. The wastewater matrix is known to be different for different locations, especially if the locations are situated in different sewer systems. The results of this study are therefor not applicable for other locations. On the other hand, there is no reason to assume that the general conclusions of this research are not valid for other (similar) sewer systems.

The analysis is performed on a data set with a certain range of parameters. The analysis is not valid outside this range and the uncertainties are likely to increase near the minimum and maximum parameter values. The results could be useful outside of the range of the data set, but extrapolating can have unexpected and unaccounted for effects. The data set boundaries are:

- a turbidity in the range of 52 to 1580 NTU
- a specific EC in the range of 320 to 440 µS/cm
- a COD in the range of 166 to 11500 mg/l

5. Conclusions and Recommendations

5.1. Conclusions

The most important conclusions of the research are summarized in this chapter.

The main question of this research was: "Can combined turbidity and conductivity measurements be used to accurately estimate the COD of wastewater in a Dutch sewer?" It is concluded that measurements of specific EC and turbidity can be used to estimate the COD of wastewater to some extend. The r^2 of the correlation found during this research is only about 0.65-0.69. The r^2 of the correlation between turbidity and COD is not unlike the r^2 values in literature. The r^2 of the correlation between turbidity and conductivity and COD is much lower then the r^2 values from Lombard et al (2010), but the r^2 of the fits with only turbidity is in line with the literature values for turbidity-COD relations (Lorenz, 2001, Moens, 2001, Veldkamp, 2002)

The turbidity is dominant in the estimation of the COD and the added value of the conductivity measurements is limited. Although the conductivity could be used to make the fit a bit better, the improvement is normally not worth the effort of the additional measurements and may even be insignificant. This is in contrast with the expectations leading to the research question and some literature in many cases.

The highest correlation is found with quadratic fits (especially $\log COD_{eq} = 0.15(\log T)^2 - 1.54 \cdot 10^{-4} C^2 - 0.42 \log T + 0.12C - 18.09$), but the r^2 of linear fits is only slightly lower then the r^2 of quadratic fits. Since the linear fit with only turbidity ($\log COD_{eq} = 1.09 \log T + 0.61$) has a similar r^2, but is simpler, the use of the linear fit is preferred. The conclusion that the added value of conductivity is limited is in contrast with the findings in literature.

The automated turbidity sensors are hard to maintain and to keep operational. Due to the problems with the turbidity sensors during this study the equipment was eventually abandoned and only samples were used to collect data. The equipment used for this study was old and second hand, but problems with turbidity sensors feature leading roles in other studies for turbidity measurements in wastewater as well.

5.2. Recommendations

Based on the research the following recommendations are done:

- Widespread use of turbidity measurements to estimate the COD of wastewater is not recommended. The correlation between turbidity and COD is too low for many purposes, such as estimating the pollution load on surface waters. The correlation is sufficient for some applications that rely mainly on the knowledge of the pollution load development over time instead of absolute COD values. The use of turbidity measurements for real time control (RTC) is a good example of this.

- The study results suggest that using conductivity measurements next to turbidity measurements to improve the estimation of COD increases the fit only slightly. It is recommended to assess whether the additional costs of the conductivity measurements outweigh the benefit of the improved estimate before installing conductivity sensors.

- The difference in findings between this research and similar research in literature is not clearly understood. Further research to explain the differences is recommended.

- Allocate enough time for maintenance of the turbidity sensors if they are used in wastewater.

- Repeating the study at several other locations will show if the results of the study are unique for the location at Ulvenhout or general.

- The correlation between the proxys and COD could be stronger and weaker for different types of storm events. Further research on this subject is advised.

Literature

Aumond M. and Joannis C., 21-26 August 2005, Turbidity monitoring in sewage, 10th International Conference on Urban Drainage, Copenhagen

Bertrand-Krajewski J.-L., Winkler S., Saracevic E., Torres A., Schaar H., 2007, Comparison of and uncertainties in raw sewage COD measurements by laboratory techniques and field UV-visible spectrometry

Butler D. and Davies J.W., 2004, Urban Drainage 2nd edition, London

CIW, Commissie Integraal Waterbeheer, June 2001, Riooloverstorten deel 2: Eenduidige basisinspanning, Nadere uitwerking van de definitief van de basisinspanning

Clemens F.H.L.R., 2001, Hydrodynamic models in urban drainage: application and calibration, Delft

CUWVO, Coördinatiecommissie Uitvoering Wet Verontreiniging Oppervlaktewateren, Group VI, 1992, Overstortingen uit Rioolstelsels en Regenwaterlozingen

Daal van-Rombouts P., Clemens FHLR, Veldhuis ter M.-C., Pothof I. and Langeveld J., October 2012, Congresverslag Urban Drainage Modelling 9 te Belgrado, WT-Afvalwater year 12 nr. 5, Valkenswaard

Daal van-Rombouts P., Schilperoort R.P.S., Langeveld J. and Clemens F.H.L.R., 2013, CSO pollution analysis based on conductivity and turbidity measurements and implications for application of RTC, Novatech 2013, Lyon

Davies-Colley R.J., Vant W.N. and Smith D. G., 1993, Colour and clarity of natural waters : science and management of optical water quality, London

Dirksen J, Goldina A., Korving J.L., 26 June 2007 , (Un)certainty in sewer inspection data: the applicability for sewer detoriation modeling, Rioleringswetenschap year 7 nr. 26, Valkenswaard

Feitenonderzoek Commissie, 29 March 2010, Doelmatig beheer waterketen, final report

Flamink CML, 28 October 2004, Afvalwatersysteemoptimalisatie in het licht van de Kaderrichtlijn Water, Proceedings of the 4th Symposium Interacties binnen het afvalwatersysteem, Delft

Fletcher T.D., and Deletic A., 2007, Statistical observations of a storm water monitoring programme; lessons for the estimation of pollutant loads

Gruber G., Winkler S. and Pressl A., 2004, Quantification of pollution loads from CSOs into surface water bodies by means of online techniques, 2004

Grünning H. and Althoff M., February 2001, Bepaling van vuilvrachten uit bergingsbassins, Rioleringswetenschap year 1, nr. 1, Valkenswaard

International Organization for Standardization, 1999, ISO 7027: 1999 (E) Water quality - Determination of turbidity, Geneva

Hach, February 1999, Method 8195, Determination of turbidity by nephelometry, Revision 2.0, Loveland,Colorado

Häck M., February 2001, Het gebruik van continue kwaliteitsmetingen binnen rioolsystemen, Rioleringswetenschap year 1, nr. 1, Valkenswaard

Henckens G.J.R., November 2001, On monitoring of turbidity in sewers, MSc thesis, Delft

Henckens G.J.R. and Schuit A.D., March 2002, On monitoring of turbidity in sewers, Rioleringswetenschap en Techniek year 2, nr. 5, Valkenswaard

Irvine K.N., McCorkhill G. and Caruso J., 2005, Continuous monitoring of conventional parameters to assess receiving water quality in support of combined sewer overflow abatement plans

Lacour C., Joannis C., Chebbo G., May 2009, Assessment of annual pollutant loads in combined sewers from continuous turbidity measurements: Sensitivity to calibration data

Langergraber G., Fleischmann N., Hofstadter F., 2003, A multivariate calibration procedure for UV/VIS spectrometric quantification of organic matter and nitrate in wastewater

Langeveld J, Februari 2011, Doelmatigheid en onderzoeksbehoefte riolering: het kennisprogramma Urban Drainage van start, WT·Afvalwater year 11 nr. 1, Valkenswaard

Lepot M., Bertrand-Krajewski J.-L. and Aubin J.B., Belgrade, 2012, Accuracy of different sensors for the estimation of pollutant concentrations (Total Suspended Solids, total and dissolved Chemical Oxygen Demand) in wastewater and storm water, 9th International Conference on Urban Drainage Modelling

Levlin E., 2009, Conductivity measurements for controlling municipal wastewater treatment, International scientific seminar, Research and application of new technologies in wastewater treatment and municipal solid waste disposal, Stockholm

Lombard V., Toloméo S., Bertrand-Krajewski J.-L., Debray R. ,Comte C., de Bénédittis. J., 2010, Design and operation of pollutant loads monitoring stations for an integrated approach of sewer systems

Lorenz U., February 2001, De inzet van een continue CZV-meting (UW-sonde) voor de controle vanbedrijfslozingen, Rioleringswetenschap year 1, nr. 1, Valkenswaard

Marquart D.W., 1963, An algorithm for the least square estimation of non linear parameters, Journal of the Society for Industrial and Applied Mathematics

Ministry of Infrastructure and environment, 13 June 1979, Wet Milieubeheer, The Hague

Ministry of Infrastructure and environment, 1 December 1970, Wet Verontreiniging oppervlaktewateren, The Hague

Ministry of Infrastructure and Environment, 12 March 2009, Waterwet, The Hague

Moens M.R., February 2001, De inzet van troebelheidsmetingen voor het meten van CZV en zwevende stiffen in gemengde rioolstelsels, Rioleringswetenschap year 1, nr. 1, Valkenswaard

Mudde T, Poep is goud, 17 November 2012, Volkskrant Wetenschap, Amsterdam

Nederland Normalisatie Instituut, July 2003a, NEN-EN13508-2:2003 Investigation and assessment of drain and sewer systems outside building - Part 2: Visual Inspection Coding System, Delft,

Nederland Normalisatie Instituut, December 2003b, NEN-EN13508-1:2003 Condition of drain and sewer systems outside buildings- Part 1: General Requirements, Delft

NLingenieurs, Positioningpaper riolering, 24 March 2010, The Hague

RIONED Foundation, 2002, Het bepalen van CZV- en zwevendstofgehalte in rioolwater op basis van troebelheid

RIONED Foundation, August 2005, Rioleringsatlas van Nederland, Ede

RIONED Foundation, August 2004, Leidraai Riolering, module C2100: Rioleringsberekeningen, hydraulisch functioneren, Ede

RIONED Foundation, July 2010a, Rionedreeks 14: handreiking emissiemetingen, bepaling van buigemiddelde concentraties van relevante emissies uit het afvalwatersysteem en registratie van systeem- en gebiedskenmerken, Ede

RIONED Foundation, November 2010b, Riolering in beeld, Benchmark rioleringszorg 2010, Ede

Sadar, M.J., 1998, Turbidity Science

Schellart A., October 2003, Sewer sludge measurements in the field and under laboratory conditions, Rioleringswetenschap en techniek year 3, nr 11, Valkenswaard

Schilperoort, R.P.S., 2007a, Temperatuur en geleidbaarheid als indicatie parameters voor verdunning in het rioolsysteem, 59e Vakantiecursus in Drinkwatervoorziening en 26e vakantiecursus in Riolering en Afvalwaterbehandeling, Delft

Schilperoort R.P.S., May 2007b, Temperature and conductivity as indicator parameters for the dilution process of wastewater, Afvalwaterwetenschap, year 6 nr. 2, Valkenswaard

Schilperoort, R.P.S., 2011, Monitoring as a tool for the assessment of wastewater quality dynamics, PhD thesis, Delft

Schilperoort R.P.S., Sikkes M., Weijers S. and Langeveld J., February 2012, Lessons learnt during 6 years of RTC-research at water board De Dommel:2004-2009. Performance of a monitoring network. Performance of a wastewater system, WT·Afvalwater yeat 12 nr 1, Valkenswaard

Tchobanoglous G., Burton Franklin L. and Stensel, H. David, 2003, Metcalf&Eddy, Wastewater Engineering, treatment and reuse, fourth edition, New York

U.S. Environmental Protection Agency, August 1993, Method 180.1 Determination of turbidity by nephelometry , revision 2.0, Cincinnati

Veldkamp R.G. and Moens M.R., December 2002, The use of turbidimeters to determine COD and suspended solids in combined sewer systems, Rioleringswetenschap en -Techniek year 2, nr. 8, Valkenswaard

Veldkamp R.G., Henckens G.J.R., Langeveld J. and Clemens F.H.L.R., March 2003, Field measurement of turbidity related to sediment transport in sewer systems, Rioleringswetenschap- en Techniek, year 3 nr 9, Valkenswaard

Ven, van der F.H.M., January 2007, Watermanagement in Urban Areas, Delft

Walter S., 2008, Conceptualizing overfit or overparameterization, Rasch Measurement Transactions 22:2 p. 1165

Waterschap Brabantse Delta, 26 May 1997, drawing: Transportleiding Ulvenhout-Breda, Rioolgemaal Ulvenhout

Witteveen+Bos, 3 June 2009, Basisrioleringsplan Ulvenhout, Breda

WRW, Werkgroep Riolering West-Nederland, 1996, Regels voor het bepalen van de vuiluitworp uit gemengde stelsels

Appendices

Appendix I Calibration of turbidity meters

The turbidity meters are calibrated to a lab turbidity meter. First the lab turbidity meter is calibrated and then the calibrated lab turbidity meter is used to calibrate the field turbidity meters.

First the calibration procedure is explained and then the results of the calibration are presented and discussed.

I.1. Calibration of lab turbidity meter

The calibration of the lab turbidity meters is done using a Stablcal Stabilized Formazin Turbidity Standards kit of Hach Company. The Stablcal kit is cited as a primary calibration standard in Hach method 8195 (Hach, 1999). Hach method 8195 is an USEPA accepted version of USEPA method 180.1 (USEPA, 1993) for the calibration of turbidity meters. The Stablcal kit contains formazin standards of <0.1, 20 (+/- 1), 200(+/-10), 1000(+/- 50) and 4000(+/- 200) NTU. The 4000 NTU standard is not used to calibrate the lab turbidity meter. The field turbidity meters range is approximately 0 to 500 NTU and only the first standard exceeding the maximum of the range of the field turbidity meter contributes to the calibration of the instrument.

The lab turbidity meter has calibrated before every experiment and checked for degradation after every experiment with a similar procedure. None of the calibrations or checks resulted in deviations exceeding the Stablcal kits accuracy.

The Stablcal kit is valid until September 2010. This means that the kit is formally expired at the time of use. Since no alternative calibration is available and the calibration residues are small, the kit is assumed to be still accurate enough for the calibration.

I.2. Procedure

The field turbidity meters are fixed in a vessel with a diameter of 51 cm. The vessel is filled with approximately 30 l of demineralized water. The experiment set up is shown in Figure I.1.

Figure I.1a and b - Set up of calibration experiment. Both of the sensors are fixed in a vessel with water that is stirred during the experiment

The turbidity of the water is measured with both of the field turbidity meters at an interval of 8s. The flask of the lab turbidity meter is rinsed three times with the water and then a sample is taken and measured with the lab turbidity meter. After measuring the turbidity with the lab turbidity meter, the sample is returned to the vessel. Every five minutes a sample is taken and analyzed for turbidity.

After the fourth sample is acquired, a small amount of Goudse Clay is added to the water to increase the turbidity. Mechanical stirring equipment is used during the entire test to keep the Goudse Clay suspended. The stirring equipment revolves at 108 to 299 rpm during the experiment depending on the Goudse Clay concentration. The turbidity measurements are repeated as described before.

The experiment is repeated until the maximum range of the sensor is reached.

I.3. Number of experiments

The calibration curve is known to be linear. The number of performed experiments determines the accuracy of the calibration. Literature is not consistent on the number of experiments needed for a calibration. Table I.1 shows several examples.

Table I.1 - Minimum sample size for calibration according to literature

source	sample size
ISO standard 11095:1996	3
ISO standard 15302:2007	4
EURACHEM guide	6
Commission Decision 2002/657/EC	6
LGC/VAM guide "Preparation of Calibration Curves	7
ISO 8466-1:1990	10

Assuming a normal distribution of the deviations, the relation between the number of experiments and the confidence interval is calculated by:

$$P[\overline{X} - z(\alpha/2)\frac{\sigma}{\sqrt{n}} \leq \mu \leq \overline{X} + z(\alpha/2)\frac{\sigma}{\sqrt{n}}] = 1 - \alpha \qquad \textit{(equation I.1)}$$

in which:
α = the percentile within the confidence interval
n = the number of experiments
$z(\frac{\alpha}{2})$ = the z-score of percentile α
σ = the standard deviation
μ = the unknown parameter
X = the result of the experiment

This means that the required number of experiments is:

$$n \geq \frac{\sigma^2 \cdot (z(\alpha/2))^2}{h^2} \qquad \textit{(equation I.2)}$$

in which:
h = the confidence interval

The required number of experiments depends on the standard deviation σ, the confidence percentile α and the confidence interval h. The standard deviation is unknown until the experiment is performed, but an estimation of the standard deviation is possible. The standard deviation is estimated at 10 NTU. The confidence percentile is a bit arbitrary. For this experiment a confidence percentile of 95% is used, resulting in a z(α/2) of 1.96. The confidence interval is also a bit arbitrary and chosen at 10 NTU. This results in a minimum sample size of 4. To make sure that enough samples are taken for the calibration a safety factor of 2.5 is used. A minimum of 10 experiments is performed.

I.4. Results

Based on the measurements the calibration parameters are fitted using least squares. The results of the experiments and the fit are shown in Table I.2 and Figure I.2 for both turbidity meters.

Table I.2 - Calibration results and residues. The Table shows the turbidity measured with the lab turbidity meter (tur.), the current signal of the field turbidity sensors (cur.), the fitted turbidity estimate (est.) and the residue of the fit (res.). The results and residues are also shown in Figure I.2

Sensor 1					**Sensor 2**				
nr.	**tur. [NTU]**	**cur. [mA]**	**est. [NTU]**	**res. [NTU]**	**nr.**	**tur. [NTU]**	**cur. [mA]**	**est. [NTU]**	**res. [NTU]**
1	0.37	4.31	-6.79	7.17	1	0.37	4.22	-16.35	16.72
2	0.76	4.31	-6.79	7.56	2	0.76	4.41	-9.51	10.27
3	10.40	4.71	6.46	3.94	3	10.40	4.67	-0.17	10.57
4	27.93	5.26	24.61	3.32	4	77.86	6.91	80.51	-2.65
5	77.86	7.00	82.50	-4.64	5	147.50	9.03	157.05	-9.55
6	147.50	9.16	153.96	-6.46	6	221.50	11.43	243.34	-21.84
7	221.50	11.62	235.48	-1.40	7	295.50	13.44	315.77	-20.27
8	295.50	13.65	302.87	-7.37	8	364.25	15.41	386.79	-22.54
9	364.25	15.77	373.09	-8.84	9	435.00	17.31	454.94	-19.94
10	435.00	17.83	441.40	-6.40	10	532.00	19.11	520.11	11.89
11	532.00	19.79	506.29	25.71	11	554.25	19.41	530.88	23.37
					12	566.25	19.73	542.28	23.97

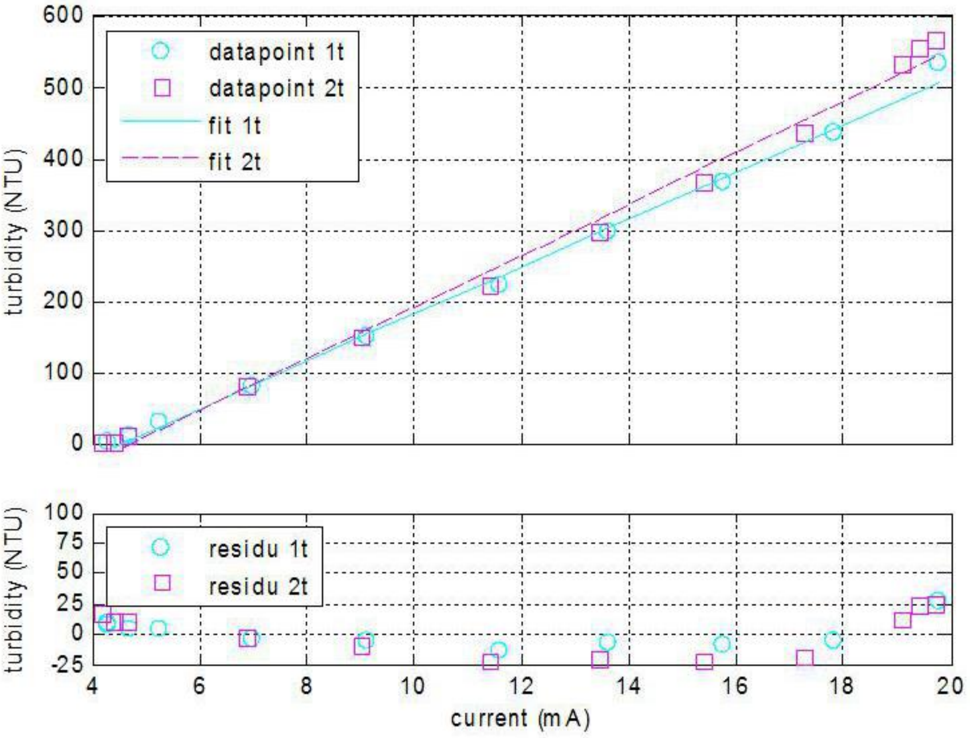


Figure I.2 - Calibration results and residues. Figure I.2a shows datapoints of the average current measured with the field turbidity meters versus the turbidity measured with the lab turbidity meter and the fit of both sensors. Figure I.2b shows the residues of the fit. The results and residues are also shown in Table I.2

The parameters for the fit are shown in Table I.3.

Table I.3 -Parameters of fit. The Table shows the parameters α and β, the variance and covariance of α and β, the maximum total variance, the r^2, the adj r^2, the root mean square error and the bias

sensor	**α**	**β**	**σ_α**	**σ_β**	**$\sigma_{\alpha\beta}$**	**$\sigma_{T;max}$**	**r^2**	**adj r^2**	**RMSE**	**bias**
1	-149.62	33.14	50.7	0.37	-3.83	6.68	0.997	0.996	3.18	-0.085
2	-168.34	36.02	144.8	0.80	-9.69	8.79	0.993	0.992	5.03	0.594

The Table shows that the standard deviation, $\sigma_{T;max}$, is 6.68 NTU for sensor 1 and 9.22 NTU for sensor 2. Based on the standard deviation and the formulas given in section I.3 the required number of experiments is calculated. This is respectively 2 and 3 for sensor 1 and 2 to have a 95%-confidence interval of 10 NTU. Given the number of experiments performed the 95%-confidence interval is: 3.94 and 4.97 NTU for respectively sensor 1 and 2.

I.5. Discussion

Earlier experiments showed a decline of the turbidity over time. This may be due to sedimentation of particles in the vessel and in the drain of the vessel. Sedimentation in the vessel is minimized by increasing the revolutions of the stirring device as the mass of Goudse Clay in the vessel increases. The sedimentation in the drain of the vessel is estimated to be insignificant.

As the number of revolutions increases, the variance in the turbidity measurements also increases. This may have several causes. The harder working stirring device or heavier stirring could interfere with the measuring equipment or the heavier stirring could cause unforeseen fluctuation in the turbidity due to turbulence effects. The variance is shown in Figure I.3 as a function of the revolutions and the mean measured current.

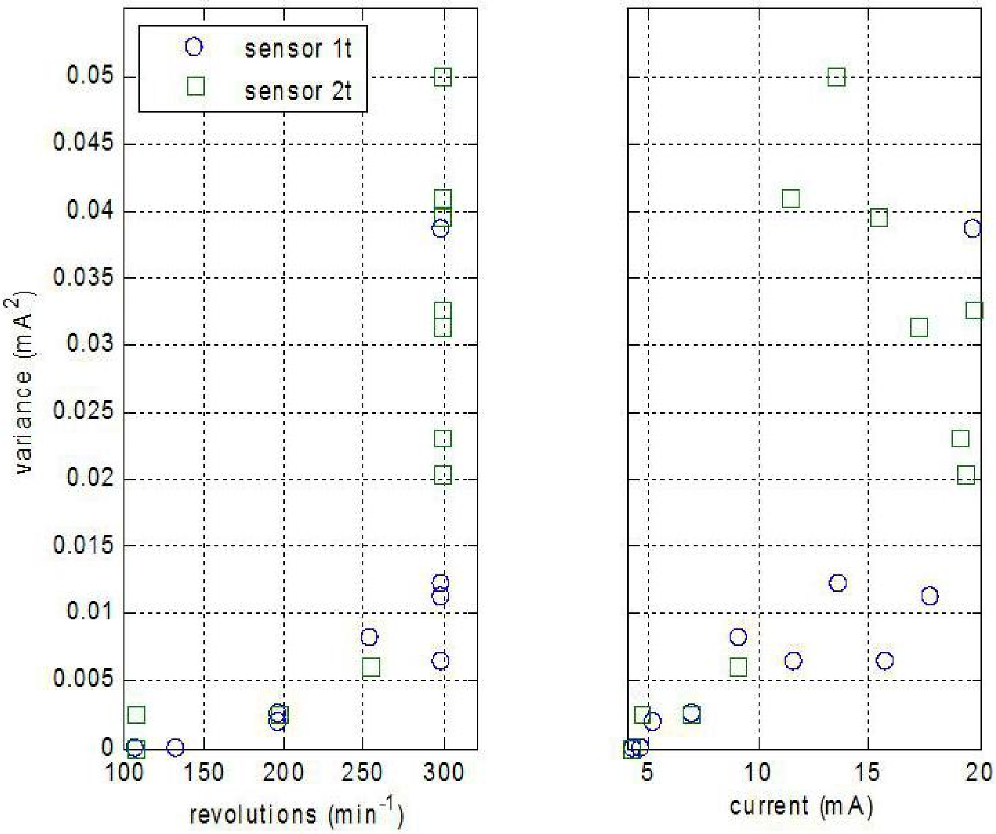


Figure I.3 - Variance in measured current. The variance increases with the number of revolutions per minute (Figure I.3a). There is no clear direct relation between the variance and the measured current (Figure I.3b)

Appendix II Errors in turbidity measurements

Errors in the measurements originate from the following sources

- sensor errors
- sampling errors

II.1. Noise

Noise is an unwanted perturbation in the signal of a sensor. The noise gives fluctuations in the data with no meaning. The noise thereby introduces uncertainty to the measurements. The most important sources of noise is thermal noise. Thermal noise, or Johnson-Nyquist noise, is caused by random movement of electrons. At lower temperatures the electrons movement is limited, but at higher temperatures the electrons movement is less restricted. This may cause electrons to move randomly and stray from the desired path. The electrons are used to carry the current signal. Random movement of the electrons affects the level of the current and thus the signal.

Given certain instruments noise is hard to fight, but it is possible to measure the extent of the noise. This is done by measuring a water sample with a certain constant water quality over some time. Because of the constant water quality, the signal should also stay constant, with the exception of the noise.

The amplitude of the signal is the noise of the signal. The noise is assumed to be white noise. White noise is a random effect so it should be normally distributed. According to the law of large numbers: if the number of measurements is large enough, the mean of the signal is the value that is representative for the measured value. It is calculated with:

$$\bar{c}_n = \frac{\sum_{i=1}^{n} c_{n;i}}{n} \qquad \text{(equation II.1)}$$

in which:

$\bar{c}_n$ = the mean value of the noise measurements

$c_{n;i}$ = the ith value measured by the sensor

n = the number of measurements

The uncertainty caused by noise is also known as sensor uncertainty. The variance is the measurement for the uncertainty of the noise and can be calculated with:

$$\sigma_n{}^2 = \frac{\sum_{i=1}^{n} (c_{n;i} - \mu_n)^2}{n-1} \qquad \text{(equation II.2)}$$

in which:

σ_n^2 = the variance for the noise measurements

To check if the noise is indeed white noise a residuals analysis is done. This analysis determines if the noise is normally distributed. This means that the measurements should resemble the normal distribution $N(\bar{c}_n, \sigma_n{}^2)$. The model distribution function of the measurements is:

$$F(x) = \int_{-\infty}^{x} \frac{1}{\sigma_n \sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\bar{c}_n}{\sigma_n}\right)^2} dx \qquad \text{(equation II.3)}$$

in which:
F(x) = the model distribution function of the noise measurements

The empirical distribution function of the measurements is:

$$F_n(i) = \frac{c_{n;i}}{n} \qquad \text{(equation II.4)}$$

in which:
$F_n(i)$ = the empirical distribution function of the measurements
$c_{n;i}$ = the value of the x^{th} measurement

If the number of measurements is large enough and assuming that our model distribution is chosen correctly:

$$F_n(i) \approx F(x) \quad \text{for x=i} \qquad \text{(equation II.5)}$$

The sample Pearson correlation coefficient or Pearson's r gives an indication of the goodness of fit of the model distribution function to the empirical distribution function. The closer Pearson's r-squared gets to unity, the better the fit. Pearson's r is calculated with:

$$r = \frac{n \cdot \sum_{i=1}^{n}(F_{n;i} \cdot F_i) - \sum_{i=1}^{n} F_{n;i} \cdot \sum_{i=1}^{n} F_i}{\sqrt{\left[n \cdot \sum_{i=1}^{n}(F_{n;i}{}^2) - (\sum_{i=1}^{n} F_{n;i})^2\right] \cdot \left[n \cdot \sum_{i=1}^{n}(F_i{}^2) - (\sum_{i=1}^{n} F_i)^2\right]}} \qquad \text{(equation II.6)}$$

in which:
r = the sample Pearson correlation coefficient

II.2. Systematic sensor variations

The sensors express their measurement in a value between about 4 mA and 20 mA. A value of 0 mA is logged when the datalogger is operational, but receives no signal from the sensor. The current signal is ideally linearly linked to the measurement quantity. For example: the turbidity is measured between 0 NTU and 500 NTU. The relation is expressed as:.

$$T=(I-4)\times\frac{500}{16}=-0.128+0.032\times I \qquad \text{For } 4 \le I \le 20 \text{ mA} \qquad \text{(equation II.7)}$$

$$\text{no signal} \qquad \text{for } I \le 4 \text{ mA or } I \ge 20 \text{ mA}$$

in which:
T = the turbidity (NTU)
I = the current (mA)

Due to various reasons, the relation above is not exactly true for every sensor. Small sensor specific differences can occur. These differences can cause systematic errors in the results. Therefore it is important to calibrate each sensor and establish a sensor specific relation to gain a higher accuracy (Schellart, 2002). The sensors are calibrated before use and after the field experiment. The calibration after the field experiment is to check the degeneration of the sensors. The calibration is done in two steps.

Step one is the calibration of laboratory instruments. The turbidity meter has been calibrated with formazin solutions in accordance to the procedure supplied by the instrument manufacturer. The calibration of the specific EC meter was performed according to the ISO 7888 standard with a standard solution. The calibration of the laboratory instruments is repeated after step two to see if there has been any degeneration during step two.

In step two the field equipment is calibrated to the laboratory equipment. Two series of calibration samples are prepared. One with suspended Goudse Clay and one with dissolved sodium chloride. Respectively the turbidity and the specific EC of the calibration samples are determined with the calibrated laboratory instruments and with the field sensors. The multiple measurements are done to filter out noise.

It is assumed that the relation can be modeled as linear relation.

$$T = \hat{\alpha} + \hat{\beta} \cdot I \qquad \textit{(equation II.8)}$$

in which:
$\hat{\alpha}$ = estimated intercept parameter (NTU/mA)
$\hat{\beta}$ = estimated slope parameter (NTU)

Parameters $\hat{\alpha}$ and $\hat{\beta}$ are estimated by linear regression using a least squares approach. This approach gives:

$$\hat{\beta} = \frac{n \cdot \sum_{i=1}^{n}(c_{c;i} \cdot c_{l;i}) - \sum_{i=1}^{n} c_{c;i} \cdot \sum_{i=1}^{n} c_{l;i}}{n \cdot \sum_{i=1}^{n} c_{c;i}^{2} - (\sum_{i=1}^{n} c_{c;i})^{2}} \qquad \textit{(equation II.9)}$$

and

$$\hat{\alpha} = \frac{\sum_{i=1}^{n} c_{l;i} - \hat{\beta} \cdot \sum_{i=1}^{n} c_{c;i}}{n} \qquad \textit{(equation II.10)}$$

in which:
n = the number of measurements
$c_{l;i}$ = the i^{th} value measured by laboratory equipment
$c_{c;i}$ = the i^{th} value measured by the field sensor

Pearson's r gives an indication of the goodness of fit of the regression relation. Pearson's r is calculated with:

$$r = \frac{n \cdot \sum_{i=1}^{n}(c_{c;i} \cdot c_{l;i}) - \sum_{i=1}^{n} c_{c;i} \cdot \sum_{i=1}^{n} c_{l;i}}{\sqrt{\left[n \cdot \sum_{i=1}^{n}(c_{c;i}^{2}) - (\sum_{i=1}^{n} c_{c;i})^{2}\right] \cdot \left[n \cdot \sum_{i=1}^{n}(c_{l;i}^{2}) - (\sum_{i=1}^{n} c_{l;i})^{2}\right]}} \qquad \textit{(equation II.11)}$$

in which:
r = the sample Pearson correlation coefficient

The parameters $\hat{\alpha}$ and $\hat{\beta}$ cannot be determined without some uncertainty. The mean and the variance of the calibration are calculated with:

$$\bar{c}_c = \frac{\sum_{i=1}^{n} c_{c;i}}{n} \qquad \textit{(equation II.12)}$$

and

$$\sigma_c^{\ 2} = \frac{\sum_{i=1}^{n} (c_{c;i} - \bar{c}_c)^2}{n-1} \qquad \textit{(equation II.13)}$$

in which:
$\bar{c}_c$ = the mean value of the calibration measurements
$c_{c;i}$ = the ith value measured by the field sensor
n = the number of measurements
σ_c^2 = the variance in the calibration

The calibration parameters α and β are not independent. Therefore the variance per parameter and the covariance between the two parameters also has to be calculated. Each of them follows from the covariance matrix:

$$\underline{\mathrm{cov}} = \sigma_c^{\ 2} \cdot (J^T \cdot J)^{-1} = \begin{pmatrix} \sigma_\alpha^{\ 2} & \sigma_{\alpha\beta}^{\ 2} \\ \sigma_{\alpha\beta}^{\ 2} & \sigma_\beta^{\ 2} \end{pmatrix} \qquad \textit{(equation II.14)}$$

in which:
$\underline{\mathrm{cov}}$ = the covariance matrix
σ_c^2 = the variance in the calibration
$\underline{J}$ = the Jacobean
σ_α^2 = is the variation of the calibration error of $\hat{\alpha}$
$\sigma_{\alpha\beta}^2$ = the covariance between the calibration parameters $\hat{\alpha}$ and $\hat{\beta}$
σ_β^2 = is the variation of the calibration error of $\hat{\beta}$

The Jacobean is defined as:

$$J = \begin{pmatrix} \frac{\partial r_i}{\partial \alpha} & \frac{\partial r_i}{\partial \beta} \\ \vdots & \vdots \\ \frac{\partial r_n}{\partial \alpha} & \frac{\partial r_n}{\partial \beta} \end{pmatrix} \qquad \textit{(equation II.15)}$$

in which:
r_i = the ith residual of the measurement, which is calculated with:

$$r_i = c_{l;i} - c_{c;i} \qquad \textit{(equation II.16)}$$

The uncertainty in the calibration parameters is also known as calibration uncertainty.

II.3. Sampling errors

The water that is sampled and analyzed is different from the water that is measured by the sensors. The samples are taken from the manhole where the sensors are installed and the time of sampling is logged carefully. Despite this fact, the water sampled is a bit different from the water that is measured by the sensors. To get an indication of the sampling error the following test is executed.

After the installation of a sensor, a sample is taken. Simultaneous to the sampling, measurements are done in the sewer. Measurements are done in the sample and compared to the measurements done in the sewer. To minimize errors due to measuring with different equipment, both measurements are done with the same type of equipment.

The sampling error appears to be a random effect so it is normally distributed. The mean and the variance can be calculated with:

$$\bar{c_s} = \frac{\sum_{i=1}^{n} c_{s;i}}{n} \qquad \text{(equation II.17)}$$

and

$$\sigma_s^2 = \frac{\sum_{i=1}^{n} (c_{s;i} - \bar{c_s})^2}{n-1} \qquad \text{(equation II.18)}$$

in which:
$\bar{c}_s$ = the mean value of the difference between the sensor measurements
$c_{s;i}$ = the i^{th} difference measured between the sensors
n = the number of measurements
σ_s^2 = the variance for sampling

II.4. Error propagation

The total uncertainty consists of the three sources of uncertainty: sensor uncertainty, calibration uncertainty and sampling uncertainty. The sensor uncertainty is also assessed during calibration. As a result it does not contribute to the total uncertainty separately. The parameters of the calibration are dependent on each other so their covariance has to be taken into account when calculating the total uncertainty. The other sources of uncertainty are independent. The total uncertainty is calculated with (Schilperoort, 2011):

$$\sigma_t^2 = \left(\frac{\partial f}{\partial c_\alpha}\right)^2 \cdot \sigma_\alpha^2 + \left(\frac{\partial f}{\partial c_\beta}\right)^2 \cdot \sigma_\beta^2 + \left(\frac{\partial f}{\partial c_s}\right)^2 \cdot \sigma_s^2 + \left(\frac{\partial f}{\partial c_\alpha}\right) \cdot \left(\frac{\partial f}{\partial c_\beta}\right) \cdot \sigma_{\alpha\beta}^2$$

(equation II.19)

in which:
σ_t^2 = is the variation of the total error
f = the calibration relation $= \hat{\alpha} + \hat{\beta} \cdot I$
$\hat{\alpha}$ = the intercept of the calibration relation
$\hat{\beta}$ = the slope of the calibration relation

I = the current measurement
c_c = the sensor measurement during calibration
σ_α^2 = is the variation of the calibration error of $\hat{\alpha}$
σ_β^2 = is the variation of the calibration error of $\hat{\beta}$
σ_s^2 = is the variation of the sampling error
$\sigma_{\alpha\beta}^2$ = the covariance between the calibration parameters $\hat{\alpha}$ and $\hat{\beta}$

This can easily be derived into:

$$\sigma_t^2 = \sigma_\alpha^2 + I^2 \cdot \sigma_\beta^2 + I^2 \cdot \sigma_s^2 + 2 \cdot I \cdot \sigma_{\alpha\beta}^2 \qquad \textit{(equation II.20)}$$

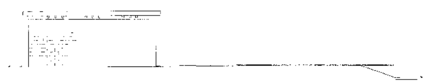
With the variation known confidence intervals can be established. The 95%-confidence interval is given as:

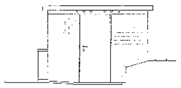
$$c_{95} = 2 \cdot 1.96 \cdot \frac{\sigma_t}{\sqrt{n}} \qquad \textit{(equation II.21)}$$

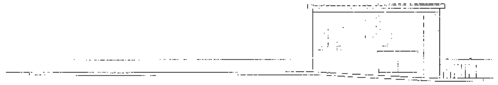
in which:
c_{95} = the size of the 95%-confindence interval
σ_t = the standard deviation estimation for noise
n = the number of measurements

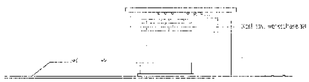
The 95%-confidence interval is always centered by the measurement.

Appendix III Drawing of main pumping station Ulvenhout

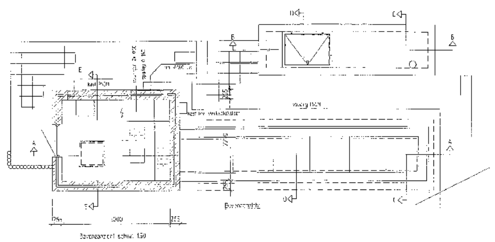

Z.W. Gevel schaal 1:50

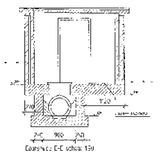

N.W. Gevel schaal 1:50

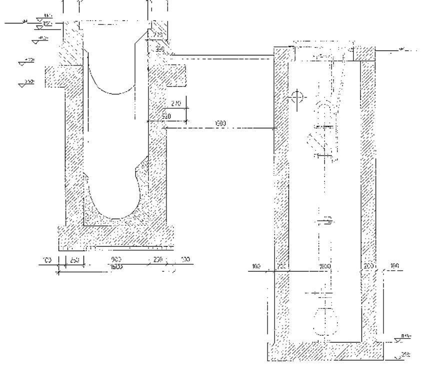

N.O. Gevel schaal 1:50

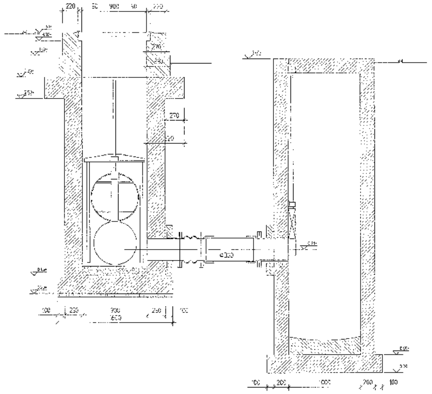


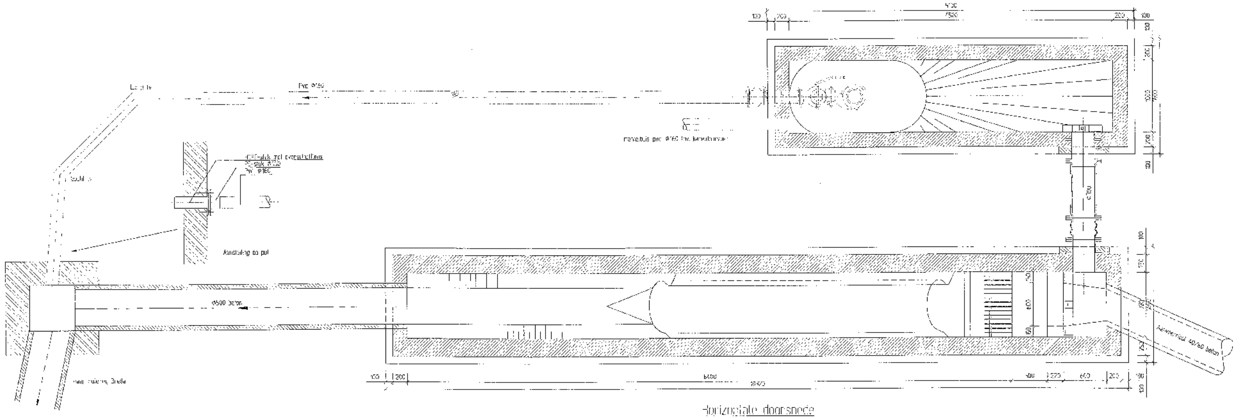
Z.O. Gevel schaal 1:50


Bovenaanzicht schaal 1:50

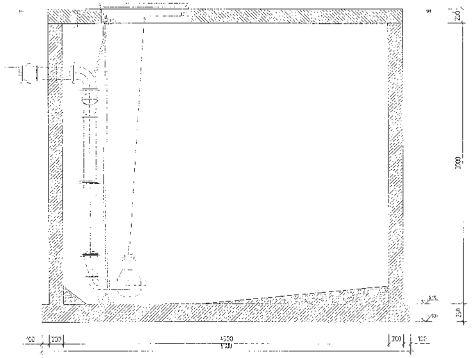

Doorsnede E-E schaal 1:50

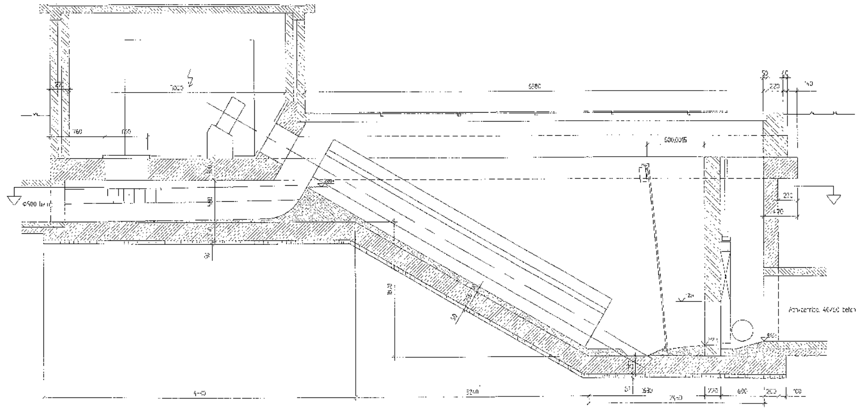

Doorsnede D - D


Doorsnede C - C

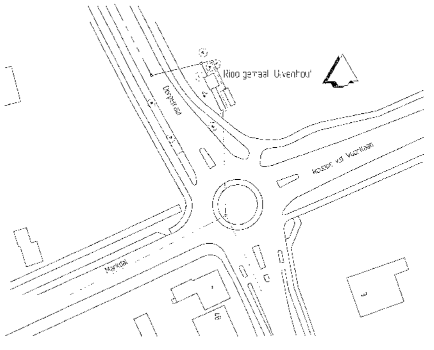


Horizontale doorsnede


Doorsnede B - B



Doorsnede A - A



Situatie schaal 1:500

Waterschap
Brabantse Delta

Opdracht
Transportleiding Uivenhout - Breda
Rioolgemaal Uivenhout

Tekening nr.
105 2 1

Appendix IV Results laboratory test

Table IV.1 and Table IV.1 (continued) show the data resulting from the laboratory test.

Table IV.1 - Results laboratory test

number	**Turbidity**	**EC**	**CODf**		**number**	**Turbidity**	**EC**	**CODf**
	NTU	mS/cm	mg/l			NTU	mS/cm	mg/l
1	0.0	27.0	0.00		26	43.9	696	43.22
2	0.0	147.7	7.03		27	42.3	1356	76.86
3	0.0	294	14.08		28	42.3	2660	145.58
4	0.0	427	20.42		29	43.7	3900	212.41
5	0.0	562	27.52		30	42.7	6380	349.90
6	0.0	697	34.39		31	61.3	1.9	13.44
7	0.0	1363	68.32		32	62.7	148.1	20.48
8	0.0	2710	138.72		33	69.8	287	27.23
9	0.0	3920	204.76		34	64.6	426	34.18
10	0.0	5130	271.45		35	64.5	556	40.60
11	19.0	1.1	4.50		36	67.9	696	47.38
12	26.2	146.9	11.47		37	69.2	1363	81.44
13	25.8	295	18.77		38	65.2	2660	149.37
14	24.5	427	25.34		39	64.7	3910	217.40
15	21.8	560	32.06		40	63.0	6340	351.91
16	24.6	692	38.84		41	70.4	5.5	17.95
17	18.8	1359	72.77		42	79.4	154.2	25.06
18	22.7	2670	141.17		43	75.6	293	31.79
19	19.2	4540	243.48		44	76.7	436	38.71
20	22.8	6380	346.30		45	79.0	570	45.31
21	40.5	2.2	9.03		46	85.9	706	51.98
22	40.2	148.5	16.10		47	83.3	1385	85.57
23	44.7	288	22.90		48	84.3	2700	153.33
24	42.9	426	29.80		49	82.4	3990	221.54
25	47.4	557	36.23		50	81.7	6420	353.01

Table IV.1 - Results laboratory test (continued)

number	**Turbidity**	**EC**	**CODf**		**number**	**Turbidity**	**EC**	**CODf**
	NTU	mS/cm	mg/l			NTU	mS/cm	mg/l
51	90.7	2.4	22.47		76	407	719	125.26
52	93.7	147.0	29.35		77	403	1419	159.94
53	104	286	36.10		78	407	2780	229.43
54	94.6	426	43.17		79	400	4090	299.58
55	105	557	49.48		80	401	6590	436.07
56	103	700	56.68		81	744	9.4	136.56
57	97.1	1371	90.87		82	728	160.3	141.25
58	100	2630	156.56		83	733	298	149.43
59	103	3300	181.37		84	697	444	156.85
60	98.3	6930	422.51		85	685	586	163.82
61	203	4.7	45.12		86	691	721	170.44
62	203	155.7	52.53		87	709	1429	204.83
63	201	291	58.79		88	786	2790	274.54
64	192	436	66.14		89	726	4060	341.79
65	207	574	72.94		90	728	6590	480.68
66	214	711	79.38		91	1446	14.5	219.34
67	201	1300	108.86		92	1468	162.9	231.89
68	198	2720	178.45		93	1424	304	235.24
69	204	4080	253.67		94	1415	443	244.83
70	201	6580	390.09		95	1428	580	250.61
71	407	7.6	86.37		96	1424	725	258.29
72	401	272	102.33		97	1443	1430	293.83
73	414	302	105.16		98	1410	2780	363.09
74	406	436	111.10		99	1392	4110	432.04
75	410	578	122.78		100	1352	6490	558.64

Appendix V Results fits laboratory test

Table V.1 shows the goodness of fit statistics of the laboratory test.

Table V.1 - Goodness of fit statistics of laboratory test. SSE is an abbreviation for sum squared error and RMSE is an abbreviation for root mean squared error

Fit	SSE	r^2	Adjusted r^2	RMSE
1	1890	>0.99	>0.99	4.46
2	9624	0.99	0.99	9.96
3	2714	>0.99	>0.99	5.34
4	$6.251 \cdot 10^5$	0.61	0.60	80.3
5	$7.82 \cdot 10^5$	0.51	0.51	89.4
6	$4.57 \cdot 10^5$	0.72	0.71	68.7

Figures V.1 to V.6 show the fits of the laboratory test, Figures V.7 to V.12 show the residuals of the fits of the laboratory test and Figures V.13 to V.18 show the contours of the fits of the laboratory test.

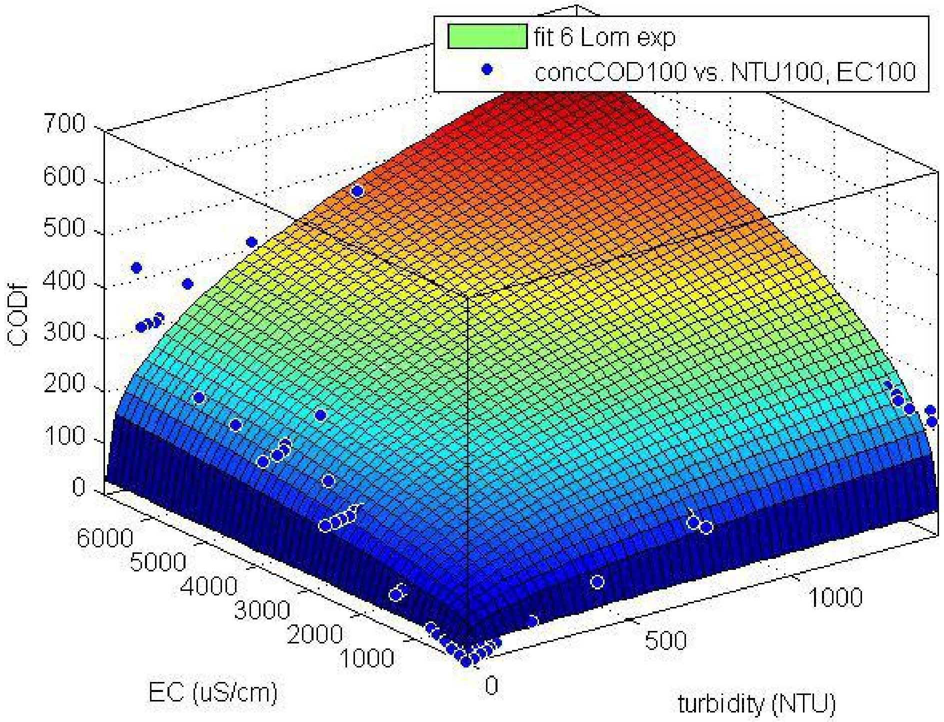


Figure V.1 - Fit 1

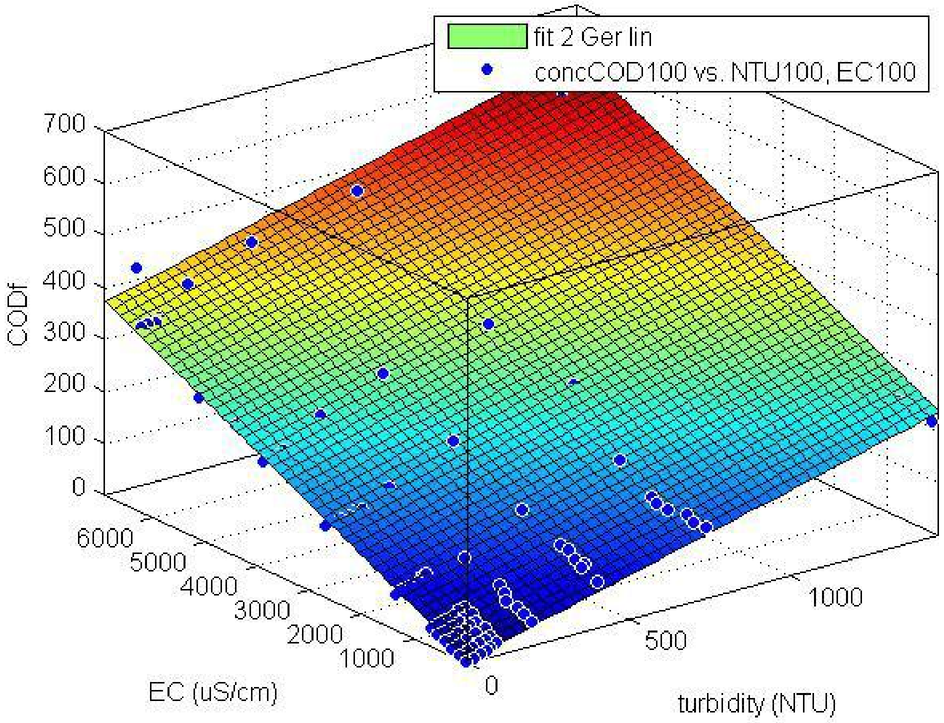


Figure V.2 - Fit 2

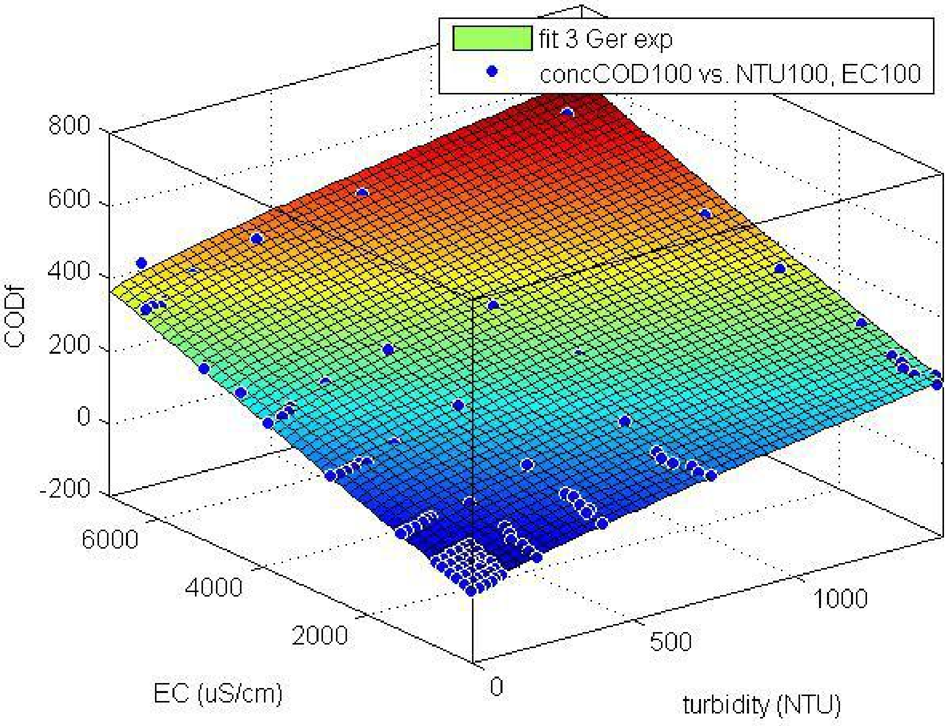


Figure V.3 - Fit 3

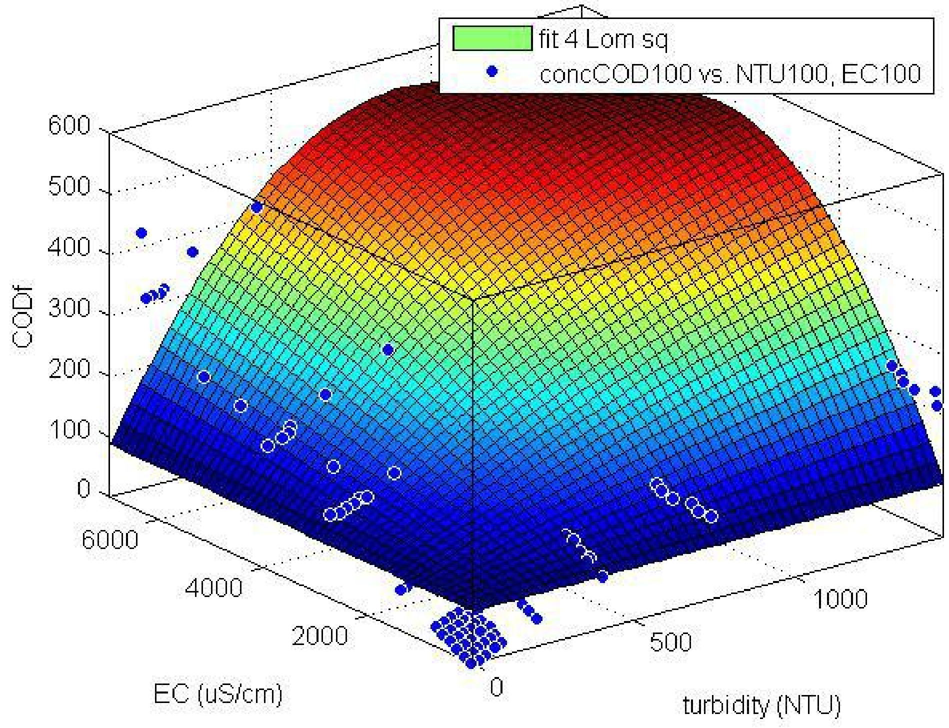


Figure V.4 - Fit 4

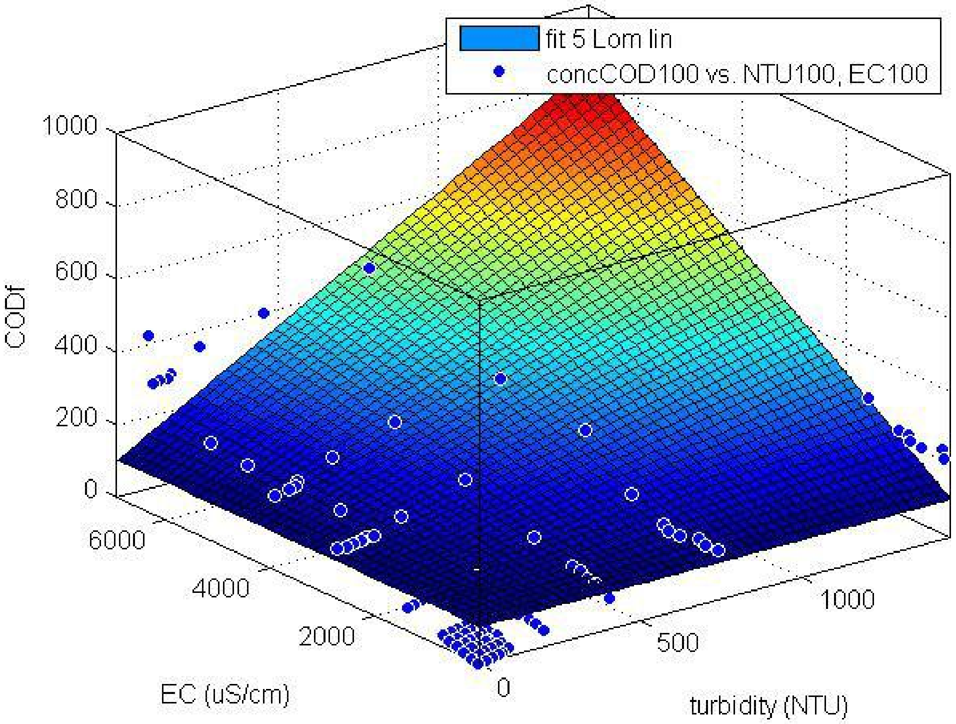


Figure V.5 - Fit 5

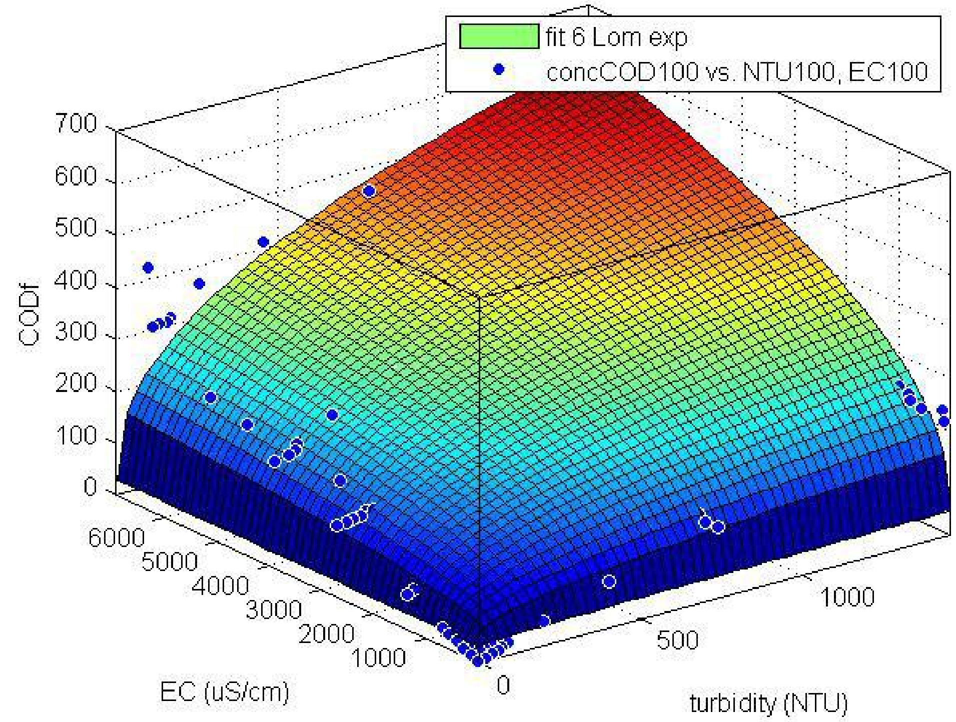


Figure V.6 - Fit 6

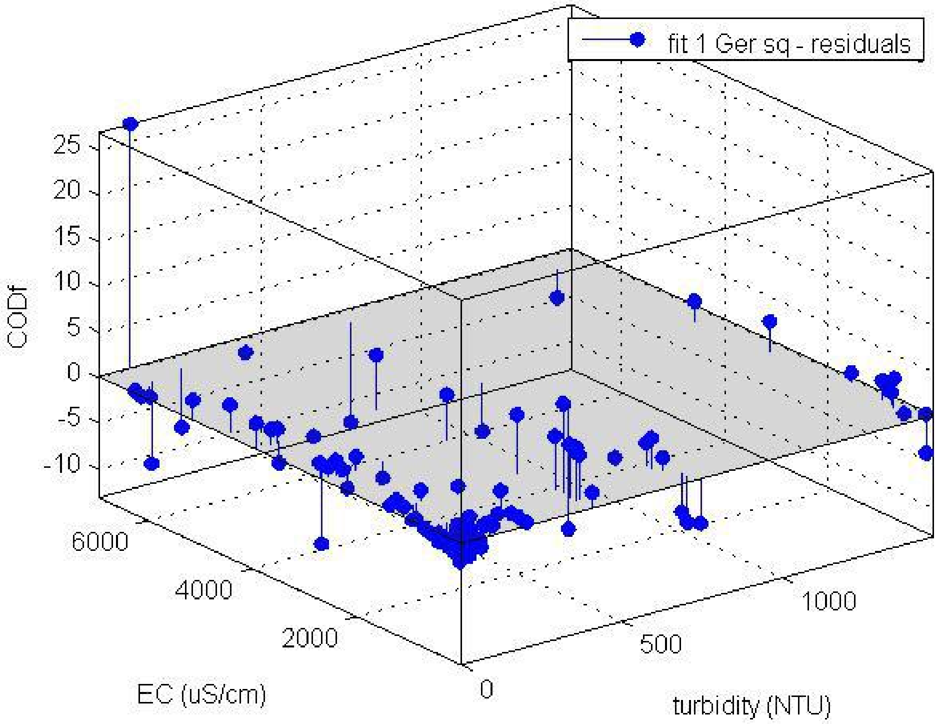


Figure V.7 - Residuals fit 1

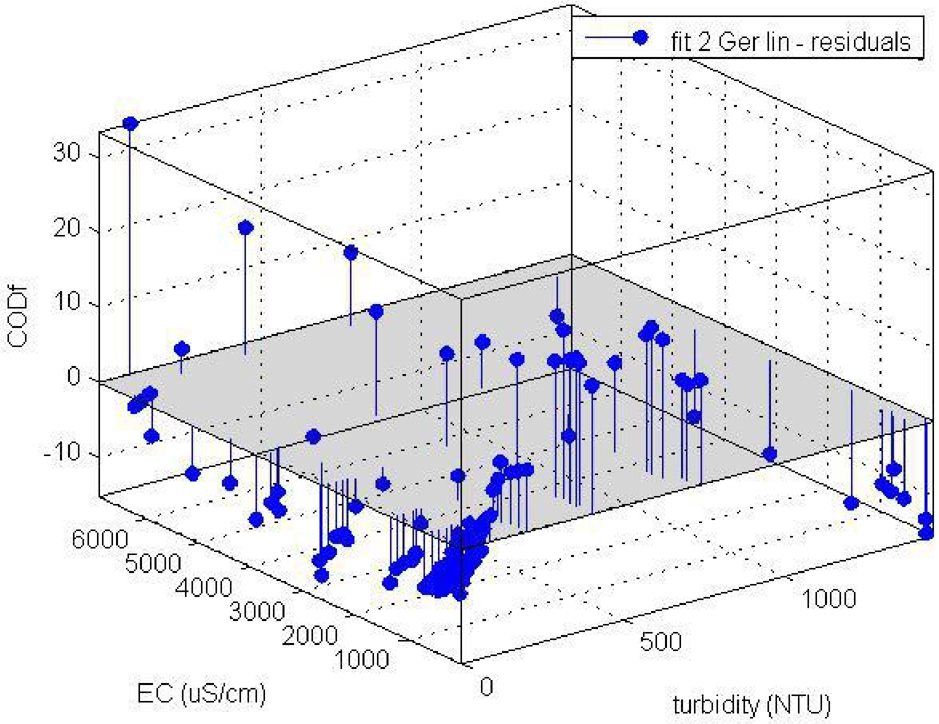


Figure V.8 - Residuals fit 2

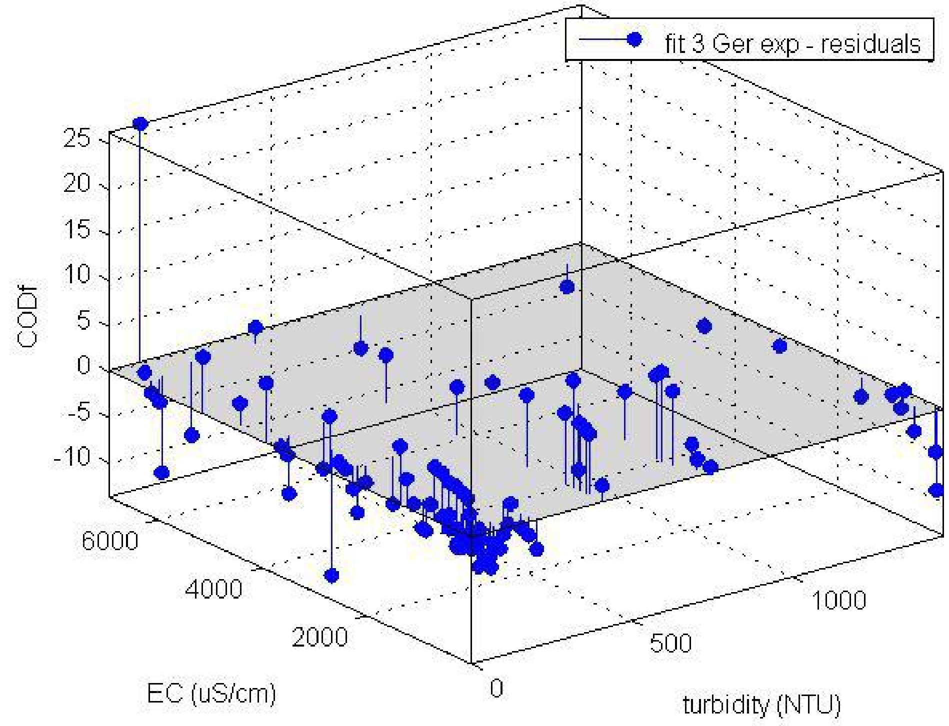


Figure V.9 - Residuals fit 3

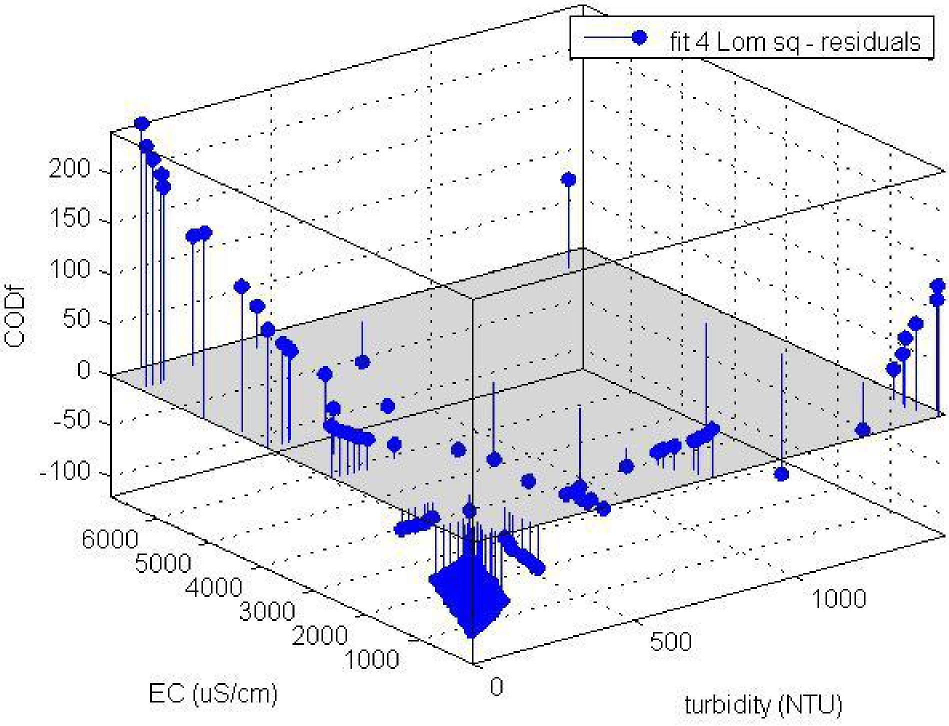


Figure V.10 - Residuals fit 4

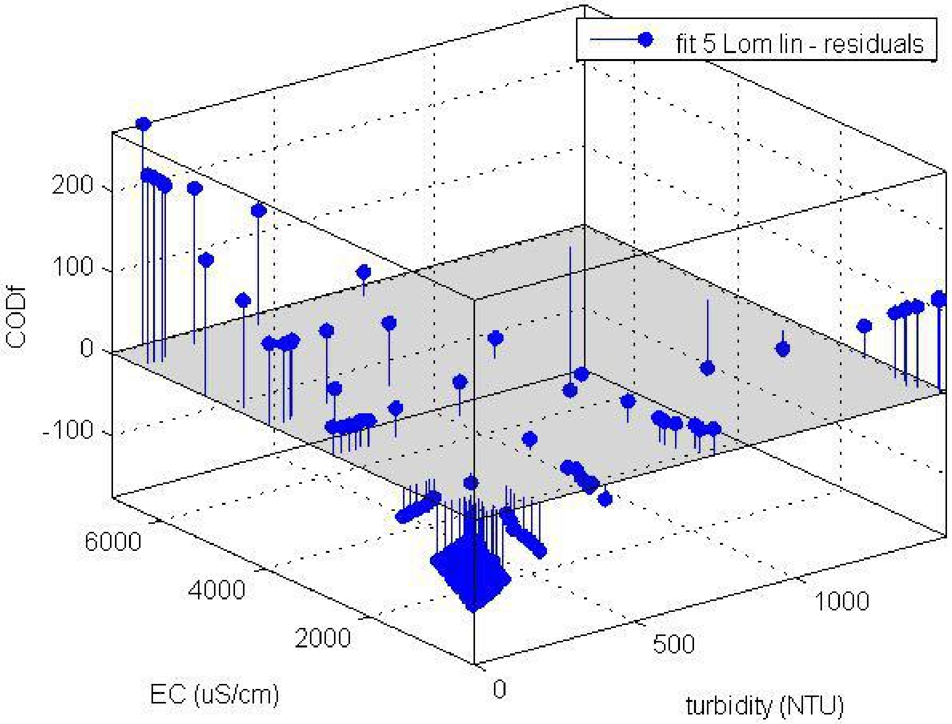


Figure V.11 - Residuals fit 5

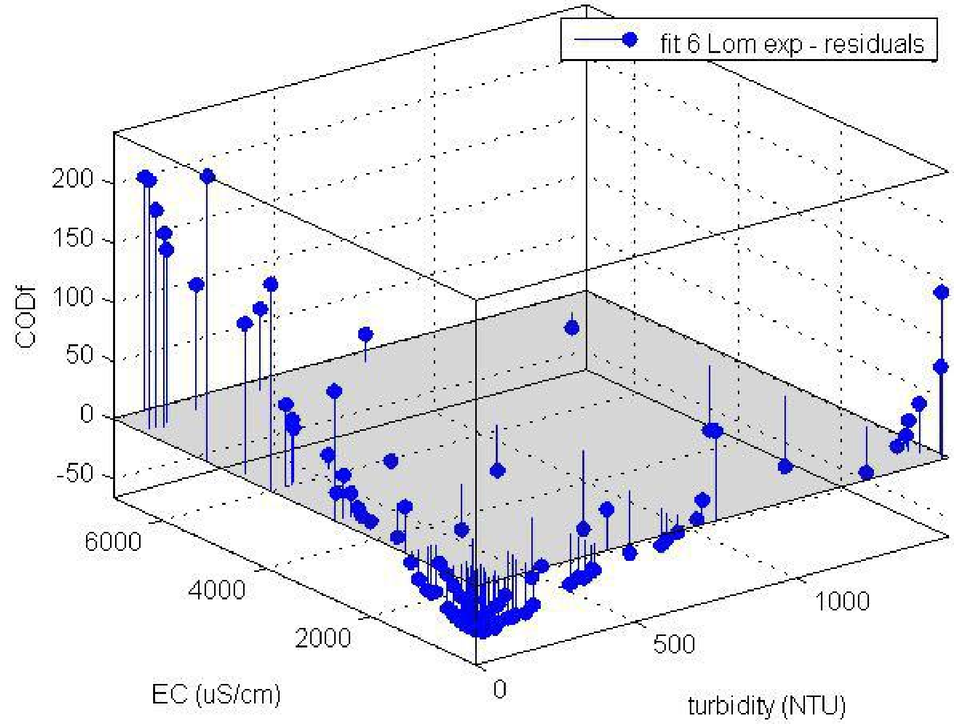


Figure V.12 - Residuals fit 6

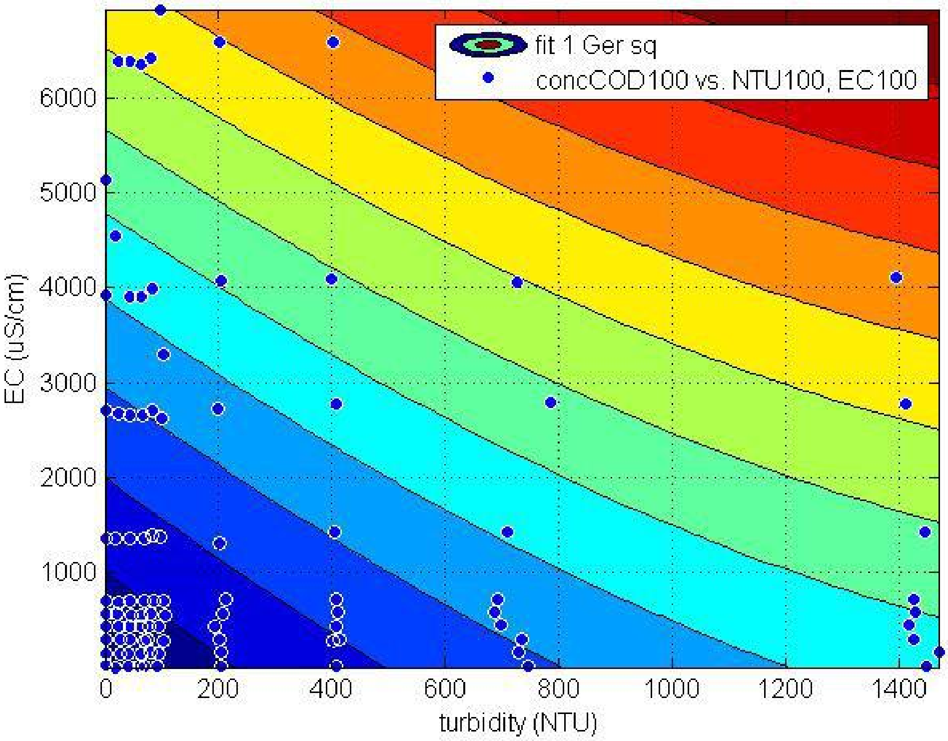


Figure V.13 - Contours fit 1

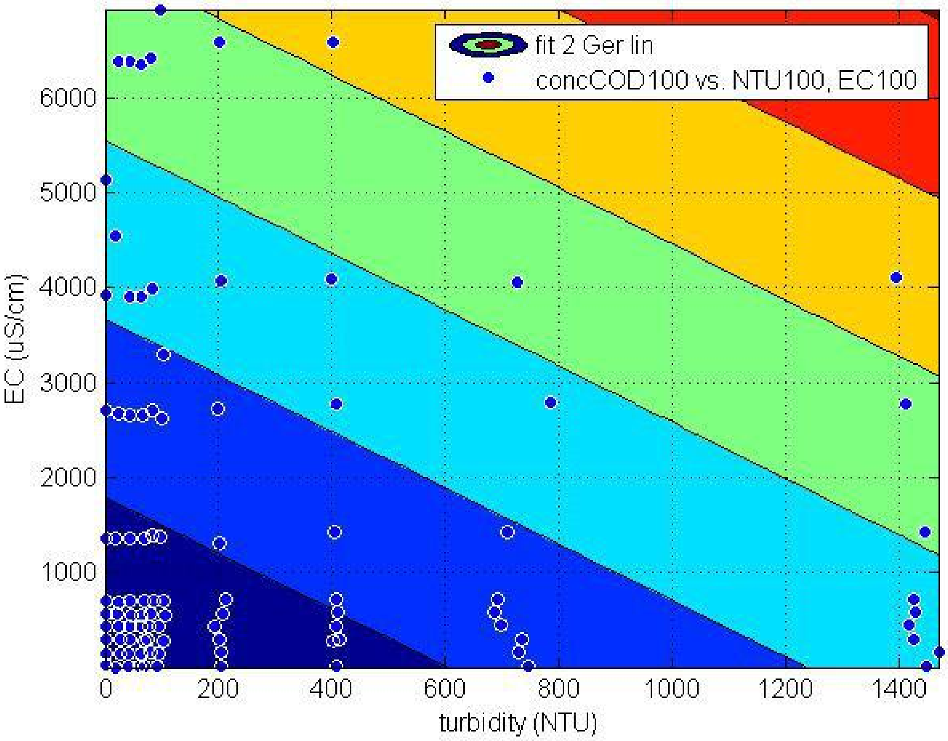


Figure V.14 - Contours fit 2

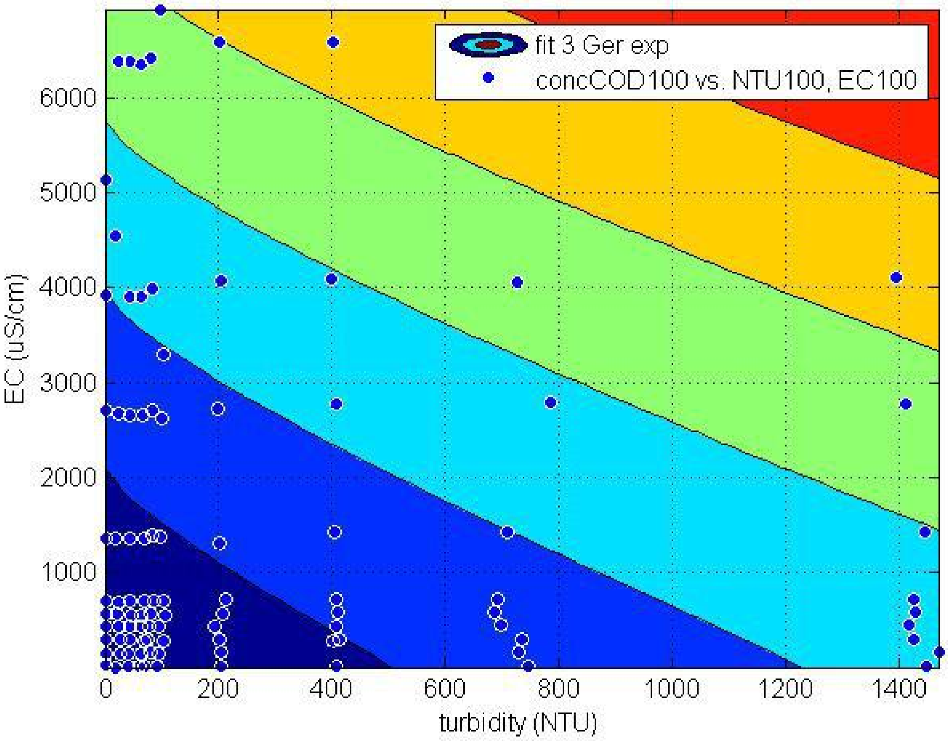


Figure V.15 - Contours fit 3

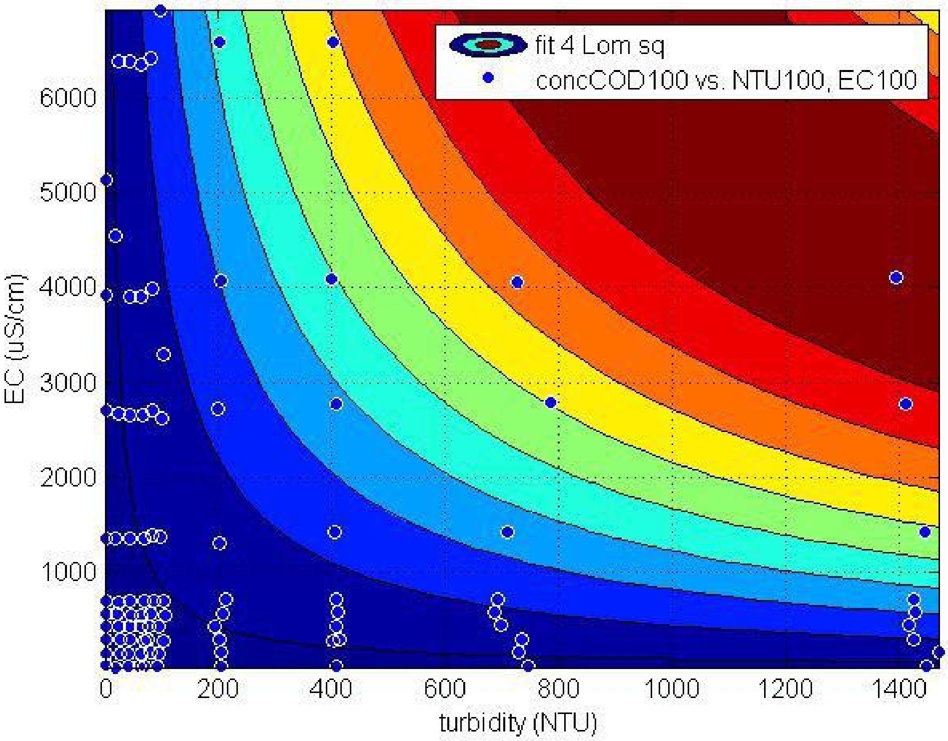


Figure V.16 - Contours fit 4

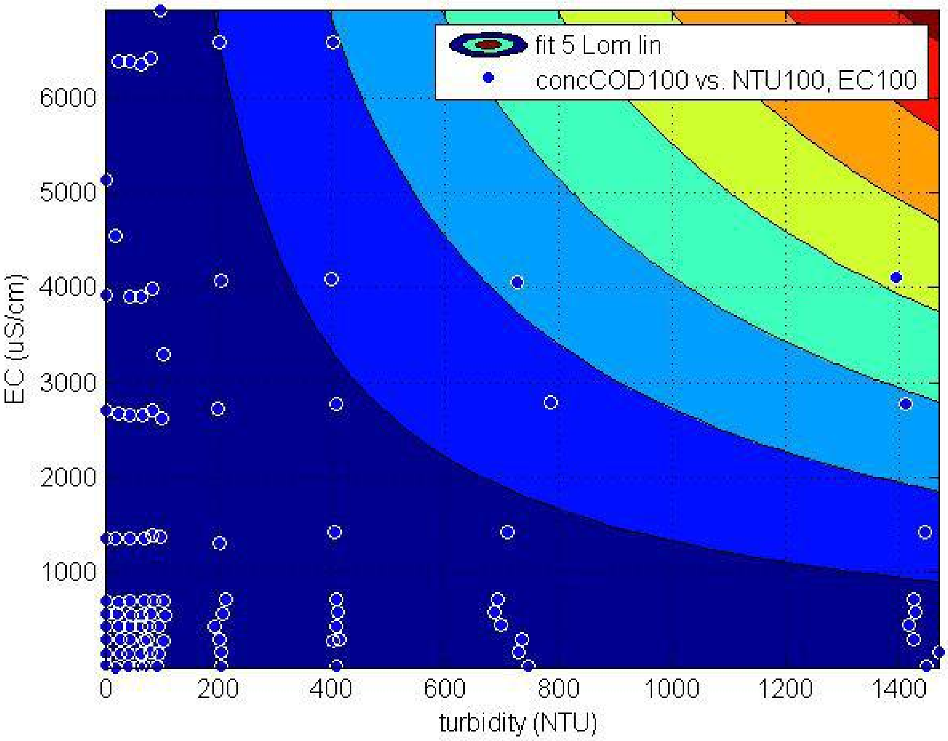


Figure V.17 - Contours fit 5

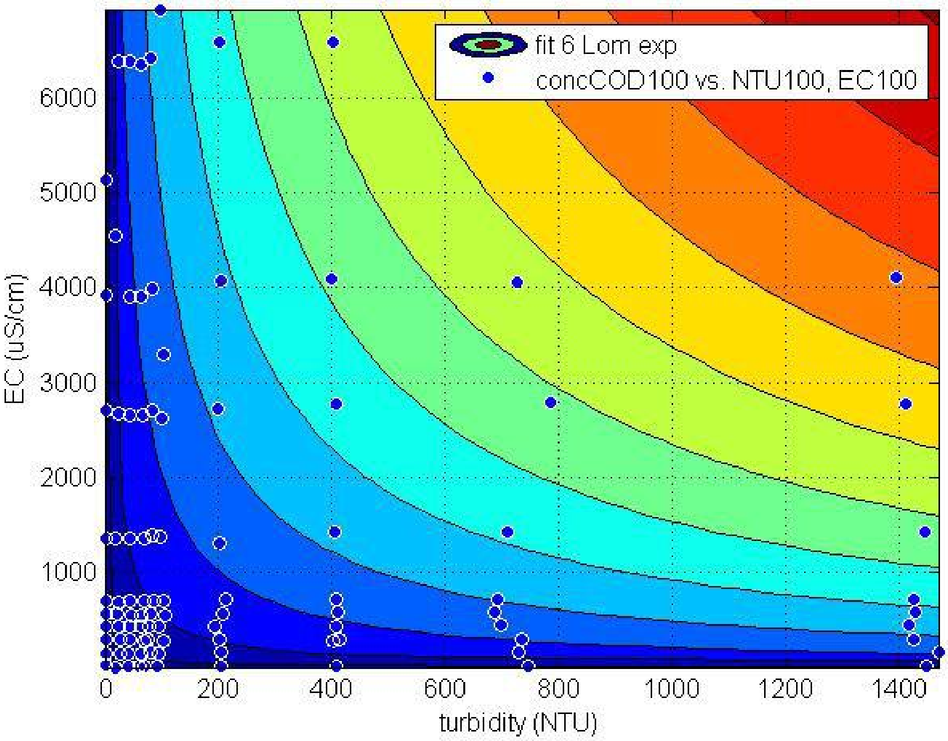


Figure V.18 - Contours fit 6

Appendix VI Results field test

Tables VI.1, VI.2 and VI.3 show the sample data from the field test for the storm events of 10 May 2012, 20 December 2012 and 24 December 2012.

Table VI.1 - Results field test 10 May 2012

Date and time	COD	EC	Turbidity
	mg/l	µS/cm	NTU
05/10/2012 19:00:00	11500	330	710
05/10/2012 19:05:00	400	330	220
05/10/2012 19:10:00	1760	330	450
05/10/2012 19:15:00	1450	320	560
05/10/2012 19:20:00	510	330	250
05/10/2012 19:25:00	510	330	260
05/10/2012 19:30:00	370	330	200
05/10/2012 19:35:00	320	340	200
05/10/2012 19:40:00	380	340	230
05/10/2012 19:45:00	400	340	240

Table VI.2 - Results field test 20 December 2012

Date and time	COD	EC	Turbidity
	mg/l	µS/cm	NTU
12/20/2012 17:30:00	2370	430	500
12/20/2012 17:35:00	3750	400	400
12/20/2012 17:40:00	3470	390	450
12/20/2012 17:45:00	4610	390	460
12/20/2012 17:50:00	1520	420	170
12/20/2012 17:55:00	2610	440	1580
12/20/2012 18:00:00	769	420	170
12/20/2012 18:05:00	1810	420	280
12/20/2012 18:10:00	226	430	130
12/20/2012 18:15:00	408	430	210
12/20/2012 18:20:00	287	420	130
12/20/2012 18:25:00	362	420	420
12/20/2012 18:30:00	371	400	130
12/20/2012 18:35:00	250	390	140
12/20/2012 18:40:00	235	390	120
12/20/2012 18:45:00	338	380	130
12/20/2012 18:50:00	302	360	110
12/20/2012 18:55:00	286	360	160
12/20/2012 19:00:00	263	360	120
12/20/2012 19:05:00	679	360	94
12/20/2012 19:10:00	482	360	280
12/20/2012 19:15:00	375	360	120
12/20/2012 19:20:00	567	360	110
12/20/2012 19:25:00	208	360	97
12/20/2012 19:30:00	193	360	94
12/20/2012 19:35:00	166	360	83
12/20/2012 19:40:00	166	350	80
12/20/2012 19:45:00	215	350	83
12/20/2012 19:50:00	348	350	160
12/20/2012 19:55:00	282	350	110
12/20/2012 20:00:00	223	350	89
12/20/2012 20:05:00	232	330	110

Table VI.3 - Results field test 24 December 2012

Date and time	COD	EC	Turbidity
	mg/l	µS/cm	NTU
12/24/3912 01:00:00	636	340	360
12/24/3912 01:05:00	429	360	130
12/24/3912 01:10:00	284	370	84
12/24/3912 01:15:00	244	370	52
12/24/3912 01:20:00	186	360	53
12/24/3912 01:25:00	189	360	52
12/24/3912 01:30:00	251	360	110
12/24/3912 01:35:00	221	350	54

Appendix VII Results fits field test

Table VII.1 shows the goodness of fit statistics of the field test.

Table VII.1 - Goodness of fit statistics of field test. SSE is an abbreviation for sum squared error and RMSE is an abbreviation for root mean squared error

Fit	SSE	r^2	Adjusted r^2	RMSE
1	14.73	0.69	0.67	0.57
2	16.44	0.66	00.64	0.59
3	16.94	0.65	0.62	0.61
4	21.80	0.55	0.53	0.68
5	22.90	0.52	0.51	0.69
6	22.26	0.54	0.52	0.69
7	16.26	0.66	0.65	0.59
8	16.67	0.65	0.65	0.59
9	16.27	0.66	0.65	0.59
10	44.22	0.078	0.040	0.97
11	45.65	0.049	0.029	0.98
12	45.99	0.042	0.001	0.99

Figures VII.1 to VII.18 show the fits and fits with confidence intervals of the field test, Figures VII.19 to VII.30 show the residuals of the fits of the field test and Figures VII.31 to VII.36 show the contours of the fits of the field test.

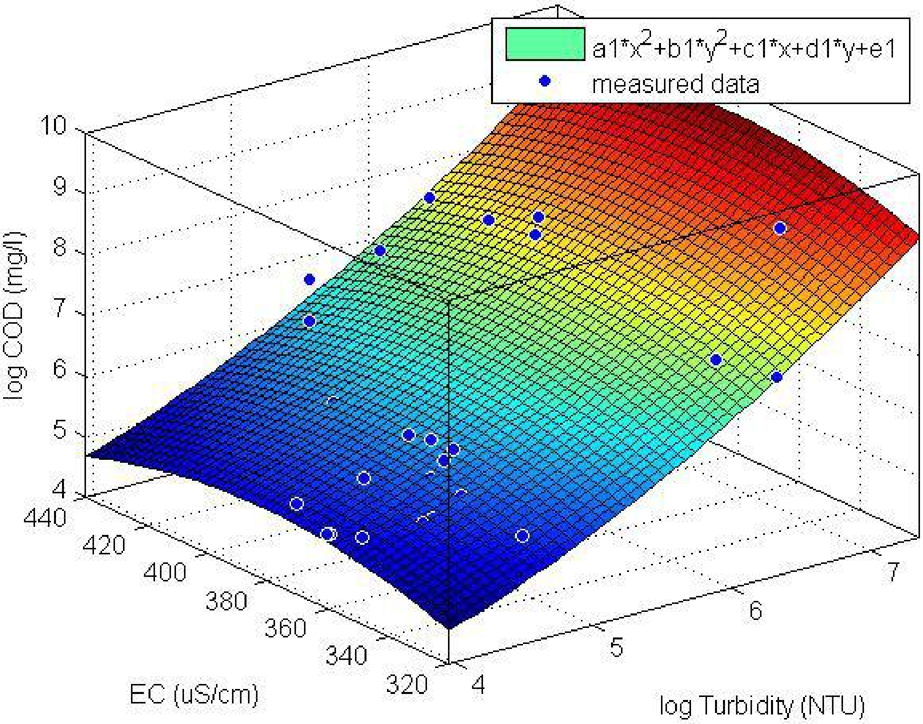


Figure VII.1 - Fit 1

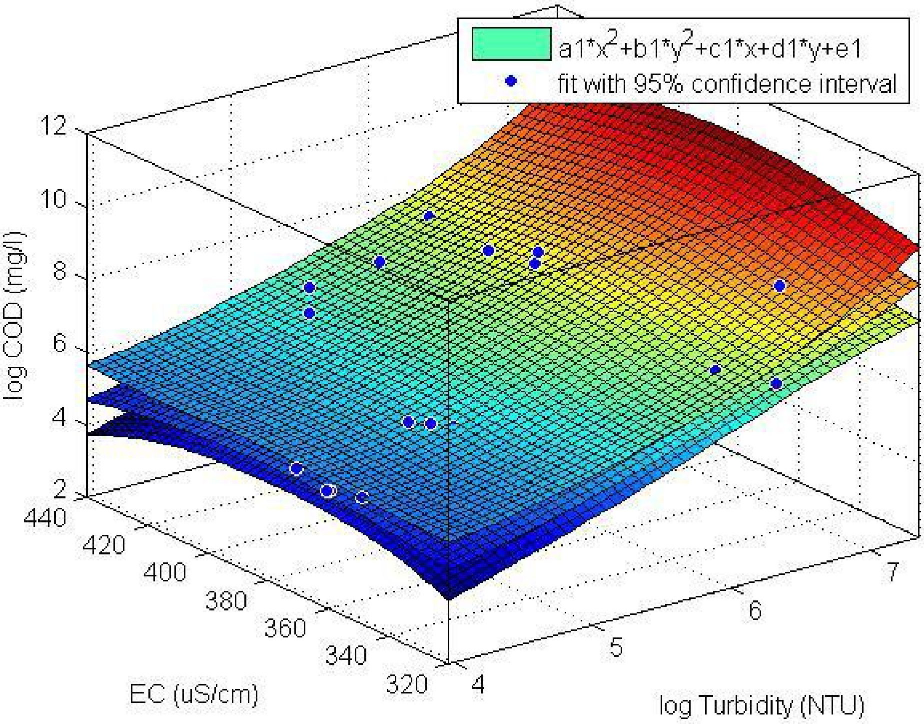


Figure VII.2 - Fit 1 with confidence intervals

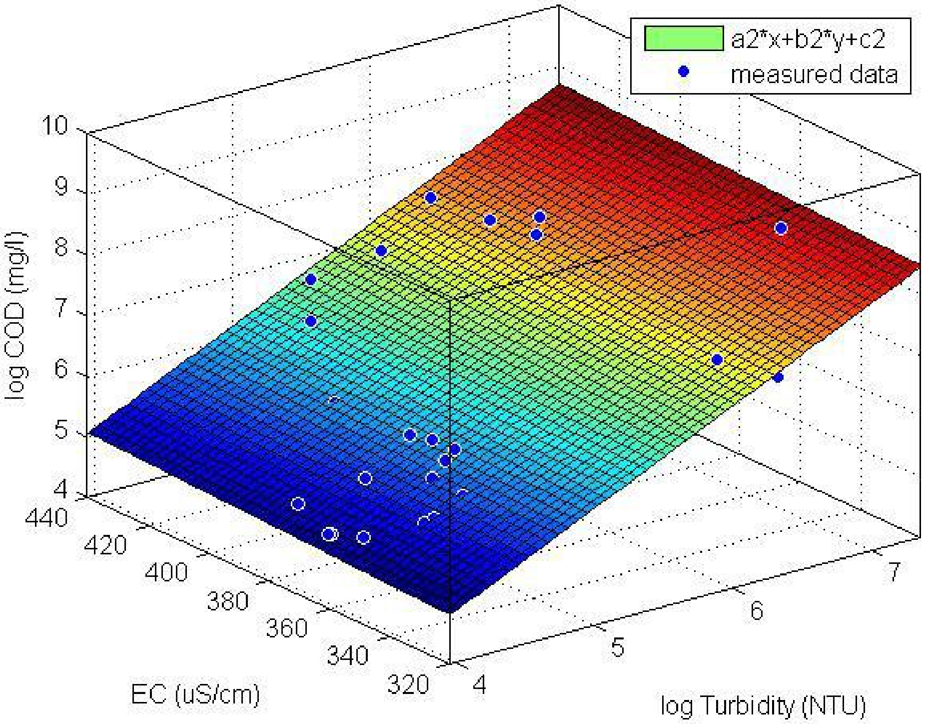


Figure VII.3 - Fit 2

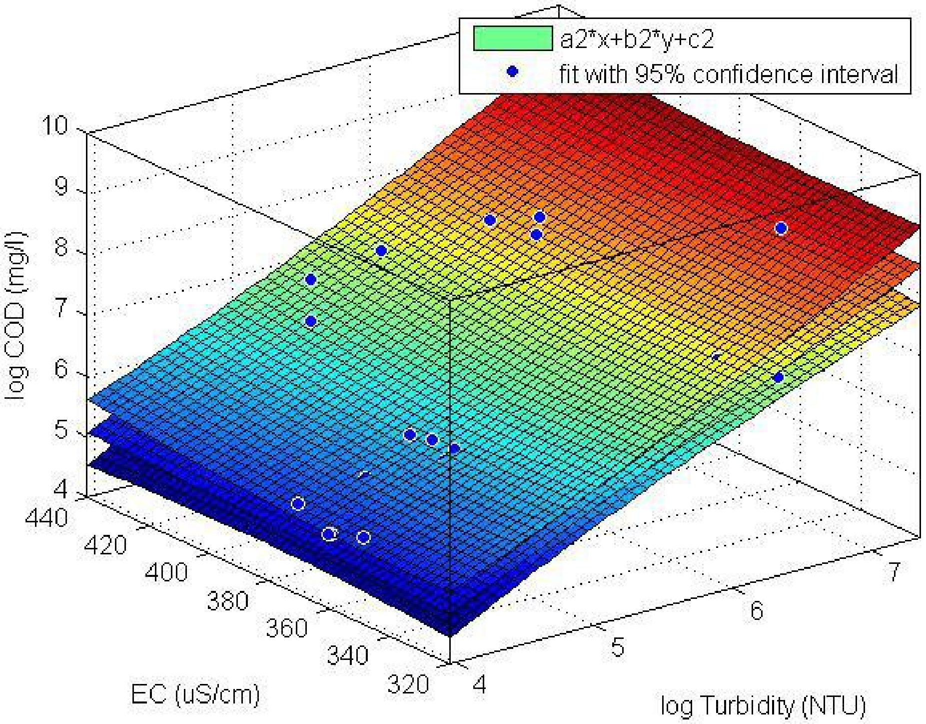


Figure VII.4 - Fit 2 with confidence intervals

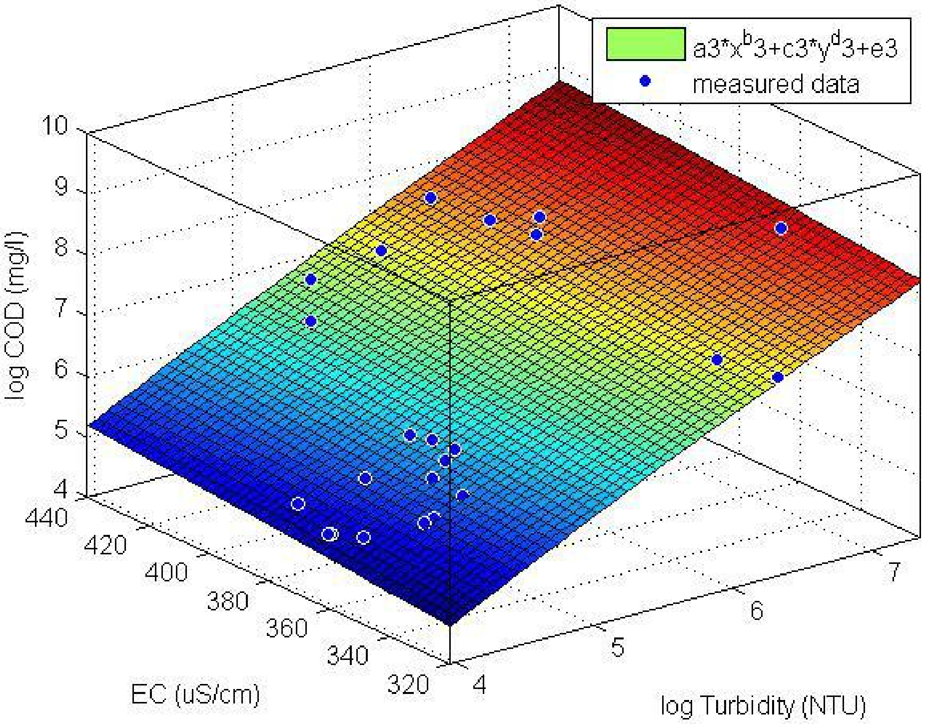


Figure VII.5 - Fit 3

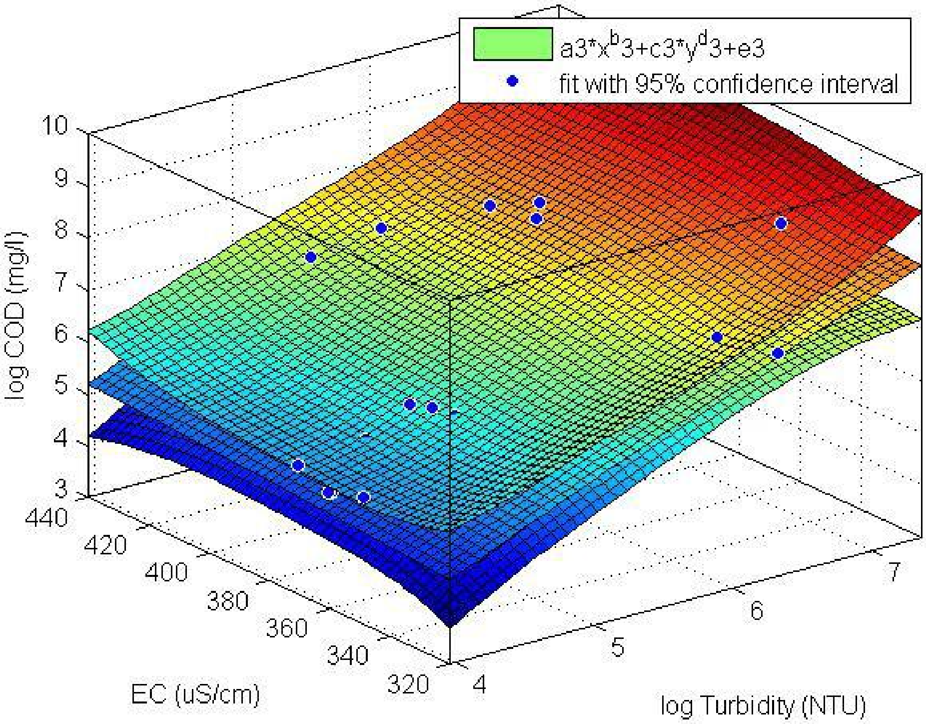


Figure VII.6 - Fit 3 with confidence intervals

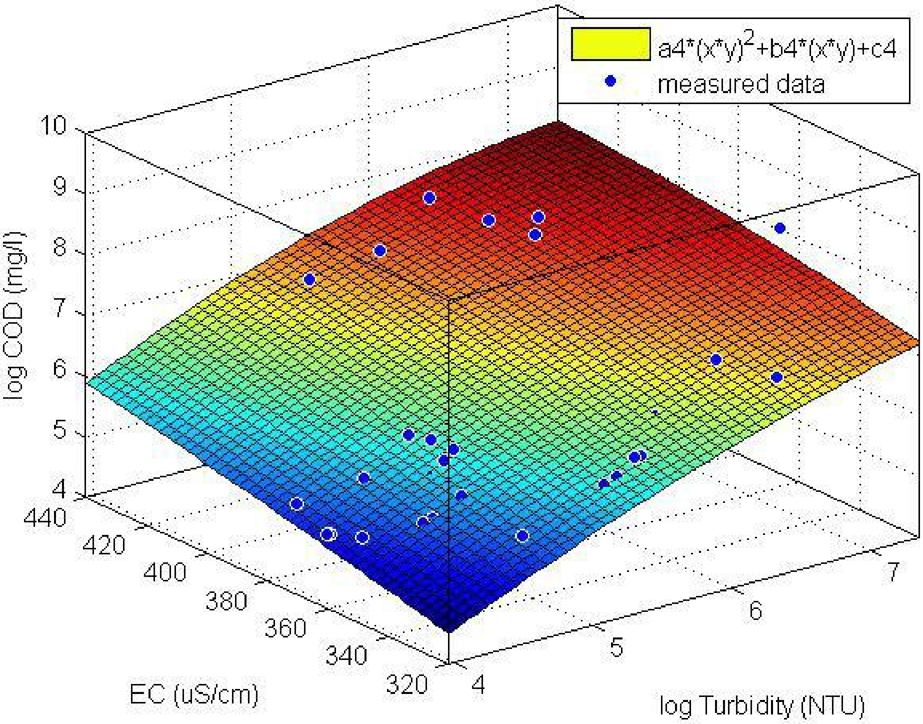


Figure VII.7 - Fit 4

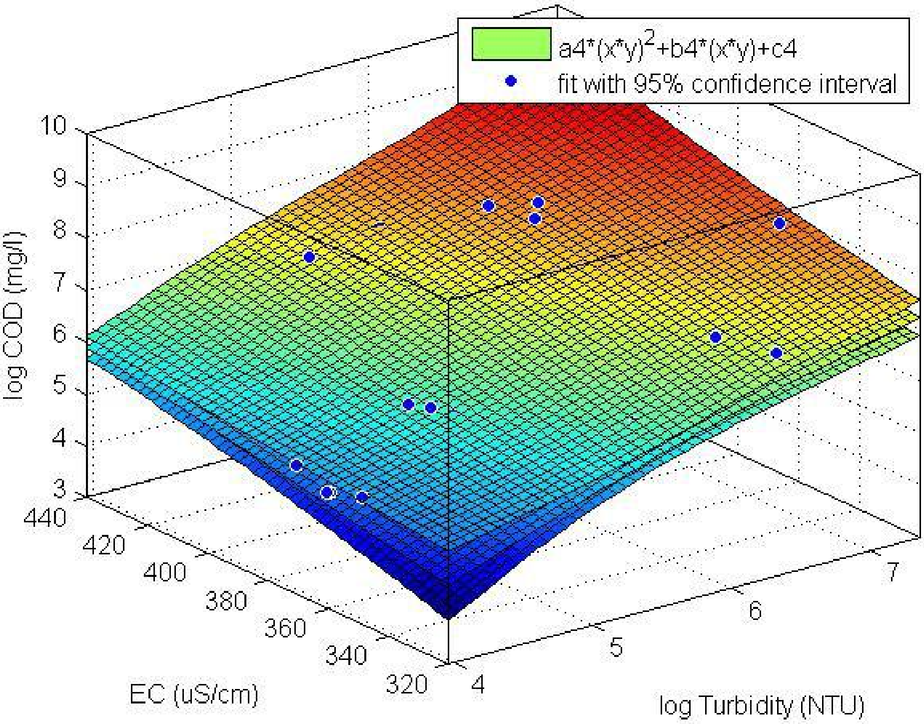


Figure VII.8 - Fit 4 with confidence intervals

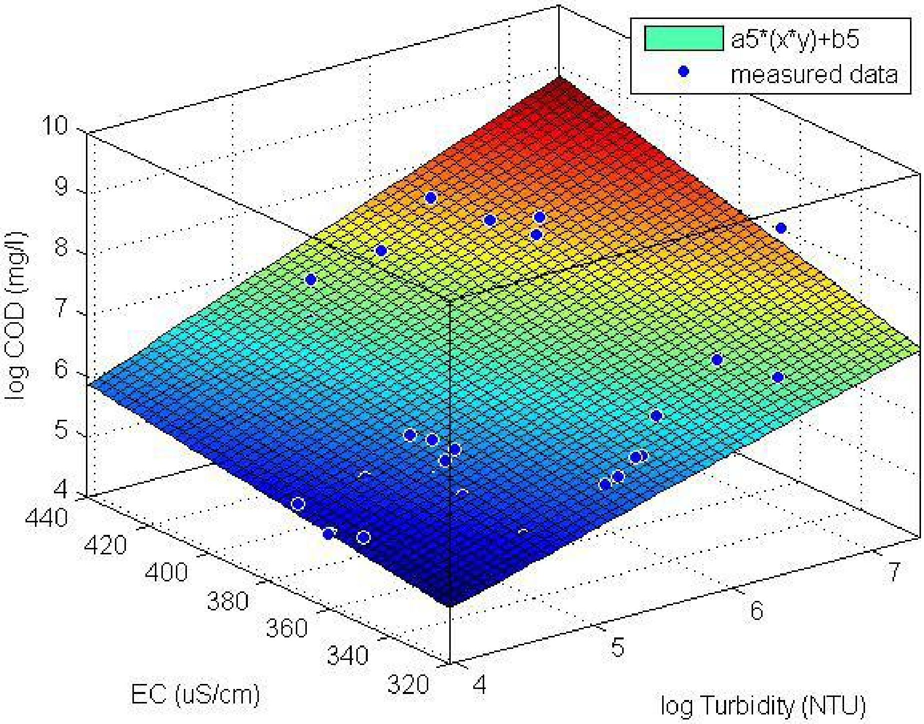


Figure VII.9 - Fit 5

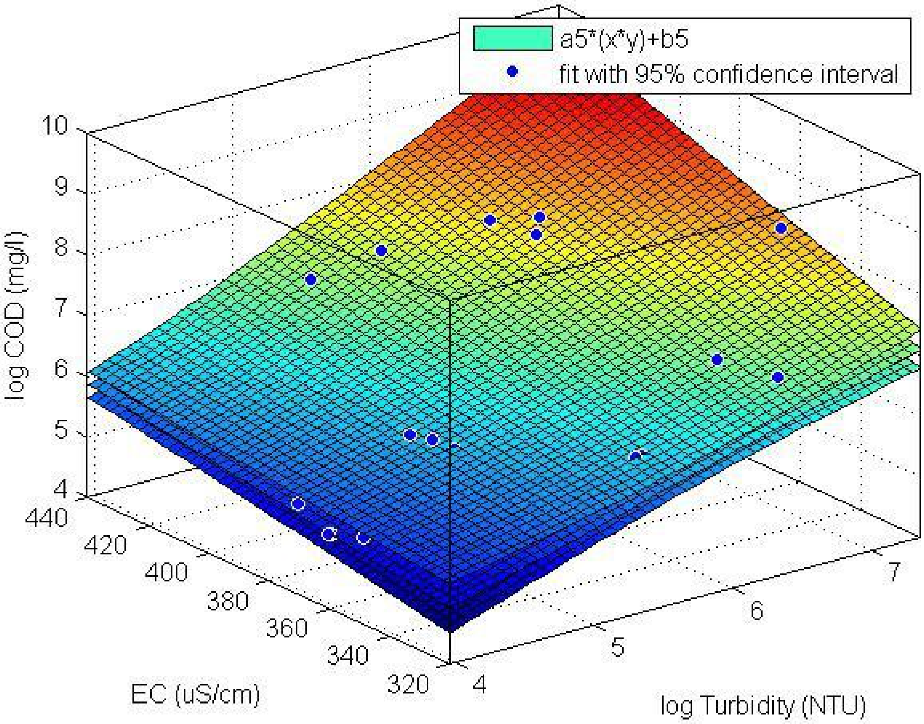


Figure VII.10 - Fit 5 with confidence intervals

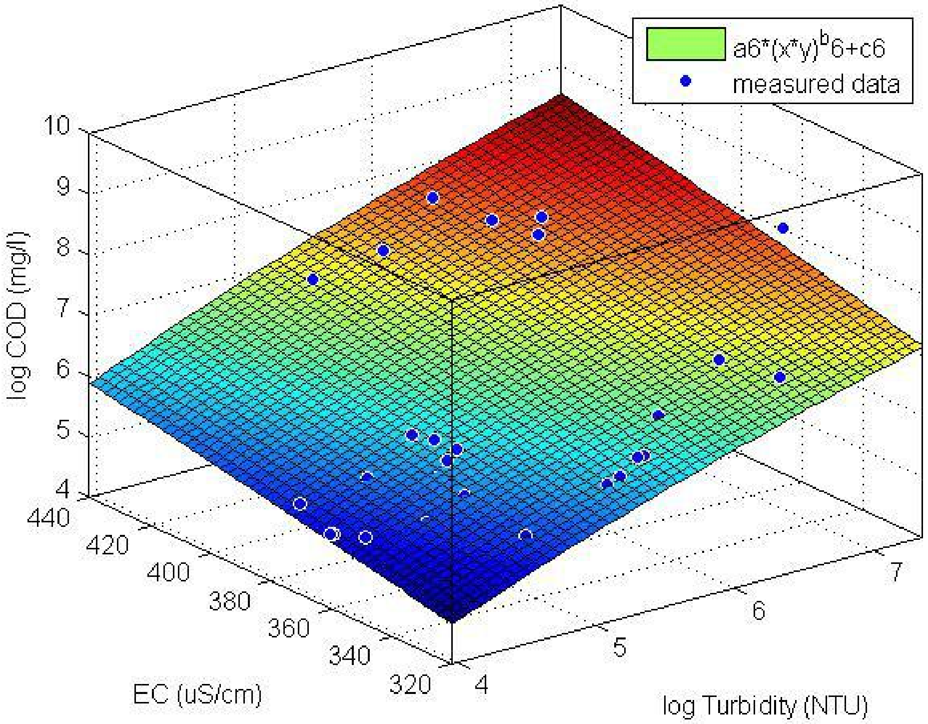


Figure VII.11 - Fit 6

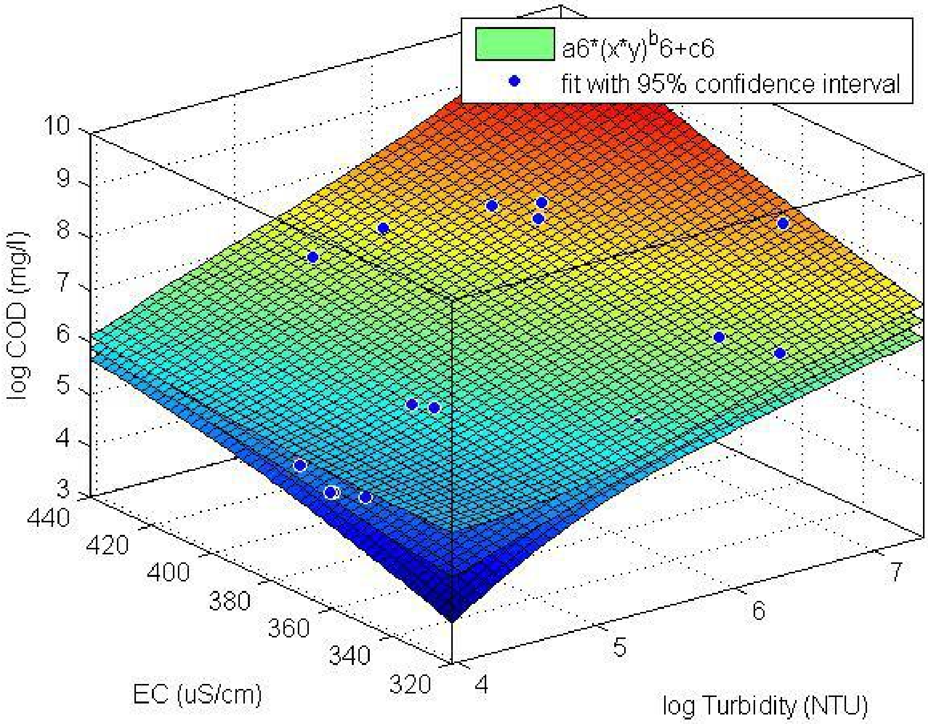


Figure VII.12 - Fit 6 with confidence intervals

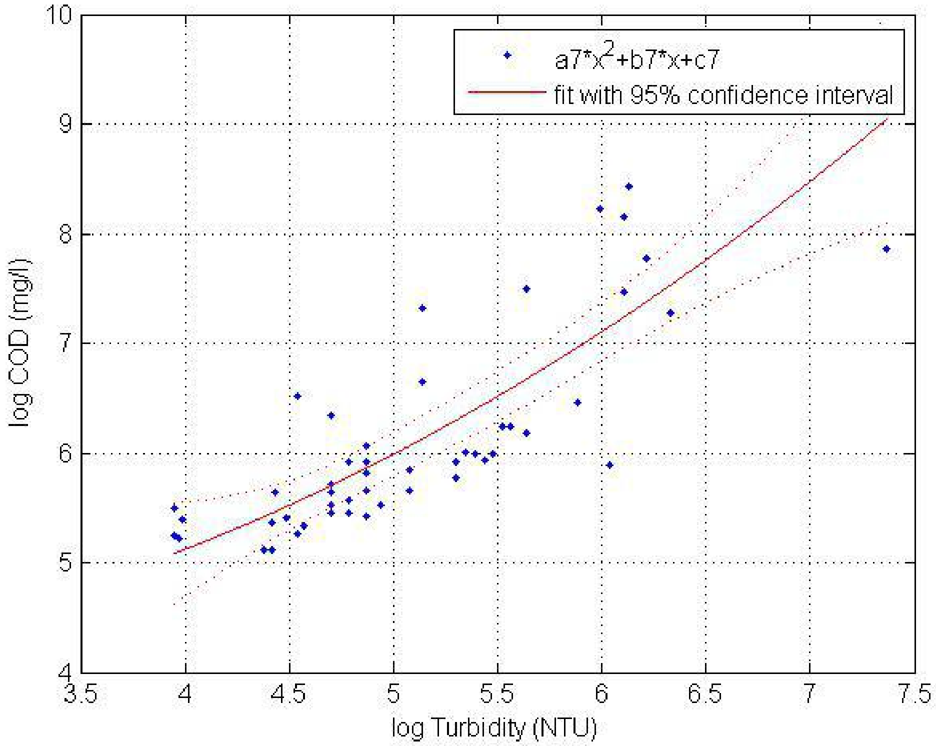


Figure VII.13 - Fit 7 with confidence intervals

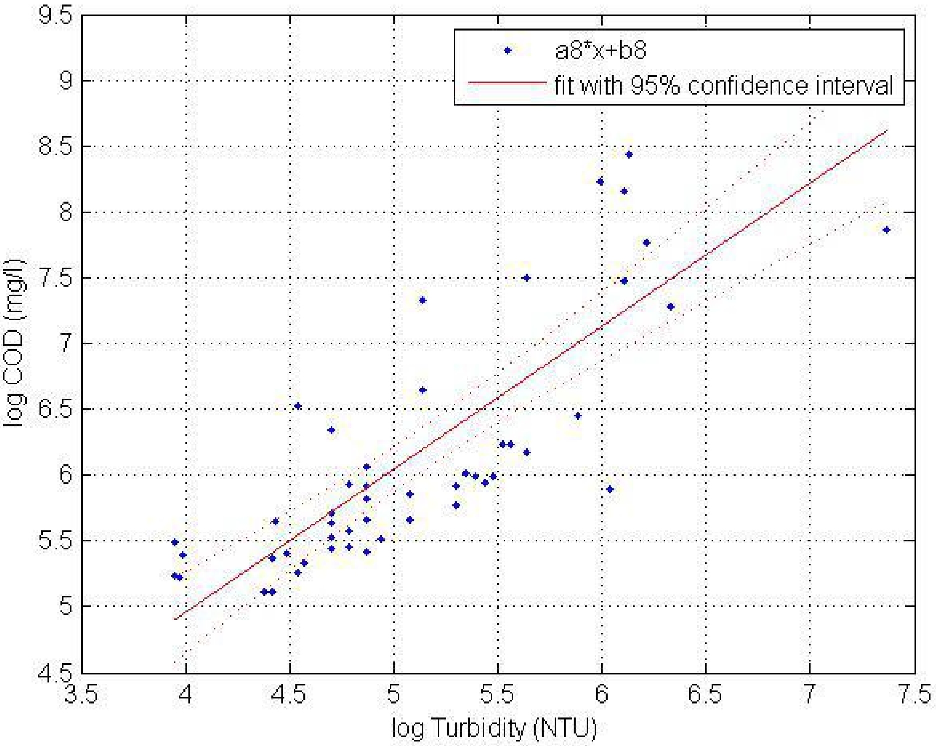


Figure VII.14 - Fit 8 with confidence intervals

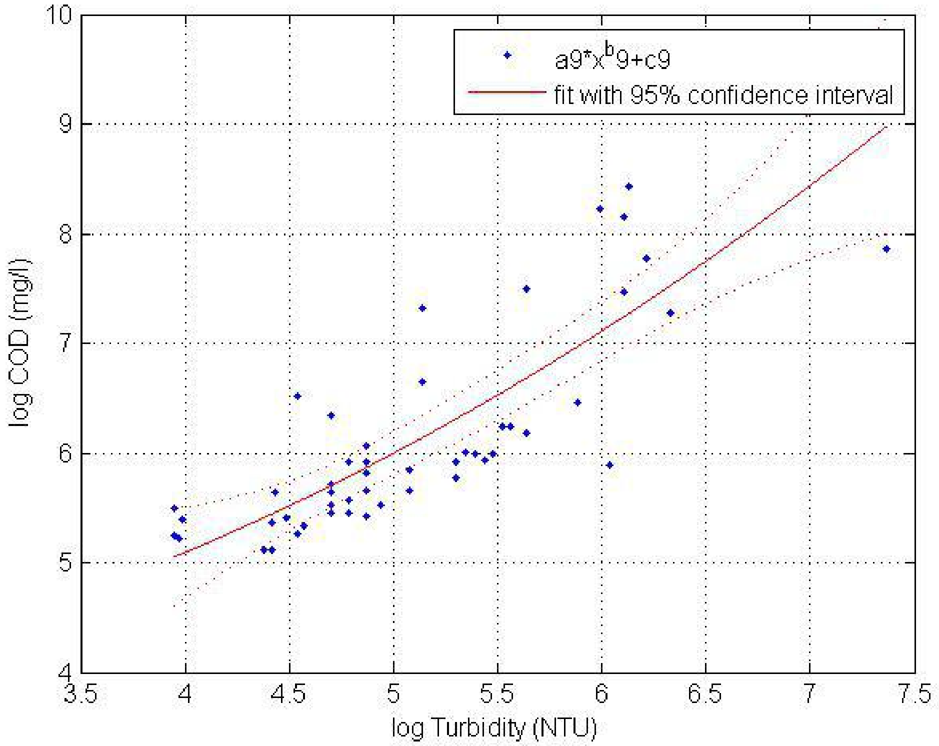


Figure VII.15 - Fit 9 with confidence intervals

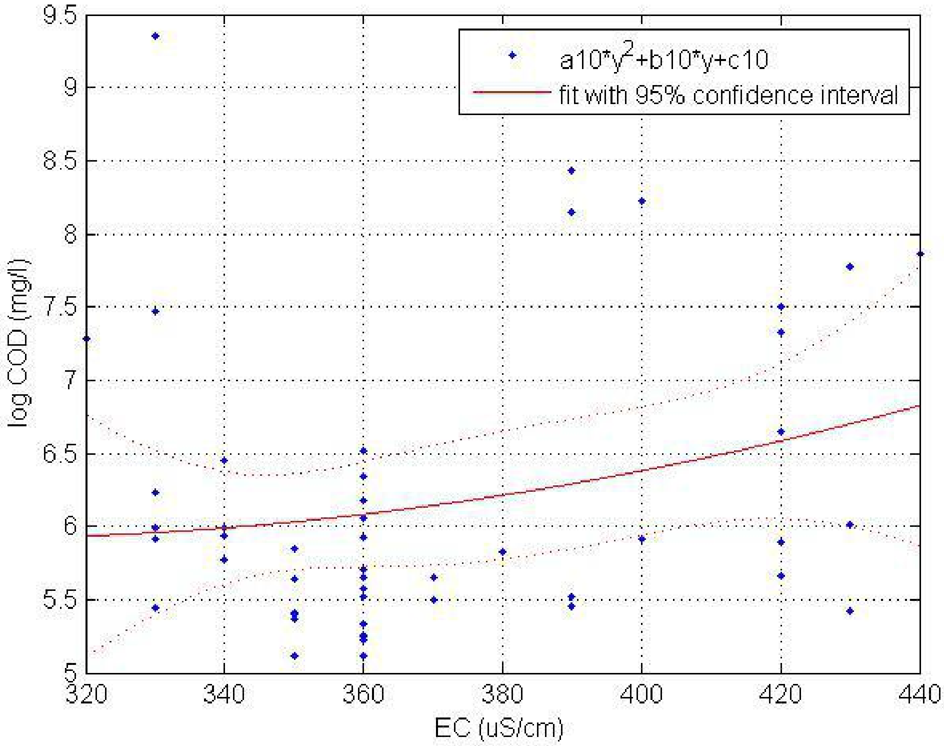


Figure VII.16 - Fit 10 with confidence intervals

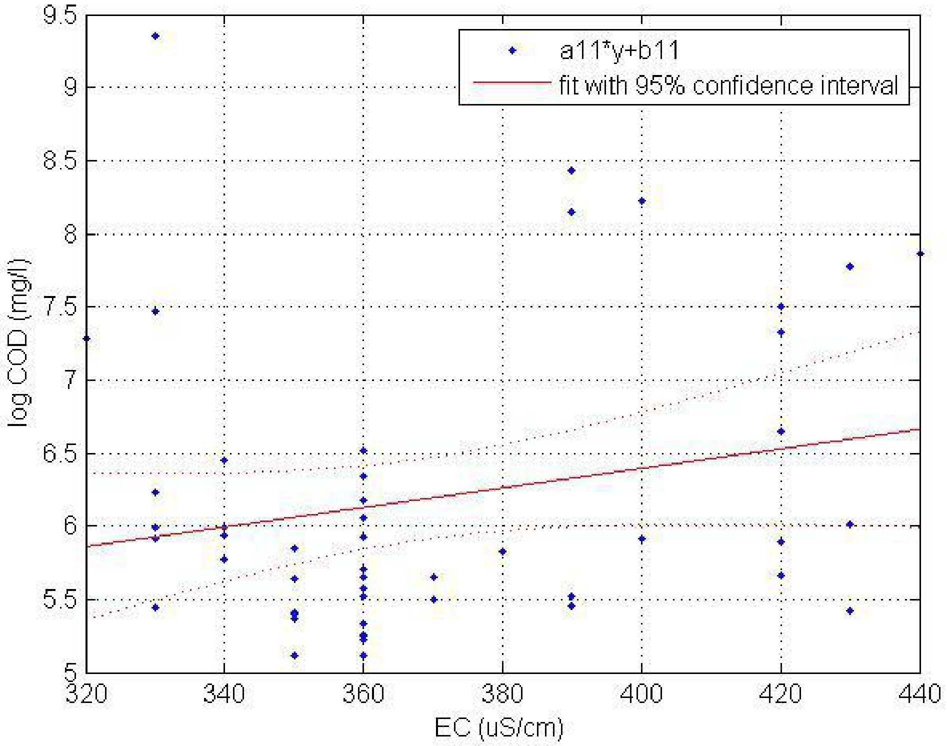


Figure VII.17 - Fit 11 with confidence intervals

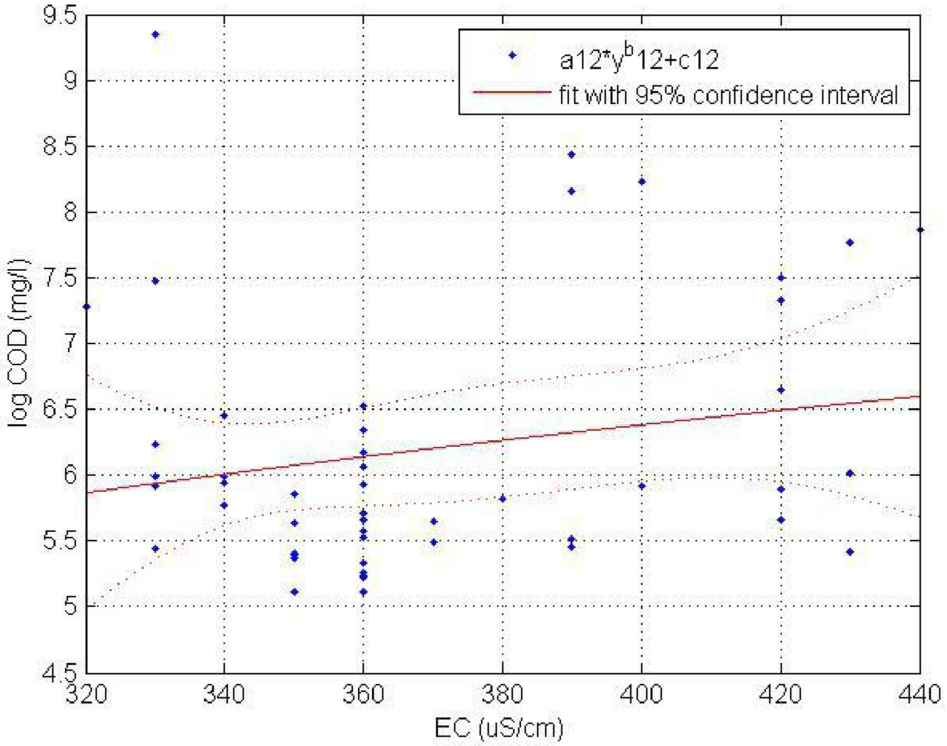


Figure VII.18 - Fit 12 with confidence intervals

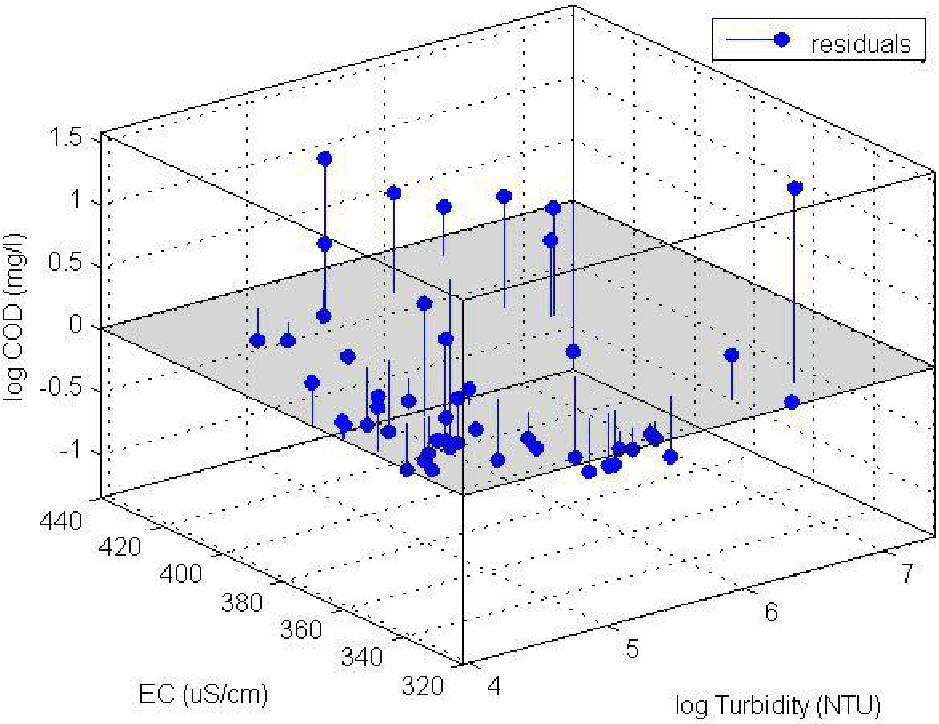


Figure VII.19 - Residuals fit 1

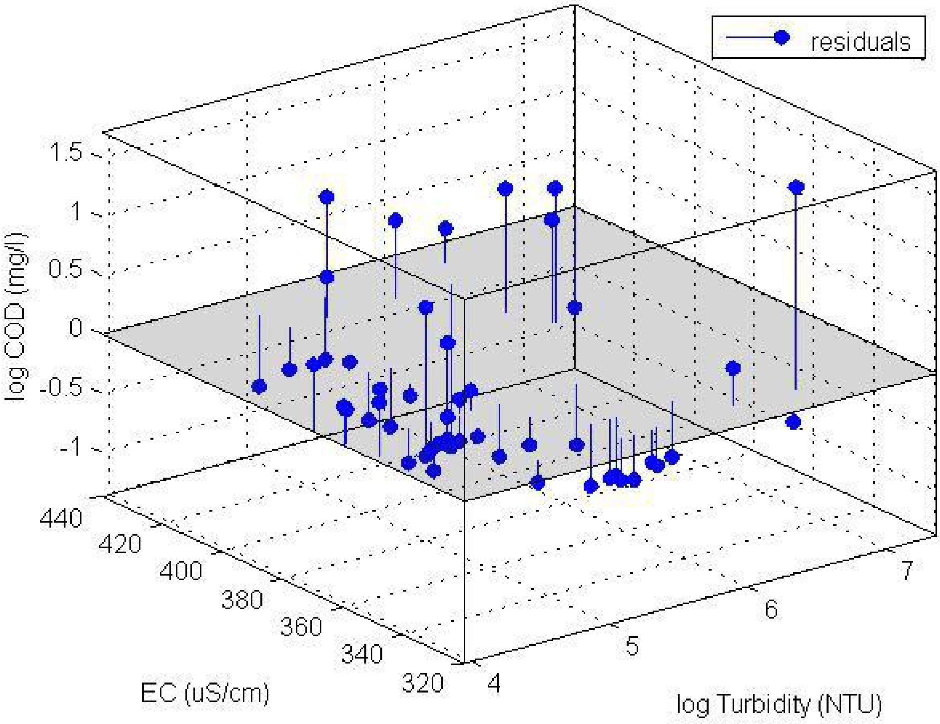


Figure VII.20 - Residuals fit 2

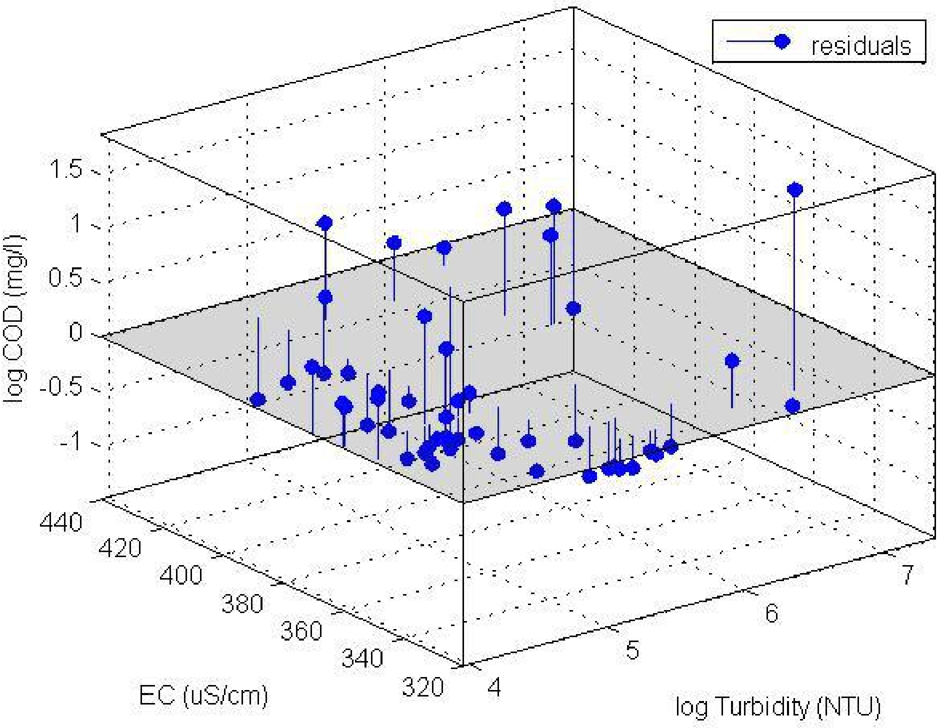


Figure VII.21 - Residuals fit 3

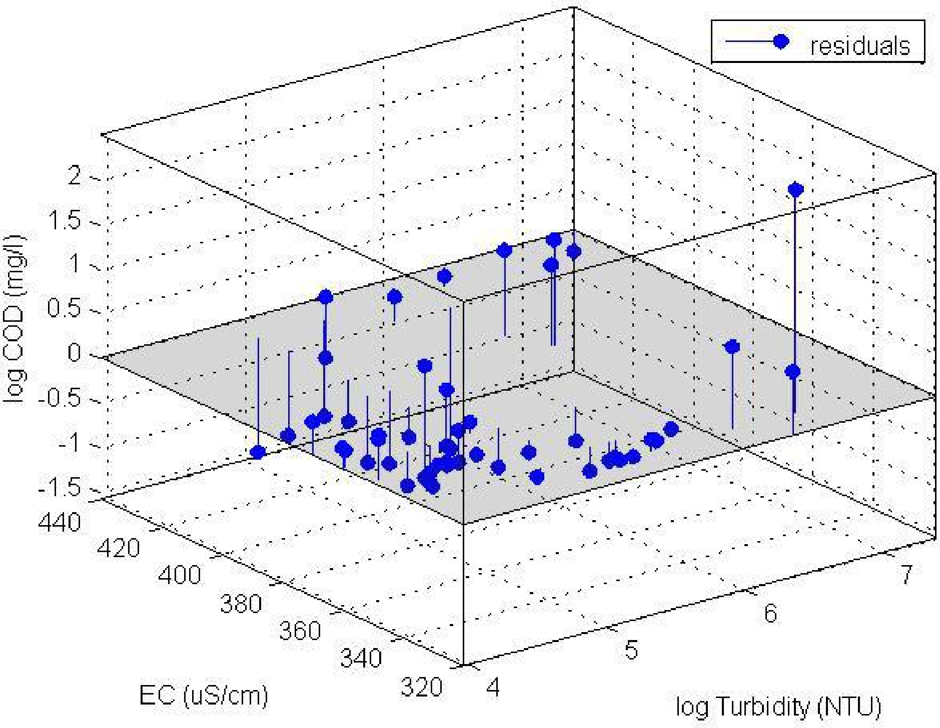


Figure VII.22 - Residuals fit 4

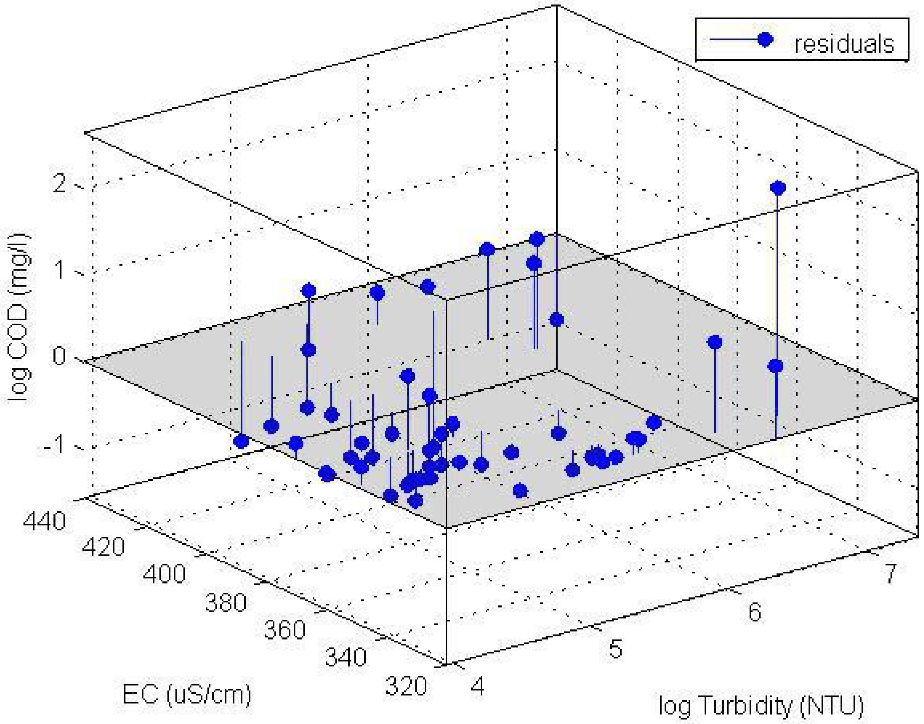


Figure VII.23 - Residuals fit 5

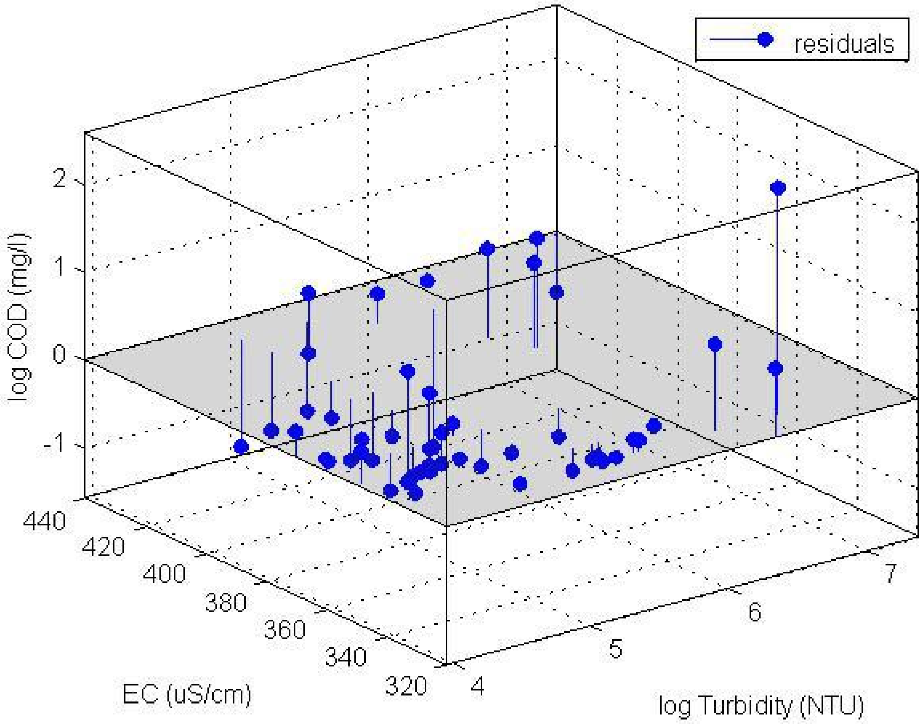


Figure VII.24 - Residuals fit 6

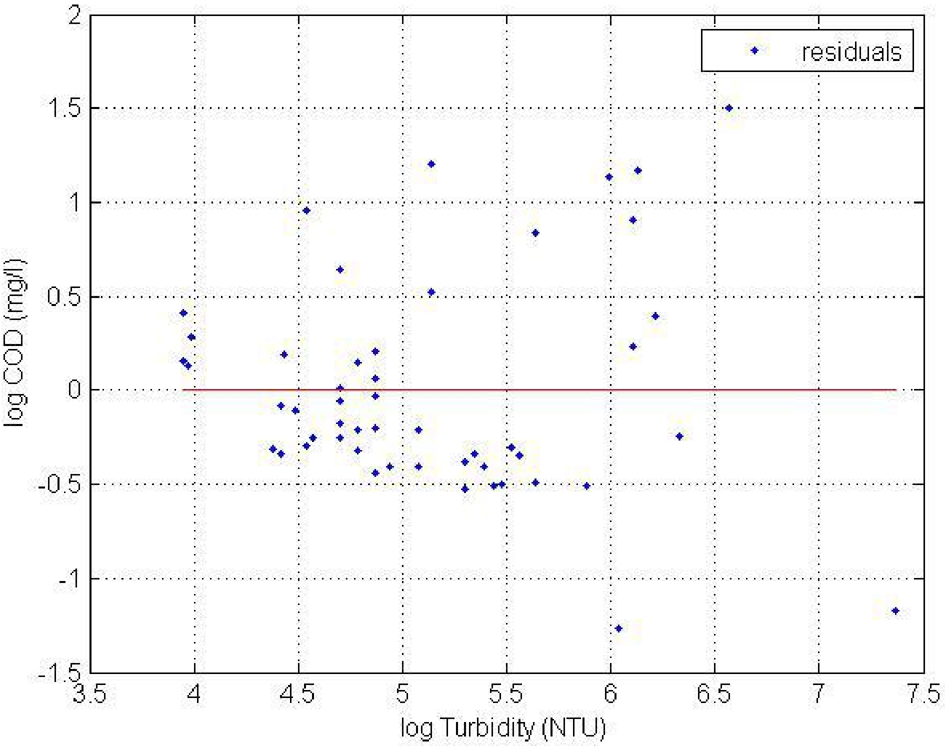


Figure VII.25 - Residuals fit 7

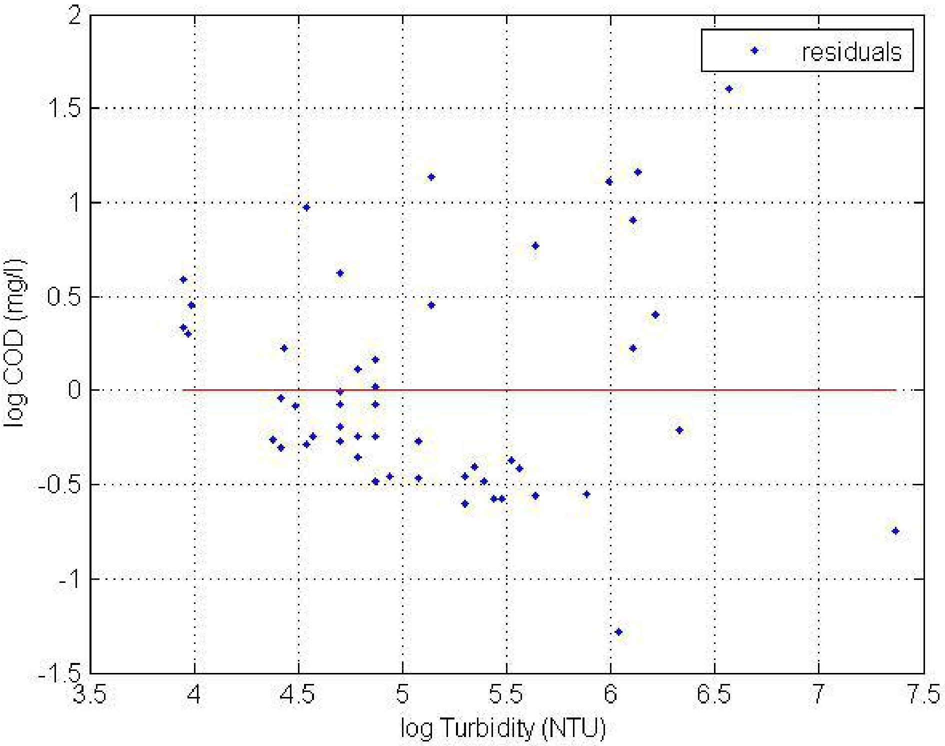


Figure VII.26 - Residuals fit 8

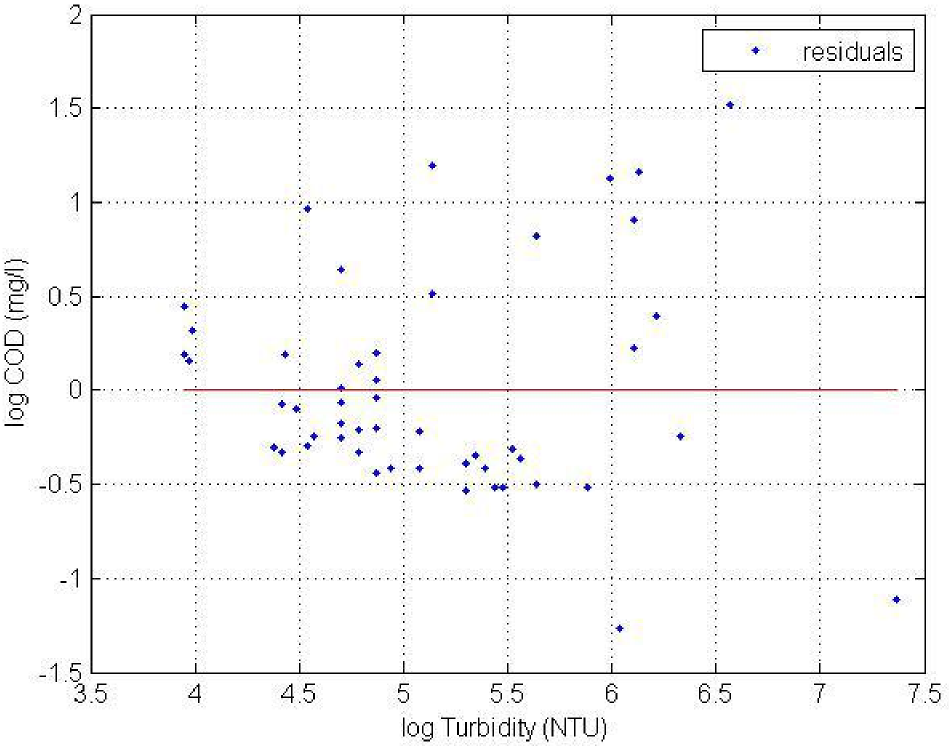


Figure VII.27 - Residuals fit 9

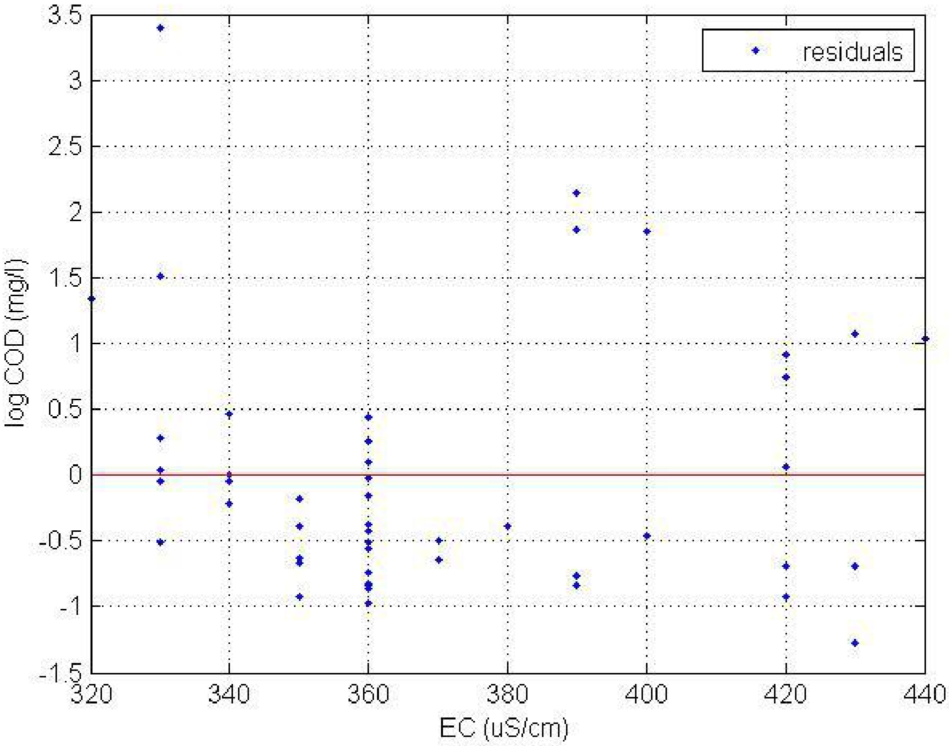


Figure VII.28 - Residuals fit 10

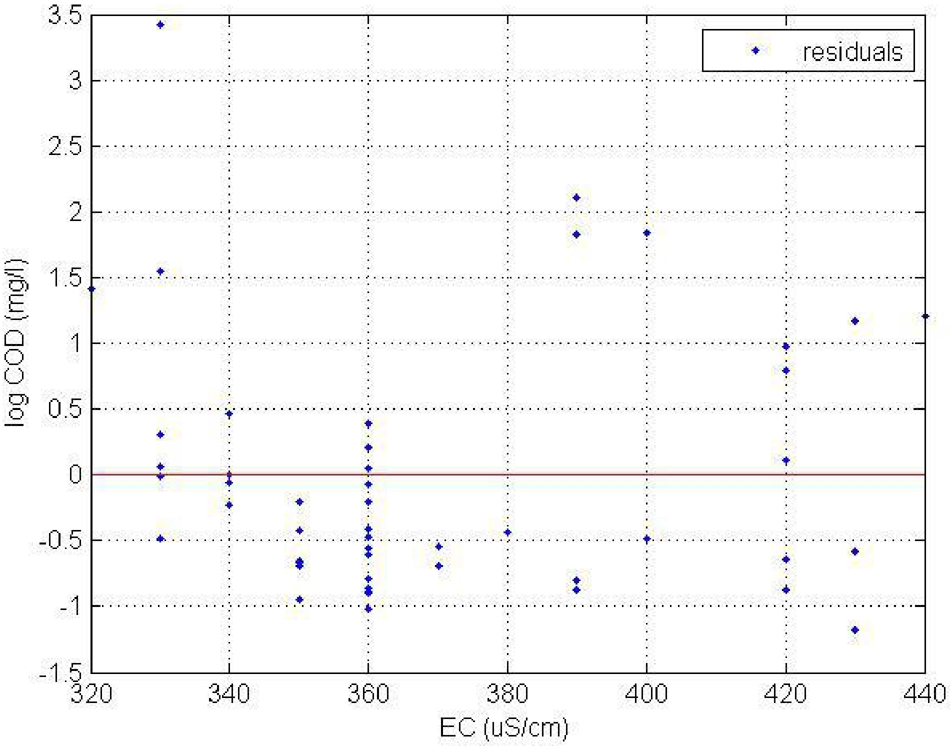


Figure VII.29 - Residuals fit 11

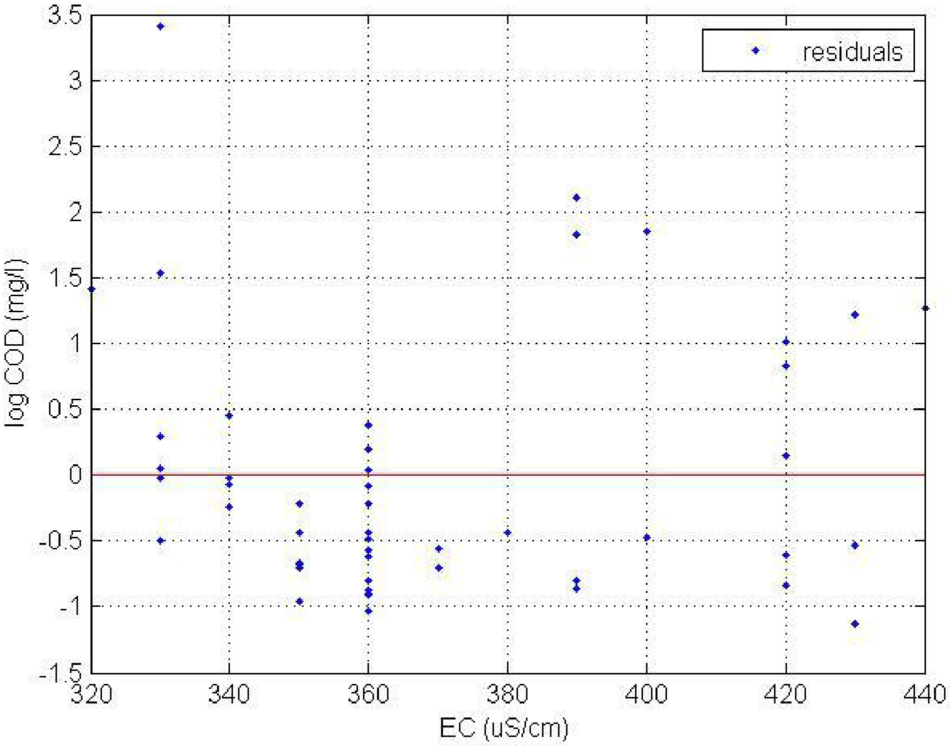


Figure VII.30 - Residuals fit 12

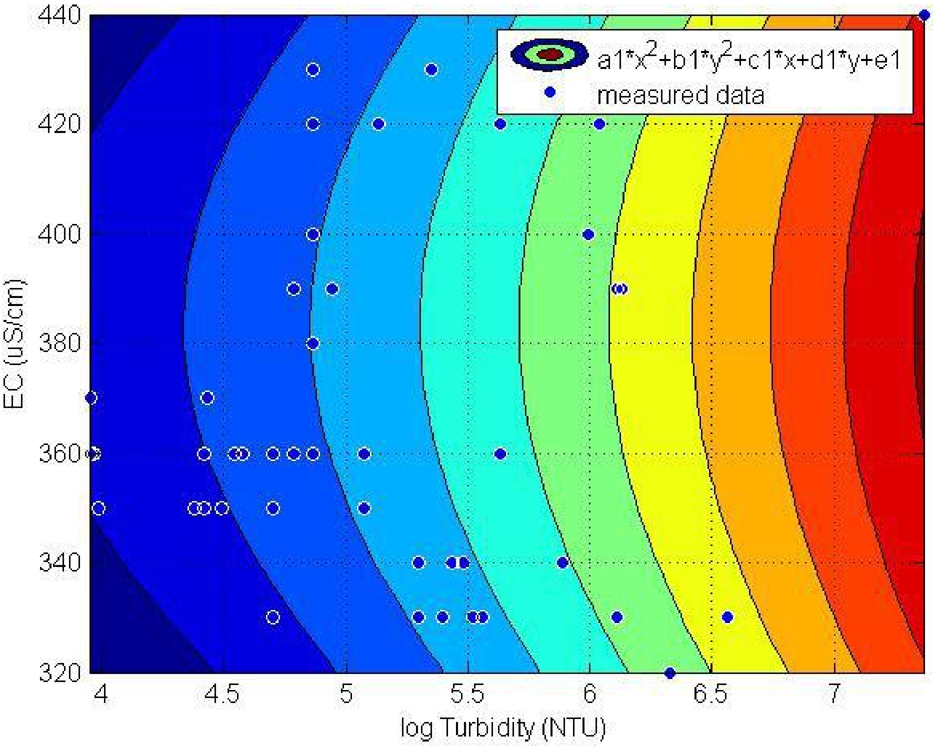


Figure VII.31 - Contour plots fit 1

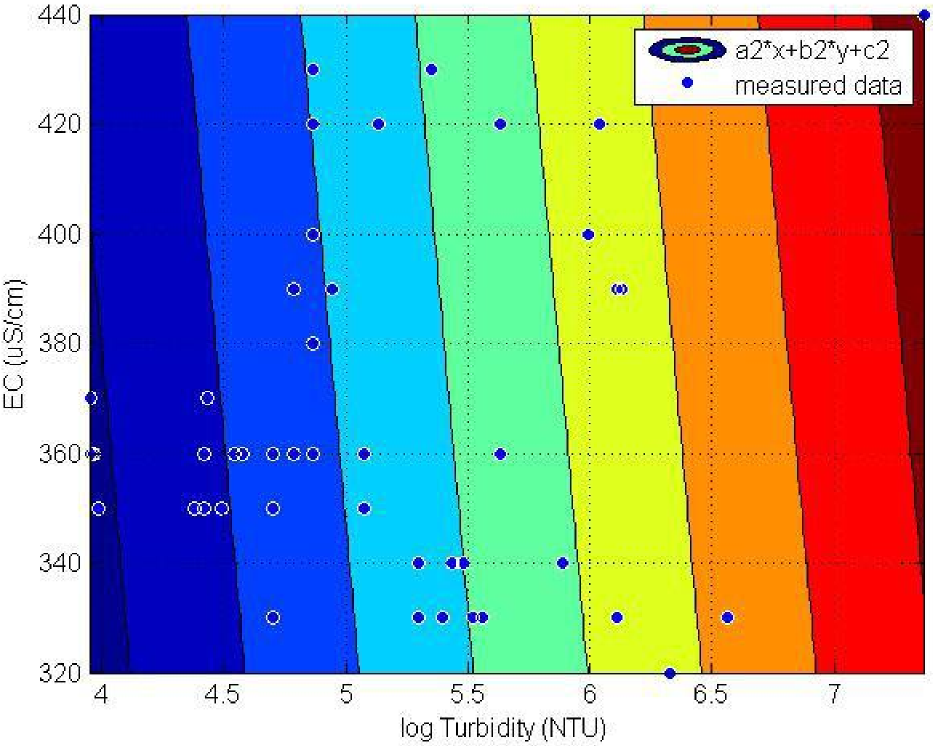


Figure VII.32 - Contour plots fit 2

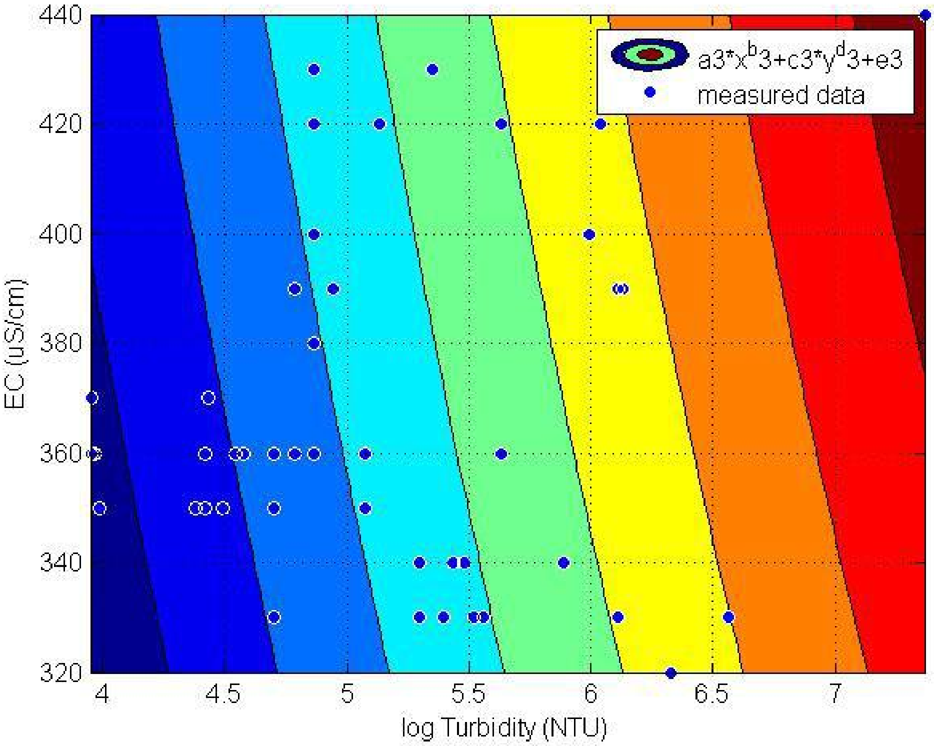


Figure VII.33 - Contour plots fit 3

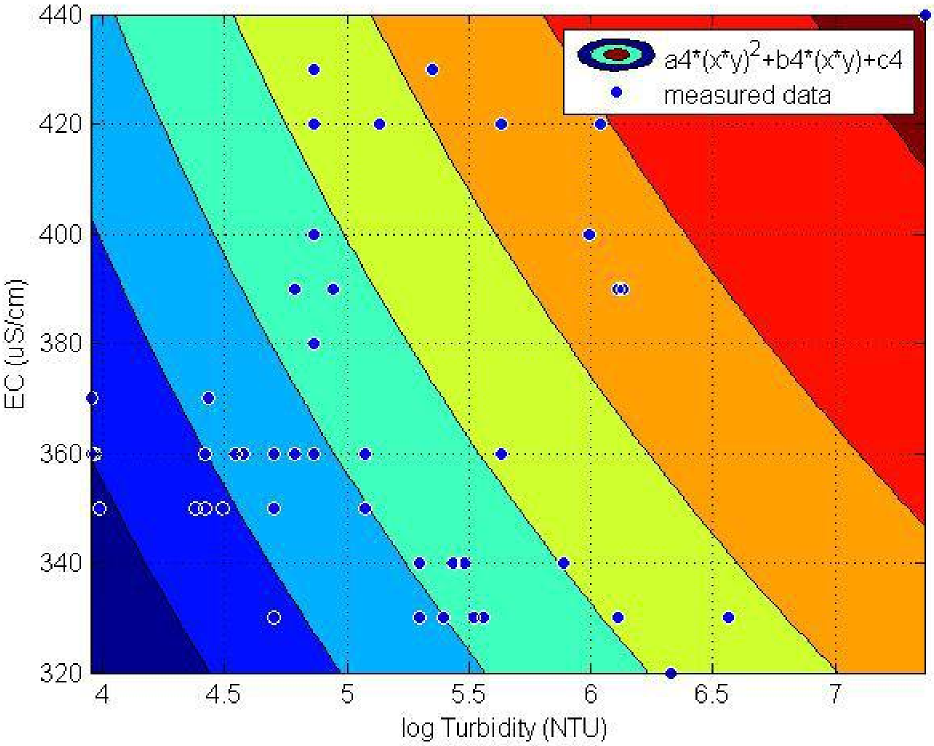


Figure VII.34 - Contour plots fit 4

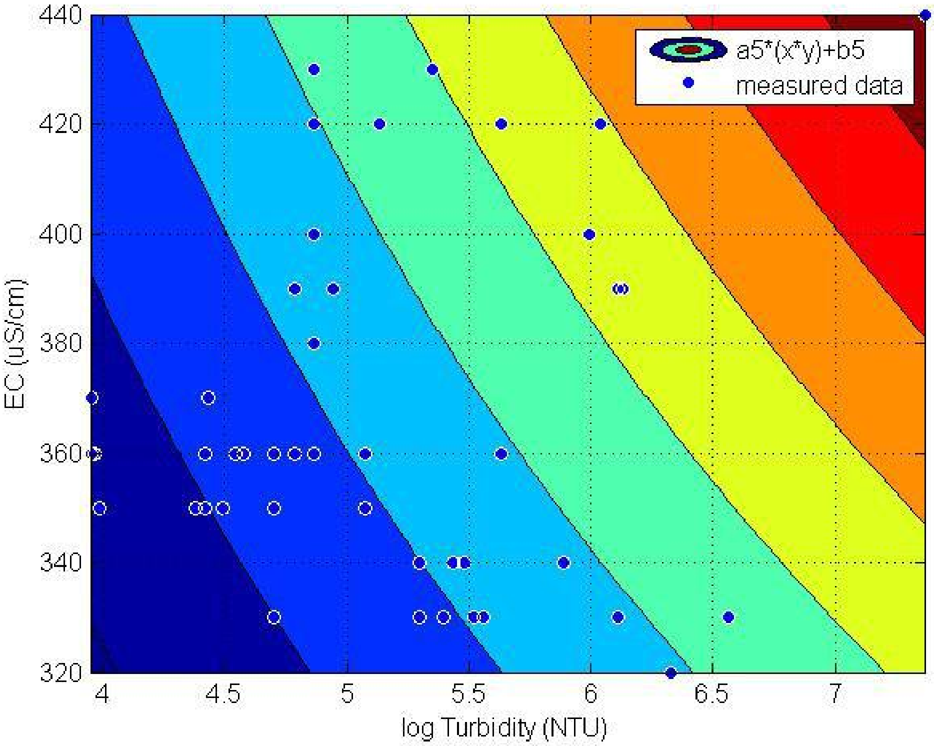


Figure VII.35 - Contour plots fit 5

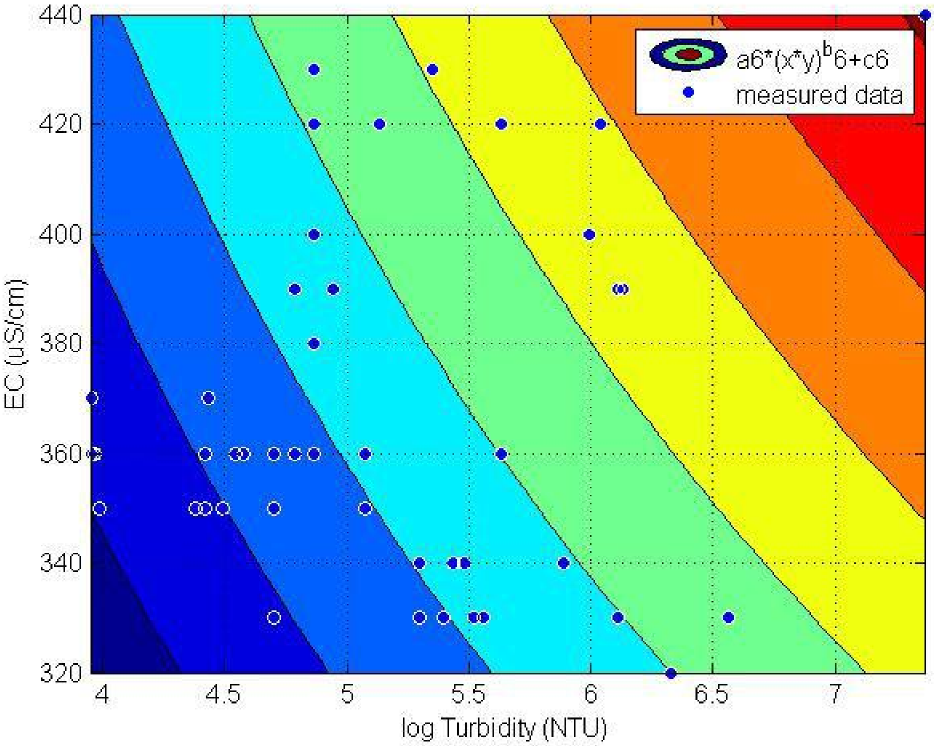


Figure VII.36 - Contour plots fit 6

Appendix VIII MATLAB script lab test

The script of the lab test consist of the script Fit100.m with the subscript concCOD100script.m.

Fit100.m

```
%Fit100(NTU100,EC100,concCOD100)
%  Fit surfaces to data.
%
%  Data for 'fit 1 Ger sq' fit:
%  Data for 'fit 2 Ger lin' fit:
%  Data for 'fit 3 Ger exp' fit:
%  Data for 'fit 4 Lom sq' fit:
%  Data for 'fit 5 Lom lin' fit:
%  Data for 'fit 6 Lom exp' fit:
%  Output:
%      fitresult : a cell-array of sfit objects representing the fits.
%      gof : structure array with goodness-of fit info.

%% Initialization.

% Initialize arrays to store fits and goodness-of-fit.
format compact
format shortG
concCOD100script;
fitresult = cell( 6, 1 );
gof = struct( 'sse', cell( 6, 1 ), ...
    'rsquare', [], 'dfe', [], 'adjrsquare', [], 'rmse', [] );

%% Fit: 'fit 1 Ger sq'.
[xInput, yInput, zOutput] = prepareSurfaceData( NTU100, EC100, concCOD100 );

% Set up fittype and options.
ft = fittype( 'a1*x^2+b1*y^2+c1*x+d1*y+e1', 'indep', {'x', 'y'}, 'depend', 'z' );
opts = fitoptions( ft );
opts.Algorithm = 'Levenberg-Marquardt';
opts.Display = 'Off';
opts.Lower = [-Inf -Inf -Inf -Inf -Inf];
opts.StartPoint = [0.983930621903036 0.277847208933994 0.158900814626516
0.870510605441892 0.261889223114753];
opts.Upper = [Inf Inf Inf Inf Inf];

% Fit model to data.
[fitresult{1}, gof(1)] = fit( [xInput, yInput], zOutput, ft, opts );

%% Fit: 'fit 2 Ger lin'.
[xInput, yInput, zOutput] = prepareSurfaceData( NTU100, EC100, concCOD100 );

% Set up fittype and options.
ft = fittype( 'a2*x+b2*y+c2', 'indep', {'x', 'y'}, 'depend', 'z' );
opts = fitoptions( ft );
opts.Algorithm = 'Levenberg-Marquardt';
opts.Display = 'Off';
opts.Lower = [-Inf -Inf -Inf];
opts.StartPoint = [0.934660987513619 0.272799877364135 0.831998396790234];
opts.Upper = [Inf Inf Inf];
```

```
% Fit model to data.
[fitresult{2}, gof(2)] = fit( [xInput, yInput], zOutput, ft, opts );

%% Fit: 'fit 3 Ger exp'.
[xInput, yInput, zOutput] = prepareSurfaceData( NTU100, EC100, concCOD100 );

% Set up fittype and options.
ft = fittype( 'a3*x^b3+c3*y^d3+e3', 'indep', {'x', 'y'}, 'depend', 'z' );
opts = fitoptions( ft );
opts.Algorithm = 'Levenberg-Marquardt';
opts.Display = 'Off';
opts.Lower = [-Inf -Inf -Inf -Inf -Inf];
opts.StartPoint = [0.577418358172537 0.948967010814119 0.645131878769838
0.223910658073472 0.141480043835504];
opts.Upper = [Inf Inf Inf Inf Inf];

% Fit model to data.
[fitresult{3}, gof(3)] = fit( [xInput, yInput], zOutput, ft, opts );

%% Fit: 'fit 4 Lom sq'.
[xInput, yInput, zOutput] = prepareSurfaceData( NTU100, EC100, concCOD100 );

% Set up fittype and options.
ft = fittype( 'a4*(x*y)^2+b4*(x*y)+c4', 'indep', {'x', 'y'}, 'depend', 'z' );
opts = fitoptions( ft );
opts.Algorithm = 'Levenberg-Marquardt';
opts.Display = 'Off';
opts.Lower = [-Inf -Inf -Inf];
opts.StartPoint = [0.412866035837833 0.99161666812192 0.0117062409795607];
opts.Upper = [Inf Inf Inf];

% Fit model to data.
[fitresult{4}, gof(4)] = fit( [xInput, yInput], zOutput, ft, opts );

%% Fit: 'fit 5 Lom lin'.
[xInput, yInput, zOutput] = prepareSurfaceData( NTU100, EC100, concCOD100 );

% Set up fittype and options.
ft = fittype( 'a5*(x*y)+b5', 'indep', {'x', 'y'}, 'depend', 'z' );
opts = fitoptions( ft );
opts.Algorithm = 'Levenberg-Marquardt';
opts.Display = 'Off';
opts.Lower = [-Inf -Inf];
opts.StartPoint = [0.603216840484718 0.0862266441713112];
opts.Upper = [Inf Inf];

% Fit model to data.
[fitresult{5}, gof(5)] = fit( [xInput, yInput], zOutput, ft, opts );

%% Fit: 'fit 6 Lom exp'.
[xInput, yInput, zOutput] = prepareSurfaceData( NTU100, EC100, concCOD100 );

% Set up fittype and options.
ft = fittype( 'a6*(x*y)^b6+c6', 'indep', {'x', 'y'}, 'depend', 'z' );
opts = fitoptions( ft );
opts.Algorithm = 'Levenberg-Marquardt';
```

```
opts.Display = 'Off';
opts.Lower = [-Inf -Inf -Inf];
opts.StartPoint = [0.259258973416716 0.373815955588613 0.865777800708306];
opts.Upper = [Inf Inf Inf];

% Fit model to data.
[fitresult{6}, gof(6)] = fit( [xInput, yInput], zOutput, ft, opts );

%% Plot data

% Plot fit with data.
%3D scatterplot
Figure( 'Name', 'data' );
scatter3(NTU100, EC100, concCOD100 );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf (g/l)' );
grid on

%2d scatterplot
Figure( 'Name', 'inputdata' );
scatter(NTU100,EC100)
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
grid on

%% Plot fits with data

% Plot fit with data.
Figure( 'Name', 'fit 1 Ger sq' );
h = plot( fitresult{1}, [xInput, yInput], zOutput );
legend( h, 'fit 1 Ger sq', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on

% Plot fit with data.
Figure( 'Name', 'fit 2 Ger lin' );
h = plot( fitresult{2}, [xInput, yInput], zOutput );
legend( h, 'fit 2 Ger lin', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on

% Plot fit with data.
Figure( 'Name', 'fit 3 Ger exp' );
h = plot( fitresult{3}, [xInput, yInput], zOutput );
legend( h, 'fit 3 Ger exp', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
```

```
zlabel( 'CODf' );
grid on

% Plot fit with data.
Figure( 'Name', 'fit 4 Lom sq' );
h = plot( fitresult{4}, [xInput, yInput], zOutput );
legend( h, 'fit 4 Lom sq', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on

% Plot fit with data.
Figure( 'Name', 'fit 5 Lom lin' );
h = plot( fitresult{5}, [xInput, yInput], zOutput );
legend( h, 'fit 5 Lom lin', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on

% Plot fit with data.
Figure( 'Name', 'fit 6 Lom exp' );
h = plot( fitresult{6}, [xInput, yInput], zOutput );
legend( h, 'fit 6 Lom exp', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on
%}

%% Plot residuals

% Plot residuals.
Figure ( 'Name', 'fit 1 Ger sq res' );
h = plot( fitresult{1}, [xInput, yInput], zOutput, 'Style', 'Residual' );
legend( h, 'fit 1 Ger sq - residuals', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on

% Plot residuals.
Figure ( 'Name', 'fit 2 Ger lin res' );
h = plot( fitresult{2}, [xInput, yInput], zOutput, 'Style', 'Residual' );
legend( h, 'fit 2 Ger lin - residuals', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on

% Plot residuals.
```

```
Figure ( 'Name', 'fit 3 Ger exp res' );
h = plot( fitresult{3}, [xInput, yInput], zOutput, 'Style', 'Residual' );
legend( h, 'fit 3 Ger exp - residuals', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on

% Plot residuals.
Figure ( 'Name', 'fit 4 Lom sq res' );
h = plot( fitresult{4}, [xInput, yInput], zOutput, 'Style', 'Residual' );
legend( h, 'fit 4 Lom sq - residuals', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on

% Plot residuals.
Figure ( 'Name', 'fit 5 Lom lin res' );
h = plot( fitresult{5}, [xInput, yInput], zOutput, 'Style', 'Residual' );
legend( h, 'fit 5 Lom lin - residuals', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on

% Plot residuals.
Figure ( 'Name', 'fit 6 Lom exp res' );
h = plot( fitresult{6}, [xInput, yInput], zOutput, 'Style', 'Residual' );
legend( h, 'fit 6 Lom exp - residuals', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
zlabel( 'CODf' );
grid on


%% Plot contours

% Make contour plot.
Figure ( 'Name', 'fit 1 Ger sq cont' );
h = plot( fitresult{1}, [xInput, yInput], zOutput, 'Style', 'Contour' );
legend( h, 'fit 1 Ger sq', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
grid on

% Make contour plot.
Figure ( 'Name', 'fit 2 Ger lin cont' );
h = plot( fitresult{2}, [xInput, yInput], zOutput, 'Style', 'Contour' );
legend( h, 'fit 2 Ger lin', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
```

```
ylabel( 'EC (uS/cm)' );
grid on

% Make contour plot.
Figure ( 'Name', 'fit 3 Ger exp cont' );
h = plot( fitresult{3}, [xInput, yInput], zOutput, 'Style', 'Contour' );
legend( h, 'fit 3 Ger exp', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
grid on

% Make contour plot.
Figure ( 'Name', 'fit 4 Lom sq cont' );
h = plot( fitresult{4}, [xInput, yInput], zOutput, 'Style', 'Contour' );
legend( h, 'fit 4 Lom sq', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
grid on

% Make contour plot.
Figure ( 'Name', 'fit 5 Lom lin cont' );
h = plot( fitresult{5}, [xInput, yInput], zOutput, 'Style', 'Contour' );
legend( h, 'fit 5 Lom lin', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
grid on

% Make contour plot.
Figure ( 'Name', 'fit 6 Lom exp cont' );
h = plot( fitresult{6}, [xInput, yInput], zOutput, 'Style', 'Contour' );
legend( h, 'fit 6 Lom exp', 'concCOD100 vs. NTU100, EC100', 'Location', 'NorthEast' );
% Label axes
xlabel( 'turbidity (NTU)' );
ylabel( 'EC (uS/cm)' );
grid on


%% Summarize results
%fitresult{1}

formulas=char(formula(fitresult{1}),
formula(fitresult{2}),formula(fitresult{3}),formula(fitresult{4}),formula(fitresult{5}),formula(fi
tresult{6}))
coefficients= ...
    [coeffvalues(fitresult{1}) numcoeffs(fitresult{1}) ...
    ;coeffvalues(fitresult{2}) 0 0 numcoeffs(fitresult{2}) ...
    ;coeffvalues(fitresult{3}) numcoeffs(fitresult{3}) ...
    ;coeffvalues(fitresult{4}) 0 0 numcoeffs(fitresult{4}) ...
    ;coeffvalues(fitresult{5}) 0 0 0 numcoeffs(fitresult{5}) ...
    ;coeffvalues(fitresult{6}) 0 0 numcoeffs(fitresult{6})]
'sum suare error|  r square |deg of freedom| adj r square|root mean square error'
gofsum=transpose([gof.sse; gof.rsquare;gof.dfe;gof.adjrsquare;gof.rmse])
```

concCOD100script.m

```
%%load data
load data/Md100.csv
load data/Me100.csv
load data/Mc100.csv
load data/Ms100.csv
load data/Mf100.csv
load data/NTU100.csv
load data/EC100.csv

%% clay concentration in 'father' sample
concclay= ... %g clay/g h20
    (51.48083-43.0276) ... %gclay=mass of (clay+container)-container
    /(2100.66-190.41-6.9); %gh20=mass of total-(clay+container)-stirring magnet

concclayl=...%g clay/ml H2O
    (51.48083-43.0276) ... %gclay=mass of salt
    /((2100.66-190.41-6.9)/.9982); %mlh20=volume of water = mass of water*density at 20 degrees

%% salt concentration in 'mother' sample
concsalt= ... %g salt/g h20
    (47.04358-35.66567) ... %gsalt=mass of (salt+container)-container
    /(1857.88-169.5); %gh20=mass of total-(salt+container+stirring magnet)

concsaltl=...%g salt/ml H2O
    (47.04358-35.66567) ... %gsalt=mass of salt
    /((1857.88-169.5)/.9982); %mlh20=volume of water = mass of water*density at 20 degrees

%% calculate CODf
concCOD100= ...
    (concclay.*(Mc100-Me100)./(Mf100-Md100) ... % father concentration * amount
    +concsalt.*(Ms100-Mc100)./(Mf100-Md100) ... %
    ).*10^5; %dimensionless parameter to make the CODf of the same order of magnitude as the turbidity and the EC
```

Appendix IX MATLAB script field test

The script for the field test consists of the script Ulvfit2.m with the following subscripts:

- Ulvimport2.m
- Ulvplotdata2.m
- Ulvuserchoices.m
- Ulvplotmoredata.m
- Ulvfitscript.m
- Ulvplotfit.m
- Ulvsum.m
- Ulvfitdata4.m

Ulvfit2.m

```
% Ulvfit(NTUUlv,ECUlv,CODUlv)
%  Fits surfaces and curves to data.
%  Output:
%      fitresult : a cell-array of fit objects representing the fits.
%      gof : structure array with goodness-of fit info.


%% Initialization

clear
close all
clc

% insttructions: use "1" to switch on

switchh.userquestions=0; % default is used when "0"
switchh.plotdata=1;
    switchh.plotdatatime=1;
    switchh.plotdatahist=1;
    switchh.plotdatascatter=1;
switchh.plotresult=0;
    switchh.plotfit=1;
    switchh.plotresi=1;
    switchh.plotconf=1;
    switchh.plotcont=1;
switchh.summary=1;
switchh.fitdata=1;
switchh.clearup=0;

errorcount=0;
errors=struct();
format compact
format shortG

Ulvimport2;                    % imports data from Ulvenhout

%% Plot data and fit prepare

if switchh.plotdata==1;
    Ulvplotdata2;
end

Ulvuserchoices;
```

```
if switchh.plotdata==1 & switchh.plotdatascatter==1
    Ulvplotmoredata;
end

%% Fit

Ulvfitscript;

%% Results

if switchh.plotresult==1;
    Ulvplotfit;              %plot fits, residuals and contours (if switches are 1)
end
if switchh.summary==1;
    Ulvsum;                        %shows summary of fit results
end
if switchh.fitdata==1;
     Ulvfitdata4;               %creates structures with fit data
end

%% cleanup

if errorcount>0                  %creates errorlog
    errorcount
    for i=1:errorcount
        errors(i).error
    end
end

if switchh.clearup==1;
    clear dataUlv3 timevector timestr timeserial events event tsc name axis %clears abundant import files
    clear xInput yInput zOutput customlabel legendname                      % clears abundant fit files
    clear n ans k i errorcount                                              % clears abundant misc files
end
```

Ulvimport2.m

```
%{
IMPORT ULVENHOUT
created: 10 February 2012 by Gerben Tommassen
last update: 9 May 2014 by Gerben Tommassen

imports the data of samples taken in Ulvenhout on:
- 5 may 2012
- 20 dec 2012
- 24 dec 2012

and stores it into workspace files usable for fitting (Ulvfit2.m).
%}

% Load data
load data/dataUlv3.csv; %loads the raw data

%% put time and data in structure

timeserial=datenum(dataUlv3 (3,:)+2000,dataUlv3 (2,:),dataUlv3 (1,:),dataUlv3
(4,:),dataUlv3 (5,:),0); %serial time of the sample
timedata=sortrows(transpose([timeserial;dataUlv3(7,:);dataUlv3(6,:)*1000;dataUlv3(8,:)]));
%serial time of the sample
timestr=(datestr(timedata(:,1)));
timevector=(datevec(timedata(:,1)));

events=unique(timevector(:,3));
%event(1:length(events))=struct('samples',[],'start',[],'end',[],'startnum',[],'endnum',
[],'startvec',[],'endvec',[]); % create empty data structure
for i=1:length(events)
    events(i,2)=sum(timevector(:,3) ==events(i,1));
    event(i).samples=events(i,2);
    event(i).start=timestr(sum(events(1:i,2))-events(i,2)+1,:);
    event(i).end=timestr(sum(events(1:i,2)),:);
    event(i).startnum=datenum(event(i).start);
    event(i).endnum=datenum(event(i).end);
    %event(i).startvec=datevec(event(i).start);
    %event(i).endvec=datevec(event(i).end);
end

data(1:21)=struct('name',[],'data',[],'axis',[]);                    % create empty data
structure

% mg/l, EC in uS/cm and Turbidity in NTU
data(1:3)=struct('name',{'COD','EC','Turbidity'},...
    'data',{transpose(timedata(:,2)), transpose(timedata(:,3)), transpose(timedata(:,4))},...
    'axis',{'COD (mg/l)','EC (uS/cm)','Turbidity (NTU)'});

try
    tsc=tscollection(timedata(:,1));                                % creates time series collection
    tsc.name='Raw data Ulvenhout';
    tsc.timeinfo.units='days';
    customlabel=['COD (mg/l), EC (uS/cm) & Turbidity (NTU)'];
    for i=1:3                                                   % puts raw data in time series
collection
        tsc= addts(tsc,timedata(:,i+1),data(i).name);
```

```
    end
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error='error: crash creating timeseries';
end

%% create adjusted datasets

%creates log datasets
for i=1:3
    name=['logarithmic ' data(i).name];
    axis=['log ' data(i).axis];
    data(9+i)=struct('name',name,'data',log(data(i).data),'axis',axis);
end

% creates standardized datasets
for i=[1:3 10:12]
    name=['standardized ' data(i).name];
    axis=['nom ' data(i).axis];
    data(3+i)=struct('name',name,'data',(data(i).data-
mean(data(i).data))./std(data(i).data),'axis',axis);
end

% creates uniformized datasets

for i=[1:3 10:12]
    name=['uniformized ' data(i).name];
    axis=['unif ' data(i).axis];
    data(6+i)=struct('name',name,'data',(data(i).data-min(data(i).data))./(max(data(i).data)-
min(data(i).data)),'axis',axis);
end
```

Ulvplotdata2.m

```
%{
PLOT DATA ULVENHOUT
created: 26 July 2013 by Gerben Tommassen
last update: 30  April 2014 by Gerben Tommassen
plots imported data in several ways
%}

%% plot data in timesteps
 if switchh.plotdatatime==1;
    Figure ('Name', 'Data over time1');
    for i=1:3
        for k=1:length(events)
            titlefig=['Event ' datestr(floor(event(1,k).startnum))];
            subplot(1,length(events),k); plot(tsc.(data(i).name),'-s');
            title(titlefig);
            xlim([event(k).startnum event(k).endnum])
            ylabel(customlabel);
            grid on
            hold on
        end
    end

    Figure ('Name', 'Data over time2 (x-axis not to scale)');
    for i=1:3
        for k=1:length(events)
            titlefig=['Event ' datestr(floor(event(1,k).startnum))];
            subplot(3,length(events),k+3*(i-1)); plot(tsc.(data(i).name),'-s');
            xlim([event(k).startnum event(k).endnum])
            ylim([min(data(i).data) max(data(i).data)])
            datetick('x','HH:MM','keeplimits', 'keepticks')
            if k==1
                ylabel(data(i).axis)
            elseif k==2
                ylabel ('')
            elseif k==3
                ylabel ('')
                legend(data(i).name,'Location','NE')
            end
            if i==1
                title(titlefig);
                xlabel ''
            elseif i==2
                title ''
                xlabel ''
            elseif i==3
                title ''
                xlabel 'time'
            end
            grid on
            hold on
        end
    end

    Figure ('Name', 'Data over time3');
    for i=1:3
```

```
        titlefig=[data(i).name ' over time'];
        subplot(4,1,i); stairs (data(i).data);
        title(titlefig);
        ylabel( data(i).axis );
        grid on
    end
    subplot (4,1,4); stairs(timeserial);
    title('date and time');
    xlabel('samples')
    ylabel( 'time (days)' );
    grid on
end

%plot histograms
if switchh.plotdatahist==1;
    for a=[1 10]
        if a==1;
            Figure ( 'Name', 'histograms data' );
        else Figure ( 'Name', 'histograms log data' );
        end
        for i=a:a+8;
            titlefig=['histogram ' data(i).name];
            subplot (3,3,i-a+1); hist(data(i).data,min(data(i).data):(max(data(i).data)-
min(data(i).data))/10:max(data(i).data));
            title(titlefig);
            xlabel(data(i).axis);
            grid on
        end
    end
end

%scatterplot
if switchh.plotdatascatter==1;

    for i=(1:6)*3-2
        titlefig=['Scatterplots ' i];
        Figure ('Name', titlefig);
        subplot (2,2,1); scatter3 (data(i+2).data,data(i+1).data,data(i).data);
        xlabel(data(i+2).axis);
        ylabel(data(i+1).axis);
        zlabel(data(i).axis);
        grid on
        for a=1:3
            if a<3
                b=a;
            else
                b=0;
            end
            subplot (2,2,a+1); scatter(data(i+a-1).data,data(i+b).data);
            xlabel(data(i+a-1).axis);
            ylabel(data(i+b).axis);
            grid on
        end
    end
end

clear a titlefig b extraswitches
```

Ulvuserchoices.m

```
%prepares user choices structure
choice=struct('name',{'autonorm' 'COD' 'EC' 'Turbidity' 'lowlimits' 'confidenceinterval'},...
    'choice',[],'default',{'n' 'l' '0' 'l' 'n' 0.95});

%asks the user if data should be autonormalized
if switchh.userquestions==1;
    choice(1).choice=input('auto normalize data? y/n [n]: ','s');
end
if isempty(choice(1).choice);
    choice(1).choice=choice(1).default;
end

%asks the user what inputdata should be used for fit
 for i=1:3
    if switchh.userquestions==1;
        question=['What inputdata should be used for ' data(i).name '? 0/n/u/l/ln/lu ['
choice(i+1).default ']: '];
        choice(i+1).choice=input(question,'s');
        %raw data, standard normalized, uniformed or their natural log (e) versions
    end
    if isempty(choice(i+1).choice);
        choice(i+1).choice=choice(i+1).default;
    end

    if strcmp(choice(i+1).choice,'0')==1;
        c=0;
    elseif strcmp(choice(i+1).choice,'n')==1;
        c=3;
    elseif strcmp(choice(i+1).choice,'u')==1;
        c=6;
    elseif strcmp(choice(i+1).choice,'l')==1;
        c=9;
    elseif strcmp(choice(i+1).choice,'ln')==1;
        c=12;
    elseif strcmp(choice(i+1).choice,'lu')==1;
        c=15;
    end
    % 'c' is the identifier for the type of inputdata.

    data(i+18).name=data(i+c).name;
    data(i+18).data=data(i+c).data;
    data(i+18).axis=data(i+c).axis;
    %puts inputdata for fit in collumns 19 to 21 of 'data' structure
end

%ask whether the lower limit for dependend parameter coefficients should be zero
if switchh.userquestions==1;
    choice(5).choice=input('Use realistic values for parameter coefficients? y/n (if y, trust-
region instead of L-M algorithms will be used) [n]:','s');
end
if isempty(choice(5).choice);
    choice(5).choice=choice(5).default;
end

clear c question
```

Ulvplotmoredata.m

```
%{
PLOT DATA ULVENHOUT
created: 1 May 2013 by Gerben Tommassen
last update: 1  May 2014 by Gerben Tommassen

plots imported data in the chose way

%}

titlefig=['Scatterplots ' 19];
Figure ('Name', titlefig);
subplot (2,2,1); scatter3 (data(21).data,data(20).data,data(19).data);
xlabel(data(21).axis);
ylabel(data(20).axis);
zlabel(data(19).axis);
grid on
for a=1:3
    if a<3
        b=a;
    else
        b=0;
    end
    subplot (2,2,a+1); scatter(data(18+a).data,data(19+b).data);
    xlabel(data(18+a).axis);
    ylabel(data(19+b).axis);
    grid on
end

clear a b
```

Ulvfitscript.m

```
%{
ULVFITSCRIPT does 3D-fits for data of Ulvenhout

created: 26 June 2013 by Gerben Tommassen
last update: 2  May 2014 by Gerben Tommassen

%}

%% Initialization

n=12;
fitresult = cell( n, 1 );       %Initialize arrays to store fits
gof = struct('sse', cell( n, 1 ), 'rsquare', [], 'dfe', [], 'adjrsquare', [], 'rmse', [] ); %Initialize arrays to store goodness-of-fit.
[xInput, yInput, zOutput] = prepareSurfaceData( data(21).data, data(20).data, data(19).data ); %% Prepare inputdata for fit
Ulvfit(1:12)=struct('num',[],'name',[]);

%% Fit: 'fit 1 Ger sq'.

try
    % Set up fittype and options.
    i=1;
    Ulvfit(i).num=i;
    Ulvfit(i).name='Gerben Square';
    ft = fittype( 'a1*x^2+b1*y^2+c1*x+d1*y+e1', 'indep', {'x', 'y'}, 'depend', 'z' );
    opts = fitoptions( ft );
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 0 0 0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf -Inf -Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 1 1 1 0]; %starting point iteration
    opts.Upper = [Inf Inf Inf Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit( [xInput, yInput], zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

%% Fit: 'fit 2 Ger lin'.

try
    % Set up fittype and options.
    i=2;
    Ulvfit(i).num=i;
    Ulvfit(i).name='Gerben Linear';
    ft = fittype( 'a2*x+b2*y+c2', 'indep', {'x', 'y'}, 'depend', 'z' );
```

```
    opts = fitoptions( ft );
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 1 0]; %starting point iteration
    opts.Upper = [Inf Inf Inf];%upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit( [xInput, yInput], zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

%% Fit: 'fit 3 Ger exp'.

try
    % Set up fittype and options.
    i=3;
    Ulvfit(i).num=i;
    Ulvfit(i).name='Gerben Exponential';
    ft = fittype( 'a3*x^b3+c3*y^d3+e3', 'indep', {'x', 'y'}, 'depend', 'z' );
    opts = fitoptions( ft );
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 0 0 0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf -Inf -Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 1 1 1 0]; %starting point iteration
    opts.Upper = [Inf Inf Inf Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit( [xInput, yInput], zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

%% Fit: 'fit 4 Lom sq'.

try
    % Set up fittype and options.
    i=4;
    Ulvfit(i).num=i;
    Ulvfit(i).name='Lombard Square';
```

```
    ft = fittype( 'a4*(x*y)^2+b4*(x*y)+c4', 'indep', {'x', 'y'}, 'depend', 'z' );
    opts = fitoptions( ft );
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 1 0]; %starting point iteration
    opts.Upper = [Inf Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit( [xInput, yInput], zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

%% Fit: 'fit 5 Lom lin'.

try
    % Set up fittype and options.
    i=5;
    Ulvfit(i).num=i;
    Ulvfit(i).name='Lombard Linear';
    ft = fittype( 'a5*(x*y)+b5', 'indep', {'x', 'y'}, 'depend', 'z' );
    opts = fitoptions( ft );
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 0]; %starting point iteration
    opts.Upper = [Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit( [xInput, yInput], zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

%% Fit: 'fit 6 Lom exp'.

try
    % Set up fittype and options.
    i=6;
    Ulvfit(i).num=i;
```

```
    Ulvfit(i).name='Lombard Exponential';
    ft = fittype( 'a6*(x*y)^b6+c6', 'indep', {'x', 'y'}, 'depend', 'z' );
    opts = fitoptions( ft );
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 1 0]; %starting point iteration
    opts.Upper = [Inf Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit( [xInput, yInput], zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

%% Fit: 'fit 7 COD-turbidity square'

try
    % Set up fittype and options.
    i=7;
    Ulvfit(i).num=i;
    Ulvfit(i).name='COD-Turbidity Square';
    ft=fittype('a7*x^2+b7*x+c7', 'indep', 'x', 'depend', 'z' );
    opts=fitoptions(ft);
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 1 0]; %starting point iteration
    opts.Upper = [Inf Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit(xInput, zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

%% Fit: 'fit 8 COD-turbidity linear'

try
    % Set up fittype and options.
    i=8;
```

```
    Ulvfit(i).num=i;
    Ulvfit(i).name='COD-Turbidity Linear';
    ft=fittype('a8*x+b8', 'indep', 'x', 'depend', 'z' );
    opts=fitoptions(ft);
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 0]; %starting point iteration
    opts.Upper = [Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit(xInput, zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

%% Fit: 'fit 9 COD-turbidity Exponential'

try
    % Set up fittype and options.
    i=9;
    Ulvfit(i).num=i;
    Ulvfit(i).name='COD-Turbidity Exponential';
    ft=fittype('a9*x^b9+c9', 'indep', 'x', 'depend', 'z' );
    opts=fitoptions(ft);
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 1 0]; %starting point iteration
    opts.Upper = [Inf Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit(xInput, zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end


%% Fit: 'fit 10 COD-EC square'

try
```

```
    % Set up fittype and options.
    i=10;
    Ulvfit(i).num=i;
    Ulvfit(i).name='COD-EC Square';
    ft=fittype('a10*y^2+b10*y+c10', 'indep', 'y', 'depend', 'z' );
    opts=fitoptions(ft);
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 1 0]; %starting point iteration
    opts.Upper = [Inf Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit(yInput, zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

%% Fit: 'fit 11 COD-turbidity linear'

try
    % Set up fittype and options.
    i=11;
    Ulvfit(i).num=i;
    Ulvfit(i).name='COD-EC Linear';
    ft=fittype('a11*y+b11', 'indep', 'y', 'depend', 'z' );
    opts=fitoptions(ft);
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 0]; %starting point iteration
    opts.Upper = [Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit(yInput, zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

%% Fit: 'fit 12 COD-turbidity linear'
```

```
try
    % Set up fittype and options.
    i=12;
    Ulvfit(i).num=i;
    Ulvfit(i).name='COD-EC Exponential';
    ft=fittype('a12*y^b12+c12', 'indep', 'y', 'depend', 'z' );
    opts=fitoptions(ft);
    if choice(1).choice=='y'
        opts.normalize = 'on';
    end
    opts.Algorithm = 'Levenberg-Marquardt';
    opts.Display = 'Off';
    if choice(5).choice=='y'
        opts.Lower = [0 0 -Inf]; %lower limit of coefficients
    else
        opts.Lower = [-Inf -Inf -Inf]; %lower limit of coefficients
    end
    opts.StartPoint = [1 1 0]; %starting point iteration
    opts.Upper = [Inf Inf Inf]; %upper limit of coefficients

    % Fit model to data.
    [fitresult{i}, gof(i)] = fit(yInput, zOutput, ft, opts );
catch
    errorcount=errorcount+1;
    errors(errorcount,1).error=['error: crash during fit ' Ulvfit(i).name];
end

clear ft opts
```

Ulvplotfit.m

```
%{
ULVPLOTFIT plots fits, residuals and contours for data of Ulvenhout

created: 26 June 2013 by Gerben Tommassen
last update: 9  May 2014 by Gerben Tommassen

%}

%% Plot fits with data
if switchh.plotfit==1
    for i=1:n
        try
            if length(indepnames(fitresult{i}))==2;
                titlefig=['fit ' Ulvfit(i).name];
                Figure( 'Name', titlefig );
                h = plot( fitresult{i}, [xInput, yInput], zOutput );
                legend( h, formula(fitresult{i}), 'measured data', 'Location', 'NorthEast' );
                % Label axes
                xlabel( data(1,21).axis );
                ylabel( data(1,20).axis );
                zlabel( data(1,19).axis );
                grid on
            elseif strcmp(indepnames(fitresult{i}),'x')==1;
                titlefig=['fit ' Ulvfit(i).name];
                Figure( 'Name', titlefig );
                h = plot( fitresult{i}, xInput, zOutput );
                legend( h, formula(fitresult{i}), 'measured data', 'Location', 'NorthEast' );
                % Label axes
                xlabel( data(1,21).axis );
                ylabel( data(1,19).axis );
                grid on
            elseif strcmp(indepnames(fitresult{i}),'y')==1;
                titlefig=['fit ' Ulvfit(i).name];
                Figure( 'Name', titlefig );
                h = plot( fitresult{i}, yInput, zOutput );
                legend( h, formula(fitresult{i}), 'measured data', 'Location', 'NorthEast' );
                % Label axes
                xlabel( data(1,20).axis );
                ylabel( data(1,19).axis );
                grid on
            end
        end
    end
end

%% plot residuals

if switchh.plotresi==1
    for i=1:n
        try
            if length(indepnames(fitresult{i}))==2;
                titlefig=['residuals ' Ulvfit(i).name];
                Figure ( 'Name', titlefig );
                h = plot( fitresult{i}, [xInput, yInput], zOutput, 'Style', 'Residual' );
                legend( h, 'residuals', 'Location', 'NorthEast' );
```

```
                % Label axes
                xlabel( data(1,21).axis );
                ylabel( data(1,20).axis );
                zlabel( data(1,19).axis );
                grid on
            elseif strcmp(indepnames(fitresult{i}),'x')==1;
                titlefig=['residuals ' Ulvfit(i).name];
                Figure ( 'Name', titlefig );
                plot( fitresult{i}, xInput, zOutput,'residuals' );
                legend( 'residuals');
                % Label axes
                xlabel( data(1,21).axis );
                ylabel( data(1,19).axis );
                grid on
            elseif strcmp(indepnames(fitresult{i}),'y')==1;
                titlefig=['residuals ' Ulvfit(i).name];
                Figure ( 'Name', titlefig );
                plot( fitresult{i}, yInput, zOutput,'residuals' );
                legend( 'residuals');
                % Label axes
                xlabel( data(1,20).axis );
                ylabel( data(1,19).axis );
                grid on
            end
        end
    end
end

if switchh.plotconf==1
    choice(6).choice=input('set confidence interval [0.95]: ');
    if isempty(choice(6).choice);
        choice(6).choice=choice(6).default;
    end
    legendname=['fit with ' num2str(choice(6).choice*100) '% confidence interval']
    for i=1:n
        try
            if length(indepnames(fitresult{i}))==2;
                titlefig=['confidence fit ' Ulvfit(i).name];
                Figure( 'Name', titlefig );
                h = plot( fitresult{i}, [xInput, yInput],
zOutput ,'Style','PredFunc','Level',choice(6).choice );
                legend( h, formula(fitresult{i}), legendname, 'Location', 'NE' );
                % Label axes
                xlabel( data(1,21).axis );
                ylabel( data(1,20).axis );
                zlabel( data(1,19).axis );
                grid on
            elseif strcmp(indepnames(fitresult{i}),'x')==1;
                titlefig=['confidence fit ' Ulvfit(i).name];
                Figure( 'Name', titlefig );
                h = plot( fitresult{i}, xInput, zOutput , 'predfunc', choice(6).choice );
                legend(formula(fitresult{i}), legendname, 'Location', 'NorthEast' );
                % Label axes
                xlabel( data(1,21).axis );
                ylabel( data(1,19).axis );
                grid on
            elseif strcmp(indepnames(fitresult{i}),'y')==1;
```

```
                titlefig=['confidence fit ' Ulvfit(i).name];
                Figure( 'Name', titlefig );
                h = plot( fitresult{i}, yInput, zOutput, 'predfunc', choice(6).choice  );
                legend(formula(fitresult{i}), legendname, 'Location', 'NorthEast' );
                % Label axes
                xlabel( data(1,20).axis );
                ylabel( data(1,19).axis );
                grid on
            end
        end
    end
end

if switchh.plotcont==1;
    for i=1:n
        try
            if length(indepnames(fitresult{i}))==2;
                titlefig=['contour ' Ulvfit(i).name];
                Figure ( 'Name', titlefig );
                h = plot( fitresult{i}, [xInput, yInput], zOutput, 'Style', 'Contour' );
                legend( h, formula(fitresult{i}), 'measured data', 'Location', 'NorthEast' );
                % Label axes
                xlabel( data(1,21).axis );
                ylabel( data(1,20).axis );
                grid on
            end
        end
    end
end

clear titlefig h
```

Ulvsum.m

```
%{
ULVSUM summarizes fits of data of Ulvenhout

created: 26 April 2014 by Gerben Tommassen
last update: 30  April 2014 by Gerben Tommassen

%}

for i=1:n
    try
        formula(fitresult{i})
    end
end

coefficients(i,7)=zeros;
for i=1:n
    try
        coefficients(i,1)=i;
        coefficients(i,2:1+numcoeffs(fitresult{i}))=coeffvalues(fitresult{i});
        coefficients(i,7)=numcoeffs(fitresult{i});
    end
end
coefficients

'sum square error|  r square |deg of freedom| adj r square|root mean square error' % creates Table head
gofsum=transpose([ gof.sse; gof.rsquare; gof.dfe; gof.adjrsquare; gof.rmse]) % creates Table with statistic parameters


clear gofsum
```

Ulvfitdata4.m

```
%{
ULVFITDATA makes a file with data of fits for Ulvenhout

created: 30 April 2013 by Gerben Tommassen
last update: 1 May 2014 by Gerben Tommassen

%}

%build structure
dataout(1:n)=struct('name',[],'rawvalue',zeros(50,1), 'CODproxy', zeros(50,1), 'residual',
zeros(50,1), 'meanresi',[],'stdresi',[]);


for i=1:n
    dataout(i).name=Ulvfit(i).name;
end

%generate raw data
for i=1:n
    try
        if length(indepnames(fitresult{i}))==2;
            dataout(i).rawvalue=fitresult{i}(xInput, yInput);
        elseif strcmp(indepnames(fitresult{i}),'x')==1;
            dataout(i).rawvalue=fitresult{i}(xInput);
        elseif strcmp(indepnames(fitresult{i}),'y')==1;
            dataout(i).rawvalue=fitresult{i}(yInput);
        end
    end
end

%generate proxy COD
if strcmp(choice(1).choice,'0')==1;
    for i=1:n
        dataout(i).CODproxy=dataout(i).rawvalue;
    end

elseif strcmp(choice(1).choice,'n')==1;
    for i=1:n
        dataout(i).CODproxy=dataout(i).rawvalue.*std(data(1).data)+mean(data(1).data);
    end

elseif strcmp(choice(1).choice,'u')==1;
    for i=1:n
        dataout(i).CODproxy=dataout(i).rawvalue.*(max(data(1).data)-min(data(1).data))
+min(data(1).data);
    end

elseif strcmp(choice(1).choice,'l')==1;
    for i=1:n
        dataout(i).CODproxy=exp(1).^(dataout(i).rawvalue);
    end

elseif strcmp(choice(1).choice,'ln')==1;
    for i=1:n
```

```
        dataout(i).CODproxy=exp(1).^(dataout(i).rawvalue.*std(data(1).data)
+mean(data(1).data));
    end

elseif strcmp(choice(1).choice,'lu')==1;
    for i=1:n
        dataout(i).CODproxy=exp(1).^(dataout(i).value.*(max(data(1).data)-
min(data(1).data))+min(data(1).data));
    end
end

%generate residues

for i=1:n
    dataout(i).residual=data(1).data'-dataout(i).CODproxy;
    dataout(i).meanresi=mean(dataout(i).residual);
    dataout(i).stdresi=std(dataout(i).residual);
end
```

Delft University of Technology
Faculty of Civil Engineering and Geosciences
Department of Water Management
Section of Sanitary Engineering
Stevinweg 1
2628 CN Delft

www.sanitaryengineering.tudelft.nl