Institute of Vehicle System Technology
Institute of Mobile Machines
Prof. Dr.-Ing. Marcus Geimer

Automation and Optimization of Agricultural Soil Tillage applying Machine Learning based on Machine- and Process Sensor Systems

Master's Thesis of

cand. M.Sc. Benjamin Kazenwadel
matriculation number: 1962114

at the Department of Mechanical Engineering
Institute of Vehicle System Technology
Institute of Mobile Machines

Advisor: Prof. Dr.-Ing. Marcus Geimer
Co-Advisors: M.Sc. Alexander Stein
M.Sc. Simon Becker

01. November 2020 – 28. April 2021

Karlsruhe Institute of Technology
Department of Mechanical
Engineering Kaiserstraße 12
76131 Karlsruhe

I truthfully assure that I have made the thesis independently, that I have specied all resources used completely and accurately and that I have labeled everything that has been taken from the work of others, either unchanged or with amendments, as well as that I have observed the articles of KIT for the Assurance of Good Scientic Practice as amended from time to time.

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfs mittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Änderungen entnommen wurde sowie die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet zu haben.

Karlsruhe, 28. April 2021

. . . .
(cand. M.Sc. Benjamin Kazenwadel)

Abstract

Climate change and cost pressure lead to new environmental and economic challenges that increase the demand for innovative control systems to automate and optimize agricultural tasks.
Automating speed control during power-intensive soil tillage can increase eciency and sustainability and counteract the lack of qualied personnel in agriculture. A survey was carried out focused on tillage by cultivating to obtain an overview of the challenges farmers face during their work, including their target preferences. Based on the obtained requirements for tillage by cultivating, a system was developed automating working depth control by online Lidar plane detection to ensure tillage quality and establish a basis for good plant growth. Automated speed control is realized based on an online-parameterized draft force and traction model combined with the usage of a neural network for fuel rate prediction. The network is trained oine and adaptable to the individual preferences of the farms and varying implements. Thereby, the operator can choose and customize optimization objectives such as fuel eciency, performance, or total cost.
During the evaluation, the control system was tested with various objectives against a hu man driver and was able to perform optimization on the individual objective. Furthermore, the transferability of the system was demonstrated with the usage of another implement.

Keywords:
Speed Control, Fuel Economy, Agricultural Tillage, Articial Neural Networks

Kurzfassung

Durch die Veränderung ökologischer sowie ökonomischer Randbedingungen durch Klima wandel und Kostendruck werden innovative Kontrollsysteme in der Landtechnik benötigt, um die Arbeitsabläufe zu automatisieren sowie zu optimieren.
Hierbei kann eine Automatisierung der Geschwindigkeitssteuerung bei leistungsinten siver Bodenbearbeitung die Ezienz sowie Nachhaltigkeit des Bearbeitungsprozesses optimieren. Zudem kann dies dem Mangel an qualiziertem Fachpersonal im Landwirt schaftssektor entgegenwirken.

Um einen Überblick über die Herausforderungen sowie Präferenzen von Landwirten während deren Arbeit zu erhalten, wurde eine Umfrage mit dem Fokus auf die Bodenbear beitung durch Grubbern durchgeführt.
Auf der Basis der gewonnenen Erkenntnisse wurde ein System zur Automatisierung der Arbeitstiefensteuerung auf Basis einer Ebenenerkennung durch Lidarmessungen ent wickelt. Die Automatisierung der Geschwindigkeitssteuerung wurde durch ein online parametrisiertes Zugkraft- sowie Traktionsmodell in Kombination mit einem künstlichen neuronalen Netz zur Vorhersage des Kraftstoverbrauches umgesetzt. Das Netzwerk wurde hierbei oine auf Basis zuvor aufgenommener Messdaten trainiert und ist durch den modu laren Aufbau adaptiv auf die individuellen Präferenzen verschiedener landwirtschaftlicher Betriebe sowie für die Nutzung an unterschiedlichen Anbaugeräten anwendbar. Hierbei kann der Bediener die Zielfunktionen, nach denen die Geschwindigkeit optimiert werden soll, individuell und während des Betriebs auswählen und verändern, beispielsweise für eine Optimierung der Kraftstonutzung oder der Betriebskosten.
Bei der durchgeführten Evaluierung wurde das System gegen einen menschlichen Fahrer als Referenz getestet und konnte dabei die Optimierung auf die gewünschte Zielfunktion gewährleisten. Zusätzlich wurde auch die Transferfähigkeit durch eine Testfahrt mit einem anderen Anbaugerät gezeigt.

Schlüsselwörter:
Geschwindigkeitssteuerung, Kraftstoezienz, Landwirtschaftliche Bodenbearbeitung, Künstliche Neuronale Netzwerke

Acknowledgements

This work took a lot of time and eort, and therefore in the following, I would like to thank all those who supported me during this period.

Firstly I would like to thank Prof. Dr.-Ing. Marcus Geimer, M.Sc. Alexander Stein and M.Sc. Simon Becker for the opportunity to write this thesis at the Institute for Mobile Machines (MOBIMA). I want to express my special thanks for all the support and help during this work. It is absolutely not a matter of course that this was made possible despite the ongoing pandemic and the high amount of necessary practical activities, such as test drives.
My personal thanks go to Traugott and Karl Edel for permitting the usage of his elds during the data collection and evaluation.
While surveying cultivator usage, I experienced a lot of kindness and support from the participating farmers, which helped me a lot to obtain an overview of the requirements and problems they face during their work. I am very grateful for this.
All of this would not have been possible without the people and companies associated with the project *Learning control systems for mobile working machines II*. I want to express my thanks for allowing the usage of the data collected during the project and especially to the companies AGCO/Fendt and Horsch for providing the tractor and cultivator. For his support during the acquisition of the theoretical foundations that have been necessary for the machine model, I would like to thank Dipl.-Ing. André Grosa of the TU Dresden for supporting me with additional information regarding cultivator draft force requirements.
I would also like to thank all contributors and developers of the Ubuntu, ROS,

WebPlot Digitizer, PlotJuggler, GitLab, TeXstudio, and Zotero packages for contributing their time to open-source projects that were used during this thesis.

Last but not least, my family and friends for supporting me during the writing process.

Contents

Nomenclature

Acronyms

ROS Robot Operating System

TMS Tractor Management System

ANN Articial Neural Networks

LTS Long Term Support

API Advanced Programming Interface

GUI Graphical User Interface

ECU Electronic Control Unit

IMU Inertial Measurement Unit

GPU Graphical Processing Unit

RANSAC Random Sample Consensus

Symbols

P_T W Traction Power

P_D W Draft Power

F_P N Pulling Force

F_T N Traction Force

$\tilde{F}_T$ N Modeled Traction Force

F_W N Weight Force

$F_{W,h}$ N Horizontal Weight Force

$F_{W,v}$ N Vertical Weight Force

F_D N Draft Force

$\tilde{F}_D$ N Modeled Draft Force

F_{roll} N Rolling Resistance

$\tilde{F}_{roll}$ N Modeled Rolling Resistance

F_{air} N Air Resistance

F_{acc} N Acceleration Force

v_{theo} km/h Theoretical Speed

$\tilde{v}_{theo}$ km/h Modeled Theoretical Speed

v_{gnss} km/h GNSS Speed

$\tilde{v}_{gnss}$ km/h Modeled GNSS Speed v_{radar}
km/h Radar Speed

v_{set} km/h Setpoint Speed

m kg Mass of the vehicle g m/s^2 Gravitational

Constant n 1/min Engine Revolutions $\tau_{\%}$ %

Relative Engine Torque i Gear Ratio

p % Implement Position α ° Pulling Angle

θ ° Pitch Angle

δ ° Slope

M_w Workmode

B l/h Fuel Rate

$\tilde{B}$ l/h Predicted Fuel Rate T °C Ambient

Temperature σ Slippage

$\tilde{\sigma}$ Modeled Slippage

κ Traction Coecient ϱ Rolling Resistance Coecient

t cm Working Depth

w m Working Width

c_{mp} Multipass Constant R Reward

C Cost

s Standard Deviation of Heights MUD Mean Upslope

Depression Index

1 Introduction

Due to the environmental challenges caused by climate change and resulting demands to make processes more ecient and sustainable, existing procedures must be examined closely to reduce greenhouse gas emissions.

Although the share of total fossil fuel consumption in agriculture is comparatively low with less than 3-4.5 % of the total energy budget even in heavily industrialized countries, every improvement in eciency does also mean reducing emissions, and a much-desired counteract to the cause of global warming. [1]

At the same time, agricultural technology underwent a signicant transformation from small farms, with a high demand for manual labor, to large farms, with correspondingly higher machinery usage. These machines require well-educated operators, and in com bination with the need to meet the stricter environmental regulations, cost pressure is increasing for the farmers.

All these challenges require the total usage of the existing optimization potential. This potential includes technical improvements of the machinery on the one hand but also control optimization on the other. Since many agricultural tasks are still partly controlled by a human operator, improved automated control systems can increase eciency and decrease operating costs.

Simultaneously, due to innovations in technologies, Articial Neural Networks (ANN) open new ways to optimize and automate processes that were tied to the possibilities of strictly human-designed control loops so far.

This work focuses on improving the power and fuel-intensive soil tillage with this new potential. As an exemplary use case, the choice of implement fell on the cultivator for its exible usage from stubble cultivating to seedbed preparation. Human drivers currently operate these machines, and while steering control is already often handled automatically using GPS measurements, speed control is still handled manually.

There are already automatic engine and transmission optimization systems on the market, but target speeds are dened manually, and these systems solely optimize the internal variables to t the specied target speed explicitly, if possible. Adapting this target speed can yield further improvement since poorly chosen speeds can force the engine and transmission to select worse operating points to supply the resulting draft force requirements. The automation of these speed specications reveals new optimization potential. Here, solely on optimizing the speed specications up to 8 % in fuel savings can be expected. [2]

The rst chapters of this thesis are structured to give essential information about the cultivation process, and model approaches of the forces that occur during this process are presented. The following chapters give the necessary background information about ANNs

and their structure and training used during this thesis. Furthermore, currently available control systems for agricultural tillage are explained, and state-of-the-art attempts to optimize speed control and evaluate process quality. The next chapter presents the results of a survey among farmers about cultivator usage to maintain practical relevance. The following chapter describes the measurement setup and insights into the recorded datasets. From the resulting requirements and insights, a new system for working depth and speed control is presented that combines the modeling of vehicle forces with the usage of an ANN to predict the fuel rate of the system. The concluding chapters include an evaluation of the system as well as a summary and an outlook for further research.

2

2 Fundamentals

This chapter describes the necessary fundamentals relevant to this work. Firstly, general information about the cultivation process on which this work is based is given. Secondly, vehicle and implement modeling techniques are described. Furthermore, relevant software for data collection and ltering methods are presented, followed by point cloud processing techniques. Finally, information about the structure and training of articial neural networks is given.

2.1 Cultivators

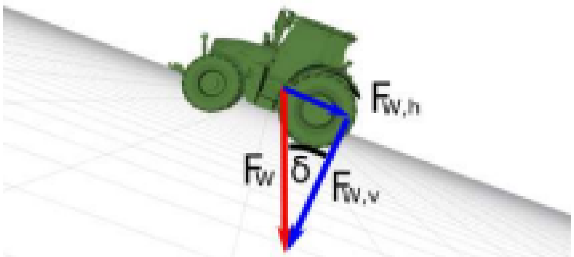
Cultivators are tillage implements that loosen, mix and re-compress the soil. Blades or tines are mounted on a frame to loosen and mix the soil, depending on the exact design. Flexible spring elements secure the implement from mechanical overload. Cheaper options feature shear bolts that are usually installed on the individual tines, which absorb excessive force by shearing to prevent damage to the whole implement. After loosening the soil, disks for further mixing and a wide variety of packers, ring-shaped metal elements, can be mounted to ensure the soil's subsequent compression and an even surface.
As no exact working depth is necessary or specied, cultivators can be used in a wide variety of use cases, just like stubble cultivation as well as seedbed preparation. Due to these exible application possibilities, cultivators' use steadily increasing, on some farms even replacing slow and performance-intensive plowing.
There is a wide variety of specialized cultivators on the market, adapted to the specic use case. [3][4]

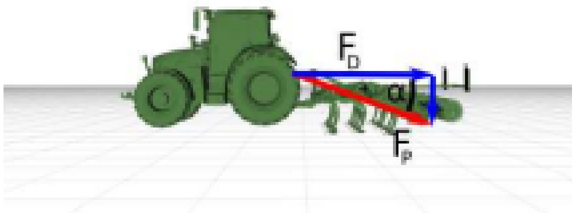
2.2 Vehicle Forces

This section lists the relationships and parameters necessary to describe the power trans mission between wheel and ground surface. Thereby it describes the inuence of the weight of the vehicle and implement combination on traction and drag forces caused by the implement and transmission resistances.

The Weight Force (F_W**)** of a vehicle is calculated using the vehicle's mass m and the acceleration due to gravity g.

$$F_W = m \cdot g \quad (2.1)$$





(a) Slope and Weight Force (b) Pulling Force and Draft Force

Figure 2.1: Vehicle Forces, Models extracted from [5] and [6]

The inuence of the slope(δ) on the distribution of the weight force (F_W) to horizontal weight force ($F_{W,h}$) and vertical weight force($F_{W,v}$) horizontally and vertically to the ground is described in gure 2.1a.
This leads to the following relationships:

$$F_{W,v} = cos(\delta) \cdot F_W \quad (2.2)$$

$$F_{W,h} = sin(\delta) \cdot F_W \quad (2.3)$$

The Slippage (σ) describes the relationship between eective speed (GNSS Speed (v_{gnss}) or Radar Speed (v_{radar})) and theoretical speed (v_{theo}) of the vehicle. [7] In the following v_{gnss} will be used instead of v_{radar}, because of superior accuracy.

$$\sigma = \frac{v_{theo} - v_{gnss}}{v_{theo}} \quad (2.4)$$

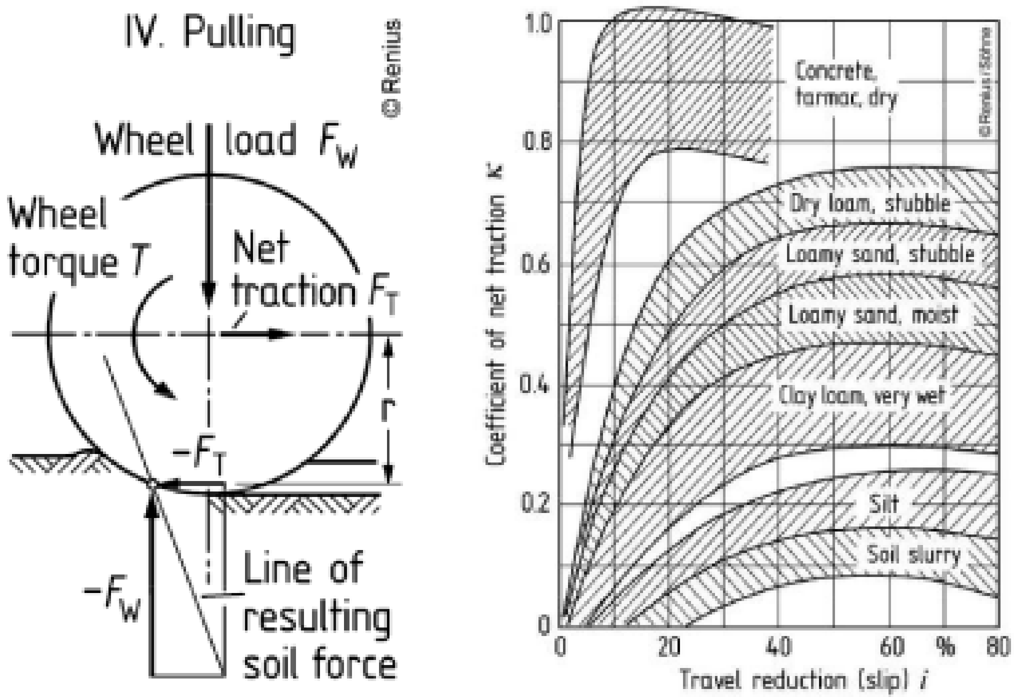
The Traction Coeicient (κ) is the relationship between the traction force (F_T) and vertical weight force ($F_{W,v}$) on a single wheel-ground contact.

$$\kappa = \frac{F_T}{F_{W,v}} \quad (2.5)$$

These forces are illustrated in gure 2.2a.
κ is often brought into relationship with σ and highly depends on the contents and hu midity of the soil as shown in gure 2.2b. [7]

For a 4-wheel drive tractor, the total force the tractor can provide in order to move forward, the traction force, can be calculated using the sum of the individual axle forces $F_{T,f}$ (front) and $F_{T,r}$ (rear) in the following formula: [8]

$$F_T = F_{T,f} + F_{T,r} \quad (2.6)$$



(a)

Tire-soil mechanics in pulling mode [7] (b) Relationship between κ and σ [7]

Figure 2.2: Traction Modeling

Horizontal Dra Force (F_D) that is caused by the drag of the implement can be calculated from the total pulling force (F_P) using the dependency on the pulling angle (α) as visualized in gure 2.1b. [7]

The characteristics of this parameter highly depend on the choice of implement and the speed of the vehicle.

$$F_D = cos(\alpha) \cdot F_P \quad (2.7)$$

Multipass-Eect describes the changes in soil caused by a consecutive rollover in the same trajectory. The soil is compacted due to the load of the tire/vehicle. During subse quent passes more traction force can be used as shown in gure 2.3a. The rst pass shows the highest eect on soil density change. [9][10]

The negative eects on plant growth caused by increased pressure on the soil during consecutive passes will not be discussed here since the focus here is solely the power transmission. [11]

Rolling Resistance (F_{roll}) is another impact factor mainly caused by deformation of the tires, the soil, and slip sinkage. It is calculated using equation 2.8 using the included rolling resistance coecient (ϱ). [7]

σ

σ

(b) Rolling Resistance, edited after [12]

(a) Multipass Eect, edited after [9]

Figure 2.3: Inuences on Traction

$$F_{roll} = \varrho \cdot F_{W,v} \quad (2.8)$$

The parameter is aected by the slippage as described in gure 2.3b.

The Acceleration Force (F_{acc}**)** counteracting the acceleration of the vehicle, is calculated by equation 2.9.

$$F_{acc} = m \cdot \frac{dv_{gnss}}{dt} \quad (2.9)$$

Air Resistance (F_{air}**)** does also reduce the amount of force that can be used to pull the implement forward, but due to the relatively low speeds of tractors during tillage, the impact can be neglected. [13]

2.2.1 Traction Force Simplifications

The tractive behavior of driven wheels is well known, and this section describes the con nection between the behavior of a single wheel and the tractor as a whole.

The traction force (F_T) of a vehicle is described in relation to κ in *Fundamentals of Tractor Design*. Adapting the relationship to the exclusive usage on a 4-wheel-drive tractor using equation 2.6 leads to equation 2.10. [7]

$$F_T = \kappa_f \cdot F_{W,v,f} + \kappa_r \cdot F_{W,v,r} \quad (2.10)$$

Renius proposed that ratio between κ_f and κ_r caused by the multi-pass eect can be described with formula 2.11 using the multipass constant (c_{mp}), that depends only on the tire dimensions and soil conditions. [7]

$$\kappa_r \; c_{mp} \cdot \kappa_f \quad (2.11)$$

Using 2.10 this leads to:

$$F_T = \kappa_f \cdot (F_{W,v,f} + c_{mp} \cdot F_{W,v,r}) \quad (2.12)$$

In the simplied case, when conditions and weight forces remain constant during obser vation, this relationship proposes that the function for κ of the whole vehicle will also follow the characteristic behavior estimated for a single wheel because all other parameters remain constant.

For this case equation 2.12 can be simplied further:

$$F_T = \kappa \cdot \mathit{const} \quad (2.13)$$

2.2.2 Traction Modeling

The relationship between κ and σ follows a characteristic behavior as shown in gure 2.2b. There have been many empirical approaches in modeling this relationship to describe the eects of power transfer between wheel and ground.

Jahns and Steinkampf describe the relationship between κ and σ with the following parameters: [14]

$$\kappa(\sigma) = a + b \cdot e^{c \cdot \sigma} \quad (2.14)$$

This equation has been improved by Schreiber and Kutzbach [15] to improve usage in simulations, because the previous model cannot represent the decrease of the traction coecient at high σ values:

$$\kappa(\sigma) = a - b \cdot e^{-c \cdot \sigma} - d \cdot \sigma \quad (2.15)$$

The included parameters were further improved by Meiners; Böttinger and Regazzi [16], measuring not single tires, but on a two-wheel-drive tractor. They concluded that it is possible to calculate the traction force from the pulling force and a measured or assumed rolling resistance of the front tires.

Their approach allows for a realistic simulation of κ under actual operating conditions as shown in gure 2.4.

σ

σ

Figure 2.4: Simulated κ/σ-curves for front and rear wheel of a standard tractor under dierent ground conditions, edited after [16]

Pacejka presented another, more complex yet accurate empirical model known as the *Magic Formula*. Using this formula, traction forces can be modeled using factors for stiness, shape, curvature as well as a peak value. [17][18]

In forestry technology the relationship $F_T(\sigma)$ for a whole vehicle is described by

Jacke and Drewes using second-degree polynomials in equation 2.16. This equation yields a simplied relationship between Traction Force (F_T) and σ that only requires three parameters. In this study the pulling force was measured on a 4-axle forwarder that was set under load by an attached braking tractor. It shows that characteristic behavior of the $\kappa(\sigma)$-curve (gure 2.2b) also applies to whole vehicles F_T (σ)-curve (gure 2.5) even if the multipass eect inuences tractive forces of several axes. [19]

$$F_T(\sigma) = a_T + b_T \cdot \sigma + c_T \cdot \sigma^2 \quad (2.16)$$

This direct description of the relationship using $F_T(\sigma)$ conrms the simplications resulting in equation 2.13.

2.2.3 Dra Force Modeling

The implement choice also decides upon the draft force (F_D) caused by an increase of the eective speed (here: v_{gnss}). This correlation diers signicantly between various implements.

σ

Figure 2.5: Traction Force and Slippage (without brush-wood), edited after [19]

Harrigan and Rotz parameterized this relationship using equation 2.17. A comprehensive collection of parameters for dierent implements can be found in “Draft Relationships for Tillage and Seeding Equipment”. [20]

The parameters important for cultivators have been listed in table 2.1. Therefore also working depth (t) and working width (w) have to be considered. For some implements like cultivators, the number of rows or tools is used instead of the width of the processing area.

In general for non-turning tillage the parameter c_D is set to zero. [2]

$$F_D = s_D \cdot (a_D + b_D \cdot v_{gnss} + c_D \cdot v_{gnss}^2) \cdot w \cdot t \quad (2.17)$$

Table 2.1: Implement Parameters for Cultivating [20]

Process	a_D	b_D	c_D
Primary tillage	46.4	2.77	0
Secondary tillage	32.0	1.94	0

The included parameter s_D depends on soil conditions that are hard to measure continu ously.

These ndings were later incorporated into the standard “ASAE D497.4 Agricultural Ma chinery Management Data”. [21]

Furthermore, studies by Getzla [22] and Gebresenbet [23] showed that increasing working depths result in a progressive increase in draft force for the plowing process. On the cultivating process Reich [24] also expected a progressive increase in draft force with increasing working depth due to higher static shear resistance in deeper soil layers. However, he could not prove this eect during measurements.

To t the model to the above presented issues in the context of online simulation based on currently observed conditions, Rößler; Kautzmann and Geimer [25] proposed equation 2.18 for cultivators, as well as dierent equations for modeling other implements. The parameter $q = a_D/b_D$ is thereby set to 20 because it remains constant during operation. This simplication means that all unknown variables can now be combined into one ($X = s_D \cdot c \cdot w$), which is determined during operation based on currently measured values.

$$F_D = s_D \cdot (c \cdot q + c \cdot v_{gnss}) \cdot w \cdot t^2 \quad (2.18)$$

Another modeling approach is presented in as equation 2.19. According to Al-Neama [26] this equation can be traced back to the Soviet agricultural technology researcher Gor jatschin and was originally designed to describe the horizontal forces of moldboard plow usage.

$$F_D = w \cdot t \cdot (k + e \cdot v_{gnss}^2) \quad (2.19)$$

Grosa [27][28] conducted measurements with cultivators based on equation 2.19. In contrast to previous evaluations of the model limited to 8 km/h, velocities up to 12 km/h were examined. The derived relationships apply to the individual tine as well as to the whole implement. He concluded that implement specic inuence in the model does also change due to changes in v_{gnss}. Therefore, the model is not usable during online simulation. However, the measured relationships describe a quadratic increase in the tractive force requirements with increasing v_{gnss} as shown in gure 2.6.

Bögel [29] conducted further experiments using single cultivator tines to evaluate F_T in relationship with v_{gnss} as well as soil prole changes during the process. Thereby a steep increase in horizontal draft forces was observed due to an increase in v_{gnss}. This work's focus was mainly on changing mounting angles on the tines to keep work quality constant despite speed changes.

Al-Neama [26] provided a extensive, tine-specic comparison between dierent draft force regression-models. He concludes that the draft force can be calculated by the sum of horizontal forces on the tines. Further he advises against the use of equation 2.19, since required coecients are dicult to determine.

Figure 2.6: Measured relationship between F_D and v_{gnss} on a cultivator, edited after [27]

2.3 Data Filtering

Several approaches can be chosen to lter noisy sensor data. The here mentioned Arith metic Mean and other alternatives are described in *The Concise Encyclopedia of Statis tics*. [31]

The Arithmetic Mean is calculated by averaging over a specied amount of values.

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \quad (2.20)$$

It can also be applied continuously on the last measured values. Then it is called Moving Average Filter. [31]

Only states from the past can be observed in live systems, so the window n cannot be chosen too wide to not lack behind in time. Another more sophisticated approach for ltering an incoming data stream would be to use a Kalman lter that is not as heavily inuenced by outliers.

2.4 Least-Squares Function Approximation

The least-squares method is a data tting method that minimizes the squared errors' sum to t a model to an observed dataset. The method is mainly used in linear regression to approximate functions.
For a simple linear regression using ϵ as a non-observable error:

$$y = \beta_0 + \beta_1 \cdot x + \epsilon \quad (2.21)$$

the method yields the estimated parameters $\hat{\beta}_0$ and $\hat{\beta}_1$:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \quad (2.22)$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \cdot x_i \quad (2.23)$$

The method can also be applied to more complex regression models. [31]

2.5 Surface Estimation

2.5.1 Random Sample Consensus (RANSAC)

The RANSAC-Algorithm is used to estimate a model into a dataset containing many outliers and uncertainties. It is frequently used for plane detection and is known for being very robust.
According to Fischler and Bolles [32] the following steps have to be taken to estimate a plane model:

- Randomly select three points from the dataset
- Build the model using the selected points
- Count how many points are part of the model given a certain

tolerance • Optimize the model (often by using least squares)

- Repeat the above as often as required
- Select the best-evaluated model

2.5.2 Standard Deviation of Heights (s)

The usage of the parameter Standard Deviation of Heights is a simplistic approach for calculating and describing surface roughness.
The parameter is calculated with equation 2.24. [33]

$$s = \sqrt{\frac{\sum_{i=1}^{N} z_i^2 - \bar{z}^2}{N - 1}} \qquad (2.24)$$

z_i is the height of a specic point in the estimation dataset, $\bar{z}$ the mean height of all points, and N the number of points.

2.5.3 Mean Upslope Depression Index (MUD)

The Mean Upslope Depression Index is a parameter that was developed by Hansen; Schjønning and Sibbesen [34] to describe soil roughness of tilled surfaces. The elevation line is described by a xed amount of line sub-segments m, each containing n height measurement points.
The calculation is based on the dierence between the height of a reference point Z_r and the height of points in an upslope line sub-segment Z_a.

$$MUD = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n} \sum_{j=1}^{n} \Delta Z \qquad (2.25)$$

where $\Delta Z = Z_r - Z_a$ for $Z_a < Z_r$ and $\Delta Z = 0$ for $Z_a \leq Z_r$.

The author proposed a length of 30 cm per line sub-segment. However, other lengths can be advantageous depending on the individual circumstances. [33]
This calculation principle is visualized in picture 2.7.

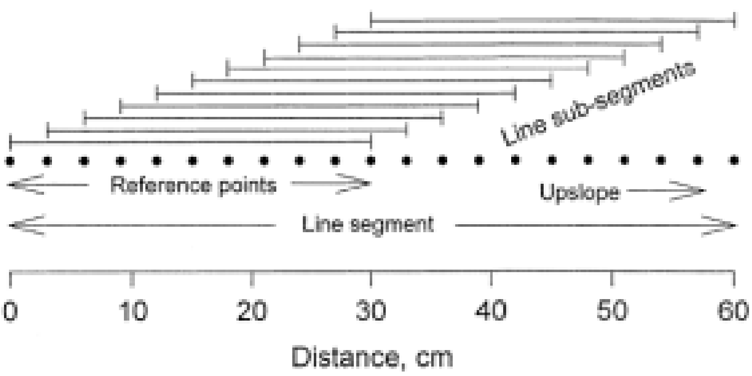


Figure 2.7: Principle of MUD calculation [34]

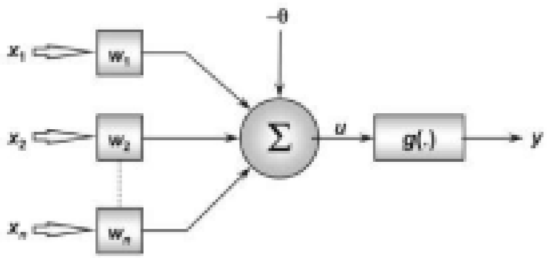
2.6 Robot Operating System (ROS)

ROS is a software framework developed for simplifying communication in robots. It allows seamless communication between sensors, programs, and other components of the robots. The data exchange is established via topics where dierent programs, known as ROS-nodes, can pass information.
The primarily used protocols are UDP and TCP, but access to other protocols (here ISOBUS and CAN-Bus) can be achieved by bridges that pass the information between the dierent information systems.
ROS1 is optimized for usage under Ubuntu, but there are also options available for other Linux distributions, Windows, and Android.

To record measurement data, ROS oers the option to save all incoming messages in a rosbag le. The storage includes the recording time, as well as the topic and content for each message. Rosbags allow easy playback for testing purposes and direct access to the messages using an included Advanced Programming Interface (API). [30]

2.7 Artificial Neural Networks (ANN)

The fundamental theory behind ANNs is based on biological nervous systems that can acquire and handle knowledge. The nervous system structure is thereby represented by articial neurons that are connected using weighted interconnections. The output is determined by a xed, nonlinear activation function $g()$, which calculates a neuron's specic output based on the sum of the weighted incoming values. The neurons, as shown in gure 2.8a, feature a individual activation threshold, also known as bias θ, that shifts the activation function to the left or right, so not only the slope of the activation function can be changed by the weights of the incoming connections, but also the positioning (connections x_i, weights w_i). [35][36][37]



(a) Articial neuron [35] (b) Activation Functions

The individual neurons can then be combined in arrangements diering in complexity and afterward trained on previously or interactively acquired data to predict values, groups, or actions in previously unknown situations.

The following sections include information related to ANNs, starting with the dierent available groups of algorithms and possible network architectures. Furthermore, learning strategies, as well as parameter optimization algorithms, are pre sented.
Finally, specic software related to the training of ANNs is described to close the funda mentals.

2.7.1 Online and Oline Learning

Oine Learning, also known as batch learning, describes algorithms that are only trained based on a xed and already known data set. If the data set changes or new data should

be added to it, the whole model needs to be discarded to be trained again. As a necessity, all data needs to be available during the entire learning process. [35][38] In contrast to this approach, Online Learning describes a method to split the training data into smaller subsets and feed them to the network one by one. Previously recorded data can also be used for this purpose. Therefore, new data can improve an existing model, and the model can discard storage-intensive data after calculation. [38]

2.7.2 Learning Strategies

ANNs can be sectioned into dierent groups depending on the specic use-case and training strategy. These groups feature unique characteristics and are therefore described in the following sections.

2.7.2.1 Supervised Learning

Supervised Learning algorithms contain features (input) and labels (output) and attempt to learn the relationship in between. The term "supervised" comes from the idea that a supervisor teaches the algorithm an appropriate output for given input signals. Training requires many examples of features and their respective labels.
These labels can either be the group where the training instance belongs to (Classication: Figure 2.9a), or a numeric output value of the training instance (Regression: Figure 2.9b). After the training, the obtained relationship can predict the label or value for a new feature set. [38][36]

(b) Regression

(a) Classication

Figure 2.9: Supervised Learning Examples

2.7.2.2 Unsupervised Learning

This group of algorithms is used to learn the properties of the structure of a dataset. Therefore the data itself contains no labels. Most probability distributions are extracted from the learning data. Some other algorithms belonging to this group also perform clustering to divide the dataset into smaller sets with similar features. An example of this approach is illustrated in gure 2.10. [38][36]

(b) Clustered Dataset

(a) Training Dataset

Figure 2.10: Unsupervised Learning: Clustering

2.7.2.3 Semisupervised Learning

This group of algorithms is used when only a part of the dataset contains labels, and the rest is unlabeled. These algorithms cluster the data (Unsupervised) and then use the labeled data to describe each data cluster individually. This method can extend training datasets and reduce the required amount of labeled data since manual labeling can be a time and cost-intensive task. Figure 2.11 illustrated this process. Grey points represent unlabeled data, the triangles the relatively few labeled datasets. [38]

Figure 2.11: Semisupervised Learning

2.7.2.4 Reinforcement Learning

Reinforcement Learning allows the algorithm to interact with its environment due to an implemented feedback loop. The algorithm contains an *agent* that observes the en vironment and receives rewards or penalties based on the performed actions as seen in
gure 2.12. Therefore the agent develops a *policy* that describes a strategy to obtain the most reward (R) or the least cost (C) over time. [38][36]

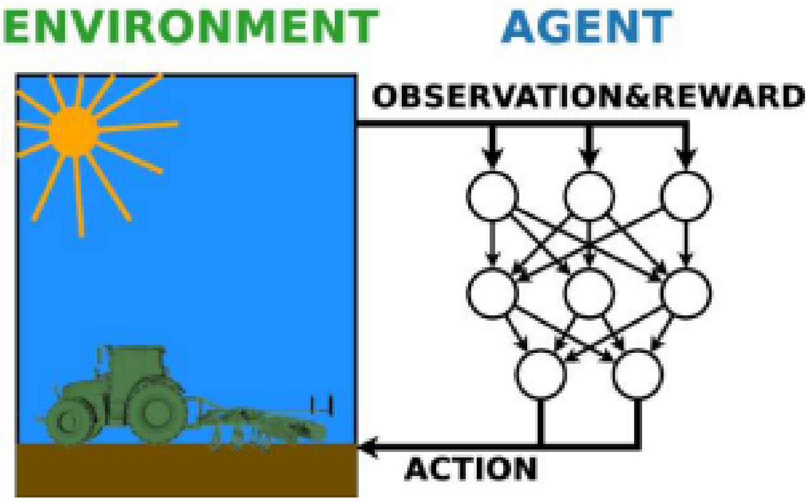


Figure 2.12: Reinforcement Learning, Tractor Model from [5] and [6]

2.7.3 Network Structure

The characteristics of an Articial Neural Networks are dened by the specic structure and organization of the individual network. Thereby, if the network does not contain any internal feedback loops and only oers a direct connection between input and output, it is called a *Feedforward Network*. Depending on the individual task the network has to perform, other important types of networks can be used, such as *Convolutional Neural Networks* mainly for image processing as well as *Recurrent Neural Networks* that can use their internal feedback loops to process sequential sensor data. [36]
The following sections describe dierent modules that can be used in the structure of a neural network.

2.7.3.1 Normalization

Incoming data can be normalized by the range of the individual inputs and outputs to minimize the inuence in weights caused by dierent number magnitudes. Normalization has to be done for training as well as test data. [35]

2.7.3.2 Hidden Layers

An ANN uses hidden layers to establish a connection between the input and the network's output data. Since the information in these layers in between is not human readable, they are called hidden layers. [36]

2.7.3.3 Regularization

A machine learning algorithm needs to perform on new test data as well as on initial training data. If an algorithm only performs well on its training data, it is overtting by not nding a general rule that will t all data. Regularization is the measurement to take against this generalization error, but not against the training error. [36]

Regularization can be achieved by adding weight decay functions that improve general ization (often L^2 or L^1 Regularization). The weights are driven closer to zero by an added penalizing term to limit the model parameters' size. Early stopping, another technique to regularize, is done by evaluating the current network on a subset of the training data to stop training when the internal validation does not improve anymore. Another option to achieve the same is the usage of dropout layers. Thereby part of the internal connections in the network is dropped by multiplying their output value with zero. [36]

2.7.4 Optimization and Training

Since the structure of a neural network oers many degrees of freedom, an important task is to optimize the network's internal parameters.
Therefore, a crucial task is the initialization of the network. If a previously working conguration is already known, the adapted initial values can enable faster optimization of the algorithm onto new data.
The real training uses an algorithm to minimize a loss function for the network to optimize internal weight and thresholds. For regression problems, the mean squared prediction error or mean absolute prediction error are often used as loss functions. [36][35]

The fundamental basis of training of most neural networks is stochastic gradient descent. Thereby, the weights and thresholds are minimized by calculating the gradient of the error $E(w)$ regarding the weight vector w, which combines bias and weights. w is adjusted step-wise to minimize the error depending on the learning rate η, which determines the speed of the training process. This iterative process is visualized in gure 2.13. [35]

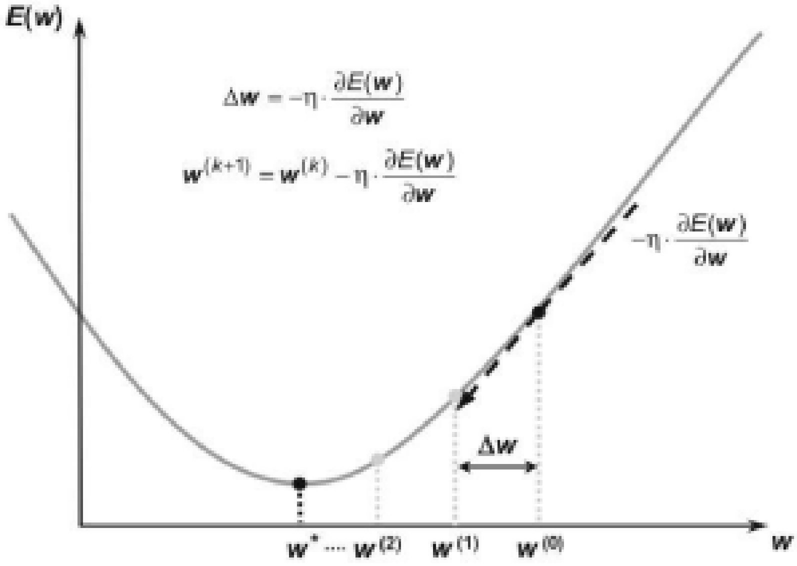


Figure 2.13: Gradient Descent [35]

Usually, multiple training iterations are performed over the entire dataset to increase the accuracy of the algorithm. Each iteration over the whole training data is called an epoch. [36]

For multi-layer networks, the complexity of the training process increases, and due to the non-linearity of the neuronal networks, a training algorithm called optimizer is used to minimize the total prediction error of the network. These algorithms often feature an adaptive learning rate to prevent the network from converging to local minima or overshooting the optimal conguration. A frequently used example of this is the opti mizer Adam, but current software libraries such as Keras also feature a wide choice of alternatives like Adadelta, Adagrad, Adamax, and SGD, that all have unique advantages and specications. [36][39][40]

2.7.5 Hyperparameter Optimization

The previously described possible contents of Articial Neural Networks grant many dif ferent possible structures. These parameter sets can be automatically tuned and evaluated to nd the optimal conguration. This process costs computational power but can also nd a not obvious solution that performs well.

2.7.5.1 Grid Search

When the number of hyperparameters is low, the search area can be explored systematically and uniformly. With an increasing number, this method gets increasingly computationally expensive because the whole search area is explored evenly, and if the optimal conguration lies in between grid points, it will not be detected. [36]

2.7.5.2 Random Search

Bergstra and Bengio [41] researched the eciency of grid search, compared with a purely random approach. They concluded that randomly chosen trails are more ecient due to nding at least equally performing models in a fraction of the necessary computational time. Random Search is thereby not bound to the programmer’s beliefs of grid arrangement but can eectively search through a large variety of parameter congurations. Figure 2.14 shows the dierences between the evaluated congurations of the two approaches.

2.7.5.3 Bayesian Optimization

Hyperparameter Search can also be conducted by modeling the validation error. This model can then be used for proposing good guesses for new congurations. Most widely Bayesian regression models are thereby used as described by Snoek; Larochelle and Adams. [42]

Since Bayesian Optimization can only be used for continuous hyperparameters and the results may vary from surpassing human experts to failing catastrophically, the usage is only recommended to a limited extend. [36][43]

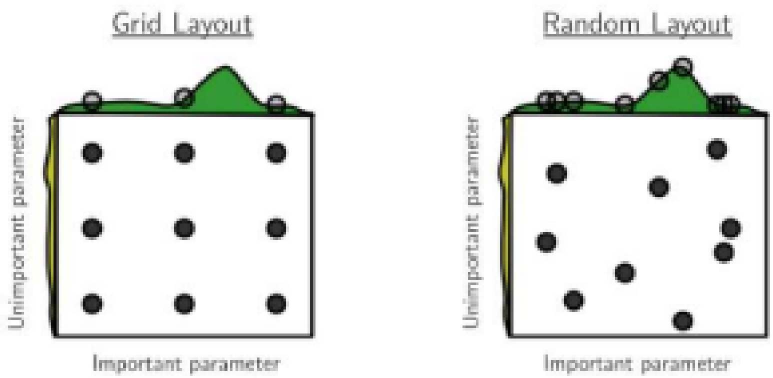


Figure 2.14: Grid- and Random Search [41]

2.7.5.4 Hyperband

A drawback of the previously described methods is that every network that shall be evaluated must be trained entirely for comparison. To x this drawback, Li et al. proposes a method used to speed up Random Search instead of proposing a model for the validation error.
The algorithm trains many random congurations for a xed number of epochs and com pares their validation loss over time. Then lowest half of the performers are discarded, and training is continued for the better performers that are then again continuously evaluated. Since not promising congurations are abandoned early,

the saved computational time and resources can be used to raise the number of total evaluated congurations, increasing the chances of optimizing the setup further. [44][43]

2.7.6 TensorFlow

TensorFlow is an open-source software package that allows easy implementation of a large amount of machine learning algorithms. Firstly published by Google, it allows the execution of machine learning software on various devices using a standard interface for coding. Also included are implementations of already known algorithms like Keras, which can be combined and used intuitively for the respective purpose. Application areas range from data analysis over speech recognition to computer vision. [40]

Keras is an API designed to provide an easy interface for deep learning applications. It is heavily interconnected with TensorFlow and provides a comprehensive library of machine learning components that can be easily interconnected to simplify the development of machine learning algorithms. It also includes the commonly used Hyperparameter Search algorithms. [45]

2.7.7 CUDA

CUDA is an API by Nvidia that enables the execution of programs on a CUDA-enabled Graphical Processing Unit (GPU). Since neural network training consists of a high quantity

of calculative operations that can be carried out simultaneously, this feature can speed up the training process by using the parallel computation power of GPUs for distributed processing. CUDA is supported by Tensorow. [46]

3 State of the Art

This chapter presents the current state of the art in machine control systems and tillage quality monitoring to dene the context and necessity of this thesis. The information describes currently available tillage-control systems and newer research activities to highlight the need for simplistic yet functional automatic control systems.

3.1 Automated Control Systems in Agriculture

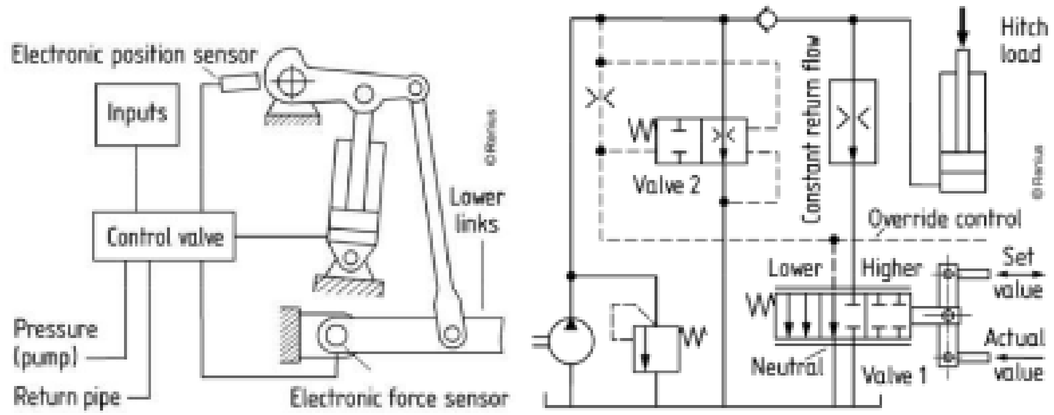
Various control systems are already on the market. This section describes systems that help maintain the operating speed of the tillage combination and hitch control systems, supporting the operator in controlling the position of the implement relative to the tractor.

3.1.1 Cruise Control

State-of-the-art cruise control systems allow the user to set static vehicle speeds while the tractor engine and transmission control optimizes RPM and gear ratio conguration. In modern power-split transmissions, these control systems such as the Tractor Manage ment System (TMS) from Fendt relieve the driver even from shifting gears during load changes.

3.1.2 Hitch Control Systems

Speed Control alone cannot adapt to soil composition changes, and the resulting changes in vertical implement pulling forces, resulting in a change in tire loads and traction. Hitch control systems were developed to adapt the implement position to increase traction and prevent sticking, relieving the driver from manual adaption. The rst proposed systems were merely hydromechanical (Figure 3.1a), later introduced systems mostly grant additional signal connectivity due to electronic components but also rising initial cost (Figure 3.1b). [7]
The complexity of such a modern control system is shown in gure 3.2. Modern electro-hydraulic hitch control systems allow the use of two control modes. The rst is Position Control, where the driver controls the hitch, and the second Draft Control, where the driver adjusts the implement position based on the draft of the implement to obtain a favorable weight distribution. These modes can also be mixed in percentage. Slimařík; Bauer and Dostál conducted a study on fuel consumption of dierent mode settings. They concluded that even the mixed draft control increases the system eciency compared to position control. Thereby also the drawback of strongly uctuating working



(a) Hydromechanical control of a Deutz rear hitch (1968) [7]

(b) Electro-hydraulic rear hitch control (EHR) [7]

Figure 3.1: Hitch Control Systems

depths of a pure draft control can be reduced. [47]

The development of radar-based measurement methods of eective vehicle speed in agriculture enabled the use of new possibilities and a new control mode for the hitch. Slippage is calculated from radar speed and theoretical speed. Implement position (p) is then adapted to limit the slippage to a maximum threshold. [47]

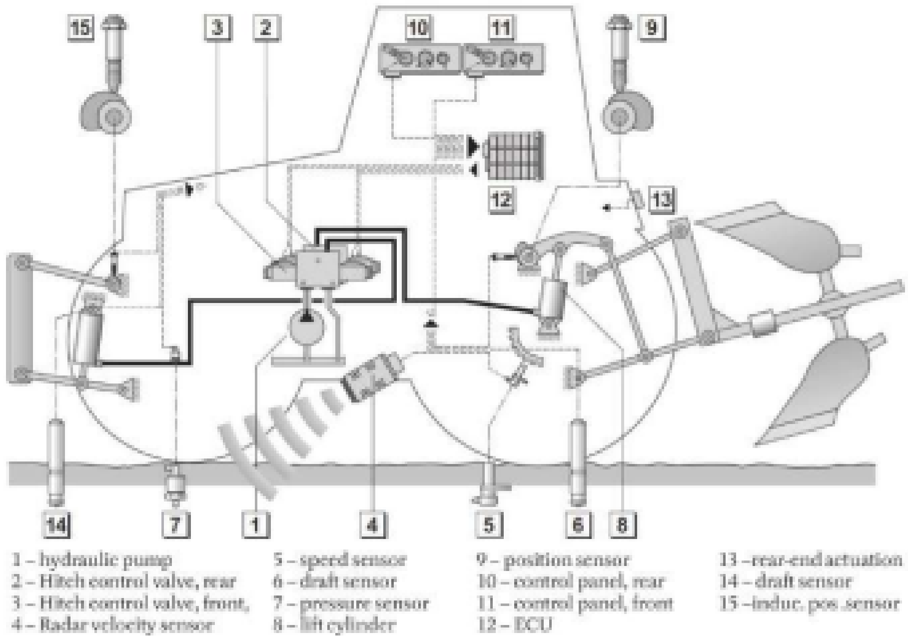


Figure 3.2: EHR system components [47]

3.2 Improving Fuel Consumption

The previously described control systems that are already widely available focus on maintaining the operating point that the operator has chosen.
During the last years, various research projects were conducted to enable further op timization by eliminating the dependency of manually dened machine settings and automatically choosing preferable operating points.
The presented ideas and projects are similar in that they either require extensive modeling or signicant computational complexity to optimize agricultural processes.

3.2.1 Holistic Eiciency Optimization

Kautzmann et al. [48][49] describe an approach to optimize fuel consumption based on the current state of an agricultural vehicle. The vehicle was thereby considered as a whole system to calculate the overall holistic eciency.
The system was based on a *System under Observation and Control* (SuOC) algorithm that consists of an *Observer* that evaluates the current state of the vehicle based on the combined eciency of a simulation model and a *Controller* that suggests a state transition based on an evolutionary algorithm.

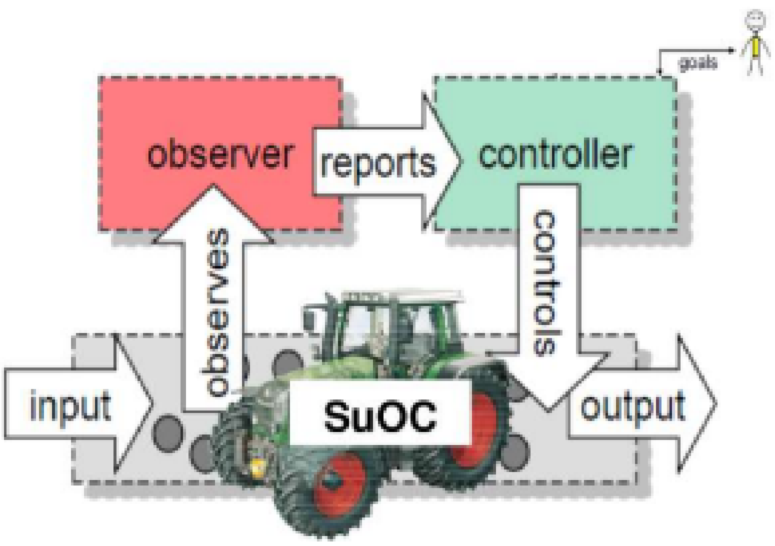


Figure 3.3: Generic Observer/ Controller architecture [48]

The architecture was able to increase holistic eciency by about 50 % in an AMESim-Model running the *PowerMix*-cycle Z5K - rotary harrow of the *German Agricultural Society (DLG)* as Input.

3.2.2 Reinforcement Learning

Becker et al. [50] used a Reinforcement Learning Approach to optimize dierent reward functions on the example of the plowing process. Based on the selection of a reward

function, the network is trained to detect the optimal state transition to optimize the reward (R). The input data was derived directly from CAN-Bus.

The action space consisted of deceleration ($-0.2\ km/h$), acceleration ($+0.4\ km/h$), and maintaining the tractor's velocity.
The reward functions 3.1 and 3.2 were chosen for state transitions, including parameters w_1 and w_2 as reward weights:

$$R_{ecient}(s_{t+1}, s_t, a_t) = -w_1 .B$$

$$v_{radar} \cdot w \quad (3.1)$$

$$R_{performant}(s_{t+1}, s_t, a_t) = w_2 \cdot v_{radar} \cdot w \quad (3.2)$$

The training process also involved an exploration phase directly on the eld and hard and soft constraints to achieve system safety.
As a result, fuel eciency and system performance can be improved with the trained agent illustrated in table 3.1. Since the agent had no continuous action space, the initial optimization period must be removed from evaluation. The increase of mean velocity by 5.17 % and decrease by 11.6 % in fuel per area, generated by the appropriate target function, show the potential of such a system.

Table 3.1: Reinforcement Learning Results edited after Becker et al. [50]

	Mean Velocity in km/h		Mean Fuel Consumption in l/ha	
	Complete Distance	After 30 meters	Complete Distance	After 30 meters
Reference	7.03	6.77	27.69	28.60
Ecient	6.11	6.29	24.68	25.29
Performant	6.72	7.12	24.34	27.10

3.2.3 Modeling Tractor-Implement Combinations

Schreiber [2] describes an approach to model tractor and implement combination in order to predict the fuel consumption. This model is primarily used to compare dierent agricul tural processes and process combinations regarding their energy requirements. A high drawback of this approach is the high required amount of knowledge about the specic setup.

Machl [51] visualized the relationship between relative pulling force and the speed of the vehicle as illustrated in gure 3.4. This study was conducted to examine the inuence of ballasting on traction power utilization but shows the vehicle speed's inuence on traction

power utilization. For lower speeds, the limiting factor is tire-soil mechanics, whereas the engine power limits the transmittable pulling power for higher speeds. Therefore higher speeds shift the optimal ratio to lower pulling forces.

Figure 3.4: Engine and tractive power relative to the respective maximum power, shown as a function of tractive eort and speed level, edited after [51]

3.2.4 Automatic Gear-Shiing based on Dra Force Characteristics

To optimize energy usage during tillage, Li et al. proposed a new theoretical method of automating gear-shifting based on real-time identication of draft force requirements. Thereby a mathematical model is presented to calculate torque requirements based on the draft force model in ASABE D497. On this foundation, an automated shifting schedule is presented to optimize fuel eciency. [13][21]

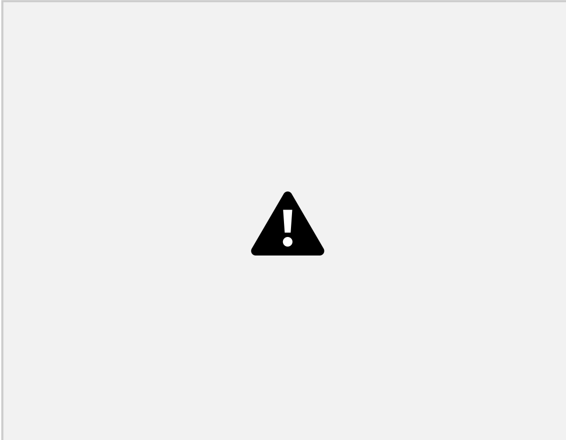
3.3 Soil Profile Estimation

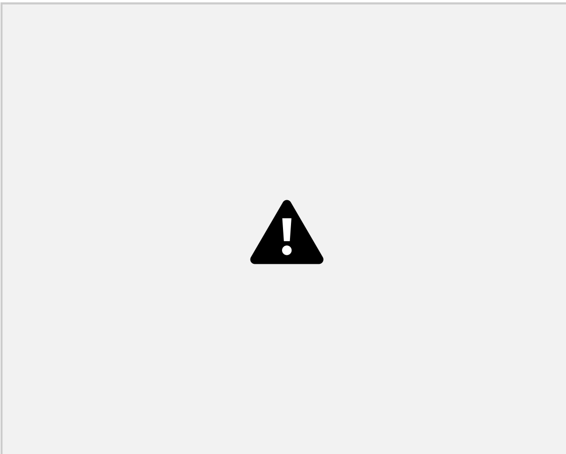
Soil prole estimation can be used to quantify the working quality during agricultural tillage. In this section, state-of-the-art estimation models are described, focusing on soil aggregate distribution and parameter calculation of a cross-section of the disturbed soil after tillage.

3.3.1 Soil Aggregate Observation

Steinhaus and Frerichs [52] describe an approach to detect soil aggregate distribution based on 3D point clouds that are detected using a stereo camera system. Their algorithm uses a Cloth-Simulation-Filtering process to remove small aggregates from the point clouds. Thereby a virtual cloth is placed on the captured cloud, and stiness and gravity adjusted so that only small aggregates remain in contact with the cloth and can thereby be detected and removed.

The remaining points are clustered by proximity using the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm. Afterward, individual aggregates are clustered by size. The results are visualized in gure 3.5.





(a) Original Point Cloud [52] (b) Detected Aggregates [52]

Figure 3.5: Aggregate Distribution

Their methods showed high accuracy during laboratory tests for aggregates with a diameter greater than 10 mm. They conclude that their method allows automated data acquisition of surface structure without the necessity of personal interaction.

3.3.2 Line-Based Surface Estimation

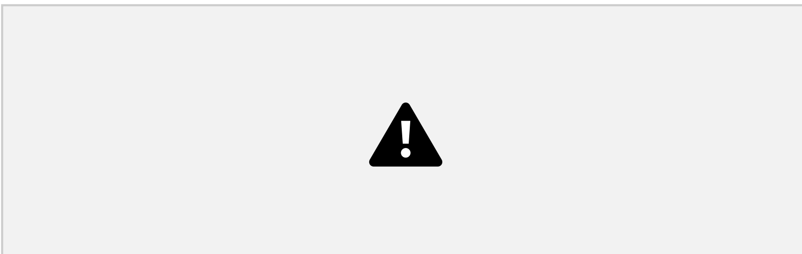


Figure 3.6: Soil prole as cross section [26]

Measuring a line prole of the surface cross-section as visualized in picture 3.6, parameters can be calculated to describe the elevation prole and enable classication of tillage processes.
Martinez-Agirre; Álvarez-Mozos and Giménez performed a comparison between a wide selection of evaluation parameters and ranked them by their individual ability to distinguish soil proles after tillage processes. They concluded with the

recommendation of the parameter Mean Upslope Depression Index (MUD), which takes horizontal as well as vertical deviations of the soil prole into account (Section 2.5.3), and Standard Deviation of Heights (s) (Section 2.5.2), for their simplicity and good performance. [33][34]

4 Cultivator Survey

To meet the practicability requirement and to evaluate the usage of current control systems in German agriculture, an anonymous opinion poll was conducted to obtain information about the cultivating process and specically about farmers' common problems and preferences during their work.
The questionnaire, which can also be found in the appendix, was sent to selected farmers using postal delivery. Of 22 sent questionnaires in total, 17 were returned, resulting in a return rate of 77.3 %.
Due to the small size of the poll, the result can only serve as an orientation of variables relevant to farmers. Furthermore, the included questions were not designed to meet psychological requirements for surveys and shall only serve as an approximate guideline. Farm sizes ranged from 22.87 ha up to 2100 ha with an average total operating area of approximately 264 ha and an average eld area of approximately 254 ha. Figure 4.1 shows the distribution of the total area of the individual farms.

Figure 4.1: Total operating Area and Field Area

Most of the participating farms have a strong focus on arable farming in terms of acreage. However, especially in relatively small farms, a high percentage of areas on which other plants are cultivated can also be observed. Due to the usually more labor-intensive cultivation of perennial plants like viticulture and cultivation in greenhouses, the respective farm focus can also be on other cultures despite their predominant land usage for arable farming.

The questions ranged from general information about the farms to precise information about the cultivating process, including machine settings, user control, and process targets. The whole questionnaire can be found in the appendix in section A.2.

Figure 4.2 visualizes the power of the most frequently used tractor of each farm during cultivating compared to their logarithmic visualized total operating area. An increase in machine power can be seen up to a level of approximately 270 kW (360 PS). This limitation can be traced to the fact that large farms use multiple machines rather than increasingly larger ones due to limited availability and drawbacks during road transition between elds.

Figure 4.2: Tractor Power compared to total Operating Area

4.1 Control Systems

Most of the questions were directly related to the cultivating process. In this regard, the use of electronic support systems for the process was evaluated. The farms were questioned on which support systems are used, and the results were ordered by the magnitude and visualized in gure 4.3.

Figure 4.3: Control System Usage

Electronic Hitch Control is in use on 76.47 % of the farms, whereas it has to be noted that the three farms with the largest cultivated area per year are not using these systems. One of those two farmers also noted that due to the silty and swampy soil caused by the farm’s location being in a terminal moraine, the driver’s skills are of irreplaceable importance. Driving conditions frequently vary due to up to four dierent soil types per

consolidated eld, forcing the driver to adapt the working process manually.

The usage of automatic GPS-based steering systems is at 58.82 %, whereby it has to be noted that such a system is in use at all surveyed farms with a cultivated area of more than 300 ha per year. Usage varies for farms with less cultivated area per year with no identiable trend.

Slippage Control is only in use on 41.18 % starting at farms with approximately 100 ha of an available eld area. The lack of usage in small farms can be explained due to the necessity of Radar- or GPS-based sensors to evaluate actual vehicle speed. These sensors are not always available on smaller machinery.
However, even for the larger farms, no denite trend can be identied.

4.2 Working Depth Configurations

To obtain information about the range of work done using cultivators, working depth congurations of the primarily used tractor-cultivator combination of each farm were analyzed. The smallest participating farm did not include information about their working depth setups and is therefore excluded from this specic analysis.
The target working depths were ranging from 2 to 30 cm, with each farm using an average of approximately 2.6 congurations.
The variety of congurations shows that some farms use one cultivator for a wide variety, if not all, their tillage processes, as stated by the fact one machine

combination covers a wide range of working depths and use-cases. Only two farms use their most-used combination solely for one specic working depth.
The survey was limited to one specic machine combination rather than all used combina tions and machines on the farm and their usage of external services to keep the survey duration low. The area covered by each machine combination was also recorded but can therefore not be compared with the total area of the respective farms, as some farmers cultivate elds of other farms as a service or utilize such a service themselves.

In summary, it must be stated that cultivator usage and settings depend on complex individual operational decisions, which result individually from the respective conditions and machine availability. Optimization must comply with every potential usage and setting of the machinery to improve the cultivating process to a great extend.

4.3 Observation Parameters

The farmers were asked to rate possible parameters on how much attention they pay on them during their work according to their priorities on a scale of 1 (No attention) to 10 (Much attention) to obtain a better understanding of the importance of specic possible observation parameters machine operators evaluate during cultivating. The results were then ordered by average rating and are displayed in table 4.1.

Table 4.1: Observation Parameters

Parameter	Average Rating
Clearing Straw and Catch Crop	9.26
Constant Working Depth	8.71
Surface Smoothness	8.53
Avoiding Clogging	8.50
Division of Large Chunks	7.71
Minimizing Fuel per Area	7.41
Minimizing Time	7.29
Slippage	6.67

The table shows that some factors, like clearing straw and catch crop and a constant work ing depth are equally signicant for all the participating farmers. One of the participants also mentioned that a constant working depth is decisive for a constant crop growing of the subsequent planting.

A smooth surface is, for most farmers, also of high priority. The only rating of 3 points comes from a farm located in a region with soils containing a high amount of clay. Due to the resulting diculties in tillage, the farm has a strong focus on a constant working depth, avoiding clogging of the implement, and clearing of the previous planting.

Most farms see high importance in the avoidance of implement clogging. The only two exceptions are from a farm whose elds have sandy to slightly loamy soils and a farm that stated that the clogging of the utilized cultivator is extremely rare due to the high spacing between the tines compared to other cultivators on the market.

The four remaining objectives were rated very broadly and dierently, so no clear pattern could be recognized. This result suggests that monetary inuences, such as lower fuel consumption or shorter working times, have only a relatively low relevance in tillage, and the quality of the process is of the highest importance.

In conclusion, it can be said that most of the suggested targets feature a high average relevance on the surveyed farms. Interestingly three of the four most relevant targets, clearing straw and catch crop, surface smoothness, and clogging avoidance, can only be perceived visually and show the open use cases for camera and laser-based detection and scanning systems. The most relevant targets are also heavily correlated with the quality of the cultivating process, but targets must be set individually depending on the respective circumstances.

4.3.1 Working Depth Evaluation

The working depth is one of the only parameters challenging to observe from the driver’s seat. Therefore the participants were asked to mark the methods they use out of a list of possibilities. The results were then ordered by usage and displayed in gure 4.4.

Figure 4.4: Working Depth Control

It has to be noted that several farmers added that they measure and recognize the working depth by the position of the soil on the cultivator blades. This option was not included in the original survey.
Most farmers tend to dig the soil until undisturbed layers get visible. This method is the safest way to ensure a constant working depth, but the driver needs to stop the machine and check it manually.
The second most used method is a visual control. It was not specied whether this happens from the driving position or by stopping the vehicle and checking it from a side angle. Both options may cause high inaccuracy due to the roughness of the soil close to the tines, which are the only close reference point on the machine to evaluate working depth.

4.3.2 Preferred Optimization Targets

In addition to the questions concerning the current operating conditions and preferences, the farmers were asked to rate dierent optimization parameters under the assumption that they would have a fully autonomous cultivating machine at their disposal that does not require any human operator input.

Table 4.2: Autonomous Cultivator Targets

Target	Average Rating
Constant Working Depth	9.06
Smooth Surface Prole	9.0
Minimize Fuel per Area	8.69
Minimize Worktime	8.19

The results of this question do further underline that the farmers prefer the targets that aect tillage quality. However, the high average rating of all of the specied targets signies that the combination of the parameters needs to fulll the individual requirements, rather than one target that can be optimized solely.

4.4 Survey Conclusion

Concluding the survey, the farmers' opinions and preferences show the need for time saving and accurate working depth measurement, as all methods currently used are either subject to inaccuracies or time-consuming hassle for the operator. Furthermore, the usage of implement position control systems varies due to varying soil conditions that depend on the total operational area of the farm. This distribution means that new control systems have to feature a high level of adaptability to obtain a high usage level. This requirement is underlined because many farmers use the same tractor-cultivator combination for various tasks and working depths on which such a system would have to function equally.
Concerning target functions during cultivation, it is apparent from the survey that the quality of work is signicantly more critical than the optimization of economic values such as a low total time or low fuel consumption per area. Before optimizing the latter operational target gures, consistent working quality must be ensured.
The fact that GPS-controlled steering is already widely used, especially on larger farms, shows farmers' acceptance of relatively new systems if the operational improvements are apparent.

5 Setup and Data Collection

This chapter describes the collection of training and evaluation data for neural networks. It was possible to use the data collected for another project. In this data, mostly cruise control was used as a driving strategy, and the drivers merely adjusted the depth of the cultivator in dense soil conditions to keep a steadier driving speed. A few additional driving strategies were used to obtain more insights into the tractor and implement. This additional data was not used for the training of neural networks.

5.1 System Setup

A *Fendt 516 Vario* in combination with a *Horsch Terrano 4 FX* cultivator and added front weights was used for data collection and for evaluation.

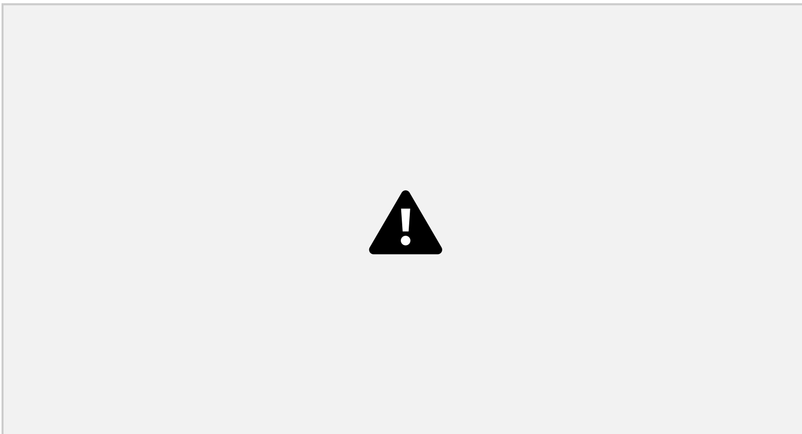


Figure 5.1: Fendt 516 Vario with Horsch Terrano 4 FX

The power-split transmission of the tractor allows the gear ratio to be changed without interruption. The tractor can continuously optimize engine and transmission settings based on a given setpoint speed (v_{set}) using the internal transmission control TMS. The tractor features a nominal power of 120 kW at 2100 1/min and a maximum power of 126 kW. The engine characteristics from the manufacturer, as well as the fuel consumption curves experimentally determined by the DLG are shown in the gures 5.2. [53][54]
Communication was established between the CAN-Bus, ISOBUS, and the LAN network using a computer running Ubuntu 20.04 with ROS noetic as a bridge. In addition to the internal parameters, additional sensors were used to collect data and control the internal signals. All sensor data was collected via ROS and saved to rosbags.

(b) Fuel Consumption, edited after [54]

(a) Engine Characteristics, edited after [53]

Figure 5.2: Fendt 516 Characteristics

5.1.1 Pitch Angle Control

To check the internal Pitch Angle (θ) sensor an *XSENS MTI 300* Inertial Measurement Unit (IMU) was mounted on top of the vehicle. The orientation and acceleration data were then published to the network using ROS.
It should be noted that high-frequency uctuations, possibly induced by the rough ground surface, are automatically corrected by the positioning of the IMU on the damped tractor cabin.

5.1.2 Visual Sensor Systems

To obtain visual information about the process, a *Realsense L515* Lidar Sensor was mounted to the cultivator frame as well as a *Realsense D415e* Stereo Camera to the top of the cabin facing backwards. Mounting position, as well as eld of view of the sensors, are visualized in gures 5.3 and 5.4.

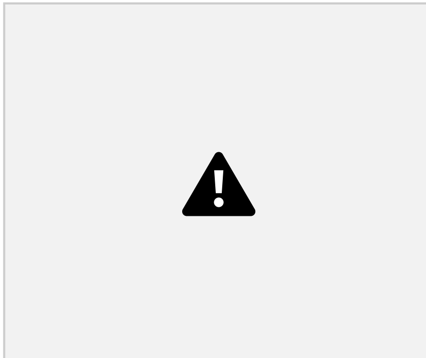


Figure 5.3: Mounting position of D415e and L515, Models from [5] and [6]

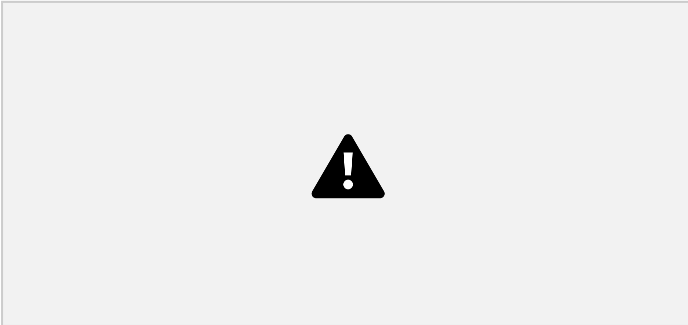


Figure 5.4: View of D415e and L515, Models from [5] and [6]

Both sensors were available during primary data collection, but only processed data was recorded and not raw data due to limitations in the recording storage capacity and the large amount of space required for recording point clouds. Therefore, the generated point clouds were not recorded to the training data, and both sensors can only generate online data during work.
The Lidar shall be primarily used for working depth calculation and the Stereo Camera for surface parameter evaluation as described in the following sections.

5.1.2.1 Working Depth Calculation

A *Raspberry Pi 4* was used to run the Lidar drivers and establish the network connection to publish the sensor data in ROS-messages.
To decrease network load, the *Pi 4* directly calculated the distance to the planar surface using a RANSAC-Algorithm and published it to the ROS-network for data collection. Figure 5.5 visualized that this algorithm is resistant to noise and can calculate planar coordinates even on stubble elds. Here, the red points are categorized as plane points and green points as outliers.

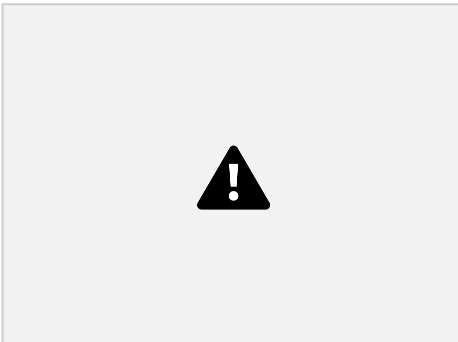


Figure 5.5: Plane Detection

The oset was then calculated by measurement on asphalt to calculate the working

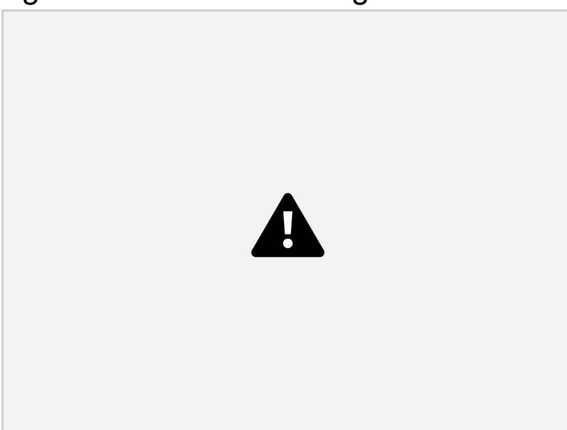
depth: t = oset − distance to plane (5.1)

5.1.2.2 Surface Parameter Calculation

For surface parameter calculation, the mounted *Realsense D415e* was directly connected to the system's central computer to obtain information about the surface structure.

The approach of obtaining a surface line of the processed soil is based on *Automatisierte Erkennung von Prozessparametern in der Landtechnik mittels Stereokamera am Beispiel Pügen* [55].
Using RANSAC, the surface plane parameters are detected from the point cloud, and measured points on the implement were removed from the observation area. A coordinate frame located in the centerline of the implement and on the soil plane was generated. A Kalman lter was applied to the coordinate transformation to correct measurement irregularities.
The point cloud was then transformed into the new coordinate system for simplied calculations to allows easy processing because the z-coordinate of the points is now orthogonal to the surface.
Point height coordinates were merged in the driving direction to observe a height line of the cultivated surface similar to one generated by a laser scanner directly mounted above. The mounting position of the camera on the tractor roof ensures easy cable management and better operating conditions due to fewer dust particles surrounding it.

The obtained height line now allows the calculation of slope-line dening parameters like the Mean Upslope Depression Index (MUD) and Standard Deviation of Heights (s), whereby the former has not been used because the parameters depend on the slope of the measurement line.
Figure 5.6 shows the usage of this camera-based surface evaluation process.




(a) Normal Working Condition (s ≈ 0.12 m)
(b) Implement lifted too high (s ≈ 0.2 m)

Figure 5.6: Working Depth Variation

If the implement is lifted too high, the leveling disks and roller following the blades cannot distribute and compact the soil after processing. Therefore, the single lines of the roller disks can be detected to determine the operation state. If these are not visible anymore

and the amount of visible lines decreases to the number of tines on the last bar, then the implement is lifted too high to ensure sucient machining quality. The directly visually observable change in surface texture also yields a change in s.

5.2 Data Collection

The data was collected in Zaisenhausen, Baden Württemberg, using a human driver with automated steering and TMS activated. Draft control and slippage control were turned o. The training data was collected during summer with ambient temperatures ranging from 14.8 to 23.3 °C, and soil composition was varying between loess soil, sandy loam, and loam.

The frequency of each signal was analyzed based on the number of recorded messages. The detected frequencies allow individual signal ltering based on timeslots. The recorded parameters and frequencies are listed in table 5.1. The numerical value for the engine torque was not available on the tractor connection during data collection. Therefore, the read-out percentage represents the current torque compared to the maximum possible torque at the associated engine revolutions.

Table 5.1: Acquired Parameters

Parameter Source Frequency in Hz Comment

Traction Force (F_T) CAN-Bus 4

Theoretical Speed (v_{theo}) CAN-Bus 4

GNSS Speed (v_{gnss}) CAN-Bus 8

Workmode (M_w) CAN-Bus 4 no lter required Implement Position (p)
CAN-Bus 4

Pitch Angle (θ) ISOBUS 12-18 not reliable Pitch Angle (θ) IMU 100 not
transferable Fuel Rate (B) CAN-Bus 4

Ambient Temperature (T) CAN-Bus 4 informational Relative Engine

Torque ($\tau_{\%}$) CAN-Bus 4

Engine Revolutions (n) CAN-Bus 4

Working Depth (t) L515 8 not in collected data Surface Parameters D415e
not in collected data

5.2.1 Slope Calculation

Since there is no slope (δ) signal available we need to calculate it using the pitch angle (θ). A simple but imprecise approach to this is equation 5.2. The oset was calculated by placing the tractor facing upwards and downwards in the same position while measuring the internal signal.

$$\delta = \theta - 3^{\circ} \quad (5.2)$$

The pitch angle signal read from the CAN-Bus depends on the pulling angle (α) of the implement, causing the rear tires to contract due to high vertical forces. The pulling angle is not directly measurable with the current setup, but due to the recently installed *L515* the accuracy of the slope angle can be improved in further validation and data collection. The *L515* is mounted on the implement frame, so it can detect the oset angle δ_{soil}. Therefore, the varying inuence of the implement position (Implement Position (p)) has to be considered and adapted using a correction term.

$$\delta = \theta + \delta_{soil} + p \cdot correction \quad (5.3)$$

The kinematic of the angle between the tractor and implement caused by variation of p can be described as a direct relationship to the actual mounting angle since the measurable percentage p is directly interconnected with the positional angle of the implement relative to the tractor due to the xed mounting to the lower and upper lift arms. During active driving, the changes of δ_{soil} can therefore be used to calculate δ more precisely in further work.

5.2.2 Extract Rosbag Data

During data collection, every signal of the system was saved as a rosbag le. In earlier works, the collected rosbags were only used via playback, therefore needing much time to train models. On the basis of example code by Speal, a script was written to extract the necessary information of the rosbags using the rosbag Python-API in order to speed up the training process. [56]

The script allows choosing topics and searches through the rosbag le to nd all published messages on the specied topics. These messages are then saved in a separate *.csv* le for each topic ordered by their timestamp.

Afterward, messages are smoothed by calculating the arithmetic mean of this sensor's data obtained in the last recorded second. An exemplary result can be seen in gure 5.7.

Furthermore, these messages in the separate les of the dierent sensors and signals are then merged by the closest recording timestamp to create complete datasets of all sensors

0

θ

0 0 0 0 0 00

Figure 5.7: Filtered Pitch Angle Signal, Measurement taken at Zaisenhausen, lange Seeacker 21/10/20

combined to use them as the system state at each point in time. If the value of a topic is missing, the whole row is dropped.

The merged le is then also saved and a le that includes the rosbag timestamp and only information from *data* or *value*-elds of the messages. This le can then be used for neural network training.

The whole collection and extraction process is visualized in gure 5.8.

Figure 5.8: Collection and Extraction

5.3 Data Analysis

The total collected and extracted datasets are described in table 5.2. The pitch angle signal published via ISOBUS is missing in some time slots of the data collection process. The missing data can be retrieved with the equations described in section 5.3.1.

It can be seen that the number of datasets per second is related to the lowest transmission frequency. The lower total dataset frequency at the sets containing θ-data from the ISOBUS can be explained by the fact that the internal signal is unreliable and often temporarily unavailable.

Table 5.2: Data Overview

Type	Time	Datasets	per Second
With Pitch Angle	60.23 min	14273	3.95
Without Pitch Angle	242.98 min	58220	3.99
Total	303.22 min	72493	

The collected data was then ltered for data analysis and network training, as described in section 5.3.2.

5.3.1 Pitch Angle

Fixing the issue of missing pitch angle (θ) from the ISOBUS can be done by using the IMU-orientation signal, which, when negated, correlates with a static oset. The IMU is only installed on the current tractor, and to ensure interchangeability between tractors, we cannot rely on installing and calibrating a sensor on each tractor. Therefore we calculate the pitch angle only for missing data points and use the internal ISOBUS signal otherwise.

The IMU-orientation signal is published as quaternion q, to obtain θ in Euler angles we need to use the conversion 5.4 as stated by Blanco. [57]

$$sin(\theta_{IMU,raw}) = 2 \cdot (q_r \cdot q_y - q_x \cdot q_z) \quad (5.4)$$

The relationship between θ_{IMU} and $\theta_{IMU,raw}$ can be calculated using a static oset:

$$\theta_{IMU} = oset - \theta_{IMU,raw} \quad (5.5)$$

The static oset was calculated and averaged for all data points where both signals were available to receive θ. The *oset* is highly inuenced by outliers, but *oset* ≈ 2,6° can be chosen as a reference.

The IMU and ISOBUS-data is visualized in gure 5.9. The signicantly smoother course of the orientation data of the IMU is due to the positioning on the suspended driver cabin.

θ

0 θ θ
θ

0 0 0 0 0

Figure 5.9: Pitch Angle Calculation from IMU Data

5.3.2 Data Filtering

For further analysis and neural network training, outliers of the data were removed by the rules described in table 5.3.

Table 5.3: Data Selection

Rule	Reason
$v_{gnss} > 1\ km/h$	Move Forward
$M_w = 1$	Workmode On
$120\ kN > F_T > 0\ N$	Positive Draft Force
$0 < \sigma < 1$	Slippage Limits

The limiting rule for traction force (F_T) limits the maximum value to 120 kN since higher values are unrealistic due to a necessary traction coecient (κ) of more than 1 ($F_W \approx 12\ t \cdot 9.81 m/s^2$). The number of data points was reduced from 72493 to 53773 during ltering.

These lters also have to be applied to the live system's incoming data stream to validate the current state for prediction models.

5.3.3 Shell Schemes

This section showcases the training data set to display optimization potential.

Figure 5.10a shows the distribution of data relative to engine parameters. Each data point was assigned to a bin where the respective number of containing data sets is displayed on a logarithmic scale. The gure showcases that the TMS only chooses specic engine states during heavy tillage operations due to the high power demand. Furthermore, the exceptionally high count of values at the maximum relative engine torque shows the tractor's under-sizing concerning the implement's pulling force requirements.

In areas where the amount of data points is small, the natural compensation of outliers by averaging cannot take place. Therefore, outliers in these areas are expected to be not representative.

(a) Data Count

(b) Fuel Rate Characteristics

Figure 5.10: General Data and Engine Information

Figure 5.10b shows the fuel consumption of the engine dependent on engine torque and revolutions. The color-coding shows the arithmetic mean of each bin.

As expected from the literature, fuel consumption increases with rising torque and revolutions. Interestingly most of the collected datasets come from operating points close to the maximum fuel consumption.

5.3.3.1 Reward Functions

In chapter 3.2.2 dierent reward functions were described, that were previously also used for a Reinforcement Learning approach by Becker et al. [50]

The therein contained dependency to eective speed (v_{gnss}) is problematic for visualization since the levels are not comparable due to changing implement position and non-static soil density. To establish comparability in visualization, the traction power (P_T) was used instead of v_{gnss}:

$$P_T = F_T \cdot v_{gnss} \quad (5.6)$$

This adaptation establishes comparability since traction limits apply. Consecutively the reward functions were changed as well using equations 5.7 and 5.8.

$$R_{perf,visual} = P_T \quad (5.7)$$

$$R_{e,visual} = \frac{P_T}{B} \quad (5.8)$$

The comparability of the functions is displayed in gure 5.11. The distribution of $R_{perf,visual}$ is similar to B. The high reward outliers have to be disregarded because they belong to sparsely populated data bins.

Furthermore, comparing the two available reward functions, the shift from performance to eciency shows a lowering engine torque, as well as a shift to lower engine revolutions due to varying gear ratio by the TMS to improve eciency instead of maximizing traction power transmission.

Due to the correlation in gure 5.11b it is possible to assume that the TMS will automatically optimize engine and transmission settings if the system is not forced to make full use of the available engine power by a too high-speed setting. In this case, it is only necessary to provide a target speed, and there is no need to adapt the transmission setting manually.

The inverse of $R_{e,visual}$ represents the specic fuel consumption of the tractor.

(a) Performance

(b) Eciency

Figure 5.11: Reward Comparison

5.3.4 Currently Available Control-Systems

To evaluate the progression of the working depth on the control systems currently avail able on the market, they were activated in parallel rows on the same eld.

Two rows per control system were measured, one upslope and one downslope, to establish comparability.

The altitude of the eld is displayed in gure 5.12.

(a) Track 1

(b) Track 2

Figure 5.12: Working Depth on changing Soil Conditions

5.3.4.1 Constant Implement Position

As a reference, the two tracks were driven using a constant setting with target speed set to 5 km/h and implement position to 50 %. Figure 5.13 visualizes that even though the implement position remained constant in reference to the tractor, working depth and draft force changed signicantly. Because this eect occurs independent of the slope, the assumption can be raised that the reason for this change is varying soil conditions.

(a) Track 1

(b) Track 2

Figure 5.13: Working Depth on changing Soil Conditions

Heavy soils lead to an increase in horizontal draft force requirements and vertical force requirements. The load of the implement and vertical forces are split between the rigidly connected roller and the tractor lift arms. The graphs suggest that this increase yields a compression of the rear tires and underlying soil and an increase in working depth. This eect further leads to increased draft force requirements until the increase in tire pressure

and denser packed soil underneath counteract the increased forces and result in a new equilibrium.

5.3.4.2 Dra Force Control

The draft force control mode usually uses force sensors in the rear lift arms to maintain constant draft force requirements by adjusting the implement position. Theoretically, in this control mode, according to equation 2.17, heavier soils in combination with a constant draft force and speed should result in a decrease in working depth since implement parameters remain constant. This eect should counteract the increase in working depth caused by tire and soil compression.

(b) Track 2

(a) Track 1

Figure 5.14: Draft Control

Figure 5.14 visualizes the eects of this control mode onto the working depth of the cultivator. In total, the working depth is less aected by changing soil conditions in this control mode. Draft force is also more consistent, as expected, due to the usage as the control parameter. Thereby it has to be noted that the results are expected to dier with non-constant speed settings.

5.3.4.3 Slippage Control

The Slippage Control lifts the implement never to exceed a specied maximum slippage. It is expected, with optimal traction conditions and limited draft force requirement, the mode would behave similarly to a constant implement position. However, with rising power demand and the necessary increase in slippage close to the specied maximum, a behavior similar to the draft control mode is expected from traction relationships (Figure 2.2b). To visualize this, gure 5.15 shows one track with a theoretical speed of approximately 5 km/h, where the traction of the tractor could easily manage draft force requirements without a signicant increase in slippage. A parallel track was measured as a comparison with increased speed settings of 7 km/h, which could also be achieved all the time. Slippage values increased as well due to the expected traction behavior.

(a) Track 1

(b) Track 2

Figure 5.15: Slippage Control

The lower speed curve has a high similarity to gure 5.13. Due to the increased draft force requirements with higher speed settings, slippage increases, and the slippage control adjusts the implement position more frequently to avoid the maximum specied slippage of 10%. The curve gets thereby increasingly similar to the draft control mode in gure 5.14.

In conclusion, the control mode analysis suggests that the working depth must be controlled automatically to maintain a constant working depth and ensure optimal plant growth since the former is not achieved by any of the currently available control systems.

6 System Modeling and Control Strategy

This chapter describes the developed system to automate working depth and speed control based on the specied requirements from the previous chapters.

6.1 Requirements

This section gives a short overview of the specic requirements of the control system. The system should

- be robust and able to adapt to a wide range of soil conditions
- always ensure the optimal operating point of the machine
- minimize the demand of a trained human operator to specify machine setup and operating conditions
- adapt to changing soil conditions fast enough so that even within rapidly changing soil conditions, uninterrupted optimization can be provided
- be adaptable to individual system targets dened by the operators' needs and re quirements
- prioritize good cultivation to optimization of the operating speed and fuel

consump tion in order to t the farmer’s priorities

- only use previously collected data for the neural network training process, since the model must be robust enough to function without having to retrain or explore in previously unknown situations

In contrast to the suggested approach by Li et al., transmission and engine control shall be controlled by the TMS of the tractor. Therefore, the control system shall not interfere with the automatic control of the power-split transmission and shall only specify target speeds. [13]

6.2 Working Depth Control

To ensure consistent working tillage quality and counteract the variance caused by chang ing soil texture, the working depth was held constant by a ROS-node that adjusts the implement position. The working depth measurements by the *L515* were used, and if a

threshold of one centimeter below or above the requested working depth is breached, the position of the implement is adjusted step-wise subsequently.
Due to the rough surface texture and the resulting variance in working depth measure ments, a lower threshold is unfeasible. The implemented control system executes the demand for a constant working depth after the cultivator is once initialized correctly. Figure 6.1 shows the necessity of this system due to the changes in implement position necessary to maintain a constant working depth.

Figure 6.1: Working Depth and Implement Position

Furthermore, the now presumably constant working depth allows a simplied online parameterization of draft force relationships.
However, using a more advanced control system might improve results and decrease variance in working depth further.

6.3 Speed Optimization Strategy

This section describes and compares dierent strategies for improving system performance or eciency.
One option for such a strategy is the usage of a Reinforcement Learning algorithm. The involved agent can evaluate measurement data and propose a policy that suggests state transitions based on a reward function. Therefore, the reward function must be specied in advance of the training process to enable the algorithm to learn from mistakes and achievements. Thereby the background behind the agent's decisions is not directly ac cessible or understandable, and the system would therefore function as a black-box. To obtain an acceptable policy, the agent must explore its environment after a possible static learning process on previously collected data.

The other strategy is to individually model or predict system states for various theoretically possible target speed settings. The modeled states can be compared based on the reward functions from the reinforcement learning approach to choose the best state transition. Furthermore, these functions can be edited since they are independent of the modeling of the state variables. Theoretically, traction and draft relationships and the characteristics of the power train and engine of the vehicle can be modeled. However, apart from traction and draft relationships, these settings are very dependent on the individual machine and drastically increase the modeling process's complexity. The usage of an ANN to predict the fuel rate based on draft and traction modeling can decrease this eort by solely requiring enough measurement data to train the algorithm to predict the fuel rate.

6.4 State Prediction Model

The second approach mentioned in 6.3 of modeling the future state of the vehicle’s forces and speeds based on the current measurement data combined with an ANN for fuel rate prediction was chosen. The ow chart 6.2 describes one iteration of this algorithm. In the next section, the required components for this optimization process are described, followed by a detailed schedule of one iteration in section 6.5. These iterations need to occur repeatedly to allow the system to adapt to changing soil conditions or slope angles.

Figure 6.2: Optimization

Figure 6.3 illustrates the ow of the information for the modeling and prediction process in more detail. The extracted measurements from the CAN-Bus and ISOBUS are used for calculating static variables like slope and weight force. Furthermore, the information is used to parameterize current traction and draft relationships to enable system modeling.

The combination of these equations allows modeling the vehicle’s tractive behavior to elaborate on possible system states. Therefore, multiple suggestions for theoretical speeds can be entered in order to generate a complete parameter set.

Figure 6.3: Information Flow

The state variables of dierent evaluated speed settings are then entered in an ANN to approximate the associated fuel rate. The resulting state variables can then be used to select the best setting based on the reward of each state.

The model assumes that the ground's soil conditions and slope do not change between the current and predicted state during one iteration of the algorithm. Therefore, the vehicle's weight forces and weight distributions are assumed not to change during the proposed state transitions in one iteration. If an occasional change occurs, the algorithm will therefore have to adapt subsequently.
The slippage cannot be evaluated independently for the dierent wheels as we do not measure the weight distribution changes between the front and rear axle. All tires are therefore assumed to have the same speeds.
Due to the proposed working depth control in section 6.2, the working depth is assumed to be constant.
This simulation part can be divided into the calculations dependent on the tractor's tractive behavior and the implements draft forces. The combined calculation can generate the parameter set described in table 6.1 which can be used to compare dierent modeled states.

Table 6.1: Parameter Calculation

Parameter	Source
Horizontal Weight Force ($F_{W,h}$)	Stationary
δ	Stationary
Modeled Theoretical Speed ($\tilde{v}_{theo}$)	Proposed
Modeled GNSS Speed ($\tilde{v}_{gnss}$)	Model
Modeled Traction Force ($\tilde{F}_T$)	Model
Modeled Draft Force ($\tilde{F}_D$)	Model
Predicted Fuel Rate ($\tilde{B}$)	Neural Network

6.4.1 Modeling System Parameters

The vehicle's forces are modeled using a combination of a traction model used to obtain power and speed relationships from the wheel to the surface and a draft force model to dene the inuence of changing speeds on draft force requirements.

6.4.1.1 Online-Parametrization of Traction Relationships

Both in the fundamentals described relationships, equations 2.15 and 2.16 have a high similarity in the relevant range of $0 < \sigma < 0.2$ proposed by traction eciency, so both can be used for modeling traction with the assumed assessment of 2.15 being more accurate due to optimization onto agricultural vehicles.

For online parameterization, however, these equations must be simplied to the extent that a single measured dataset is sucient to determine the respective curve. This approach has been chosen to limit the extent of the time slot needed to dene exterior conditions so that the system can quickly adapt to changes.

This time limitation yields diculties together with the requirement for the system's simple usage without the need of the farmer to specify soil conditions for model parameter calculation due to twice-restricted input parameter count. Thereby equation 2.16 allows the online parameterization with the fewest additional simplications due to the small number of free constants.

These still required simplications are based on correlations of F_T and κ with σ identied in previous research work (e.g. Figures 2.2b and 2.5).

The rst simplication is based on the assumption that no Traction Force can be supplied while σ is zero, based on "ASAE S296.4 General Terminology for Traction of Agricultural Tractors, Self-Propelled Implements, and Traction and Transport Devices". [58] This denition can be adopted based on dierent zero-slip conditions. Schreiber and Kutzbach provide a comparison between the commonly used denitions. [59]

The second simplication is thereby based on the assumption that the maximum F_T can be provided while σ = 0.6.

For equation 2.15 this leads to the following simplications:

$$\kappa(\sigma = 0) \approx 0 \quad (6.1)$$
$$\kappa(\sigma = 0.6) \approx 0 \quad (6.2)$$

Using the previously specied simplication of constant weight forces and distributions, these equations can be transferred directly from κ to F_T to be used in equation 2.16:

$$F_T(\sigma = 0) \approx 0 \quad (6.3)$$
$$F_T(\sigma = 0.6) \approx 0 \quad (6.4)$$

Resulting also in an equation that is solvable using only one measured dataset:

$$F_T(\sigma) = 1.2 \cdot c_T \cdot \sigma - c_T \cdot \sigma^2 \quad (6.5)$$

Figure 6.4 illustrates the progression of the traction force as a function of the parameter c_T that is parameterized online.

Figure 6.4: Traction Force Curves

The generalizations needed to solve previously described equations show further opti mization potential to improve the model’s accuracy. These values can be optimized using pulling experiments described by Holm [9].

6.4.1.2 Online-Parametrization of Dra Relationships

For modeling the draft force (F_D) of the implement the approach was chosen to calculate the current parameter set incrementally as described by Rößler; Kautzmann and Geimer. [25]

The implement model consisted of using equation 2.17 and with the in chapter 2.2.3 described literature values.

Relationships of a_D and b_D as described in table 2.1 were used to describe the pulling force requirements.

Based on the proposed relationships by Harrigan and Rotz and Rößler; Kautzmann and Geimer the following equations are proposed for tting the draft model to the tractor implement conguration.

$$F_D = s_D \cdot (a_D + b_D \cdot v_{gnss} + c_D \cdot v_{gnss}^2) \cdot w \cdot t \quad (6.6)$$

$$F_D = s_D \cdot (a_D + b_D \cdot v_{gnss} + c_D \cdot v_{gnss}^2) \cdot w \cdot t^2 \quad (6.7)$$

The working depth is held constant. Therefore, during the online calculation of s_D, the two equations are equivalent in this specic use case. The implementation used equation 6.6. Figure 6.5 illustrates the progression of the draft force as a function of the product of s_D, w and t that is parameterized online. Therefore there is no need to enter these values manually. Since w and t are constants due to the implement choice and regulated working depth, the calculation is only aected by soil specications (s_D).

0

0

0

0

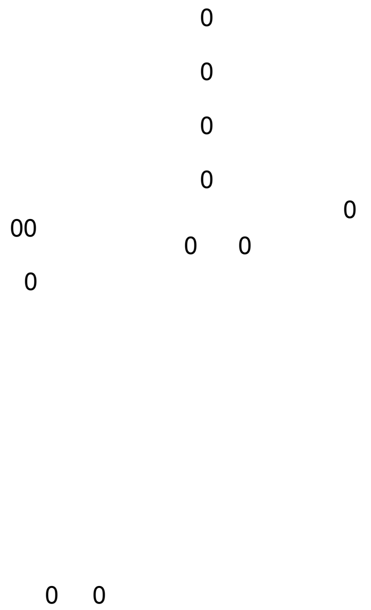


Figure 6.5: Draft Force Curves

6.4.1.3 Combined Model

To combine traction and draft forces, the simplied equation 6.8 is used, neglecting the inuence of rolling resistance, air resistance and acceleration force in a way that their sum is assumed to follow the same characteristics in regard to changes in working speed (v_{gnss}) as the implement draft force (F_D).

$$F_T = F_D + F_{W,h} \quad (6.8)$$

Air resistance and acceleration resistance can be neglected due to limited speeds and accelerations during agricultural tillage.

Rolling resistance, however, has a signicant inuence on the tractive behavior of the vehicle. The assumption of neglecting the rolling resistance can only be valid if the implements draft force characteristics feature a signicant static share independent from the vehicles working speed.

A correction term to consider this resistance into the equation, for instance, in the most simplistic way by using a static rolling resistance coecient ϱ of 0.1, could improve results further but is not considered during this work.

Some simplication is necessary because forces need to be eliminated from the equation system to determine speed relationships.

Some heavy tractors do not feature sensors for detecting the actual draft force of the implement and only allow the calculation of the traction force (F_T) based on

gearbox pressures. This equation can therefore also be used to calculate F_D from F_T . On the other hand, if only F_D is measured, F_T can be calculated for online parameterization.
Even though the tractor that has been used in this measurement setup does feature sensors for measuring both F_T as well as an equivalent value to the draft force F_D, the calculation of actual draft force values from these measurements does involve correction factors due to the lever arms on the sensor mounting points. Because of that, the method of calculating F_D from F_T has been chosen.

The relationship between the measured F_T and σ allows the calculation of c_T . Using equation 6.8, F_D can be obtained in order to calculate s_D with the usage of the measured v_{gnss}.

Subsequently entering the online-parameterized equations that have been presented during the previous sections yields a relationship between $\tilde{v}_{theo}$ and $\tilde{v}_{gnss}$ that is valid for the next state transition. After calculation of the parameters c_T and s_D based on latest measurement data, the equation can be used to calculate $\tilde{v}_{gnss}$ for hypothetical values of $\tilde{v}_{theo}$. It additionally follows that due to the obtained correlations, also $\tilde{F}_T$ and $\tilde{\sigma}$ can be calculated from this relationship.

$$1.2 \cdot c_T \cdot \frac{\tilde{v}_{theo} - \tilde{v}_{gnss}}{\tilde{v}_{theo}} - c_T \cdot \left(\frac{\tilde{v}_{theo} - \tilde{v}_{gnss}}{\tilde{v}_{theo}}\right)^2 = s_D \cdot (a_D + b_D \cdot \tilde{v}_{gnss} + c_D \cdot \tilde{v}_{gnss}^2) \cdot w \cdot t + F_{W,\hbar} \quad (6.9)$$

6.4.2 Fuel Rate Prediction

The Fuel Rate (B) is the only variable that cannot be predicted by modeling the system without further insight into the transmission and engine control. Therefore a Directional Neural Network was chosen to approximate the fuel consumption for a specic state transition. Afterward, the derived values can be used to compute rewards functions.

The prediction of B is the only part of the whole reward prediction model that needs specic customization onto the vehicle.
During tests, it was shown that the engine and transmission parameters are not necessary for predicting the vehicle's fuel consumption and that the values derived from the simula tion model are sucient for the prediction of fuel consumption.

The thereby used inputs for the training process were:

- Pitch Angle (θ)
- Theoretical Speed (v_{theo})

- Traction Force (F_T)
- GNSS Speed (v_{gnss})
- Slippage (σ)

Only these input variables and the fuel rate that needs to remain as a label for the training process remained in the training data. Any other content was purged after ltering the outliers.
Due to the dependency of σ to v_{theo} and v_{gnss}, it could probably be also eliminated from the training data in further projects.

The machine model can supply all relevant inputs for $\tilde{B}$ prediction except θ, which is assumed to be constant during state transition, as previously described. After training, these modeled traction and draft parameters are therefore entered as input parameters to obtain $\tilde{B}$:

- The latest measured Pitch Angle (θ)
- A proposed Theoretical Speed ($\tilde{v}_{theo}$)
- The associated modeled Traction Force ($\tilde{F}_T$)
- The associated modeled GNSS Speed ($\tilde{v}_{gnss}$)
- The associated modeled Slippage ($\tilde{\sigma}$)

The input parameters were normalized before being entered into the network to neutralize dierent value quantities and ranges during training and application of the neural network.

Figure 6.6 visualizes that this method allows fuel rate approximation when applied on currently measured traction and draft data. The previous chapters propose that this input data can be modeled suciently accurate for other states based on dierent proposed $\tilde{v}_{theo}$ - proposals. In that case, the prediction of the fuel rate allows the usage of this parameter for reward calculation without requiring a complete power-train model.

Figure 6.6: Fuel Rate Prediction

The optimal network parameter set was chosen by using a Hyperparameter search. There fore the structure was adapted subsequently based on the results of training and evaluating dierent architectural changes.

6.4.2.1 Network Structure Optimization

A Hyperband-Algorithm implemented in Keras was used for searching the optimal network parameter conguration.
To speedup the search, the hardware chosen for was a computer featuring a *Intel(R) Core(TM) i5-9500* CPU and a CUDA-enabled *Nvidia GEFORCE GTX 1660* GPU running Ubuntu 20.04.
The software in use was TensorFlow 2.3.0 and CUDA 10.1.

A Hyperparameter search was conducted to optimize the network's structure.

Hyperparameter Search The rst hyperparameter search was conducted to obtain the rough network structure using a hyperband algorithm. During this process, 6222 dierent network congurations were trained and evaluated.

As preparation, the data was split randomly into a training (80 %) and an evaluation set (20 %). This distribution remained constant to establish comparability between the dierent trained networks. Of the remaining training data, 20 %, meaning 16 % of the total data, was used as internal validation during training.

Furthermore, the input data was normalized to eliminate the dierent orders of magnitude of input parameters.
The learning rate was left to the initial values for each optimizer implementation in TensorFlow to be then adapted during the second hyperparameter search. Some of the evaluated optimizers also feature dynamic learning rate adaption and therefore had an advantage over optimizers with static learning rates.
The maximum number of training epochs per network was set to 1000. Table 6.2 shows the total evaluated hyperparameter congurations.

Table 6.2: Hyperparameter Search 1

Parameter	Range	Step
Number of Hidden Layers	1 - 11	2
Neurons per Layer	64 - 512	64
Activation Function	relu, tanh, sigmoid, softmax, selu	
L1 Regularization	0 - 0.2	0.01
L2 Regularization	0 - 0.2	0.01
Dropout	0 - 0.5	0.1
Optimizer	Adam, Adadelta, Adagrad, Adamax, Nadam, Ftrl	

As loss function the mean squared error was chosen. The hyperband was then sorting the networks by mean squared error on the included validation data. After successful search, the networks presented in table 6.3 performed best.

Table 6.3: Hyperparameter Search 1: Best Networks

Parameter	Network 1	Network 2	Network 3
Number of Hidden Layers	9	9	7
Neurons per Layer	256	192	256
Activation Function	relu	relu	relu
L1 Regularization	0	0.01	0.01
L2 Regularization	0.18	0.07	0.1
Dropout	0	0	0
Optimizer	Adadelta	Adagrad	Adagrad
Best MSE	0.895	0.896	0.93

For ne-tuning the parameters, a second hyperparameter search was conducted. Since the best performing network in the previous search all feature adaptive learning rates, there is no need to add the learning rate to the hyperparameter list. Furthermore, the dropout layers were removed.

The activation function was set to ReLu for all layers. In this run, the number of neurons and the regularizing terms were set individually for each layer. The total evaluated hyper parameter set is shown in table 6.4.

Table 6.4: Hyperparameter Search 2

Parameter	Range	Step
Number of Hidden Layers	6 - 10	1
Optimizer	Adadelta, Adagrad	
Per Layer: Neurons	64 - 512	64
Per Layer: L1	0 - 0.2	0.01
Per Layer: L2	0 - 0.2	0.01
Learning Rate	10^{-9}, 10^{-8}, ... , 10^{-2}	

After the second Hyperparameter Search, the initial value of the learning rate was opti mized for the best performing model. However, the second Hyperparameter Search could not yield any improvement compared to the conguration that the rst search obtained.

6.4.2.2 Final Network Architecture

The nal network architecture is displayed in gure 6.7a and consisted of a normalization layer, 9 densely connected hidden layers with 256 neurons each. Every neuron used the activation function ReLu and had a regularization term L2 of 0.18. To train the network, the optimizer Adadelta was chosen with an initial learning rate of 0.001. The training process was stopped early when the algorithm did not improve predictive performance anymore.

The nal MSE of the trained network on the evaluation dataset was 0.895. Figure 6.7c illustrates these predictions as well as their distribution. The predictions are compared to the actual fuel consumption of the dataset. The closer the values of the predictions get to the orange line, the better the performance. The graph shows that the predictions were not always perfect, but the number of outliers is minimal. The distribution is further visualized in a histogram in gure 6.7b.

It is also expected that while comparing predictions from similar environmental conditions, the associated predictions will tend to have their deviations from the actual value pointing in the same direction.

(a) Final Network Architecture
(b) Performance Histogram

- -

(c) Network Evaluation

Figure 6.7: Final Network Structure and Performance

6.4.3 State Evaluation

The previously described system can predict the vehicle's state based on various suggested values for $\tilde{v}_{theo}$. The possible states have to be evaluated and compared to select the optimal state transition.

This comparison can be established with the usage of dierent reward functions. The model's simplicity allows a comparison using dierent functions without changing the model and network training, allowing individual parameter adaption even during opera tion.

The target functions from *Machine Learning for Process Automation of Mobile*

Machines in Field Applications presented in chapter 3.2.2 were adapted to establish comparability. Furthermore, a new target function is proposed to compare total machine costs for dierent states.

6.4.3.1 Eiciency

$$R_{ecient} = \frac{\tilde{v}_{gnss} \cdot w}{\tilde{B}} \quad (6.10)$$

6.4.3.2 Performance

$$R_{performant} = \tilde{v}_{gnss} \cdot w \quad (6.11)$$

6.4.3.3 Total Cost

The additional target function includes machine costs and driver costs to perform state selection based on the total cost per area for the combined process. It shows the model's exibility to use dierent target functions without the necessity of retraining the network. The values from the billing overview of the Maschinenring Baden-Württemberg were used as orientation. The included implement costs are based on a calculation using a working width of 3 m since there was no data available for other widths. [60] These values can be adapted to t the individual operating costs of a specic farm.

Table 6.5: Operating Costs [60]

Product	Symbol	Cost Unit	Cost
4-Wheel-Drive Tractor 112-129 kW (152-175 PS)	C_T	h	33.60 €
Driver	C_D	h	18.00 €
Light cultivator with roller	$C_{C,light}$	ha	17.40 €
Heavy cultivator with roller	$C_{C,heavy}$	ha	20.30 €
Fuel	C_F	l	1.00 €

The specic total cost per area can be calculated using formula 6.12. The result has to be minimized for state selection.

$$C_{cost} = \frac{C_C + C_T + C_D + \tilde{B} \cdot C_F}{\tilde{v}_{gnss} \cdot w} \quad (6.12)$$

6.5 Usage of the State Prediction Model

This section briey describes the usage of the state prediction model. The steps necessary for a single optimization iteration are listed here in order. At the end of one optimization iteration, the next one can be started directly or after a short waiting period.

1. Extraction: All values v_{theo} , v_{gnss}, F_T , θ are read from the tractors internal ECUs

2. Processing: The obtained signals are averaged over the last second (Frequencies listed in table 5.1)

3. Filtering: Outliers are ltered using the criteria listed in table 5.3 61

4. Calculation: δ and $F_{W,\hbar}$ are calculated from θ (equations 5.2 and 2.3) Furthermore, F_D is calculated from F_T and $F_{W,\hbar}$ (equation 6.8)

5. Traction Parameterization: Traction relationship parameter c_T is calculated (Sec tion 6.4.1.1)

6. Draft Parameterization: Draft relationship parameters_D is calculated (Section 6.4.1.2) 7. Combination: Traction and draft relationships are combined (Section 6.4.1.3)

8. Modeling: Forces and speeds are modeled for a freely select-able set of Theoretical Speeds ($\tilde{v}_{theo}$) based on the combined model

9. Prediction: The fuel rate ($\tilde{B}$) is predicted for each of the modeled parameter sets (Section 6.4.2)

10. Selection: The best parameter set is chosen based on a target function (Section 6.4.3) 11. Application: The best $\tilde{v}_{theo}$ is applied as new v_{set} of the tractor

7 Evaluation

This chapter describes the evaluation process of the control system which was carried out in the area of Zaisenhausen, Baden Württemberg, equally to the collection of the training data. In this area, slope angle and soil conditions, and composition can vary to a large degree, which allows an extensive and diverse evaluation. Firstly, measurement runs were conducted using the same test setup as during training data collection. Secondly, the implement was exchanged to a light cultivator to display the capability of the control system to be transferable to various implements. Afterward, advantages and disadvantages are discussed.

7.1 Heavy Cultivator

Tests were conducted within dierent environment settings than during training data collection to evaluate the system for the dierent reward functions. Testing occurred at an ambient temperature of about -10 °C, which is signicantly lower than the lowest temperature of 14.8 °C during training data collection. The testing eld was frozen yet already partly defrosting and contained signicant dierences in slope.

The elevation of the evaluation eld is shown in gure 7.1. Tests were performed in the same working direction (Track 1), whereas the test runs with minimizing total cost as reward function used Direction 2 to establish comparability. For a better representation, the altitude values in this graphic have been adjusted to the minimum altitude contained therein.

(a) Track 1

(b) Track 2

Figure 7.1: Elevation Prole Heavy Cultivator Testing

Since the test area did not feature changing soil types and because the soil was already defrosting, the automatically lifting feature could not be evaluated, and testing was limited to comparing dierent speed setup congurations using a constant working depth.

States where the network predicted more than 34 l/h in fuel rate were declared unreachable and therefore removed from the possible list of state transitions since they never occurred during training data collection. Working depth was set to 18 cm for all automatic control modes and maintained and adjusted automatically. The human reference driver used pulling force control by 50 % and had to maintain working depth himself for a realistic comparison of state of the art.

Due to computational eort in model evaluation and fuel rate prediction, only the current theoretical speed and 0.5 km/h faster and slower were compared. The model features a continuous solution space, but due to restricted computational performance on the Thinkpad E495 notebook used to calculate state transitions, the choice was made to limit the action space to maintain a high update rate. This choice results in the problem that the algorithm takes time to increase and decrease speed levels at the beginning of each evaluation run, which can be solved using a faster computer.

The performance target was expected to be similar to full throttle since slippage could never reach values greater than 60 %. The drop in tractive power transmission typically found in this range can not be achieved due to too low tractor power. Parallel rows were driven in the same direction to deliver comparable results.

Figure 7.2 shows the results of the dierent control modes considering their performance (v_{gnss}) and their fuel eciency per area.

(a) Performance Comparison

(b) Eciency Comparison

Figure 7.2: Comparison of Control Modes

In performance, as expected, the performance control mode always gained higher speeds than human and eciency control. However, during the rst 30 meters, the speed was reduced compared to the human because of the low starting speed and the forced incre mental increase. As previously described, this can be xed using the continuous solution space of the model or by setting a higher initial speed either manually or by initializing with the speed setting of the last track.
It is clear to see that the eciency control mode succeeded over the other two

modes. This 64

results from the fact, due to lower speed settings, superior total power train eciency can be achieved by the TMS as it is not being forced to deliver full performance and therefore able to run the engine at the optimal operating point.

An additional increase in eciency can be expected by tweaking the assumed constant oset between slope angle and pitch angle or replacing the static oset with an implement position-based surface angle calculation as described in equation 5.3. Figure 7.3a shows the comparison of the individual fuel rates of the driving strategies. The lower performance of the reference run can also be explained by the fact that pulling forces can vary due to diculties during manual working depth control without exact working depth measurement. Figure 7.3b showcases this eect. Compared to the human driver, the automatic control system delivered a more constant working depth. The noise in the measurement data can be traced back to the snow and the plants on the eld. Furthermore, the measurement noise is inuenced by the relatively small eld of view of the *L515*. A wider eld of view could improve plane detection and reduce noise but is impractical due to limited mounting space between the tractor and implement.

(a) Fuel Rate Comparison
(b) Automatic Working Depth Control

Figure 7.3: Comparison of Control Modes

Table 7.1 shows the numeric results of the evaluation. Thereby, these results are due to both the automatic working depth control and speed control. A separate evaluation was not possible due to limited test capacities.

When comparing the three evaluated settings, the rst 30 m need to be excluded since the results of the automated control modes in the initial period are aected by the limitations in computational power and an unsuitable initial speed. This period is therefore omitted from the comparison to generate more representative results regarding maximum model capabilities.
Consequently, compared to the reference, the eciency target system improved fuel eciency by 18.94 %. Under the assumption of a linear eciency disadvantage caused by the increase of working depth of the implement in the reference measurement, the latter value drops to 11.68 % but still represents a signicant improvement.

Table 7.1: Results

	Mean Velocity in km/h		Mean Fuel Rate in l/ha	
	Complete Distance	After 30 meters	Complete Distance	After 30 meters
Reference	6.56	6.74	11.69	11.30
Ecient	4.09	4.20	9.55	9.16
Performant	6.40	7.32	11.38	11.10

In the same way, performance mode achieved an increase in average speed by 8.61 % compared to the human reference drive.

Due to the limited engine power of the tractor and the fact that slippage never exceeded 60 % during evaluation, performance mode resembles full throttle with working depth control active.

Cost eciency control modes were run using 1.20 €/l and a hypothetical value of 50 €/l as fuel cost to display the exibility and possibilities of the model. Figure 7.4 visualizes the speed of the vehicle when the control system is adapted to minimize total operational cost. Due to limited evaluation elds, the driving direction used to collect the data for this graph was the opposite of the previously described graphs (Track 2). The same explanation applies to the shorter range of the 50 €/l curve due to unequal track length at the edge of the eld. However, because the eld contained a hill in the center, comparability should be ensured.

Figure 7.4: Eect of Fuel Price Increase