DEVELOPMENT OF AN AUTOMATED SYSTEM FOR *IN SITU* MEASUREMENTS OF SOIL HEALTH INDICATORS

John Lan

Department of Bioresource Engineering

Faculty of Agricultural and Environmental Sciences

McGill University

Sainte-Anne-de-Bellevue, Quebec, Canada

March 2024

Abstract

Monitoring soil health is of growing significance in achieving an informed and sustainable management of agricultural and natural resources. The term “soil health” can be defined as the soil’s ability to sustainably carry out ecosystem services, emphasizing soil’s biotic components. In evaluating soil health, one can measure a range of indicators; however, in the short term, these indicators are often affected by confounding factors, which may result in high spatial variability at the field level. This project addresses the need to gather large datasets on multiple soil properties by developing a robotic sensor array coupled to an autonomous vehicle platform to provide *in situ* data on soil characteristics, focusing on shallow depths (top 0.10 m). The system uses a porous metal probe to extract air from the soil’s air-filled pores, then quantifies its CO_2 concentration using a non-dispersive infrared (NDIR) sensor. To characterize porosity, air is blown into the soil at either a fixed flow rate or fixed pressure, and the resultant pressure arising from the soil’s air resistance is measured. A pneumatic penetrometer and a soil-metal friction sensor are added, together with more conventional soil sensors (*e.g.*, moisture and temperature). The validity of each sensor was evaluated according to the nature of its response and its relationship to reference measurements. It was found that the pneumatic penetrometer could predict cone penetration resistance with an RMSE of 233 kPa. The soil’s air resistance measurement predicted bulk density with an RMSE of 0.13 g cm^{-3}. The other measurements (*e.g.*, CO_2 concentration measurement) were found to be responsive during the measurement sequence.

Résumé

La surveillance de la santé des sols revêt une importance croissante pour une gestion informée et durable des ressources agricoles et naturelles. Le terme "santé des sols" peut être défini comme la capacité du sol à fournir durablement des services écosystémiques, en mettant l'accent sur les composants biotiques du sol. Pour évaluer la santé des sols, on peut mesurer une gamme d'indicateurs ; cependant, à court terme, ces indicateurs sont souvent influencés par des facteurs confondants, ce qui peut entraîner une grande variabilité spatiale à l'échelle du champ. Ce projet répond au besoin de collecter de grands ensembles de données sur plusieurs propriétés du sol en développant un ensemble de capteurs robotiques couplé à une plateforme de véhicule autonome pour fournir des données in situ sur les caractéristiques du sol, en se concentrant sur des profondeurs peu profondes (les 0,10 m supérieurs). Le système utilise une sonde métallique poreuse pour extraire l'air des pores remplis d'air du sol, puis quantifie sa concentration en CO_2 à l'aide d'un capteur infrarouge non dispersif (NDIR). Pour caractériser la porosité, de l'air est soufflé dans le sol à un débit ou une pression fixe, et la pression résultante due à la résistance de l'air du sol est mesurée. Un pénétromètre pneumatique et un capteur de friction sol-métal sont ajoutés, ainsi que des capteurs de sol plus conventionnels (par exemple, humidité et température). La validité de chaque capteur a été évaluée en fonction de la nature de sa réponse et de sa relation avec les mesures de référence. Il a été constaté que le pénétromètre pneumatique pouvait prédire la résistance à la pénétration du cône avec une RMSE de 233 kPa. La mesure de la résistance de l'air dans le sol a permis de prédire la densité apparente avec une RMSE de 0,13 g cm^{-3}. Les autres mesures (par exemple, la mesure de la concentration en CO_2) se sont révélées réactives pendant la séquence de mesure.

Acknowledgements

I would first like to thank Ztractor, Inc. (Palo Alto, CA, USA), and their team for their partnership and support in this project. My special thanks go to Mr. Bakur Kvezereli for inspiration and assistance in mounting and testing the prototype on their systems and fields. This research was made possible thanks to funding partly through Mitacs Accelerate project with support from Ztractor, Inc. and an NSERC/Discovery project on Integrated Proximal Sensing of Soil and Crop.

Next, I would like to thank my friends and colleagues in the Precision Agriculture and Sensor Systems (PASS) research team for the collaborative and supportive community they have been to me. Thank you, Felippe Silva Hoffman Karp, Kyle Geddes, Karim Abdalla, Andres Rello, Hamed Etezadi, Md Saifuzzaman, and Philippe Leblond for keeping me company and helping me to diagnose issues when I could not otherwise see them.

I would like to thank my loving family: my mother, father, and sister, for their encouragement and for helping me to maintain perspective and focus.

Thank you to my advisory committee member Dr. Grant Clark, for giving me valuable advice and criticism.

Finally, I would like to express my heartfelt gratitude to my supervisor and mentor, Dr. Viacheslav Adamchuk, whose unwavering faith and guidance have helped me learn about myself and my potential to help others. My experience working with him throughout my academic career has allowed me to develop both as a researcher and an individual. He provides the space to learn on your own but is always there to offer words of wisdom and validation exactly when needed — that could only come from a place of experience and empathy. In moments of achievement and overcoming, his expression of endorsement and pride brings a sense of camaraderie in the team and an excitement to do more. I can see how his compassion, dedication, and balance contribute to his success, and I am so thankful to have him as someone to learn from and whose example I can hope to follow.

Contribution of Authors

The author of this thesis was responsible for the mechatronic and software design of the soil sensing implement that was to be integrated into an autonomous vehicle platform. The author also planned and conducted field testing. This study was supervised by Dr. Viacheslav Adamchuk, a professor in the Department of Bioresource Engineering at McGill University, who, together with the project partner, Ztractor's CEO, Mr. Bakur Kvezereli, conceived the idea. Dr. Adamchuk provided key support and guidance throughout the study. Ztractor, Inc. provided funding for this project, and its representative, Mr. Kvezereli, was instrumental in the final integration into their autonomous vehicle platform. Kyle Geddes was the primary lab assistant involved in troubleshooting and data collection.

Table of Contents

List of Abbreviations

AET	Autonomous Electric Tractor
CDC	Centers for Disease Control and Prevention
CO_2	Carbon Dioxide
CSV	Comma-Separated Values
DC	Direct Current
DOC	Dissolved Organic Carbon
DSM	Digital Soil Map
ECa	Apparent Electrical Conductivity
EMI	Electromagnetic Inductance
FDM	Fused Deposition Modeling
GIS	Geographic Information System
GCR	Galvanic Contact Resistivity
NDIR	Non-Dispersive Infrared
ISFET	Ion-Selective Field Effect Transistor
MAC%	Microbial Active Carbon Percent
MBC	Microbial Biomass Carbon
N	Nitrogen
NDIR	Non-Dispersive Infrared
NF	National Fine Thread
NIOSH	National Institute for Occupational Safety & Health
$\mathbf{NO_3^-}$	Nitrate Ion
NPT	National Pipe Thread
P	Phosphorus
PA	Precision Agriculture
PC	Personal Computer
PLA	Polylactic Acid
PLC	Programmable Logic Controller
PSS	Proximal Soil Sensing
PWM	Pulse-Width Modulation
RMSE	Root Mean Squared Error
RTK	Real-Time Kinematic
SOM	Soil Organic Matter
TDR	Time-Domain Reflectometry
UAV	Unmanned Air Vehicle
UTV	Utility Task Vehicle
UV	Ultraviolet
vis-NIR	Visible Near-Infrared
VRT	Variable-Rate Treatment
WEOC	Water Extractable Organic Carbon
$\boldsymbol{\theta}_{fc}$	Volumetric Water Content at Field Capacity
$\boldsymbol{\theta}_{vol}$	Volumetric Moisture Content
ρ	Bulk Density
φ	Porosity
τ	Shear Strength

List of Figures

Chapter 1 Introduction

1.1 Soil Health

Arable land is a scarce and crucial resource worldwide. While climate change may increase arable land at higher latitudes, much of the world will experience a loss of such land (Zhang and Cai, 2011). Therefore, it is important to monitor and improve the productivity of the available limited land. Soil provides several ecosystem services to aid agricultural productivity, water quality, and climate change mitigation. The ability of soil to sustainably provide these services is referred to as *soil health.* Effective evaluation of soil health requires the measurement of multiple indicators. These can be physical (*e.g*., bulk density, water infiltration rate, etc.), chemical (*e.g*., pH, other ion concentrations, etc.), or biological (*e.g*., CO_2 production, bacterial and fungal activity, etc.) (Arias et al., 2005). At present, a soil's health is largely measured through its chemical properties rather than its biological properties. Existing methods of quantifying soil biological indicators can be expensive and inconvenient as they often involve gathering soil samples, that are then shipped for laboratory testing (Lehmann et al., 2020). Biological properties (*e.g*., soil respiration) can be affected in the short term by other soil parameters (*e.g*., temperature and moisture levels). If such confounding parameters are not simultaneously measured and accounted for, the validity of measurements may suffer (Hashimoto and Komatsu, 2006). Such parameters' temporal variations can be due to weather and diurnal patterns that vary daily and over the long term (Rochette et al., 1991). Finally, soil parameters vary spatially on multiple scales, *i.e*., intra-field and inter-field (Lin et al., 2005). To properly address these issues a high volume of measurements is required to be able to advise soil management practitioners properly. This can be tedious and expensive.

1.2 Proximal Soil Sensing

While soil health and other characteristics can be measured remotely on a large scale using satellites and unmanned air vehicles (UAVs), these methods can be expensive to set up. They cannot always provide the same types of measurements that closer and more invasive sensors can provide (Adamchuk et al., 2011). Proximal soil sensing techniques allow for *active* measurements (ones that emit their own power) and *invasive* (measurements that use direct contact with the soil or are in the soil) (Viscarra Rossel et al., 2011). While these measurements can be taken *in situ* (measured in the field), saving time compared to *ex-situ* (having a soil core sample analyzed in a lab), it can still be tedious and time-consuming to achieve the necessary volume to map soil health effectively. Fortunately, advancements in automation have allowed for new and innovative ways to map soil data quickly and efficiently.

1.3 Prior Design

This study expands upon a similar project on automating proximal soil health data collection through a sensor array capable of measuring multiple soil health indicators (Dias Carlson, 2021). Three individually actuated probes were used:

(i) one that measured the CO_2 concentration and the soil's permeability to air,

(ii) a cone penetrometer that provided a profile of the soil's resistance to penetration, and

(iii) a soil moisture and temperature sensor.

A commercially available autonomous electric tractor (AET) was used as the sensor array's vehicle platform. Some challenges were faced in keeping the sensor at an ergonomic size and shape. This could be addressed by combining the actuation of the probes and focusing on shallow depths.

1.4 Research Objective

The present study's research objective was to provide a useful tool to soil researchers by developing a sensor system to automate the mapping of multiple shallow-depth soil health indicators. Specific objectives were to:

(i) develop a single sensor array prototype that could measure multiple soil health indicators: penetration resistance, soil-metal friction, CO_2 concentration in the air-filled pores, air permeability, moisture, temperature, and apparent soil electrical conductivity (EC_a).

(ii) interface the system with an autonomous vehicle platform to fully automate the process.

Chapter 2 Literature Review

2.1 Soil Health Overview

Accelerated global population growth has increased our need to tap into the planet's natural resources. This demand for scarce resources (*e.g.*, fresh water and arable land) often leads to deforestation and degradation of existing farmlands through ill-considered use. In this light, environmental quality can be seen as a limiting factor on economic growth (Cropper and Griffiths, 1994). The effects of climate change have also exacerbated the degradation of existing farmland. While areas of the world at higher latitudes might see an increase in arable land, other regions will undergo a loss of arable land. Zhang and Cai (2011) predicted the magnitude of the global decrease in arable land to depend on the increase in greenhouse gas concentrations. Moreover, some arable lands are being encroached upon by cities and other non-agricultural activities. While expanding the current expanse of agricultural lands is possible, much of the higher quality lands are already being used. Simpler ways of improving the productivity and quality of existing land, (*e.g.*, use of fertilizers and plant breeding) are already being used. More complex management practices will be required to further preserve the quality of existing lands (Hobbs, 2007). Additionally, while increasing productivity in the short term, certain practices can have adverse long-term consequences. For instance, increased fertilizer use can damage plant-synergistic arbuscular mycorrhizal fungi (Ma et al., 2021).

It is important to monitor the effectiveness of different farming techniques and management practices to select the best practices. One principle to follow is to foster *soil health*. Soil health is like the older concept of *soil quality*, which describes the soil's ability to provide valuable ecosystem services like erosion resistance, water flow and quality regulation, plant growth support, carbon sequestration, and nutrient recycling. The term soil health, on the other hand, is not just focused on the utility of the soil, but also acknowledges its biological component and treats soil as something akin to a living system (Laishram et al., 2012).

2.2 Soil Health Measurements

Soil health is evaluated by measuring soil health indicators. These are quantifiable parameters that are representative of health. This definition is useful because measuring a soil's quality in response to management practices or environmental changes requires sensitive measurement techniques, *i.e.*, capable of detecting a significant change in a relatively short time after some treatment has been applied. Some soil health indicators (*e.g.*, soil respiration or nitrogen mineralization) that focus on a soil system's biological elements tend to be sensitive to and highly variable under variable external conditions (Doran and Zeiss, 2000). At present, soil health evaluation is largely focused on chemical indicators, as biological indicators are often difficult to use due to the labor-intensive and invasive soil core sampling methods that need to be conducted by qualified individuals. Then, these cores need to be sent to a lab to be analyzed *ex situ* which incurs an additional cost and turnaround time (Lehmann et al., 2020).

The degree to which these properties relate to important soil functions and the level at which they respond to different management practices determines their usefulness. These properties can be categorized as physical, chemical, or biological but are often related (Allen et al., 2011).

2.2.1 Soil physical properties

Soil's physical properties are important because they relate to crucial functions, such as providing support for plants to grow and allowing the cycling of water and air. Useful physical parameters related to this include bulk density (ρ), shear strength (τ), porosity (φ), and soil available water, *i.e.*, the difference between the soil's water content at field capacity and that at the permanent wilting point: $\theta_{fc} - \theta_{pwp}$ (Allen et al., 2011). These parameters can be used to estimate mechanical impedance to root growth, resistance to compaction (Bengough and Mullins, 1990) and draft force required to drive certain agricultural implements (Al-Hamed et al., 2014). Compacted soil offers a poor medium for crop production as it makes it harder for air, water, and roots to pass through. This can affect biological properties such as the respiration of roots and soil microorganisms. Infiltration will be reduced leading to pooling and reduced θ_{fc}. The rooting depth can also be reduced which impairs plants' resilience (Nawaz et al., 2013). Additionally, rooting depth has been hypothesized to relate to the soil's ability to sequester carbon (Pett-Ridge et al., 2018).

ρ is a useful soil health indicator related to soil compaction that represents the mass of the soil per volume as it exists *in situ*. Soil bulk density can be measured as either wet bulk density (ρ_{wet}) or dry bulk density (ρ_{dry}). The ρ_{dry} is typically measured by hammering a metal cylinder of known volume into the soil and then digging out the cored soil while taking care not to compress it or lose any of it during extraction. This soil core is then dried in an oven at 105°C. The ρ_{dry} is an established soil health indicator since it can be indicative of compaction and aggregate stability (Håkansson and Lipiec, 2000). Fairly inexpensive, measurements of ρ benefit from using widely available tools (Jabro et al., 2020). Additionally, θ can be measured simultaneously by comparing the original weight of the sample with the dry weight.

Using the same soil core sample, φ, a characteristic of soil structure, can also be measured using the ρ_{dry} and particle density (Danielson and Sutherland, 1986). This soil structure parameter also has a profound effect on the soil's biological properties. Finding that microarthropod density was greater in soils with larger pores (*e.g*., coarse-textured and less compacted soils), Erktan et al. (2020) suggested that this arose because size distribution and interconnectedness of pores determined different species of soil fauna's accessibility to the soil, and therefore, their trophic interactions The nature and interconnectedness of the pores has also been found to relate to biodiversity in soil micro- and macro-fauna. This is because the varied sizes and separation of pores can affect accessibility by soil fauna and thus, govern trophic interactions (Erktan et al., 2020).

One method of measuring the nature of soil pores is by using pneumatics. Yildirim et al., (2006) studied the ability of pressurized air to cut through a soil core sample. They found that, at a given pressure, the primary factors affecting the results were the ρ_{dry} and θ of the soil core sample.

2.2.2 Soil chemical properties

Soil's chemical properties relate largely to the nutrient and elemental content of the soil, along with the balance of these nutrients and their availability.

One commonly assessed feature is the potential of hydrogen (pH) or the acidity and alkalinity of the soil. The pH is important since it determines the solubility of certain minerals and metals in the soil. In acidic soils (pH < 4.2), aluminum ions (Al^{3+}) are more dominant, and their toxicity becomes a major yield-limiting factor (Rengel, 1996). This can be remedied by liming the soil to increase the pH or using a different fertilizer (Li et al., 2019).

The existence of ions in the soil results in the presence of electrical characteristics. One quick way of measuring this is by measuring the soil's electrical conductivity (EC). The EC of the soil is affected by several factors, including soil texture, θ, salinity, organic matter content, ρ_{dry} and size of pores (Rhoades and Corwin, 1990). Since most of these properties do not change quickly relative to each other, soil maps offer a valid depiction of the field's spatial variability over many years (Farahani et al., 2022).

In terms of chemical composition, there exist direct measurements for the quantity of the soil's chemical components, including nutrients [*e.g.*, nitrogen (N) and phosphorus (P)] and organic matter content. However, some traditional methods for quantifying N and P use harsh chemicals to extract the elements, making it generally unfeasible to measure them *in situ*. These traditional measurements can also quantify the percent of the soil comprised of organic matter (SOM%), allowing one to get a general idea of the quantity of biologically available carbon (Haney et al., 2018).

2.2.3 Soil biological properties

Soil physical parameters such as ρ_{wet} and θ, and chemical characteristics such as EC and CEC, show spatial variations that remain largely unchanged for many years. However, this characteristic is also a drawback, as these parameters are often slower to respond to changes in the environment and crop production practices than biological characteristics (*e.g.*, microbial biomass C and N, biodiversity, and CO_2 from respiration (Cardoso et al., 2013). Soil respiration and C and N levels are often measured using commercially available soil health tests (Ward Laboratories, Inc., 2024; A&L Canada Laboratories, Inc., 2024).

While SOM% and total nitrogen were previously discussed as chemical properties of the soil, breaking down these large sinks into constituents/pools such as microbial biomass carbon (MBC) or water extractable organic carbon (WEOC) can account for biological interactions and paint a more complete picture. The WEOC is measured by shaking a soil core sample in an aqueous slurry, extracting the water, and measuring its C content (Guigue et al., 2014). Not all carbon is readily available for biological use. Dissolved organic carbon (DOC) is the most mobile form of C and occurs in the form of sugars that are readily available to microbes. In contrast, MBC is the portion of the WEOC that is found in the microbes themselves (Corvasce et al., 2006). The relative amount of MBC to WEOC is sometimes called the percentage of microbially active carbon (MAC%) and can suggest if available carbon is a limiting factor in microbial activity (Ward Laboratories, Inc., 2024).

Similarly, when applying fertilizers not all the nitrogen is immediately available for plants to use. The plant available nitrogen (PAN) of a fertilizer can be estimated by modelling decomposition by exposing it to soil and taking periodic measurements of CO_2 evolution over time (Gale et al., 2006). Decomposition rates of soil organic matter (SOM) can be greatly affected by relatively small additions and changes in the substrate. This phenomenon is known as the soil priming effect (PE) and requires breaking down the assessment of C and N into their different pools (Kuzyakov, 2000).

Generally, good biological measurements are often a combination of measurements that give a ratio (*e.g.*, the microbial biomass C to N ratio (Spohn, 2015). In other cases, additional soil parameters can confound certain measurements in the short term. For example, soil respiration, an energy-producing metabolic activity that produces CO_2 as a byproduct, can be influenced by several more-or-less spatiotemporally variable soil characteristics. Soil respiration can be indirectly measured using different methods. Some methods involve taking a core sample and placing a chamber around it to measure the flux of CO_2 using either non-dispersive infrared sensors (NDIR) or measuring CO_2 absorption into a caustic solution (Bekku, 1997). However, soil respiration can be affected not only by the soil's health but also by short-term changes in temperature and θ. Increases in temperature, such as diurnal fluctuations, can increase respiration rates by helping to achieve activation energies. This effect is even more significant in soils in relatively cool areas (Lloyd and Taylor, 1994). Regarding θ, desiccation of soils reduces the respiration rate, while rehydration re-establishes it (Orchard and Cook, 1983). Respiration measurement can be further confounded since the concentration of one of its products, CO_2, can be affected by θ and temperature, irrespective of respiration itself. Moisture can affect the permeability of the soil, locking in CO_2, while a higher temperature can increase gas diffusivity. These effects are dependent on the soil depths which have been sampled, and which combinations of temperature and θ are occurring at the time (Min et al., 2020).

Temporal variation also affects soil biodiversity. As previously mentioned, soil physical properties such as soil structure and pore size influence the biodiversity of soil fauna. As a biological soil health indicator, this diversity can also be measured directly. The presence of specific microorganisms can be detected by using 16S RNA sequencing (Sharma et al., 2011). In agricultural lands, soil conditions change with time, especially in terms of biological characteristics. This can happen on a monthly scale, for example when changes in microbial diversity occur on a seasonal basis. In some cases, these changes are predictable, but in other cases, such as in the early succession of grasslands, temporal variability may be difficult to resolve. According to Lauber et al., this is likely owing to the complex interactions of plant and microbial communities under these land use types (Lauber et al., 2013).

The value of comparing ratios of parameters and the need to acknowledge the confounding effect factors have on each other explains the necessity of measuring multiple parameters simultaneously. Spatial and temporal variations of these parameters increase the demand for sampling density and frequency.

2.3 Proximal Soil Sensing

The use of soil sensors dates to at least as early as 1908, with the use of tensiometers for monitoring changes in θ. This tool helped farmers decide when and where to water their crops (Or, 2001). By the 1920s, fertilizer recommendations were being made based on field sampling. It was known that multiple samples must be taken from a field to get an accurate representation of the field’s needs. However, this was still done in the interest of averaging out the values to create a prescription that would be applied uniformly over a field (Mulla and Khosia, 2016). By the 1960s, researchers such as Dow et al. found that the variation between samples in a field and between fields was not random, but spatially dependent. This meant that areas closer together tended to be more similar than areas farther apart (1973 as cited in Mulla and Khosia, 2016).

Further properties were found to be spatially dependent as well. In 1925, Keen and Haines developed a system to measure soil strength, a soil physical characteristic, on the go. This meant that they could sample without stopping. The machine consisted of an implement that was pulled through the ground, and the equivalent resistance was measured with a dynamometer. This is an early example of a proximal soil sensor, a sensor that works when in contact with or near the soil (Minasny and McBratney, 2016).

When measurements are taken from further away, they are termed remote sensing. This often involves using imaging techniques, which became easier with the advent of satellites and drone imagery (Campbell and Whyne, 2011). The results from soil sensing need to be referenced in time and space to provide valid insights into spatial and temporal variability. This was made easier after 1983 when Navstar, the constellation of satellites for what is now known as the Global

Positioning System (GPS), was made publicly available (Aerospace, 2024). This spatially referenced information could be combined into a digital soil map (DSM). With powerful computers, Geographic Information Systems (GIS) tools could be used to perform computationally expensive operations such as interpolation and store the data in a regular grid/raster (Minasny and Bratney, 2016).

However, sensors closer to the soil can provide resolutions and types of measurements that are not available through remote sensing. These types of sensors benefit from being able to be *active* (able to emit their own energy) rather than *passive* (*e.g*., cameras). They can also be *non-invasive* — using remote sensing techniques where the sensor is not in direct contact with the soil — or *invasive* — when the sensor is in contact or in the soil. The practice of using sensors in these ways in proximity (within 2 m) of the soil is called proximal soil sensing (PSS). This includes the practice of taking measurements *in-situ* (when a sensor measures soil parameters in the field) and *ex-situ* (such as when a soil core sample is brought to a lab to be analyzed) (Viscarra Rossel et al., 2011).

In terms of mechanical properties, composite properties such as penetration resistance are the result of the relationships between basic soil mechanical properties such as soil particle size and cohesion (Koolen and Vaandragar, 1984). A soil's τ was of interest to the US Army Corps which developed an instrument called the cone penetrometer to determine a soil's load-carrying capacity (Davidson, 1965). The tool is composed of a long shaft with a conical tip that is to be inserted steadily into the ground. Traditionally, the force required to insert the probe to a certain depth is recorded. This measurement can be multiplied by the cross-sectional area of the tip to determine the pressure required (Perumpral, 1987). Alternatively, the force required at multiple depths can be measured to generate a depth profile (Rooney and Lowery, 2000). These measurements have been used to correlate with different soil physical properties, soil compaction and crop yields (Davidson, 1965). Compaction is traditionally evaluated by measuring mechanical impedance with a penetrometer (Sanglerat, 2012).

Soil mechanical impedance is largely the result of soil cohesion and soil adhesion. Soil texture can be estimated *in situ* in several ways. Since the speed of sound is related to the media it traverses, the speed at which sound travels in soil can be described as a function of its physical characteristics such as φ, texture, and θ. Soil mechanical impedance can be measured by inserting two probes into the ground. The first, a transmitter, creates sound while the other receives it. The time between the sound being emitted and it being received can be measured to determine the speed of sound in the soil. Characteristics of the received sound can also be measured (Adamo et al., 2004). In general, the behaviour of sound in the soil is related to soil texture and ρ_{dry} (Adamchuk and Viscarra Rossel, 2010).

Pneumatic sensors can also be employed to measure both the nature of soil pores and the CO_2 concentration within them. This can be done by inserting a perforated probe into the soil and drawing air, in which a non-dispersive infrared CO_2 sensor then quantifies CO_2 concentration. Air can also be blown into the ground through this perforated probe while measuring the pressure to gain insight into the soil pore structure (Carlson et al., 2020). However, as previously mentioned, CO_2 concentration and efflux can be affected in the short term by factors such as temperature and θ, so it is important to measure these as well (Min et al., 2020).

Measurements of soil CO_2 storage have also been found to be influenced by pH levels (Flechard et al., 2007). This, and the levels of other materials of a chemical nature in the soil can also be evaluated by measuring, thanks to an ion-selective membrane, the presence of specific ions. Amongst these ions are hydrogen ions (H^+) which serve to measure pH, as well as other ions such as potassium (K^+) and nitrate (NO_3^-). These can be measured using electrodes with ion-selective field effect transistors (ISFET) (Adamchuk, 2004).

Related to soil properties such as salinity, θ, SOM% and soil texture, soil electrical conductivity (EC) is commonly used to gain a general insight into the spatial variation of soil properties (Friedman, 2005). This can provide researchers guidance in choosing where they take samples. Many methods of measuring EC use sounding equipment such as electromagnetic inductance (EMI) or galvanic contact resistivity (GVC). Since these measurements are affected by a non-discrete section of the soil, the resultant values are called the apparent soil electrical conductivity (EC_a) (Seranno et al., 2014). When resistivity is used, an array with a minimum of four electrodes is involved. Two electrodes connected to the soil generate a current through the soil. The other two are used to measure the voltage across a certain distance. Since the current travels through the ground, the EC_a reading is in response to the soil media between the electrodes. By decreasing the distance of the electrodes from one another, the EC_a reading would have a higher relative response in the shallower soil layers than in the deeper ones (Pan et al., 2014).

Since the reading on a sensor can respond to multiple parameters, it is useful to take measurements from different types of sensors and use them in tandem to determine confounding variables. This practice of combining multiple sensor measurements to get more accurate data is termed *sensor fusion* (Adamchuk et al., 2011). One example of this is measuring soil chemical content by shining visible near-infrared light (vis-NIR) and evaluating the characteristics of the reflected spectrum. In this case, while these measurements alone do not give reliable results, by combining these measurements with cone penetrometer readings and EC_a readings, more accurate results can be achieved (Veum, 2017). A summary of these sensors and the parameters they measure is shown in Table 2.1

Table 2.1: Properties of proximal soil sensors

	Qualities		Related soil health indicators								
Tool			Physical			Chemical			Biological		
	Active	Invasive	Soil structure	Soil available water	Bulk density	pH	Electrical conductivity	Plant available nutrients	SOM	Soil respiration	Soil Biodiversity
Penetrometer	■	■	■	■	■						
Reflectometer	■	■	■	■	■		■		■		
NDIR CO_2	■	■						■		■	■
EMI	■		■	■	■		■		■		
ISFET	■	■				■		■			
Vis-NIR	■					■	■	■	■		

(Viscarra Rossel et al., 2011)

2.4 Automation of the Creation and Use of Soil Maps

Research findings may not be relevant when discovered but may become important years later when technology has sufficiently advanced to utilize them (Minasny and Bratney, 2016). Early soil maps were mostly used architecturally to find sound ground to lay the foundations of buildings. A few peripheral technologies needed to be developed and made more affordable to apply this understanding of the spatial variability of soil to large-scale agriculture. These needs are being fulfilled with the advent of automation in both the mapping of data and automating the application of this knowledge. This combination of data collection and precise application became known as precision agriculture (PA).

To apply a different treatment to different places in a field, the magnitude of treatment must be variable. Spinning disc applicators can be used to apply fertilizer at a variable rate; however, these result in large-diameter circles. Smaller nozzles can be used to spray liquids in smaller diameter circles and even achieve plant-level accuracy (Hague et al., 1997 as cited in Stafford, 2000). While variable-rate treatments (VRT) can be applied to many different solid and liquid chemical treatments (*e.g.*, pesticides and herbicides), most of the focus has been on nitrogen fertilizer application. This allows farmers to reduce the number and/or quantity of inputs without reducing the overall yield. This helps to lower costs for farmers but added pressure is exerted by environmental regulations that seek to reduce nutrient pollution (Zhang, 2002). One key technological improvement has been in GPS systems; they have become more compact and more precise. Real-time kinematic (RTK) corrections allow for 20 mm accuracy which, when paired with other sensors, are sufficient for simultaneous auto-guidance and geolocation of farm vehicles (Stafford, 2000). These two technologies are coupled through controllers and automation systems. The task of outputting signals onto a bus to control individual sprayers and other actuators is managed by industrial programmable logic controllers (PLC) (Sadler, 2000). The three components of high-quality, relevant data, precise tools, and the control systems to manage these systems, form the foundation of precision agriculture.

Now that precision agriculture has proven to be feasible as well as economically and environmentally beneficial, the key limiting factor has been data. As mentioned previously, key challenges in soil data acquisition are the time, effort, and cost. Effective solutions need to have a short response time (*i.e.*, provide relevant information soon after data collection starts). They also need to be easy to use/user-friendly and reduce physical labor. They also cannot be prohibitively expensive.

As mentioned previously, bringing sensors to the field and measuring the soil there using proximal soil sensing (PSS) techniques can greatly reduce the turnaround time while still allowing for measurements that can only be taken invasively. However, the greater volume of samples to achieve high spatial and temporal resolution can still prove to be labor-intensive, so automation solutions are being developed.

One such sensor system makes use of an autonomous vehicle platform and couples a cone penetrometer, porosity measurement, soil CO_2 gas sampler, along with θ and temperature sensors to take simultaneous point measurements. However, some issues were encountered with the vehicle's suspension when large loads were implemented when trying to penetrate deeper soil depths (Carlson, 2021).

Other teams seeking to develop similar autonomous robots for field soil monitoring include a team based in Virginia that is hoping to make use of a probe to make stop-and-go soil health measurements (Piper et al., 2015). Another team is looking to use sensors and data fusion by combining sources from remote sensing and UAVs to link plant phenotyping data with soil health data to acquire a more complete picture (Barrile et al., 2022).

Automating measurements makes it easier to take more samples, thereby increasing spatial and temporal density. Temporal density can be further increased by placing gas sampling chambers that can be used on soil core samples to directly measure gas fluxes in the field. In this way, chambers can provide data at a high temporal density by sampling the flux with respect to time. The addition of wireless communication allows for a convenient method for real-time monitoring of field conditions (Debbagh, 2019).

Chapter 3 Materials and Methods

3.1 System Design

The system needed to be connectable to a Bearcub-24[1], an AET, capable of powering 12VDC or 24VDC electronic systems. This AET also features mounting options for either a category 0 three-point hitch or a 2 in. (50.8 mm) class 1 straight hitch receiver (Ztractor, Inc., 2024). Given these restrictions, the individual components of the system must be able to be carried ergonomically.

The overall system design was focused on manufacturability and ergonomics. To aid in manufacturability, the system consists of two modular pieces (Figure 3.1) (*i.e.*, a sensor probe array, and a box housing the electronics and pneumatics that control and power it), allowing the weight to be evenly distributed, thereby reducing the load that any individual person must carry at any one time.

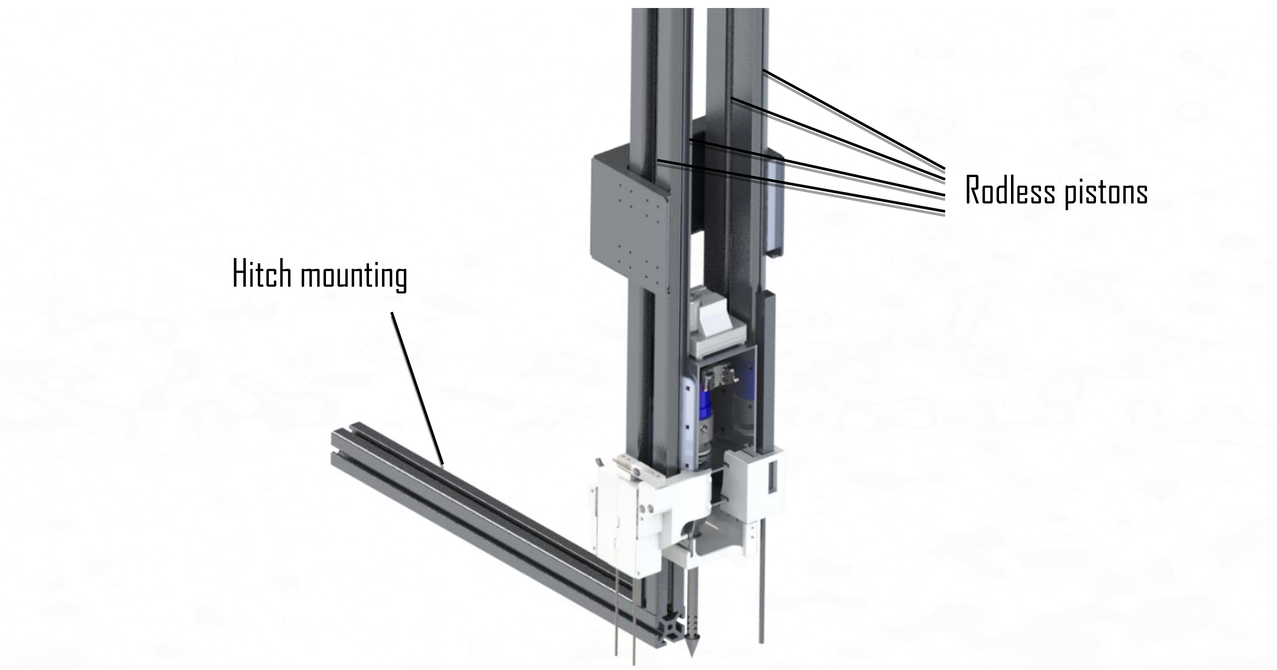


Figure 3.1: Sensor array showing location of hitch and pistons.

[1] Mention of a trade name, proprietary product, or company name is for presentation clarity and does not imply endorsement by the authors, or McGill University, nor does it imply exclusion of other products that may also be suitable.

The sensor array is designed to fit a 2 in. (50.8 mm) straight hitch receiver since it is more common than the three-point hitch. A polylactic acid (PLA) sleeve surrounds the 40 mm aluminum extrusion to make a tighter fit and improve rigidity. Aside from the plasma cut and welded ⅛ in. 6061-aluminum alloy sheet metal frame, all components were sourced from wholesale suppliers or 3D printed. The sensor probe weighs 12.7 kg (28 lbs.). Based on the National Institute for Occupational Safety & Health (NIOSH) lifting equation (Galassi 2015; Waters et al., 1994), the Centers for Disease Control and Prevention (CDC) states that workers should not be required to lift more than 23.1 kg (51 lbs.). The sensor array was lowered into the ground using four rodless pistons (Fig. 3.1) that evenly distributed the load and maintained alignment of the sensor's probes.

In this project, much of the testing was done using a John Deere (Molline) Gator, a utility task vehicle (UTV) with a diesel engine. On fossil fuel-burning vehicles, the exhaust can affect the outcome of the CO_2 concentration measurements. To address this, the sensor was moved to the front of the vehicle. However, this vehicle did not come in stock with a front hitch receiver, so a custom one was built to allow mounting the sensor in the front (Figure 3.2). An alternative to this would be to redirect the exhaust away from the sensor.

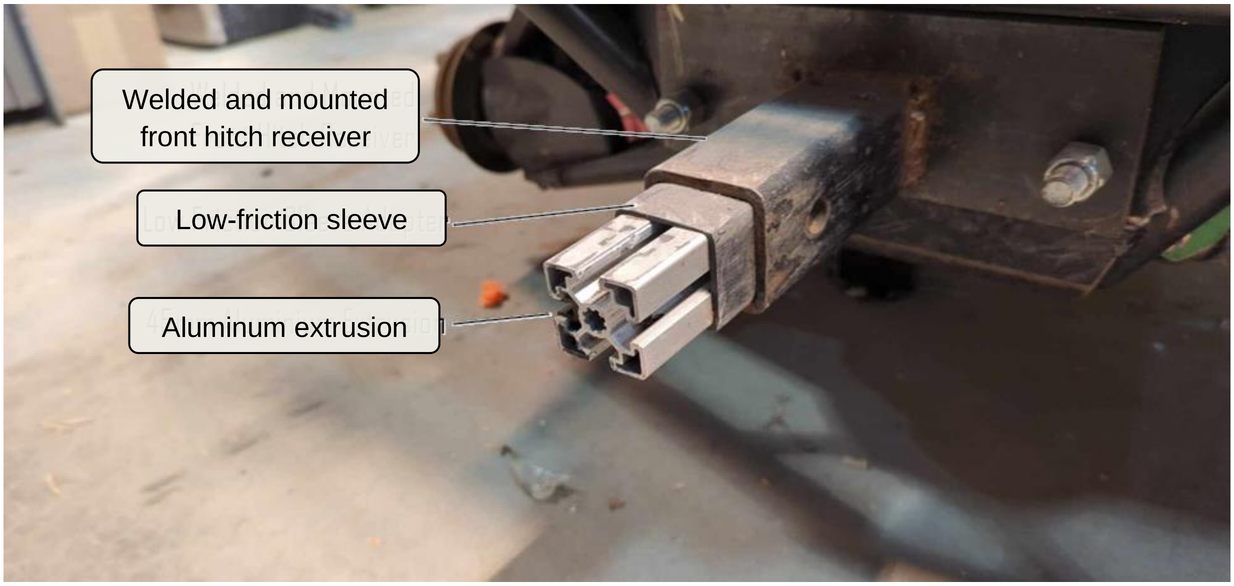


Figure 3.2: Front hitch adapter for the 45 mm extrusion to better fit the class 1 hitch receiver.

On this same UTV setup, the box housing the electronics was held in the bed. Supplying air for the system is a compressor located in the box, set to run between 620-896 kPa (90-130 psi). An 18.9 L (5 US gal) compressed air tank is used to store the pressurized air. This box provides a compact and modular way to organize and assemble the Arduino Mega microcontroller and some relays and pneumatic solenoid valves that control the sensor probe. IP67-rated Deutsch connectors serve to attach and detach the box from the sensor. Though it weighs a substantial 15 kg (33 lbs.), it features handles to aid in mounting (Figure 3.3).

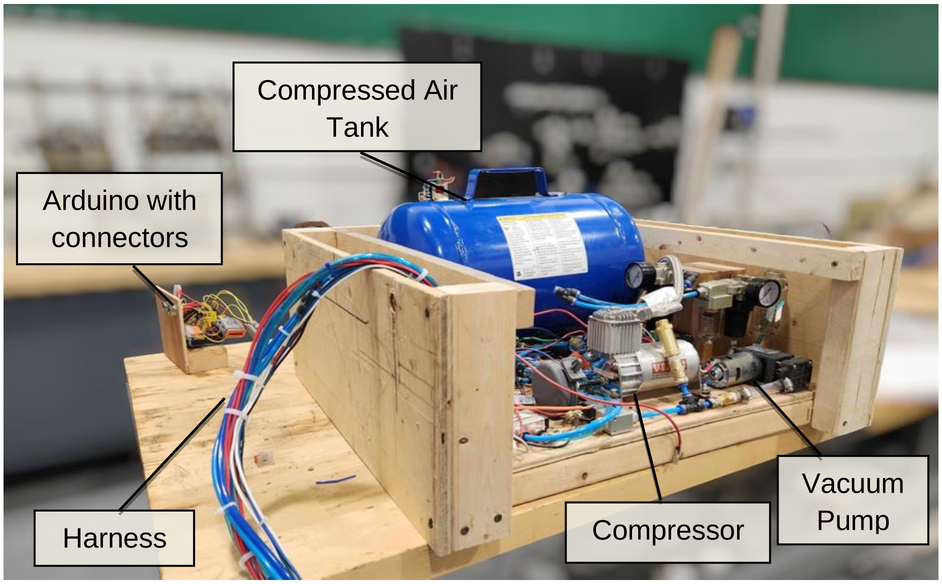


Figure 3.3: Modular housing for electrical and pneumatic components

For manufacturability purposes, as many components as possible are standard off-the-shelf components with common sizes. Components that were manufactured, including those that were 3D printed, were all designed to be machinable in metal using no more than a 3-axis mill and a lathe. However, the tradeoff for longevity and ultraviolet (UV) resistance must be compared to the lightweight, cheap cost and ease of manufacturing provided by fused deposition modeling (FDM) printing in plastic.

3.1.1 Penetrometer

Penetration resistance was measured by first driving a porous metal probe with a conical tip into the soil. A 30° cone with a 20 mm (0.798 in.) diameter was selected because it is the ASABE standard and is commonly used for soil sensing (ASABE, 2019; Rooney and Lowery, 2000). Additionally, these dimensions were those of the FieldScout SC 900 Soil Compaction Meter used for validation (Spectrum Technologies, Inc., Aurora, IL, USA). The porous probe was purchased from AMS, Inc. (American Falls, NY, USA) and originally featured pores at four heights (AMS, 2024). This was reduced to one layer to aid in cleaning (see Section 3.1.2).

Four pneumatic piston cylinders provided the driving force while maintaining alignment. Compacted soils can offer resistances of 2.76-3.45 MPa (400-500 psi) so it is necessary for a penetrometer system to supply forces that can reach at least 4 MPa (McKenzie, 2010). However, for this prototype, the piston cylinders were sized to provide sufficient force to penetrate soil offering 2.5 MPa of resistance, based on equation 3.1. This pressure was found to be sufficient for penetration of soils in nearby fields after a rainfall event and was the temporary target for this prototype based on available hardware.

$$P_{tip} = \frac{(P_{air} \cdot \pi \cdot d_{bore}^2) + F_w}{\frac{\pi}{4} \cdot d_{tip}^2} \tag{3.1}$$

where,

d_{bore} is the diameter of the piston bore (mm),
d_{tip} is the diameter of the cone base used to determine the cone area (mm)
F_w is the weight of the sensor (N),
P_{air} is the regulated pressure supplying the pistons (MPa), and
P_{tip} is the pressure total pressure at the cone tip (MPa).

P_{tip} is a result of the forces of the four pistons, F_p, and the force associated with the weight of the sensor, F_w, divided by the area of the cone tip's conical base. The pistons have 16 mm diameter bores that run at P_{air} = 620 kPa (90 psi). Based on the mass analysis in the computer-aided design (CAD) program, the weight of the sensors was estimated at around 200 N. The pressure in the cylinders was maintained using a pressure regulator and measured with a transducer. Since the pressure in the pneumatics is held constant, the maximum force is constant. The resistance is instead measured by the change in speed as the probe is inserted deeper into the soil. The speed is measured using a linear potentiometer that extends 200 mm down to the height of the tip of the probe. The position is measured with respect to time so that velocity can be calculated. These components are illustrated in Figure 3.4.

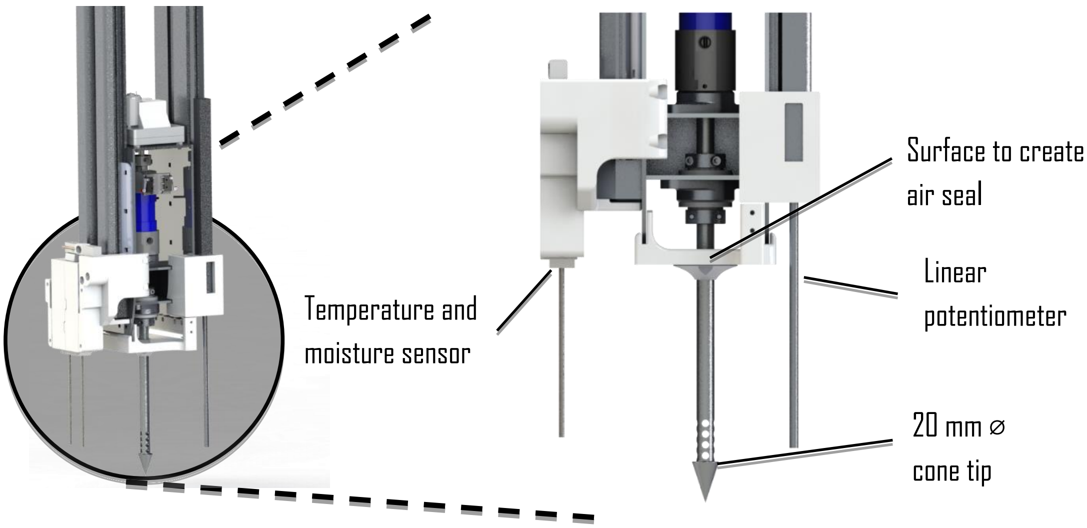


Figure 3.4: Penetration components

3.1.2 Soil-metal friction

When measuring the force required to press a metal probe into the soil, the penetration resistance is confounded by the soil-metal friction as it is inserted (ASABE, 2019). Therefore, to obtain a valid measurement of penetration resistance, both the soil mechanical impedance and the soil-metal friction must be measured. To do so, the probe was spun along its axis using torque from a compact-geared DC motor and transferred through a torque sensor (strain gauge). These two components were connected using a misalignment coupler to facilitate smooth engagement. This coupler also held two limit switches (Figure 3.5) preventing overturning in either direction. A maximum angle of rotation of 60° proved to be a good trade-off between having sufficient rotation to dilute the effects of backlash and hysteresis, but not too much so that the wiring and tubing would break. A potentiometer was mounted to the output shaft of the DC motor to measure the angle.

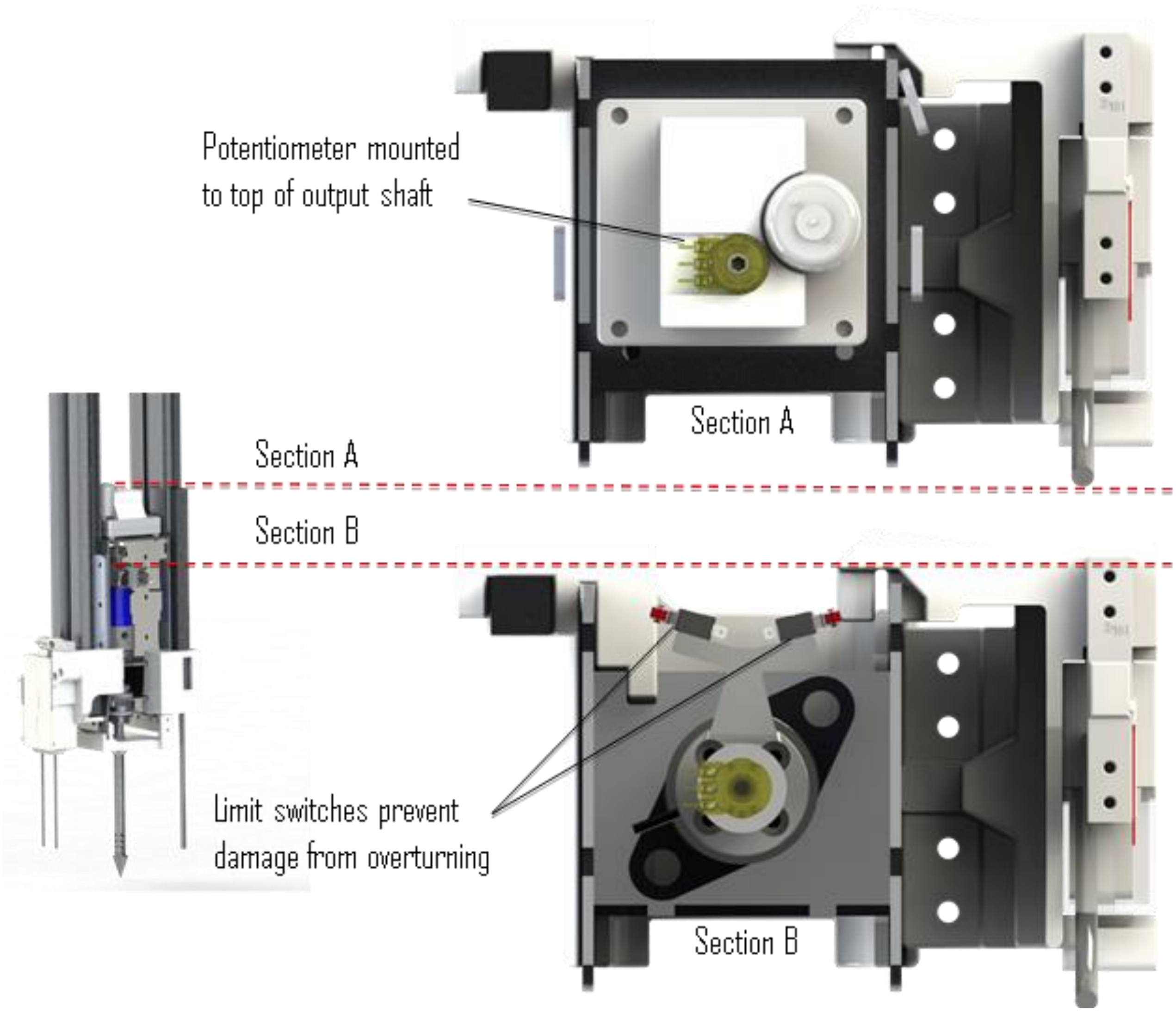


Figure 3.5: A direct top view of the sensor (Section A) and a view of the rotational components with the top panel removed exposing the limit switches and flange bearing (Section B)

The torque needed was estimated empirically by first manually pressing the probe into a high clay-content soil and then turning it by attaching a nut and then turning it with an old needle-style torque wrench. The maximum torque needed was found to be around 3 N·m, so the components were selected with a safety factor of 1.5. A DC motor (Figure 3.6.) was geared to offer a maximum torque of 4.5 N·m (40 in-lbs.), with the torque sensor sized to handle double that, with a maximum torque of 10 N·m (88.5 in.-lbs.).

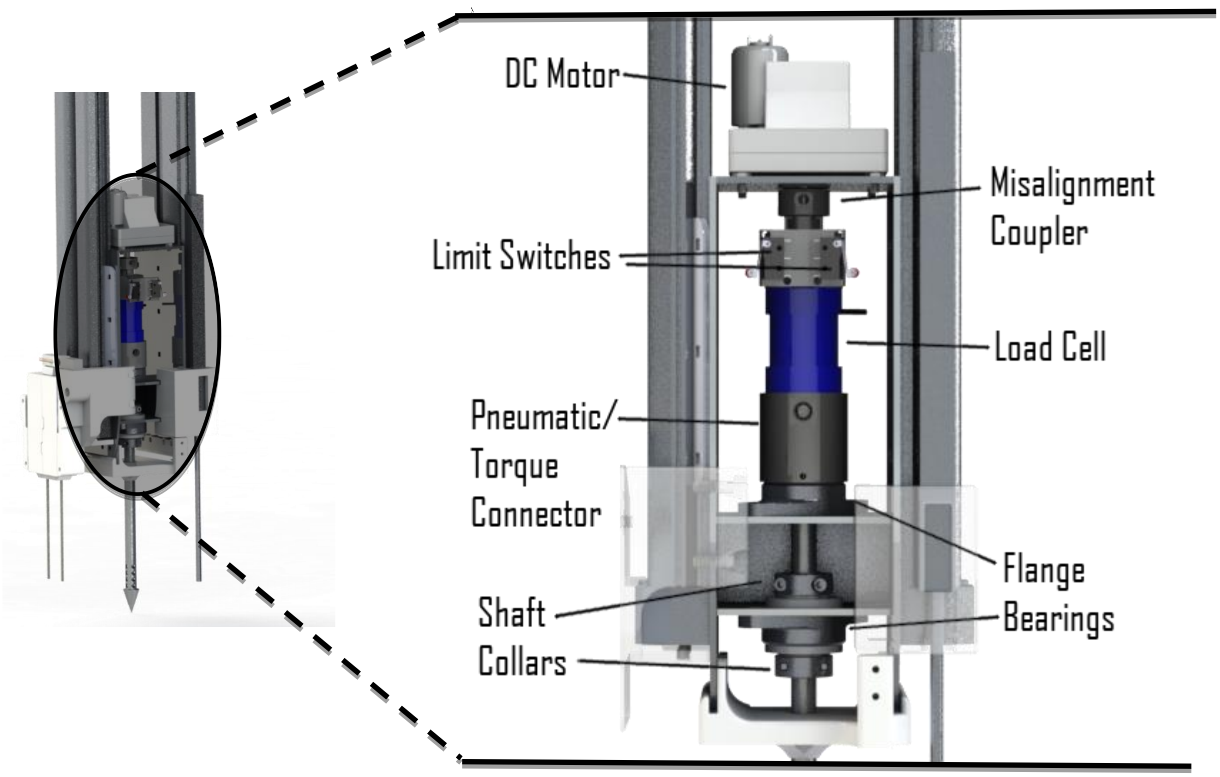


Figure 3.6: Front view of rotational components

Since the probe needs to experience an axial load to lower it into the soil while simultaneously spinning, two steel flanged oil-embedded bearings were used to keep the probe in place. These bearings were selected for their low friction and high tolerability to dust. Each bearing has a respective shaft collar that transfers the load from the sensor to the bearing housing. Between the collar and the flange bearing housing is an oil-embedded thrust bearing (washer) to reduce friction when the probe spins under axial load and a larger washer to ensure that the thrust load is evenly distributed on the face of the flange bearing housing (Figure 3.7).

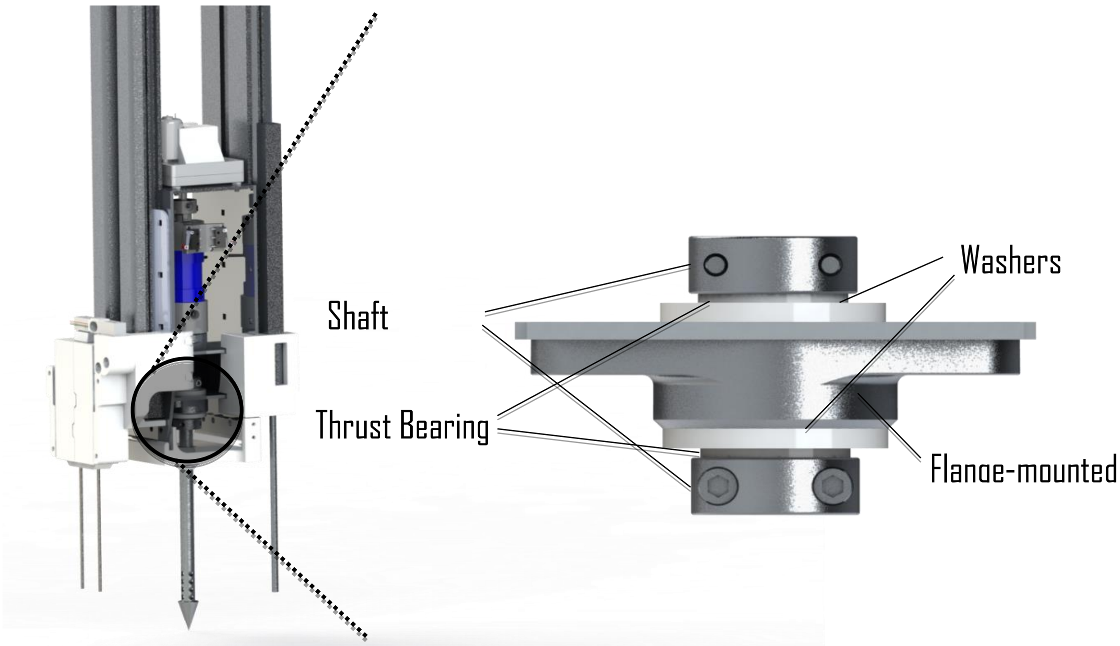


Figure 3.7: Components handling thrust load

The sheet metal piece that holds this flanged bearing is the only sheet metal component of the frame that experiences a load normal to its face. To evaluate the integrity of the design the sheet metal piece with the bearing housing was analyzed in a von Mises stress simulation. When the load parameter was set to double the estimated maximum load it would experience, the stress was far below the yield strength of aluminum (Figure 3.8a), and the displacement resulting from the strain (Figure 3.8b) was negligible.

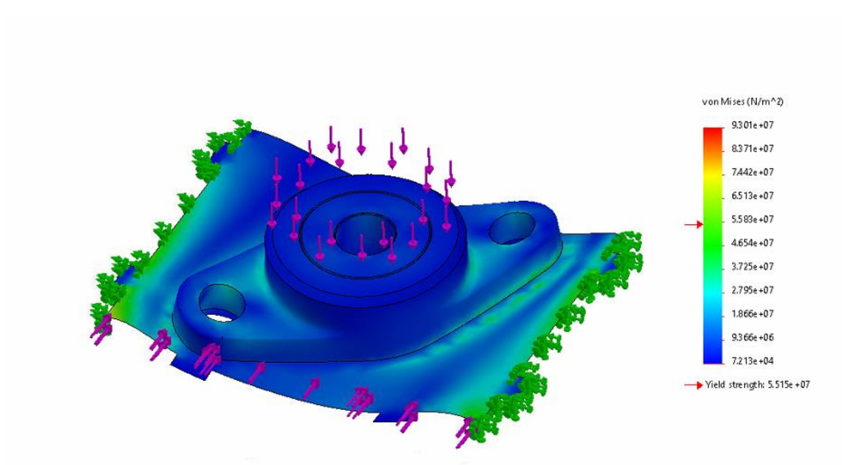


Figure 3.8a: von Mises stress results for flange bearing and plate

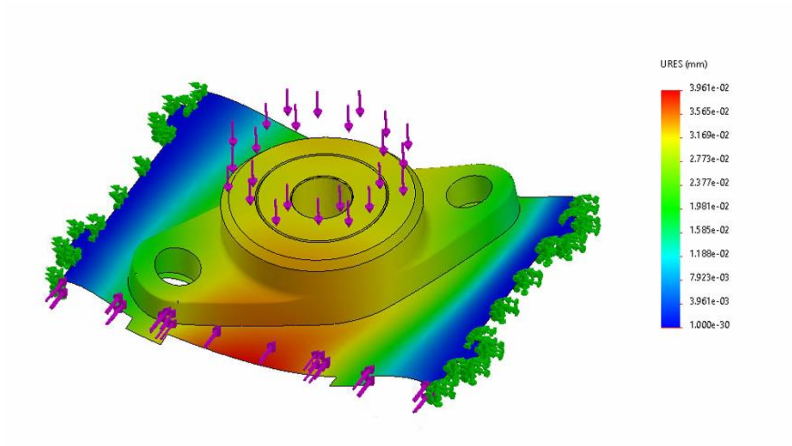


Figure 3.8b: von Mises displacement results for flange bearing and plate

Below the torque sensor is a custom machined piece that both transfers torque from the torque sensor to the probe and makes an air hose connection to the probe. The piece is currently machined out of steel, but other lighter materials may be possible. The connector features a ⅛ national pipe thread (NPT) port to attach an air hose (Figure 3.9). This connector is bolted onto the torque sensor on the top. On the bottom, the connector is threaded with a female ½-20 National Fine (NF) to accept the soil vapour probe. The torque is transferred using a size #10 set screw grub that holds against a machine's face on the probe (Figure 3.9).

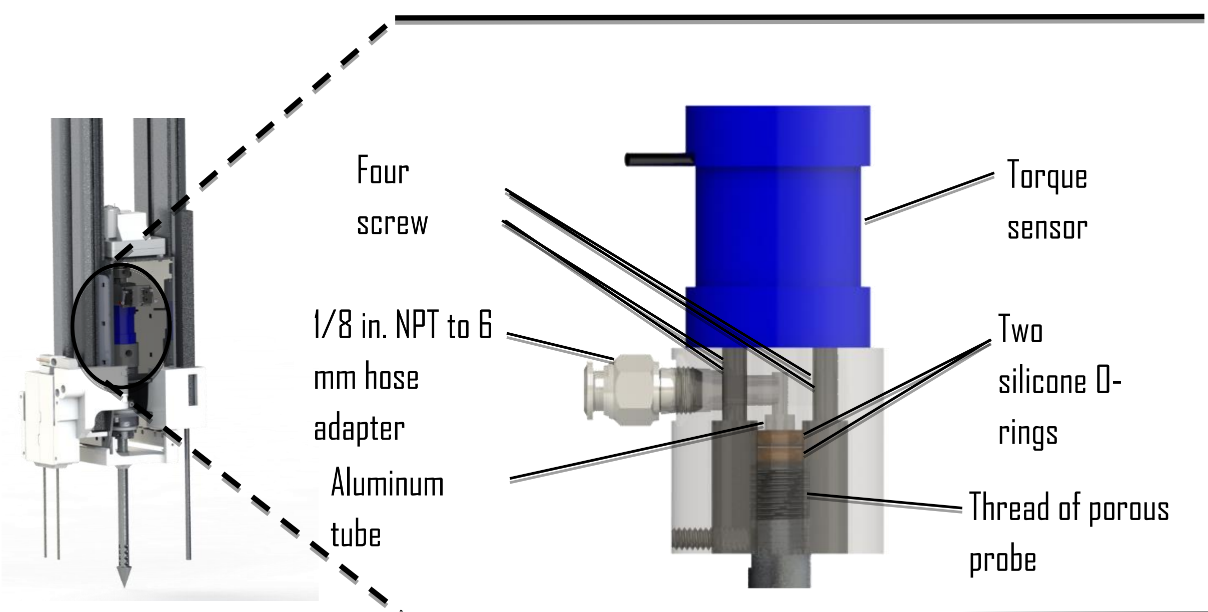


Figure 3.9: Close-up transparent view of custom connector showing attachments to hose adapter and probe

3.1.3 CO_2 Sensor

Measuring CO_2 concentration was accomplished by drawing air from the soil pores using a syringe once the probe was inserted into the soil. The special connector, shown (Figure 3.9), was designed to transfer torque while also providing a sealed pneumatic connection between the probe and the 6 mm hose circuit. This connector uses two 12.7 mm (0.5 in.) outer-diameter thick O-rings that are compressed to form a seal as the probe is threaded. An aluminum tube with an outer diameter of 6 mm (15/64 in.) and wall thickness of 0.3556 mm (0.014 in.) sits inside the O-rings to maintain alignment as they are compressed.

Used to draw air through the probe from the soil's air-filled pores, a syringe powered by a 16 mm bore pneumatic piston, controlled by a 5/2 solenoid valve with flow restrictors, was designed to ensure that the syringe did not actuate too quickly (Figure 3.10). In testing, over-rapid actuation damaged the sensor because of the high instantaneous pressure in the chamber. To ensure that the volume of air drawn in was a magnitude greater than any residual air left in the lines, the syringe was sized to hold 1.1 L of air.

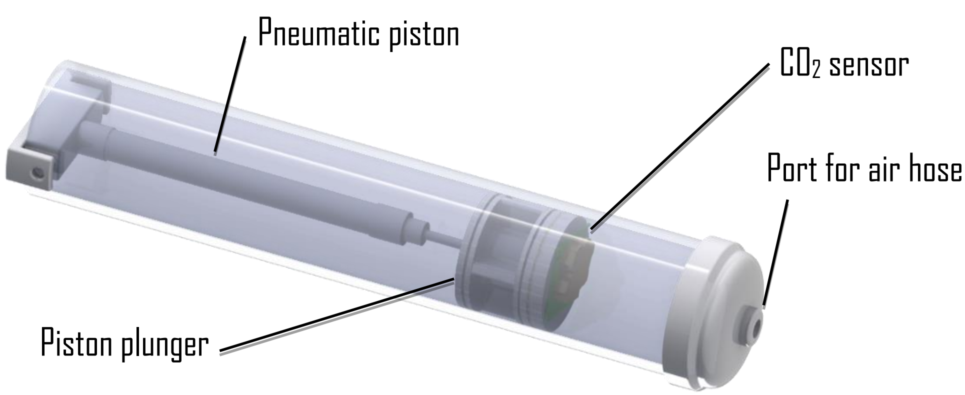


Figure 3.10: Syringe for CO_2 concentration measurement

For CO_2 concentration measurements, a K30 (CO2meter, Inc., Ormond Beach, FL, USA) with a measurement range of 0-10 000 ppm CO_2 was selected and placed inside the syringe (CO2Meter, 2022). The syringe was custom-made to accommodate the large size of the K30 sensor.

The plunger of the syringe (Figure 3.11) was 3D printed and shaped to mount the K30 sensor and allow wires to pass while minimizing leaks. Large x-profile O-rings, selected for their superior performance in environments in motion, were used to create a seal. The plunger was moved using a 16mm bore diameter piston connected to the 620 kPa (90 psi) pneumatic circuit. This piston is actuated by a 5/2 solenoid valve.

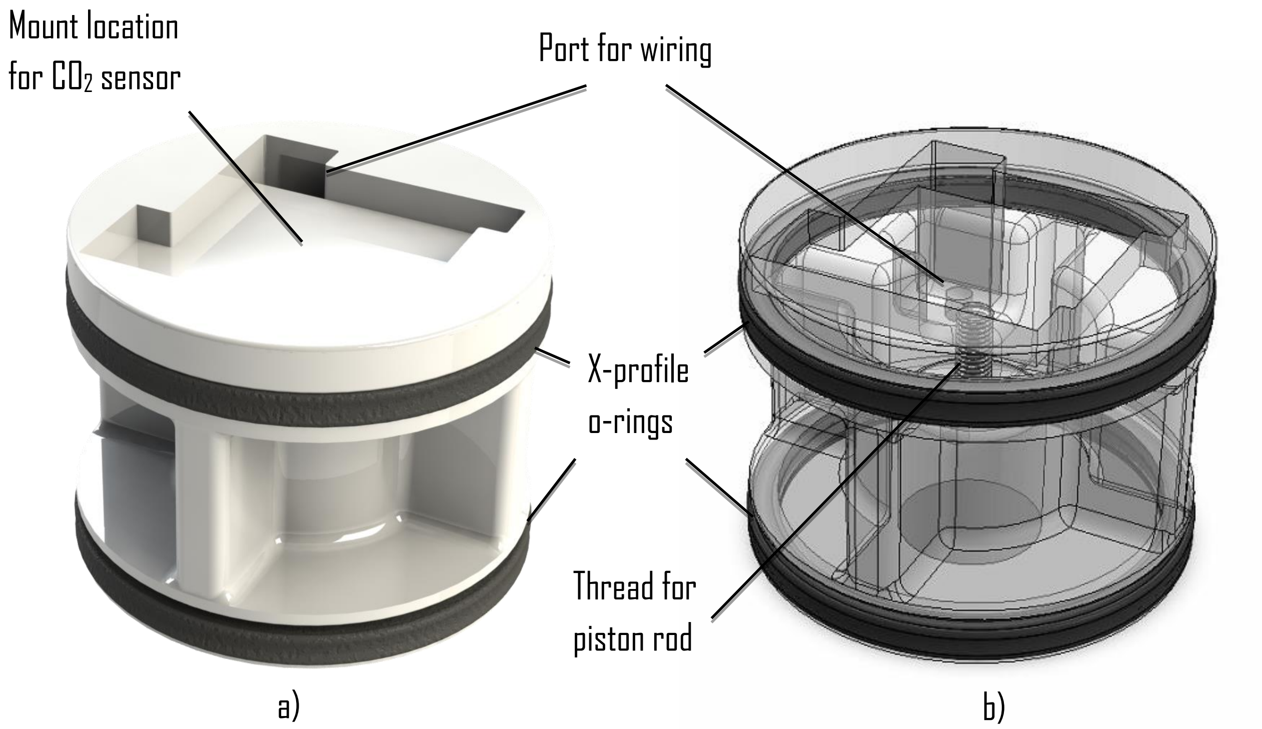


Figure 3.11: Piston plungers that hold CO_2 sensors with a solid view on the left (a) and a translucent view on the right (b)

The K30 CO_2 sensor uses a membrane and an NDIR sensor. Due to the membrane's nature, the sensor's response time is two minutes; however, this response is governed by a first-order equation. Evaluating this first-order response on a shorter timescale should reliably measure the relative CO_2 concentration from different locations.

Finally, the probe is linked to a 2/2 solenoid valve that releases air at 620 kPa (90 psi) to clean the porous probe tip from soil residue. This pressure is regulated on a separate circuit to avoid interfering with the rest of the pneumatic circuit.

The probe was cleaned after each sample by blowing pressurized air through the probe to clear the dirt in the pores. While the porous probe tip originally featured four levels of holes, this resulted in inconsistent cleaning. To address this, the top three levels were covered with aluminum tape to allow passage only through the bottom level of the holes. This configuration allowed for sufficient flow rate for the CO_2 sensor while concentrating the pressurized air for more reliable cleaning. Additionally, it was found that the larger diameter of the cone tip resulted in holes larger than the shaft when penetrating. This made it difficult to seal when taking measurements with the air permeability sensor described in the next section. To create a tighter seal, one version of the probe was lathed down to reduce the cone's diameter. The smaller base area of the cone tip also increased the penetration pressure the probe could exert. When the probe was machined from a diameter of 20 mm (0.798 in.) to a diameter of 14 mm (0.55 in.), the area was halved, doubling the exportable pressure from 2.5 MPa to 5 MPa. The probe configurations, along with the stock configuration, are shown in Figure 3.12. It was later found (see Section 4.3) that the taped cone with the stock diameter cone performed better for CO_2 concentration sampling.

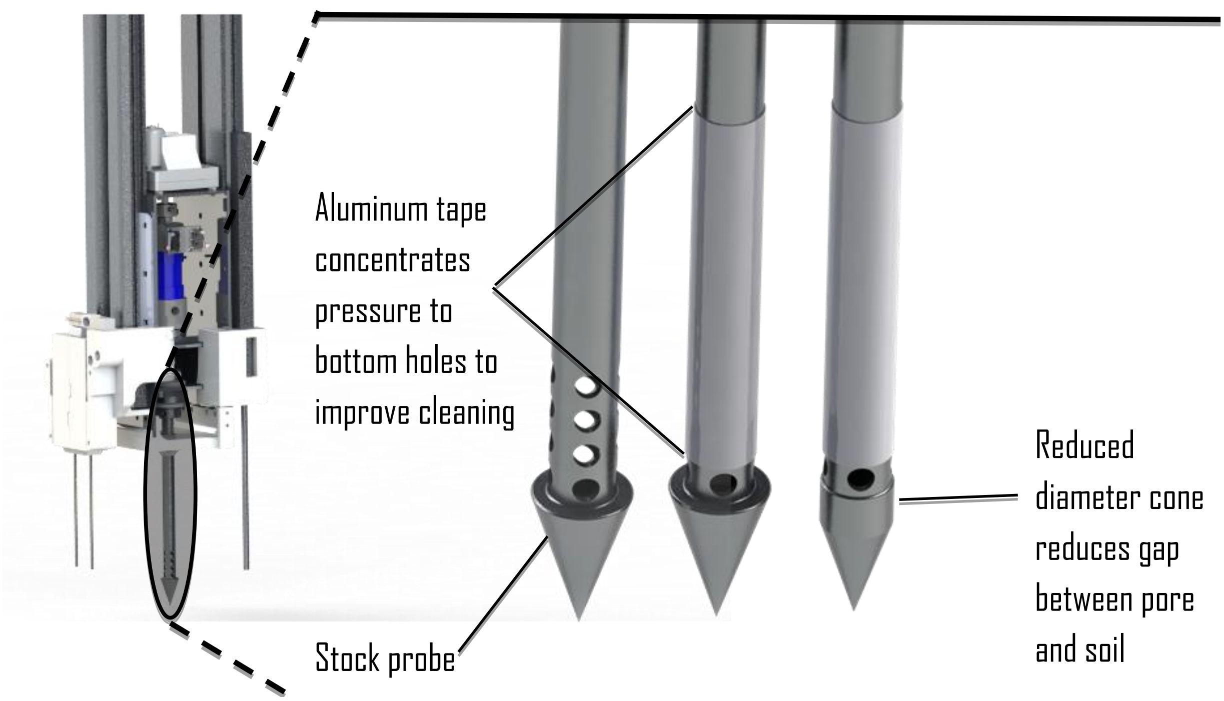


Figure 3.12: Different configurations for probes

3.1.4 Air Permeability Sensor

The air permeability sensor uses the same pneumatics as the porous probe. To measure permeability, air is pumped into the ground using a vacuum pump rated to blow air at 20 L min^{-1} at up to 85 kPa. A voltage regulator supplied the power to this pump to make the speed (and thus flow rate) more consistent. As further described in section 4.4, when operated at its maximum speed/flow rate, the vacuum pump generated pressures that were too high in certain soils. To prevent this, the speed of this pump was modulated by a motor driver through pulse-width modulation (PWM). Finally, the pump was protected from the 620 kPa (90 psi) of the high-pressure cleaning air by a 2/2 pneumatic solenoid valve since this pressure exceeds the pump's rated value (85 kPa).

3.1.5 Moisture, EC, and Temperature Sensor

Since CO_2 concentration was affected in the short term by changes in temperature and θ, it is important to measure these simultaneously. Selected as a high-quality soil reflectometer that also measures temperature, the CS655 (Campbell Scientific, Inc., Logan, UT, USA) was not as prohibitively costly as its counterparts that used time-domain reflectometry (TDR), even though this alternative may have provided more accurate results. The sensor runs on 12VDC, a voltage available on most consumer vehicles. Moreover, it communicates using SDI12, a two-way serial communication protocol that uses a single data wire. While this was convenient to wire, it did appear to result in some distorted serial messages. Finally, the CS655 was selected over the more commonly used CS650 because the former's probe length was 120 mm, which lined up with the protruding length of the porous probe.

The mount for the CS655 was designed to propel the sensor into the ground using the force from a cantilever load cell. This is done by putting the sensor in a case made of low-friction material; in this case, polylactic acid (PLA) plastic. The bottom of the case where the temperature sensor resides was left exposed. This case was placed in the mount (Figure 3.13) which allows the case to slide upwards, but not downwards. The upward motion is instead limited by the addition of the load cell which provides the downward force.

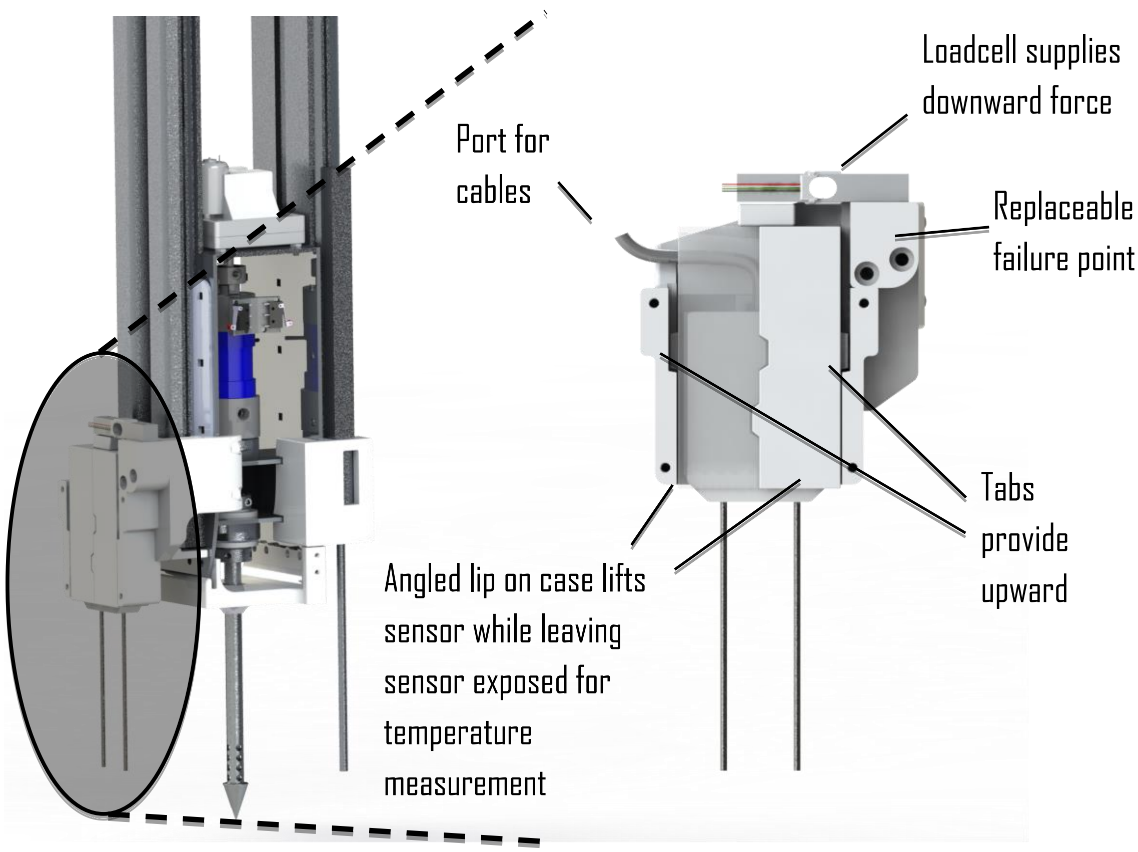


Figure 3.13: Moisture and temperature sensor assembly

Since the CS655 is propelled by the same cylinders that propel the porous probe, the force to put it in the ground interferes with the penetration resistance readings. The loadcell measures the load made by the CS655 so that it can be accounted for as a confounding factor when measuring penetration resistance. This load measurement also helps to protect the sensor if it strikes a hard surface that could otherwise damage it. When the CS655 experiences a load that is too high, pressure is relieved from the rodless pistons that lower the sensor to prevent damage. There is an additional mechanical failsafe where the load cell will shear off at an easily replaceable failure point (Figure 3.13) of the mount if the software failsafe malfunctions.

3.1.6 Pneumatic Circuit

The plumbing for pneumatic actuators includes a compressor sized to provide enough flow rate within its recommended duty cycle. It was outfitted with a pressure switch that turns on at 620 kPa (90 psi) and off at 896 kPa (130 psi). This compressor feeds an 18.9 L (5 gal) tank, selected to supply sufficient air at 620 kPa. Connected to the tank is a pressure relief valve 1034 kPa (150 psi) for safety and a pressure gauge for user convenience. From the tank, all air goes through a condenser and oil filter to prevent contaminants from entering the plumbing and damaging the pneumatic components. This air is then split into two circuits: one at 620 kPa (90 psi) that feeds the pistons, and one at 689 kPa (100 psi) used to clean the probe.

The side that runs at 620 kPa passes through a contact width flow restrictor to lower the speed of the pistons when actuated. When unrestricted, the pistons that lowered the sensors were found to move too quickly to achieve consistent penetration resistance readings. The piston in the syringe retracted enough that the sudden pressure changes in the syringe damaged the CO_2 sensor. A 5/2 solenoid served to dictate the two positions of the syringe to control the piston in the syringe. Additional adjustable flow restrictors were attached to the exhaust ports of the solenoid valve to further fine-tune the syringe’s draw and expel speeds.

A mono-stable 2/2 pneumatic solenoid valve served to control the airflow for lowering and raising the sensor, cutting off the pressure so that the sensor neither raises nor lowers when powered off. Next in line is a mono-stable 5/2 pneumatic solenoid valve that controls the direction of the pistons that raise and lower the sensor. The valve raises the sensor so that if power is cut off, the sensor safely raises using whatever residual pressure is left in the lines. The speed is also regulated by flow restrictors in the exhaust ports of this valve.

The circuit part at 689 kPa is used to clean the probe tip. Closer to the probe is a 2/2 solenoid valve that allows a burst of pressurized air to blow through the probe tip, cleaning it. Between the regulator and the solenoid valve, the normal air hose is split into four parallel air hoses that increase the volume in the lines. This acts as a chamber of sorts and was found to allow a more sustained and better cleaning burst.

Finally, an air hose was connected from the probe tip to the syringe, and between the two, a 2/2 pneumatic solenoid valve protected the syringe and its CO_2 sensor from the high-pressure cleaning air burst. A 2/2 valve also connected the syringe to the atmosphere, which helped with venting the syringe between measurements. An overall schematic of the pneumatic system is provided in Figure 3.14.

3.1.7 Electrical Connections

The electrical system (Figure 3.15) can be powered by a single 12VDC power source, making it compatible with most vehicles. Step-down converters and voltage regulators are used to power lower voltage sensors. All these direct current (DC) systems share common ground. An Arduino Mega is used to automate the sequence by controlling the actuators and stream sensor data. This data is output over a serial connection to a serial-capable device such as a personal computer (PC). As many bytes must be transferred quickly, a default baud rate of 115200 was implemented for this connection. The PC is responsible for saving/logging the streamed data and indicating to Arduino Mega when to start a measurement sequence and how to name that sequence. This simple interface design is useful since it complies with many open-source programs that read serial data (*e.g.*, PuTTY and Tera Term). However, custom programs can also be written depending on the user's needs.

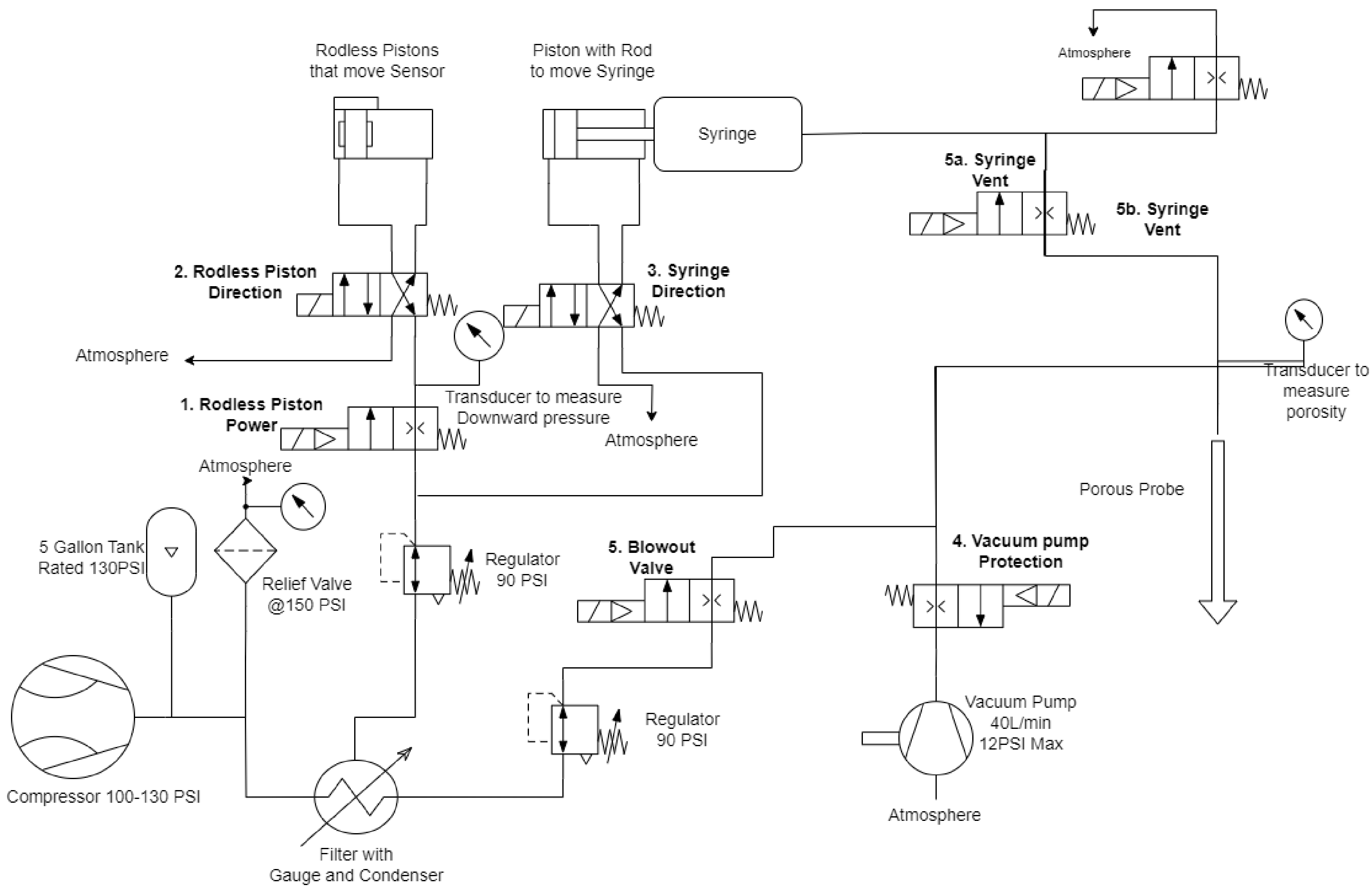


Figure 3.14: Overall schematic of the pneumatic system

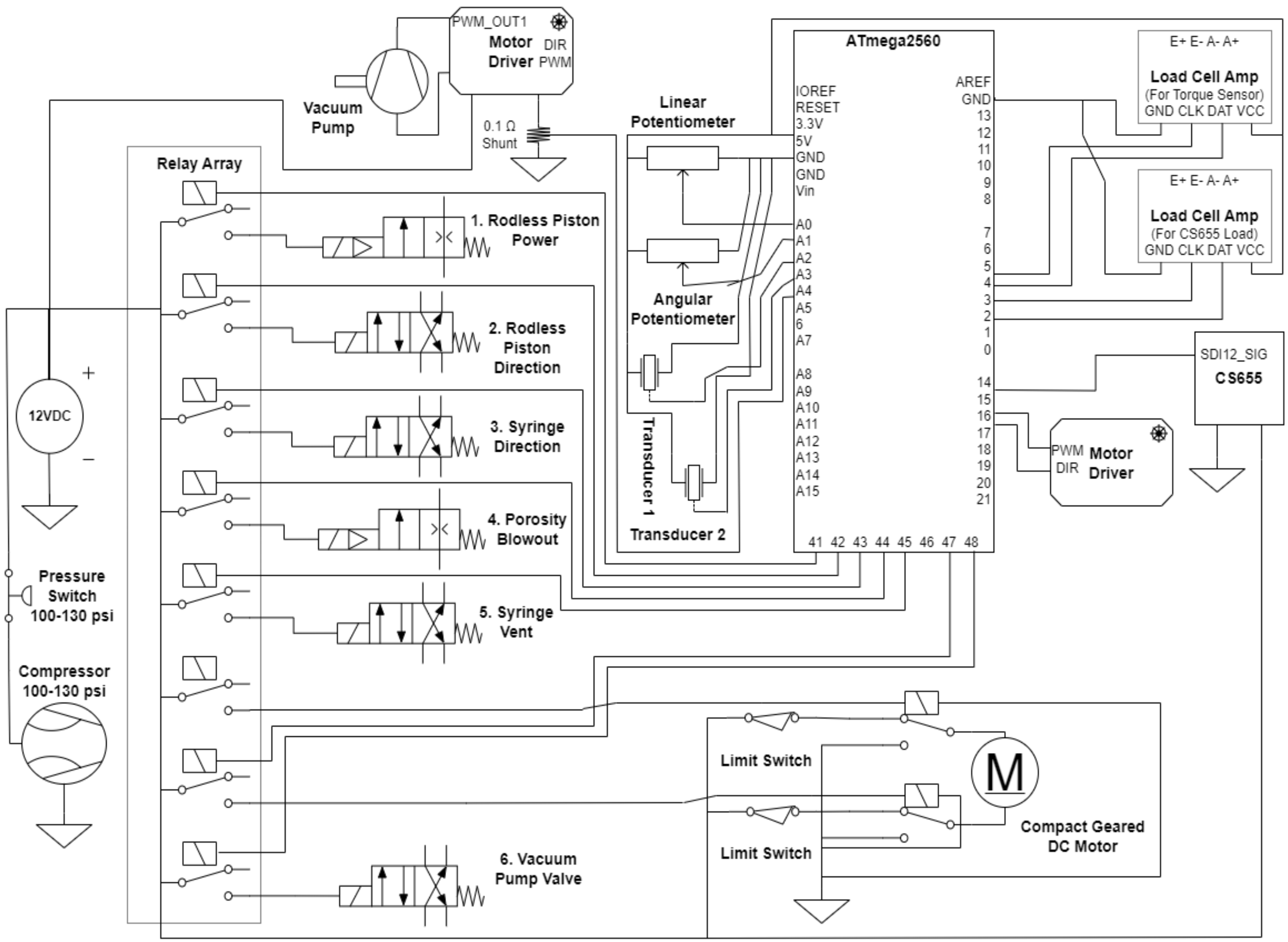


Figure 3.15: Overall schematic of the electrical system

As previously mentioned, default states are selected for safety reasons, so when power is disconnected, the sensor returns to a safe state. The Arduino Mega controls the 12V systems through 5V relays and PWM motor drives that can be actuated on a 5V system.

The 5V relays can handle 12V, have built-in flyback diodes, and actuate the 12V pneumatic solenoid valves and automotive relays. Lacking built-in flyback diodes, these larger solenoid valves and automotive relays were found in the testing phase to interrupt the Arduino Mega's processing and, in some cases, cause it to reboot. To alleviate this inductive interference, flyback diodes were soldered close to these components (not depicted in the figure).

The motor drivers accept a PWM input to control the speed of a motor (PWM output/voltage). This capacity served to control the speed of the vacuum pump used to pump air into the ground during a φ measurement. A speed controller was necessary because, in some soils, running the vacuum pump at a full 12V generated too much pressure, which then needed to be relieved. However, in other soils, running at a higher speed was necessary to generate sufficient pressure to make a measurement. A 0.1Ω shunt was also wired in line to the ground of the vacuum pump power supply to measure the current drawn when pumping air.

The 5V system is primarily used to power and read sensors. Since the testing was done primarily on a UTV where the electronics were far from the sensor, the long wires were vulnerable to inductive interference from the higher amperage 12V cables that actuated the DC motor and pneumatic solenoid valves. To avoid this, the 5V signal wires are bundled separately from the 12V power supply wires and wrapped in a metal braid grounded to the common ground shared by all the DC systems. Additionally, shielded wire was used for analog and high-frequency digital signals where possible. The shielding for these wires also shared common ground.

3.1.8 Sequence

The Arduino Mega has two main function loops. An initial sequence that runs on powerup and another sequence that runs whenever a new measurement is requested.

Initial Setup Sequence

The initial setup sequence (Figure 3.16) starts by initiating the serial connection between the PC and Arduino Mega. It then connects to the SDI-12 serial connection of the CS655 sensor and the UART serial connection of the K30. It waits until both run before sending a confirmation message and proceeding. It then ensures that all actuators are in their low/safe state by setting all the relays to their normal positions. Afterward, the sensor is raised to ensure enough pressure in the cylinders to keep the sensor up. Finally, air is blown into the probe at high pressure to clean it. The sensor sends a ready message when this is complete.

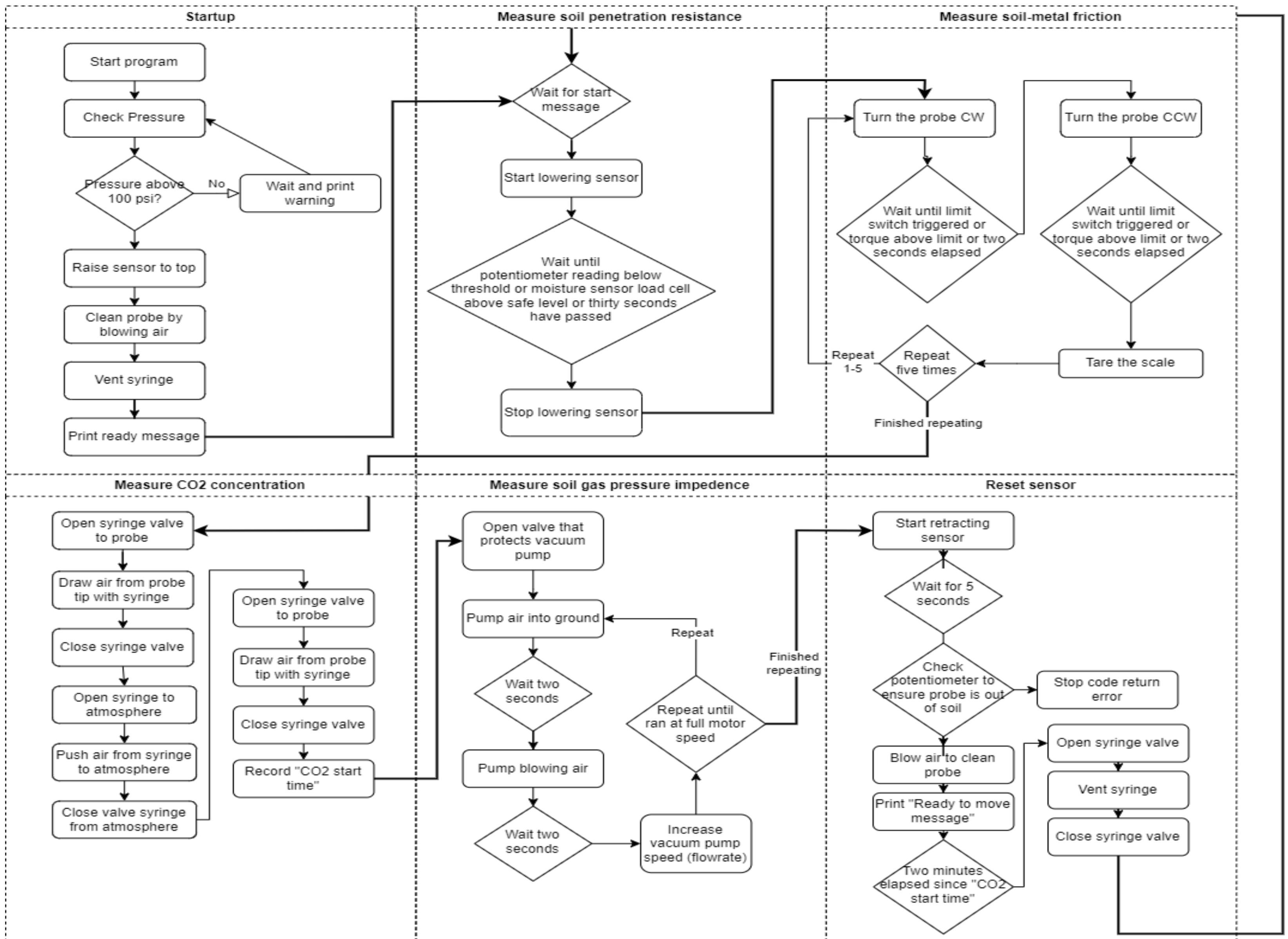


Figure 3.16: Sequence of operation, including Arduino Mega’ s two main function loops

Looped Measurement Sequence

Once the sensor is ready, it waits for any message that begins with an exclamation mark (!) and ends with a carriage return and a new line character. These two requirements act as rudimentary error detection to check if the command was sent by the user and not from random noise. When the message is received, the message is sent back to the PC so that the message can be logged as a comment or the name of the sample. Readings begin streaming as frequently as possible as comma-separated lines terminated with a carriage return and a new line character.

Using a serial streaming program such as PuTTY (Figure 3.17), this can be viewed. Each comma-separated value is indexed to represent the current sensor value such as for the CO_2 concentration sensor or another value of interest. The first value is the OS time (how long the Arduino Mega has been on for). The proceeding nine comma-separated values are measurements from the sensors. The penultimate value describes the current step, which can be seen in real-time to indicate progress. This value is also used in the data post-processing to take values from specific steps. The last value is the time since the last line of values. This is used to diagnose interruptions quickly.

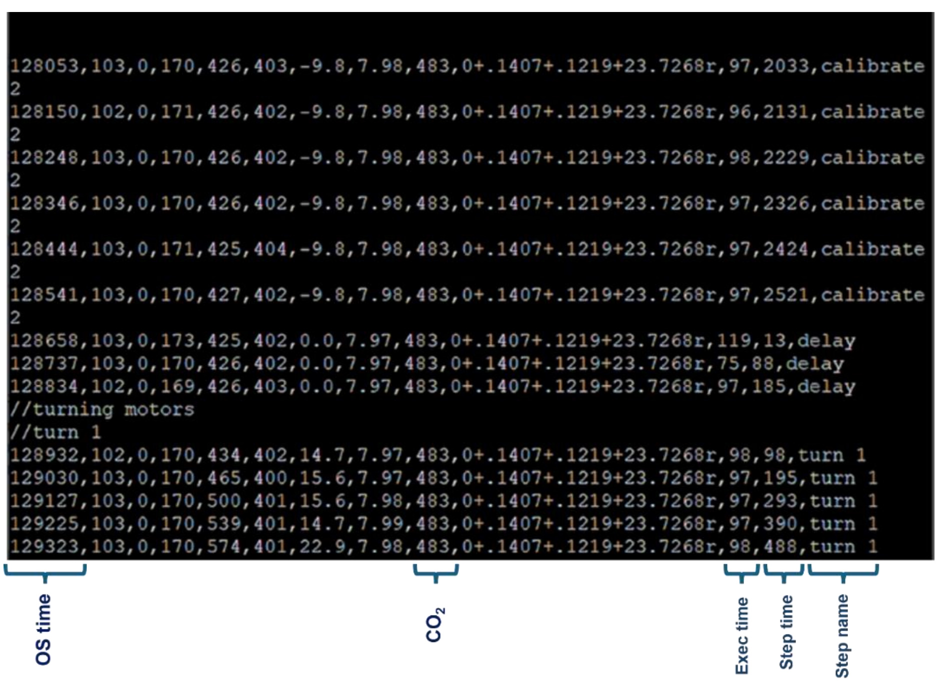


Figure 3.17: Output data as seen live from a serial interface (PuTTY)

These values are continually reported to the PC as the measurement sequence steps are followed. First air is again blown to clean the probe, then the sensor begins to lower. The lowering ends when the linear potentiometer is bottomed out; the CS655 load cell measures a load that is too high or times out after thirty seconds. The static torque sensor is tared. The probe is then spun five times to quantify the friction between the probe and the soil. In harder-to-penetrate soils such as clay, this step helped penetrate the soil more deeply and achieve a consistent sampling depth. An individual turn is stopped early if the torque exceeds a specified limit to prevent damage to the load cell. Air is drawn in by retracting the syringe. After a delay, the valve to the syringe is sealed. Air is blown into the ground using a regulated voltage/speed vacuum pump. This step's pressure response to the fixed flow rate is of key interest. The sensor is raised out of the soil. In some scenarios, the probe is spun twice to help dislodge it. Since the K30 CO_2 sensor has a 2-minute response time, pausing some fraction of that amount is specified to allow the reading to respond. Finally, the CO_2 sensor is vented 5 times using the following sequence: Air is purged from the syringe by opening a valve from the syringe to the atmosphere and squeezing the syringe. This is followed by a short blow of atmospheric air from the compressed air tank. The valve from the syringe to the atmosphere closes and then the valve between the syringe and probe opens. Air is then drawn from the probe end to the syringe.

Data processing

After the streamed serial data from the Arduino Mega has been logged, preliminary formatting of the data is performed by a Python script. This script compiles the data into two files:

(*i*) an Excel (*.xlsx) file with each point having its own sheet. Each sheet is named based on the string sent to the Arduino Mega when the sequence was started. The sheet's name was suffixed by a new, unique index to ensure that each sheet had a different name. The columns in the sheet each bore a different value output by the Arduino Mega, in the same order that it was output, starting with the time and then followed by each sensor’s measurements. The rows represent the values as time progressed.

To simplify data processing, certain excerpts from the raw output are featured in the following columns. These include the values of the linear potentiometer as the probe penetrates the soil, the values of the torque sensor when the probe turns, and the values of measurements when air is being pumped into the ground.

(*ii*) A comma-separated values (CSV) file holds some calculated values that are meant to be importable into GIS software for plotting. Each row represents a different point, and each column represents its respective computed value. These values include θ measurements when in the soil, maximum CO_2 concentration measurement reached, probe speed when penetrating soil, pressure when measuring φ, and maximum torque when turning the probe, etc.

3.2 Testing

3.2.1 Initial validation of sensors

Air permeability measurement

Some initial testing was done indoors using a bucket of soil. This was useful since the ground outside was frozen in the winter months. Additionally, the bucket allowed one to control soil's ρ_{dry}. The probe was subjected to soils with different designed resistances to validate the porosity sensor.

Under certain conditions (*i.e.*, when the soil was moderately dense and very sticky), the seal was so complete that the vacuum pump would run at a pressure higher than it was rated for. A pressure relief valve was added to protect the pump to relieve pressures beyond 70 kPa (10 psi). However, this would result in measurements hitting a cap, resulting in different soils with high pneumatic impedance having similar measurements. A motor driver was added to modulate the speed (and therefore flow rate) of the vacuum pump, allowing for varied measurements in soils with high pneumatic impedance.

Soil friction measurements

Calibration of the soil friction measurements was done using an old-style needle torque wrench. The sensor was subjected to torque until it reached the marking at 5 ft-lbs. (6.78 N·m). Two-point calibration was done by recording the sensor reading at this point and the second point was the initial point under no torque.

Field and Point Selection

Field and within-field point selection provided many conditions to assess if the sensor would respond appropriately. Accordingly, fields were selected for higher spatial variability, as well as accessibility/availability. Field 26 on the Macdonald Campus of McGill University was known to show high spatial variability. It features sections with changes in elevation, poor drainage, and different soil types. It is also safely accessible by a campus road, given that the UTV was being operated as a slow-moving vehicle. However, owing to weather and logistical issues with the farm, this field was only partially sampled.

Another nearby field, Field 30, was similarly accessible and fully harvested. It did not, however, have the levels of spatial variability that Field 26 was known for, but still had some. These fields and their electrical conductivity are depicted in Figure 3.18.

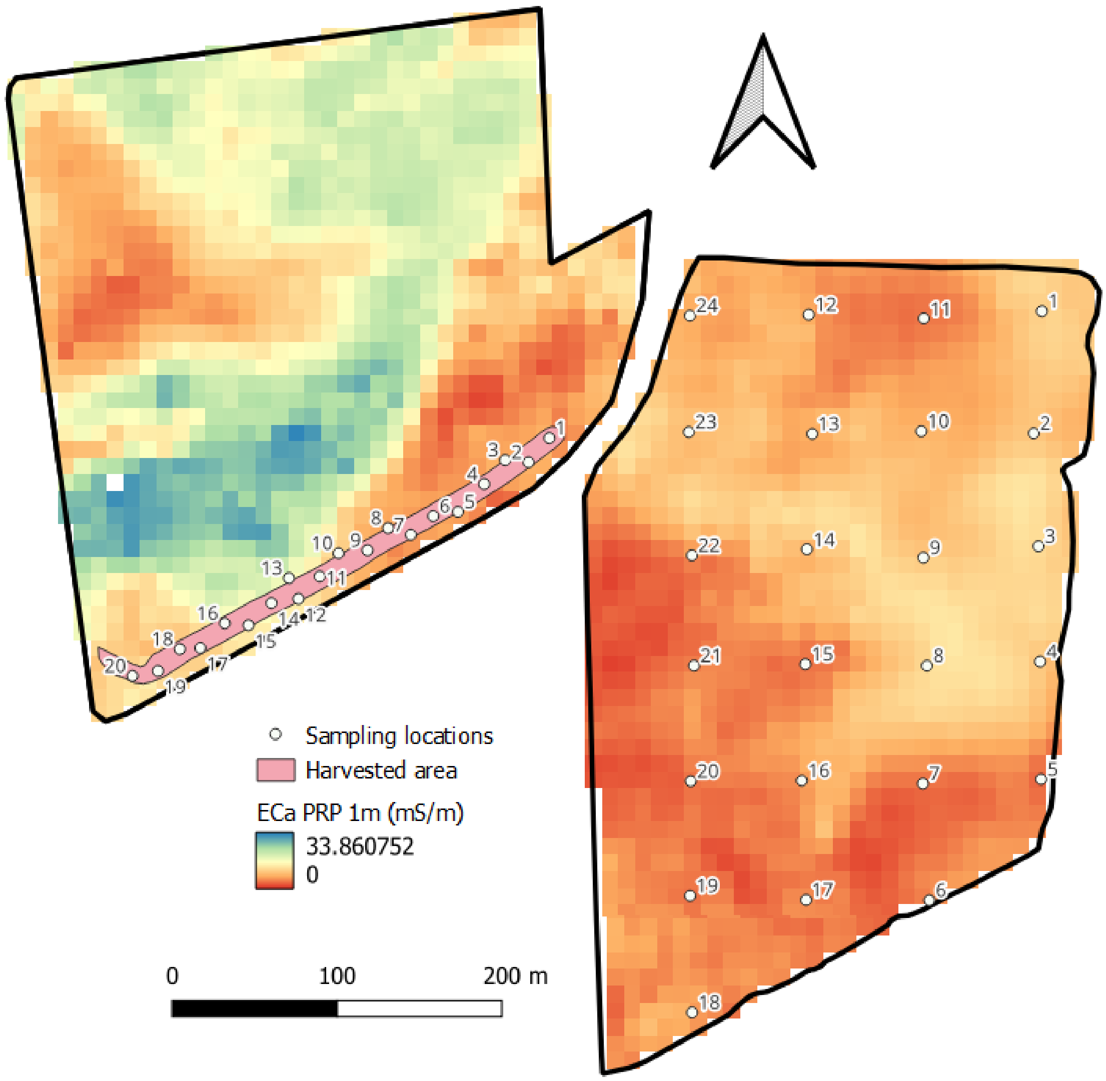


Figure 3.18: Numbered sampling locations with rasterized ECa map for Fields 26 (left) and 30 (right), on the Macdonald Campus of McGill University

3.2.2 Reference measurements

Since the purpose of this design was to provide a convenient alternative to current conventional soil sampling methods, some such methods were selected to be tested alongside the sensor array.

FieldScout SC 900 Soil Compaction Meter

Soil mechanical resistance is an important indicator of soil health, which is linked to soil compaction (bulk density) for a specific soil series under specific soil water content. Spectrum Technologies provides a meter that measures cone index (soil penetration resistance) using a probe with a force sensor and an ultrasonic sensor. The ultrasonic sensor allows the meter to relate measurements to different depths, offering a profile of resistance values (Spectrum Technologies, 2024a).

FieldScout TDR 350 Soil Moisture Meter

Spectrum Technologies also sells a moisture meter that uses TDR to obtain more accurate volumetric θ (θ_{vol}) measurements. It is important to measure θ since it may affect soil health indicators in the short term. This sensor was selected since it was ergonomic and easy to use without significantly increasing the time necessary to sample a field. It also provided a secondary measurement to compare with the oven drying method (Spectrum Technologies, 2024b).

Oven-dry Bulk Density and Moisture Measurement

The oven drying method was implemented by removing a soil core of a specific volume from the ground, measuring its mass wet, and then drying. This was important for θ since it directly measures the water content in the soil instead of using its electrical characteristics as a proxy. The oven-dry method also yielded a value for ρ_{dry} , a parameter related to compaction.

For the ρ_{dry} measurement, a core sampler consisting of a steel cylinder that could be hammered into the ground with a mallet was used. A wooden plank was held between the hammer and the cylinder to evenly distribute the impulse from the hammer and reduce the likelihood of the cylinder being hammered too deeply, thus compressing the sample.

Compression of the sample is the greatest factor in variability amongst soil core samples (Raper and Erbach, 1987). To prevent compression a wide (51 mm — 2 in.), 50 mm tall cylinder was selected (total volume = 409.7 cm^3. This was sufficiently large to reduce any effects of small variations in the soil. Additionally, the cylinder was beveled on the bottom outside edge to reduce the compression on the core sample inside (Erbach, 1987).

3.2.3 Field data collection

All components and tools were assembled and loaded onto the UTV (Figure 3.19). The sensor array was mounted onto the custom front hitch. The box housing the electronics and control components was placed in the bed. The tools for sampling; trowel, sledge, wood plank, cylinder, Ziploc® (Racine) resealable bags, compaction meter, and moisture meter were kept in the bed for convenience.

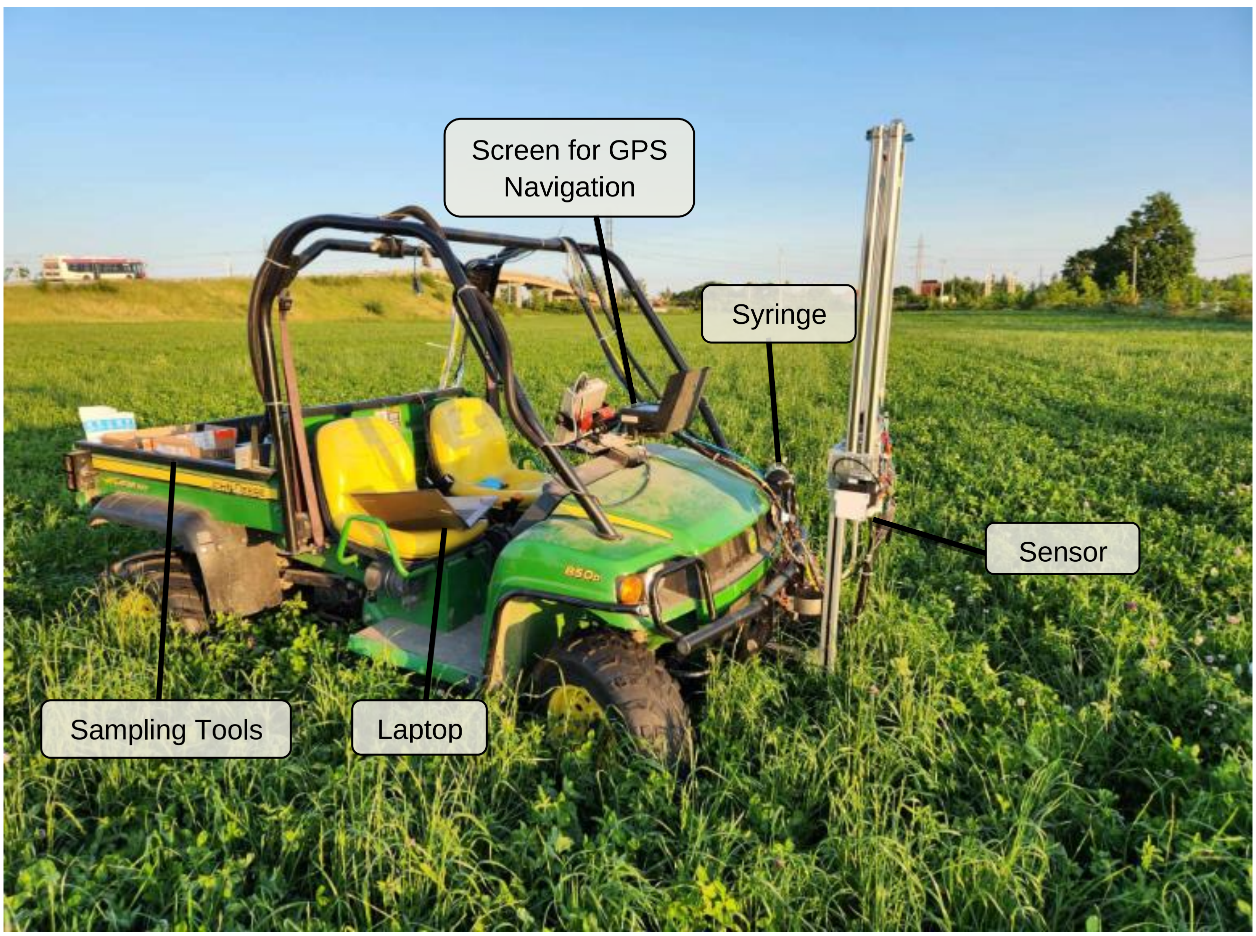


Figure 3.19: UTV with testing setup

1. The UTV was driven to the first point as selected in the previous section. A convenient open-source mobile application, 4Farm (4 Farm, Cerro Largo, RS, Brazil) was used to load the points and keep track of which ones were sampled.
2. On arriving at the sensor location, the number of the point was sent as a message to the Arduino Mega of the sensor. This started the sampling sequence and marked which point it was for future reference.
3. A more precise GPS location was recorded by placing a GPS receiver directly above the sensor probe.
4. The soil core sample was taken using the following steps:
    a. A spot within 0.6 m (2 ft) of the probe that had visibly similar soil conditions to the spot where the probe was penetrating was selected (Figure 3.20.)
    b. The cylinder was hammered into the ground using a sledge and a wood plank until its top was just level with the soil surface.
    c. The soil surrounding the cylinder was dug out until level with the bottom of the cylinder.
    d. The cylinder was removed, placed into a resealable bag, and sealed. This bag would be labeled following the formula of Field-Date-Point-Depth. For the first core of each point, the depth would be marked as “0-5” since the cylinder’s height was 0.05 m.
    e. With the soil level now at 0.05 m depth, the process was repeated at this depth to produce a sample over the 0.05-0.10 m depth.


Figure 3.20: Core sampling process showing the distance from probe sampling location

5. The other member collected the moisture meter reading simultaneously. This was done in a spot within 0.6 m (2 ft) of the probe.
6. The compaction meter reading was then taken; however, since there existed a significant amount of crop residue, the sampling individual had to step on the corn stalks to prevent them from interfering with the compaction meter's ultrasonic sensor.
7. By the time these manual measurements were taken, the sensor would have been lifted from the ground.
8. Proceed to the next point and repeat the process until all prescribed points have been sampled.

3.2.4 Oven drying

After the soil core samples were collected in the field, the resealable bags holding the cores were immediately weighed, and the weights were recorded. When drying, the samples were removed from the resealable bags and then placed into metal trays. The emptied resealable bags often still bore some residual moisture or soil and so were kept and left open to the air to dry under radiative heat lamps. Meantime, the soil cores in the metal trays were placed in welding rod ovens set to 105°C and dried for 24 hr. The soil cores were removed from the oven and returned to their respective resealable bags, which were now dry. Individual resealable bags full of dry soil were weighed to within 0.1 g using a digital balance. To calculate the dry weight of soil alone, the weight of an empty resealable bag (11.3 g) was subtracted from both the wet weight and dry weight of the samples.

3.3 Integration with a Field Robot

The prototype was originally tested on a UTV that had an available bed where the electronics box could be mounted. When moving the sensor array to the Bearcub-24, the electronics box was mounted directly to the hitch (Figure 3.21). The stronger suspension of the Bearcub-24 was able to handle the additional weight, which also added more stability when the sensor was penetrating. To hold the box, additional framing was made from aluminum extrusions and added to the hitch (Figure 3.22.)

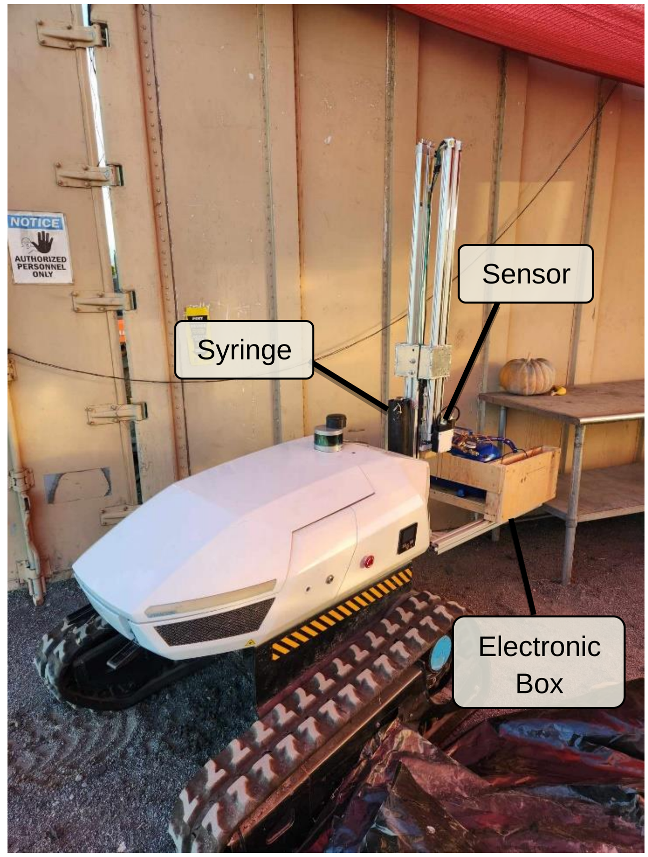


Figure 3.21: AET mounted with sensor and electronic box

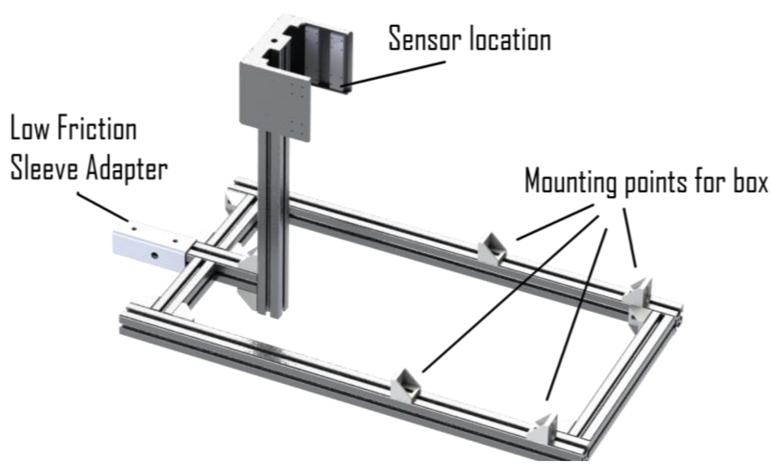


Figure 3.22: Aluminum framing to mount sensor and box

Some tests were made after attaching the sensor to the Bearcub-24 and connecting the electronics. In general, it behaved much as on the prior UTV vehicle platform. The Bearcub-24 met all the requirements, such as driving at full speed, turning, and sampling.

Chapter 4 Results and Discussion

This chapter addresses (*i*) the reliability of the sensor, (*ii*) its compatibility with the autonomous vehicle platform, and (*iii*) the accuracy of the measurements. The parameters measured by the sensor array developed in this study are, in some cases, novel and not comparable to conventional tests. Instead, the validity of the measurements is based on a heuristic evaluation of their responses.

A time series example of all sensor responses throughout a sampling sequence (Figure 4.1)., shows CO_2 concentration measurements (top trace) beginning at the ambient level, increasing when the syringe drew air in from the soil, then decreasing when the measurement was completed, and the system began venting. The loadcell measurement (Figure 4.1), representing the downforce for the CS655 moisture and temperature sensor, increased as the sensor was lowered, and then decreased when it was fully lowered, and excess pressure was purged. The torque sensor shifts at the start to find its home position and then again later to make soil-metal friction measurements. The downward pressure curve shows the pressure in the rodless piston cylinders. In this case, it is expelled early as the moisture sensor experienced a load exceeding the designated safe level. The angular potentiometer measured the turning of the probe. The linear potentiometer measures the distance from full penetration of the probe and decreases as the sensor is lowered and then increases when it is raised. The current shunt is attached to the power supply of the vacuum pump and only raises once when it is run. The pressure sensor reads a spike at the start when the sensor is cleaned. It rises negligibly when the vacuum pump is run and then it spikes when high-pressure air is blown into the ground. The subsequent spikes represent the venting of the CO_2 sensing syringe.

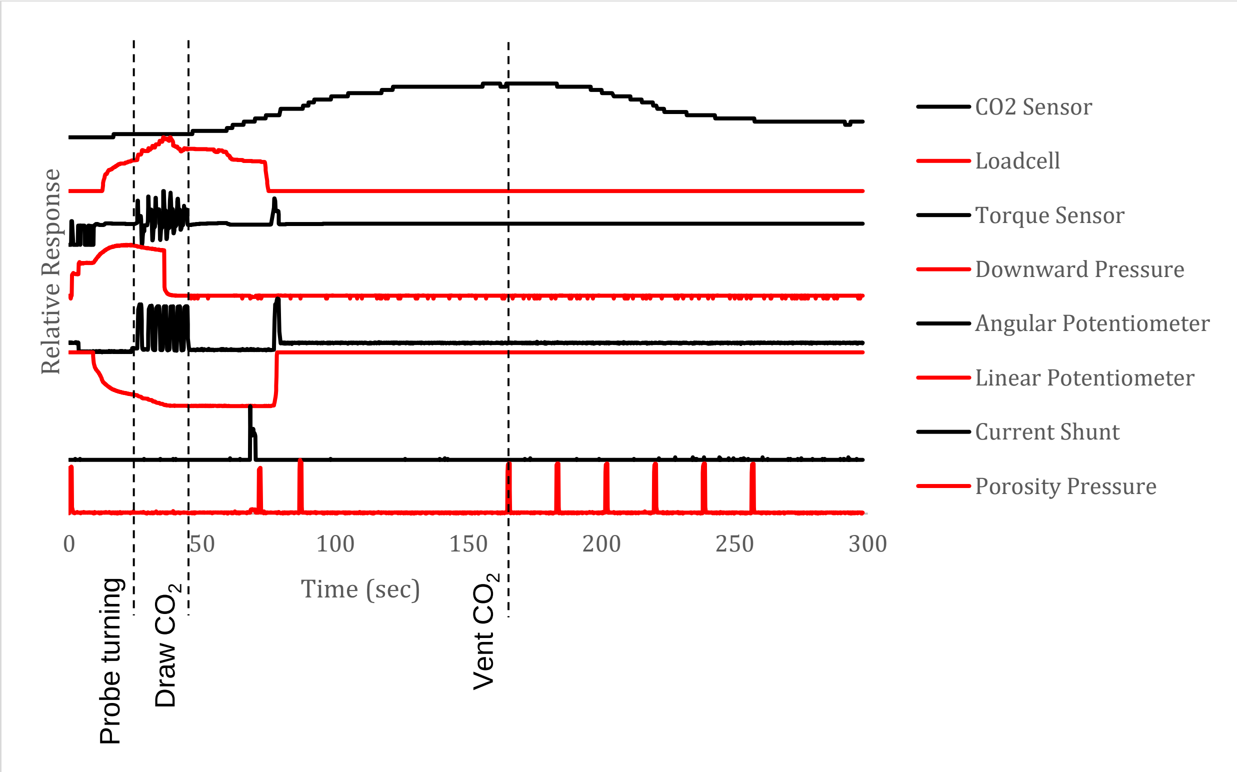


Figure 4.1: Sensors readings vs. *time, denoting key moments in time*

4.1 Penetrometer

While the penetrometer was supplied with constant pressure, it did not descend at a constant rate. Instead, at the onset, the penetrometer operates at a high speed, then rapidly decreases in speed as it begins to penetrate the soil surface (Figure 4.1). The first few centimeters of penetration often varied significantly due to crop residue and uneven terrain from seeding. Beyond that, the speed of penetration begins to plateau. Finally, the sensor further penetrates the soil when the sensor begins to turn. There is a sudden change in the slope of the black curve once the red curve starts oscillating. The penetrometer measurement focuses on the time section after the sensor has penetrated at least 40 mm, where values are less variable. It also ignores the time after the sensor begins turning since this would result in a stepwise function.

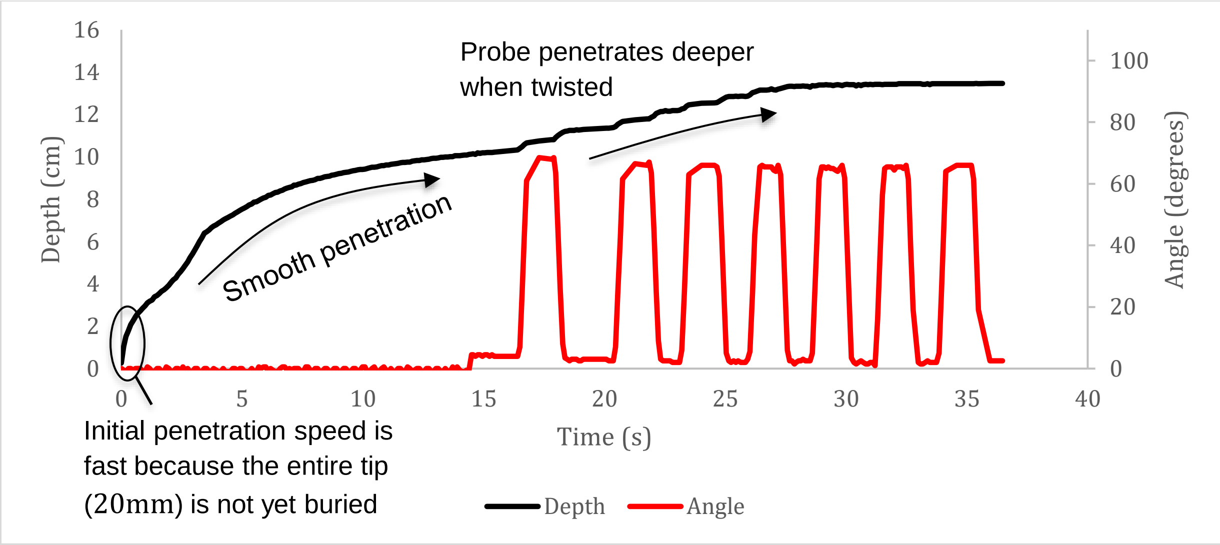


Figure 4.2: Graph of penetration sequence, showing penetration depth increasing quickly at the onset and then again when twisted.

Figure 4.3 shows the target time domain for the penetration test, when the penetration speed is smoother and more predictable. In this time domain, the pressure is more consistent, although it only builds up as the sensor meets resistance. This makes sense since the depth-resistance relationship is mostly from overburden pressure, which increases with depth (Gao et al., 2016).

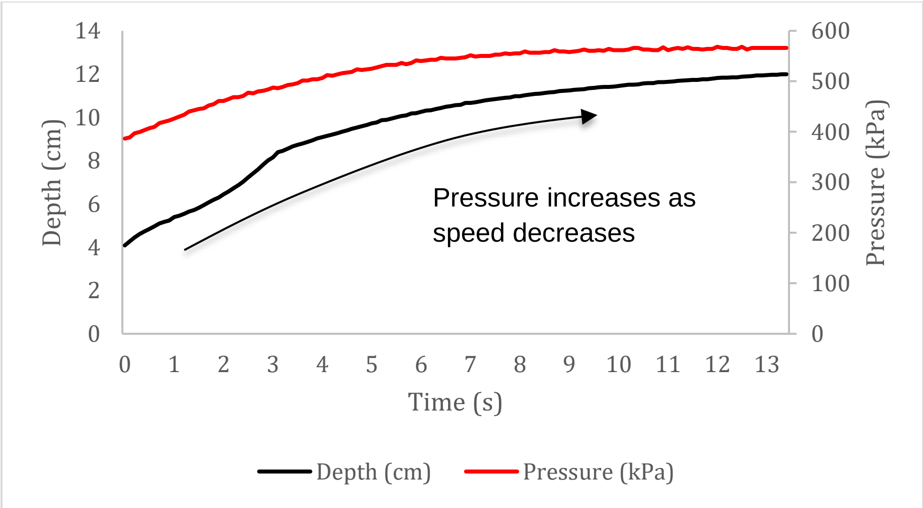


Figure 4.3: Graph of penetration pressure and depth vs time

When comparing the readings from different sites (Figure 4.4), the rate of penetration varies with penetration depth depending on the soil. The sampling sequence is programmed to move on to the next step when the sensor penetration speed has slowed sufficiently; accordingly, the time domains in Figure 4.4 vary with different samples.

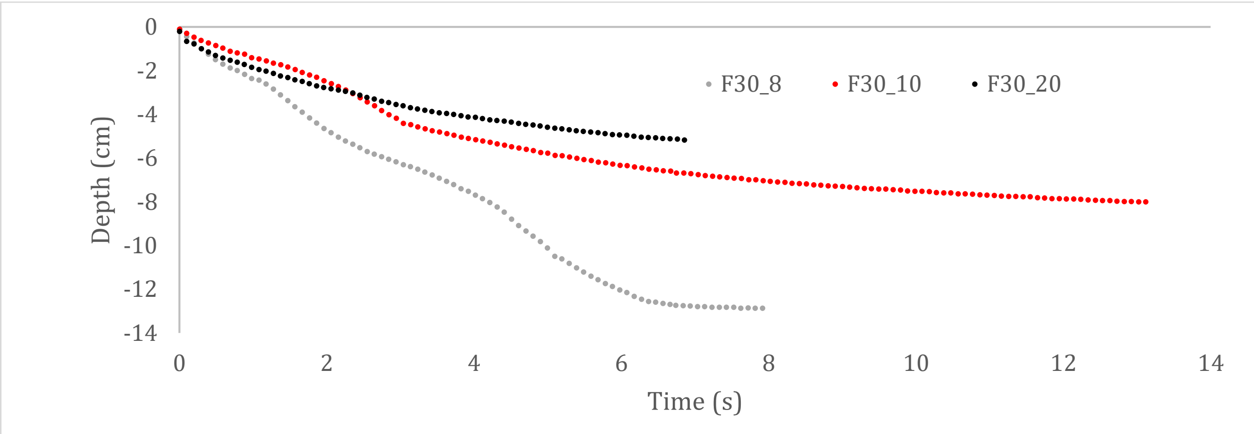


Figure 4.4: Depth vs. *time for three locations showing varying final depth and penetration rate*

For every point, the speed of penetration was taken at a 75 mm depth. This was compared to the compaction meter readings. An average was taken since there is a high deviation among individual readings at any individual point. This average ignored the first measurement at a 0 mm depth as it was often affected by crop residue and other causes of variation. The last measurement at a depth of 150 mm was also ignored since rocks were often found at this level. Figure 4.5. shows the results of these comparisons, with one point from Field 30 being removed because the sensor could not penetrate far enough to measure the penetration rate.

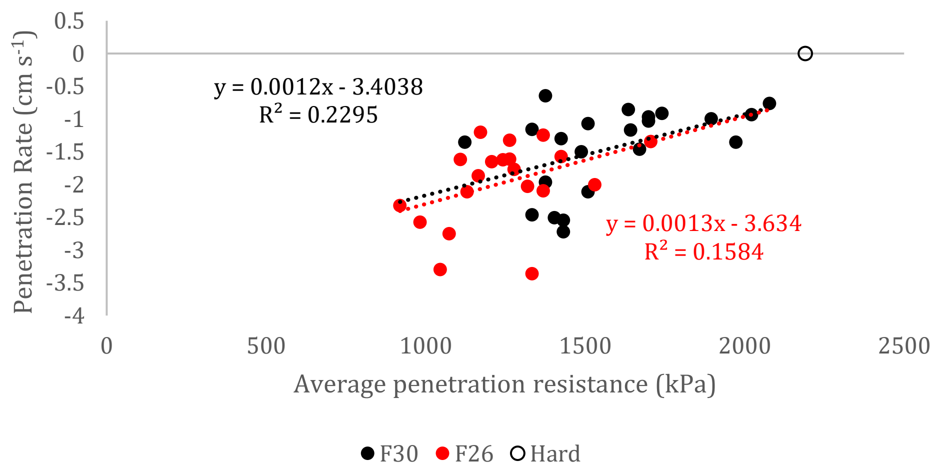


Figure 4.5: Penetration rate at a depth of 75 mm vs. *the average penetration resistance measured by the cone penetrometer at depths of 25 – 125 mm.*

For Field 30 alone, the *p*-value for the slope of the relationship between penetration rate and average penetration resistance was 0.0348, and for Field 26 alone, it was 0.0822. Since there was no significant difference between the two linear relationships, a *t*-test was performed for the two combined, resulting in a *t*-stat of 4.17 and *p*-value of < 0.05. With both datasets combined, the sensor could predict the average penetration resistance with a root mean squared error (RMSE) of 233 kPa. Moreover, the probe initially penetrated less deeply in some soils than others. However, it would penetrate deeper after the DC motor spun the probe to a more consistent depth.

The initial depth of penetration before activation of the motors was compared across the penetrometer measurements (Figure 4.6). The slope of the relationship between initial penetration depth and average penetration resistance yielded a *p*-value of 0.0568 in Field 30 and 0.00207 in Field 26 with a combined value of 0.000641. In the combined dataset, the initial penetration depth was able to predict the average penetration resistance with an RMSE of 261 kPa.

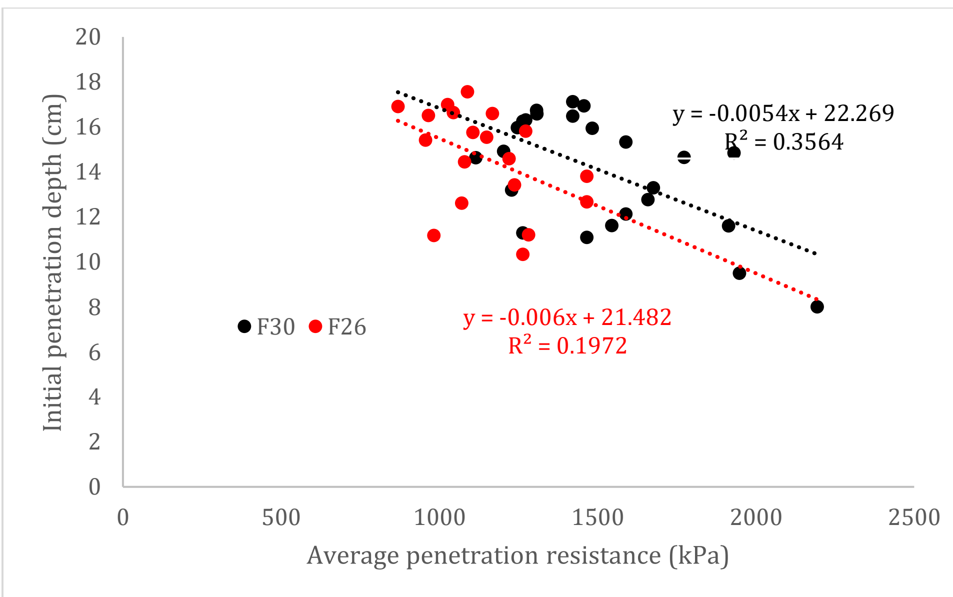


Figure 4.6: Initial penetration depth vs. *average penetration resistance as measured by a cone penetrometer from 0-0.10 m*

A map of the two fields, with rasterized EC_a measurements to show spatial variation and the initial penetration depths shown as colored circles at the sampling spots (Figure 4.7), shows that it was harder to penetrate the southern area of Field 30 since it was near the train tracks.

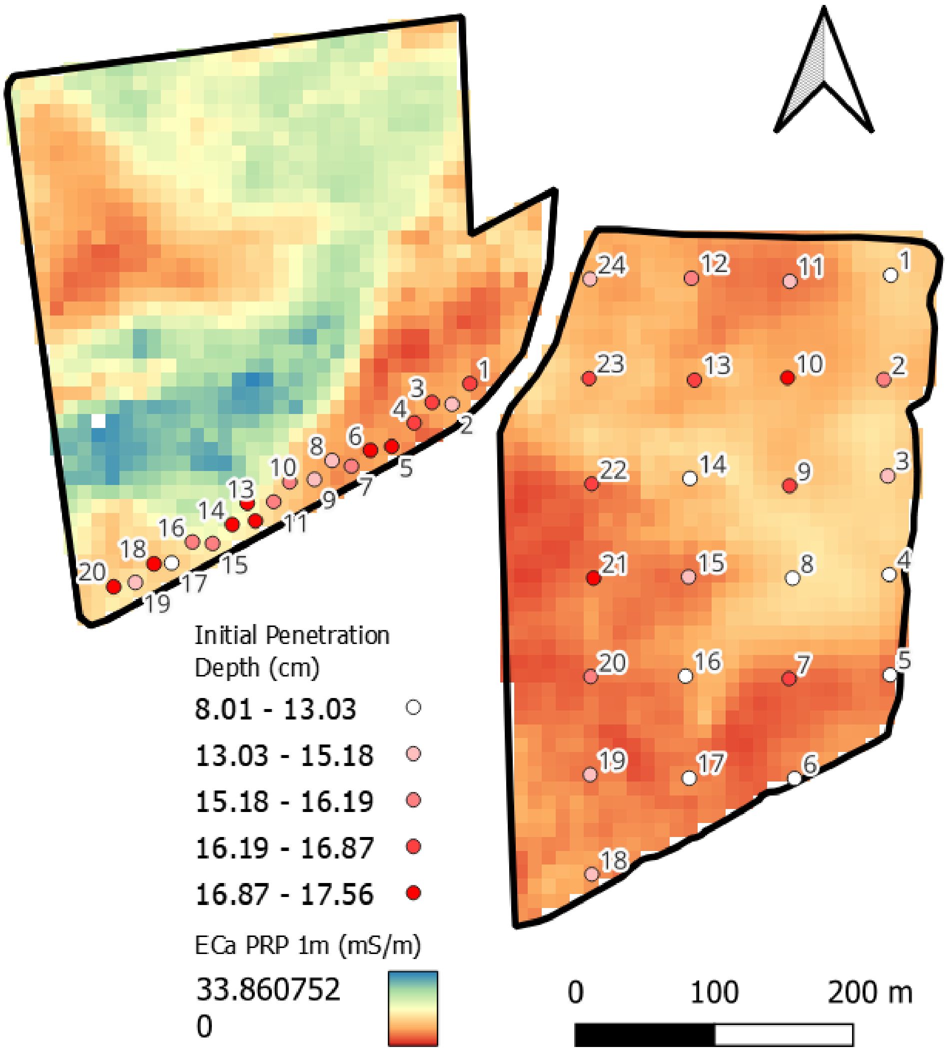


Figure 4.7: Map with rasterized EC_a and penetration depth

4.2 Soil-metal friction

Soil metal friction values are depicted in Figure 4.8. The torque increases as the motor one way turns the angle. The torque decreases when the turning stops, but not back to zero because the system's elasticity continues to impart a force. The torque then turns the other way when the motor turns the other way.

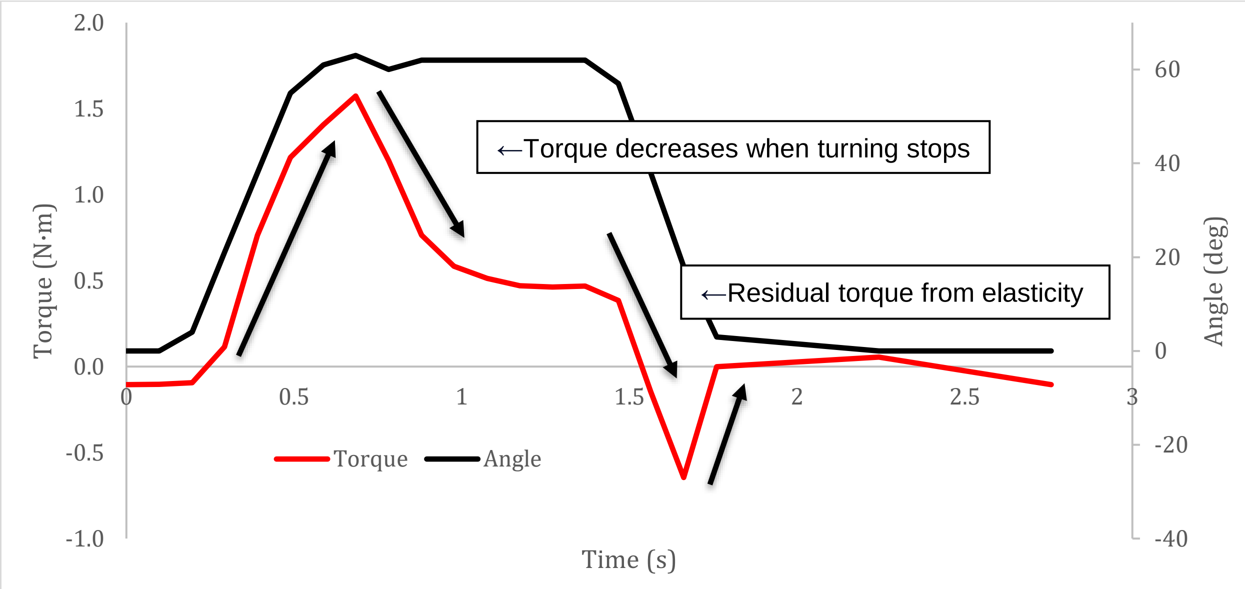


Figure 4.8: Measured torque and angle vs. *time for one twist*

For the sensor measurement, the sensor was twisted six times because it was found that it was sometimes lowered further in harder-to-penetrate soils. This response was limited, however, by the maximum torque setting in some fields. Despite this, there were still variations between measurements. Two sequences are shown (Figure 4.9): the trace in red is an example of a reading from soil that yielded low soil-metal friction, while the black trance was obtained in soil that yielded higher friction measurements.

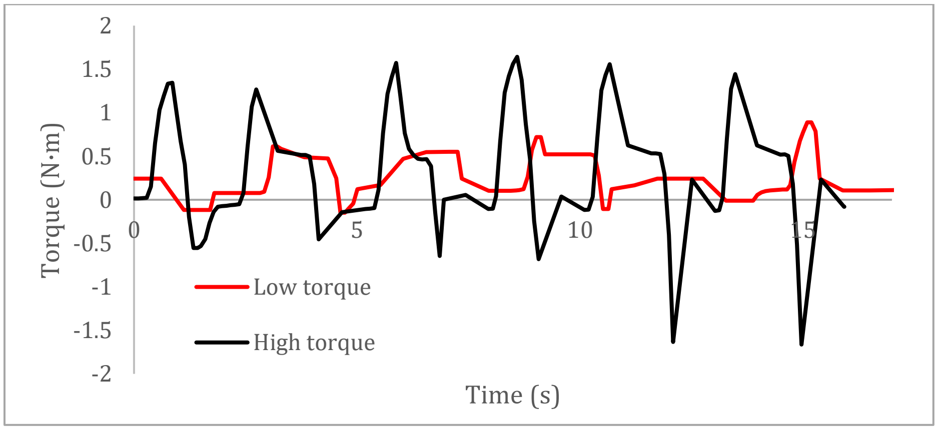


Figure 4.9: Temporal plots for a low torque area and a high torque area.

4.3 CO_2 Concentration

The CO_2 concentration measurements were not compared against a direct analog; however, there was significant variability in the results, while maintaining a predictable curve throughout a measurement sequence. Figure 4.10 shows three examples taken from Field 30 on the same day. The drawing of the sample results in an increase in the sensor reading following a first-order relationship between the sensor reading and time (Yasuda et al., 2012; as cited in Reumont, 2017). The curve hits an inflection point as the venting process begins and returns the reading to the baseline value for atmospheric CO_2 concentration levels.

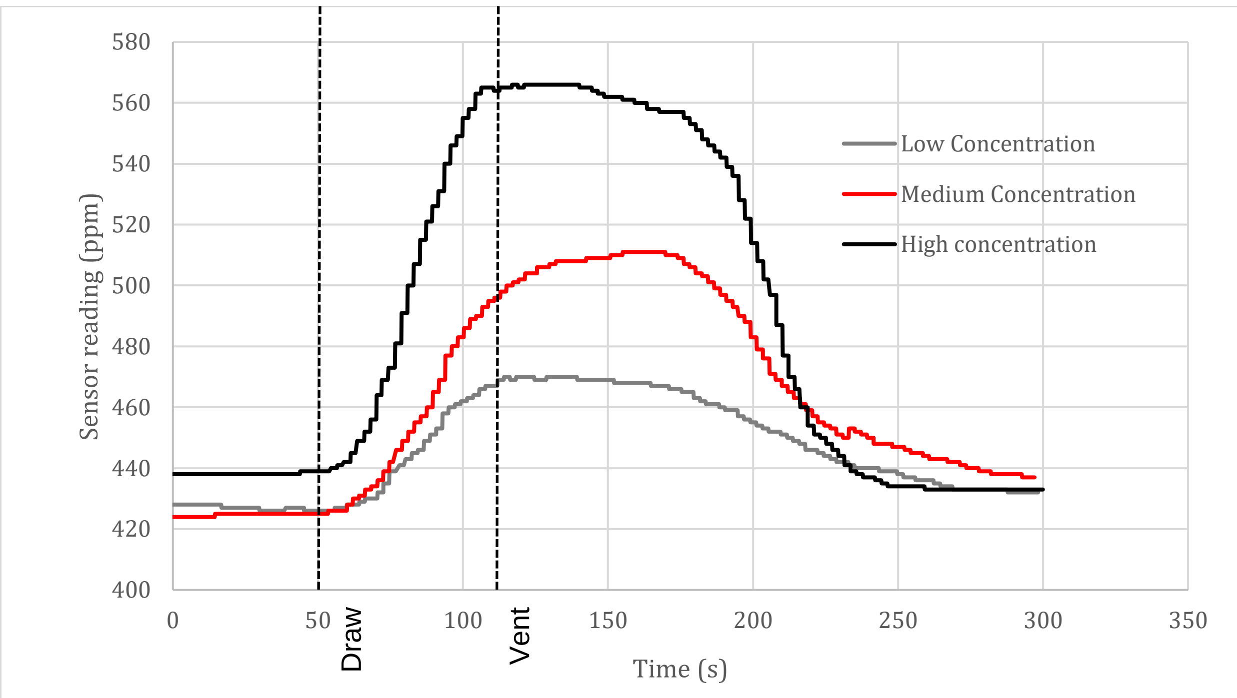
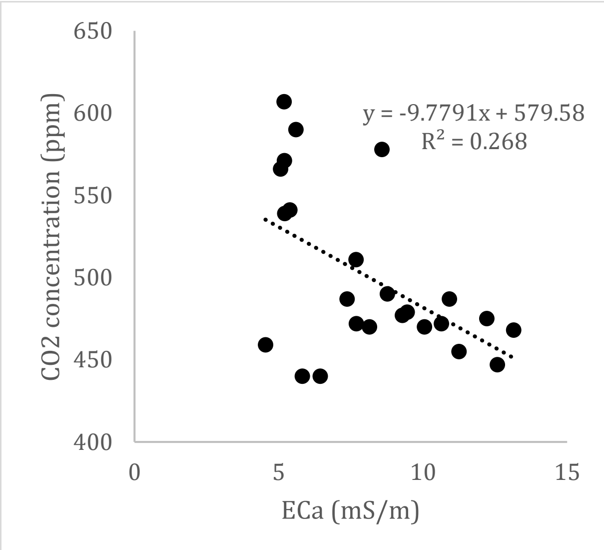


Figure 4.10: A plot of sensor reading vs time showing the times when drawing began and then when venting began

The maximum reading the sensor achieved was taken to see if the sensor readings were related to spatial variation in the field. This point was compared to the EC_a readings sampled from the raster at the same location. While EC_a is not expected to be a direct analog to measuring the concentration of CO_2 in the soil's air-filled pores, it is an established measure of spatial variation in a field, since it is affected by many important parameters (namely moisture which affects soil CO_2 concentrations) and is convenient to measure (Min et al., 2020). A closer relationship would be found with gas chromatography (GC) (Debbagh, 2019). Figure 4.11. compares measured CO_2 concentration and EC_a measured in 2013, using a DUALEM-21S (Dualem, Milton, ON, Canada) system equipped with a 1.0 m separation and a perpendicular arrangement of the receiver and transmitter windings. Out of the available data, this configuration represented the shallowest soil depths.



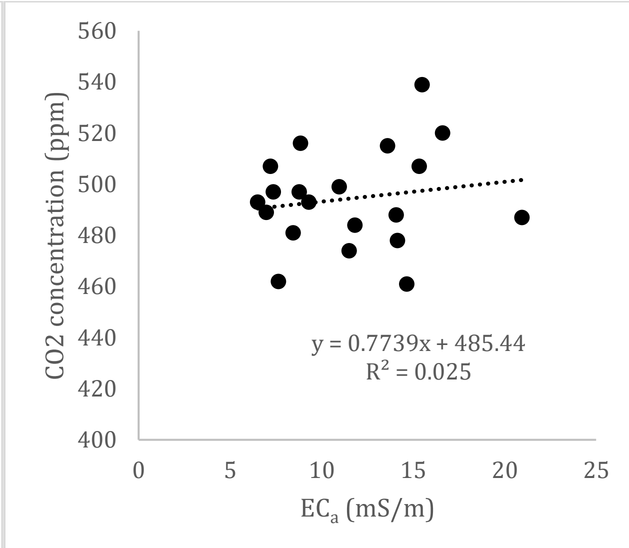


Figure 4.11: $[CO_2]_{ps}$ readings vs. EC_a in Field 30 (left) and Field 26 (right)

As shown, while there may be a significant relationship between the CO_2 concentration and EC_a in Field 30 (p = 0.00956), no significant relationship existed in Field 26 (p = 0.505). This could be partly due to the edge effect since all the samples taken in Field 26 were along the edge of the field (Hardt et al., 2013). Indeed, Figure 4.12 shows how samples in Field 26 were along an edge of the field, whereas samples in Field 30 were well distributed across the field.

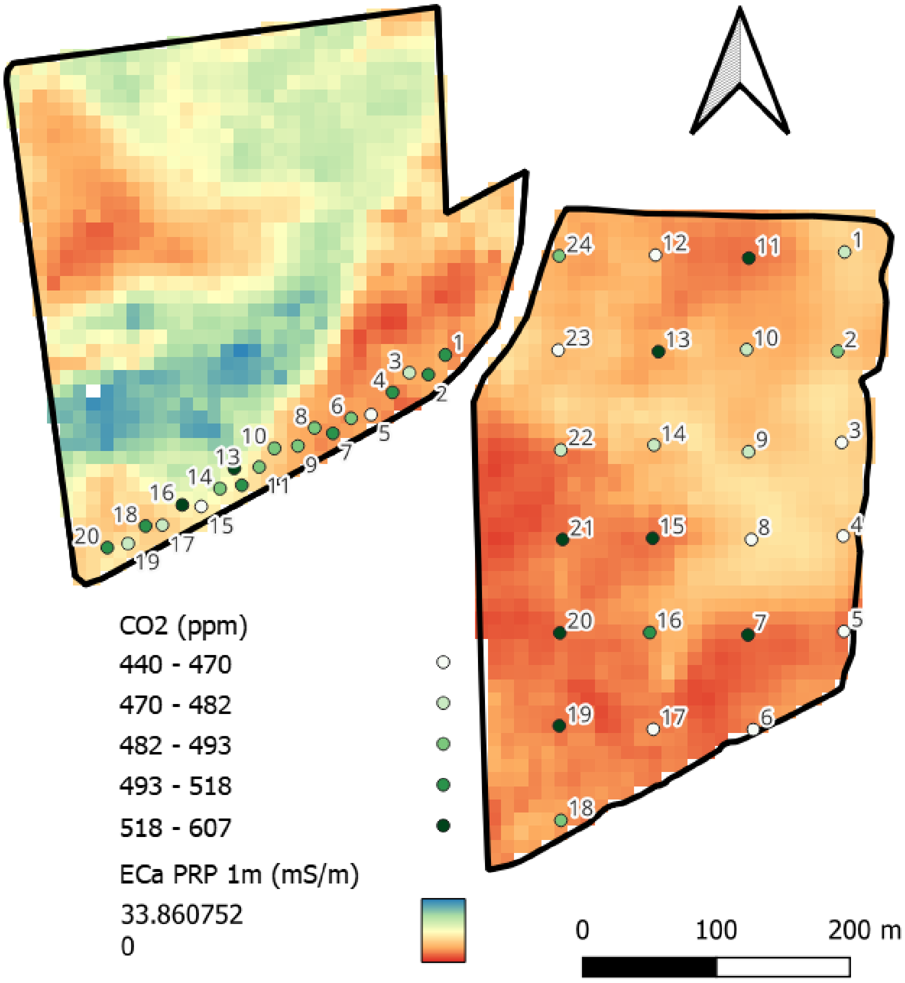


Figure 4.12: Map of rasterized EC_a and point sampled CO_2 concentration values.

4.4 Air Permeability

In testing, it was found that the responsiveness of the air permeability measurements with the vacuum pump came at the expense of the CO_2 concentration measurements. It is hypothesized that the tip that creates space for gases to diffuse well for the CO_2 concentration measurement creates too much surface area for the permeability measurement. When using the stock probe, the vacuum pump could not generate a sufficient flow rate to outpace the pressure permeating the soil. Conversely, when the cone tip of the vapour probe was lathed smoothly, the air drawing for the sample would clog the pores resulting in no response from the K30 CO_2 sensor.

Indoor testing

When testing indoors using a bucket of soil, promising responses were achieved for measuring soil air permeability when using a vacuum pump with the lathed probe. The flow rate of the vacuum pump was controllable using a PWM motor driver to control the voltage. Additionally, the pump's current draw could be measured using a shunt to have an even better picture of the soil's air resistance.

A plot of measured pressure *vs*, time as the vacuum pump was powered on (solid lines) and off (dashed lines) in multiple cycles, with increasing voltages as the probe remained in the same soil (Figure 4.13). As expected, with a lower voltage (lower flow rate), there is a lower measured air resistance pressure. This is useful because it is difficult to size an appropriate flow rate that is universal to all soil types and areas of the field. In some soils, there was so little resistance that the low flow rate was not able to accumulate a measurable pressure at all. On the other side, a high flow rate might result in pressures that are above the rated maximum pressure of the pump and thus dangerous. In testing, the vacuum pump was able to reach pressures far above its rated 85 kPa. In the figure below, the readings never exceeded 70 kPa because a relief valve was placed in line to protect the pump. The pressure also decreases past this point despite the increasing flow rate. This is likely because the flowrate out of the probe was reduced by the relief valve that was situated earlier in line and the soil surrounding the probe was visibly blown away by the previous measurements, reducing its resistance.

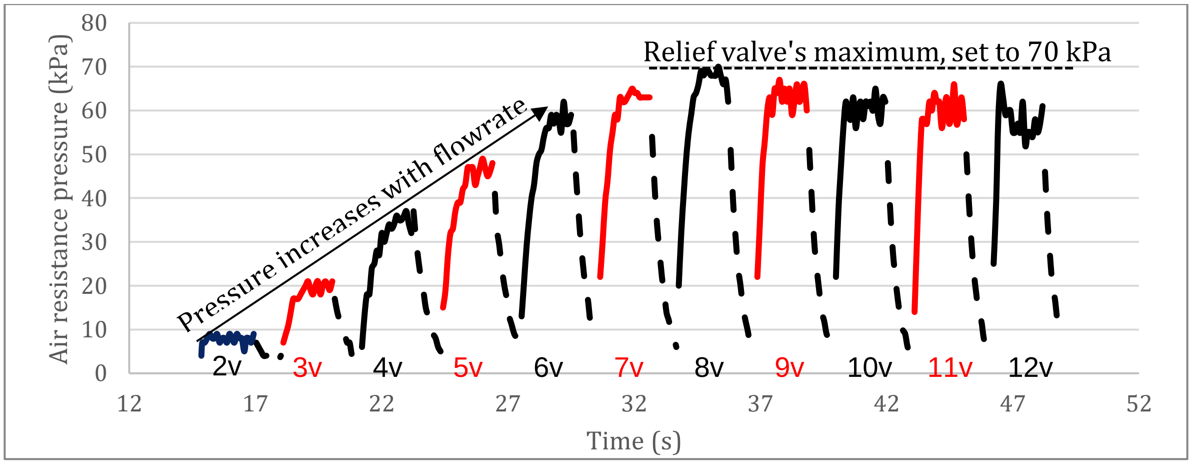


Figure 4.13: Plot of air resistance pressure vs. *time, with increasing voltages.*

Figure 4.14 shows the result of the sensor being tried three times using the same voltage of 12 volts held over 20 seconds (from 5 to 35 s). This was done once in two different soils: *softer soil* (loose garden soil) and *harder soil* where the *softer soil* in the bucket was tamped down and the probe reinserted. The *base resistance* was also measured by taking readings when the probe was not lowered into the soil at all, representing the innate resistance of the air hose and probe pores. Even with the loose, softer soil, there was a measurable response above baseline and an even higher response with the harder, tamped-down soil. In both cases, the measurements declined over time as the soil was visibly blown away — except in the base measurement, where there was no soil.

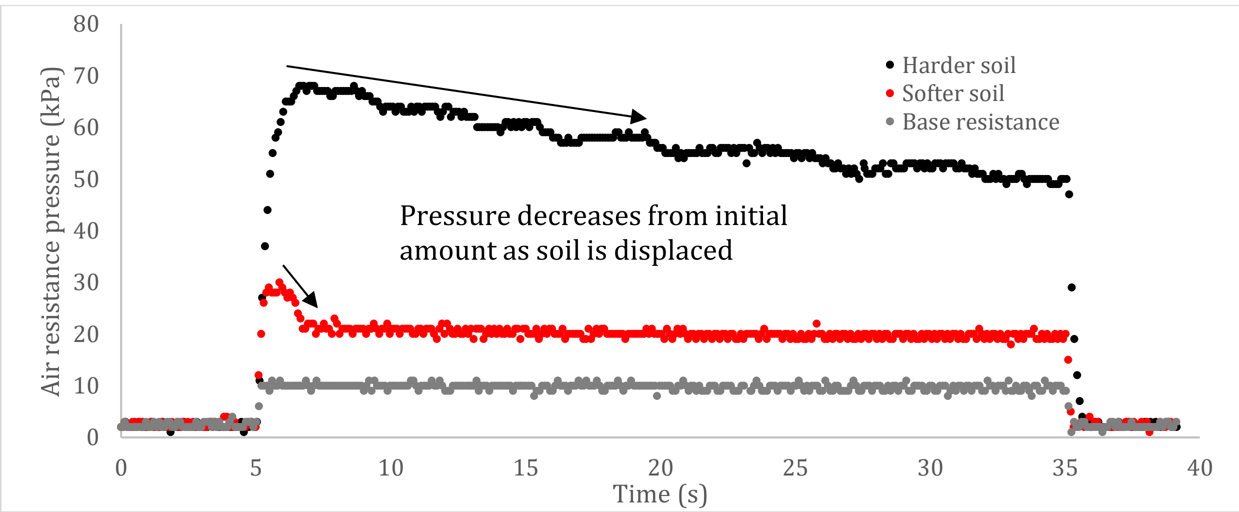


Figure 4.14: Air resistance pressure vs. *time in three different soils*

Unfortunately, this measurement method had inconsistent results when using the standard vapor probe that was necessary for CO_2 concentration measurements. The vacuum pump's flow rate was insufficient to outpace the permeation due to the poor seal. An excerpt from a sampling sequence taken in Field 30, Figure 4.15 shows how the pump was activated between 1 and 3 s, resulting in a negligible amount of pressure. This pressure rarely rose above the baseline level. At the same time, the current draw is measured. However, this measurement is erratic since the clock speed of the microcontroller is limited, and there was no analog filter. Afterward, the spike in pressure from second 4 to second 5 is from blowing pressurized air that was regulated to 620 kPa.

The measured pressure from the soil's air resistance is lower since the size of the air hose limits the flow rate.

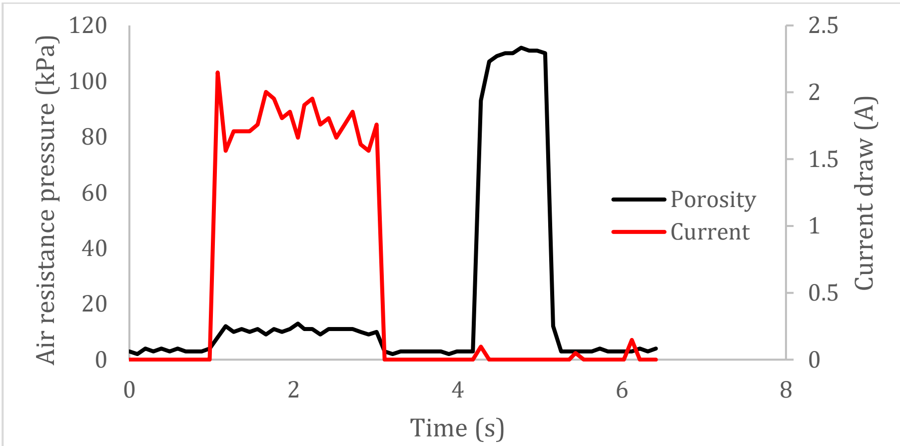


Figure 4.15. Graph of pressure vs time when a vacuum pump is used followed by a release of pressurized air

Field testing

While employing a vacuum pump that could provide known flow rates would have been preferable, such a pump could not supply sufficient flow to accumulate a significant pressure when using the default vapor probe tip. Since it was necessary to use this stock tip to sample the CO_2 concentration. The high-pressure burst was used to measure the soil permeability since it had a higher amplitude response and more variability between measurements. A plot of the maximum pressure achieved from the 22-second burst of air *vs.* ρ_{dry} The location is shown in Figure 4.16. The same five responses were removed from the F30 measurements due to a motor malfunction.

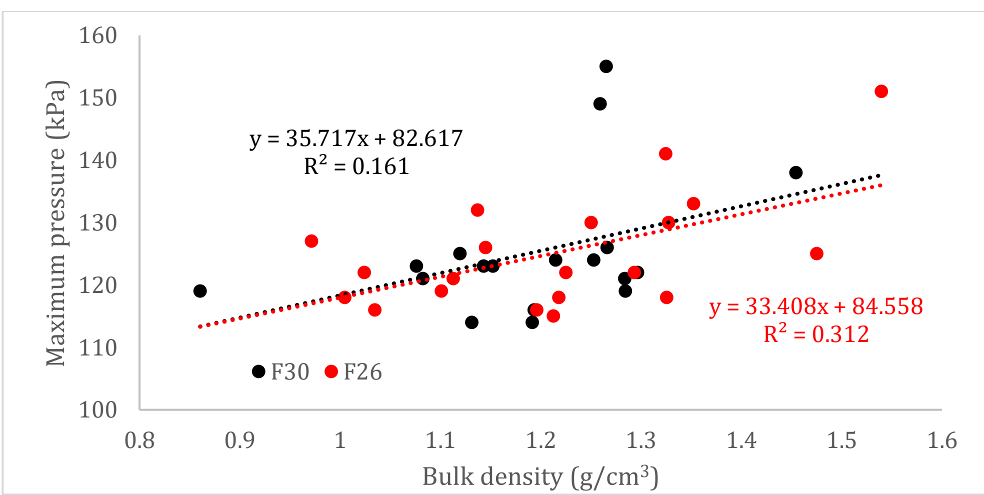


Figure 4.16. Plot of maximum pressure achieved vs. *bulk density measured using the oven dry method*

As seen in Figure 4.16, the linear trendlines for the two relationships are very close, and there is no significant difference between the models from Field 30 and Field 26. While the slope of the relationship between the pressure readings and the bulk density had a *p*-value of 0.0988 which is not significant at a significance level of $\alpha = 0.05$, there was a significant relationship in Field 26 with a *p*-value of 1.34E-05. Since there was no significant difference between the two fields, the combined measurements were also tested to have a *p*-value of 0.00268 for the slope. This combined model had an RMSE of 0.13 g/cm^3 for the bulk density prediction.

The soil moisture levels were also suspected to affect air permeability. However, as seen in Figure 4.17, no relationship was observed.

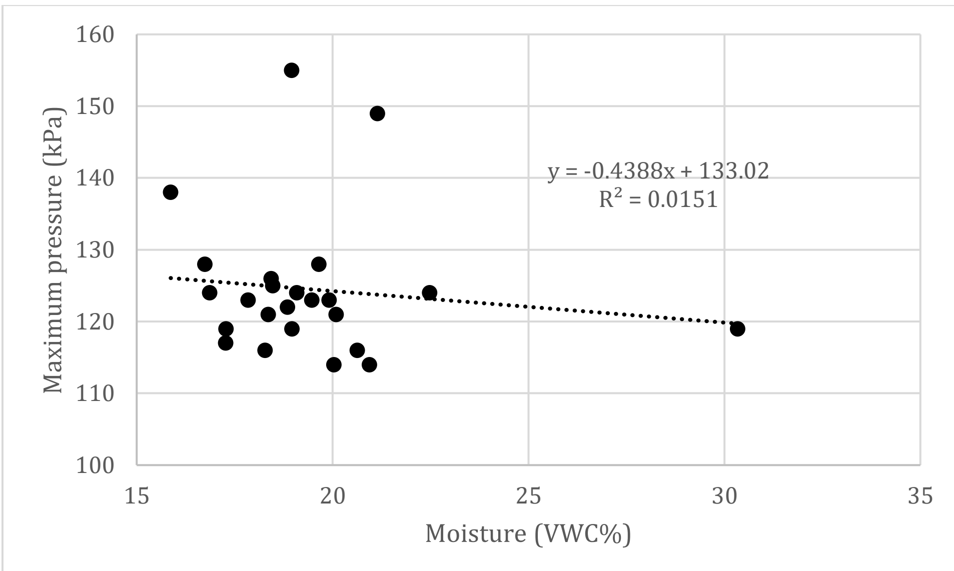


Figure 4.17. Plot of maximum pressure achieved vs. *volumetric water content using the oven dry method*

4.5 Moisture

Since the moisture probe was a commercially available sensor, the sensor itself was assumed to produce valid measurements when used as designed. However, since the sensor system detects when there is an unsafe load on the sensor, it sometimes does not allow for full penetration. A plot of the moisture sensor reading *vs*. the actual moisture determined using the oven-dry method for Field 30 (Figure 4.18), showed some lower-valued responses when the probe did not fully penetrate the soil. The readings at locations where there was insufficient penetration (more than a 50 mm distance from the soil) are highlighted and a second best-fit line is drawn without these outliers. The sensor reading could predict the sensor reading with an RMSE of 2.44%, and the relationship was significant at p = < 0.05.

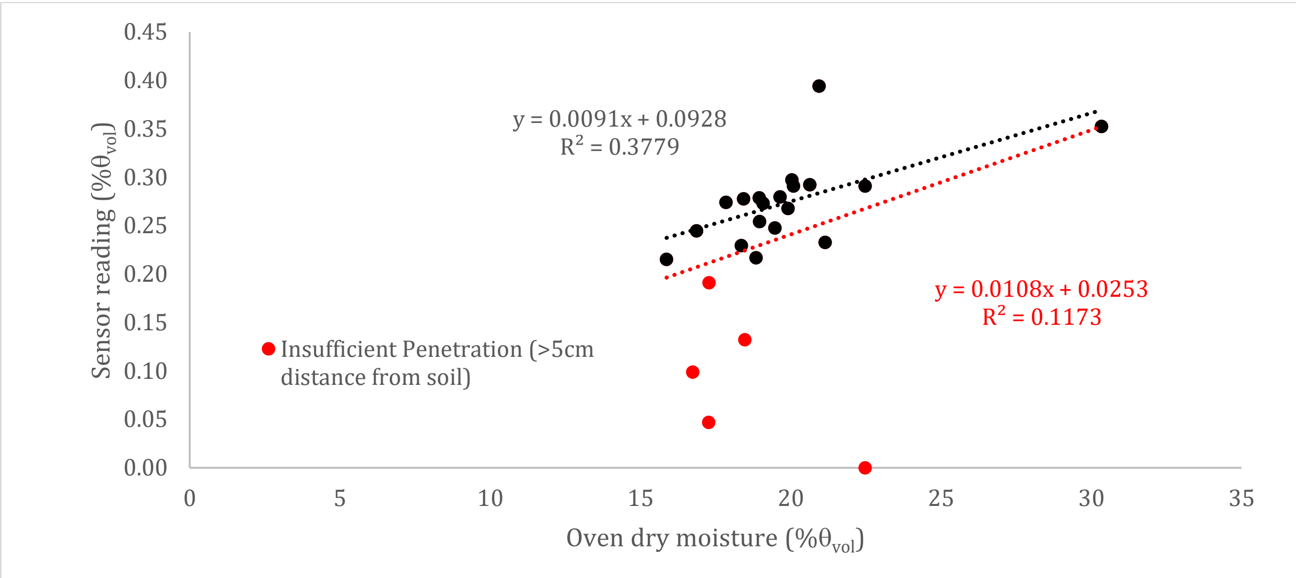


Figure 4.18: Sensor reading for VWC vs actual VWC attained from the oven dry method

4.6 Apparent Electrical Conductivity

The electrical conductivity measurement was done using the same probe as the one employed to measure soil moisture. These sensor readings for EC were compared with 2014 EC_a measurements taken by an induction sensor with a perpendicular orientation targeting 1.0 m. In Figure 4.18, the same insufficient penetration points as in Figure 4.17 are highlighted. However, in this case, there was no significant relationship between these two different measurement methods and times.

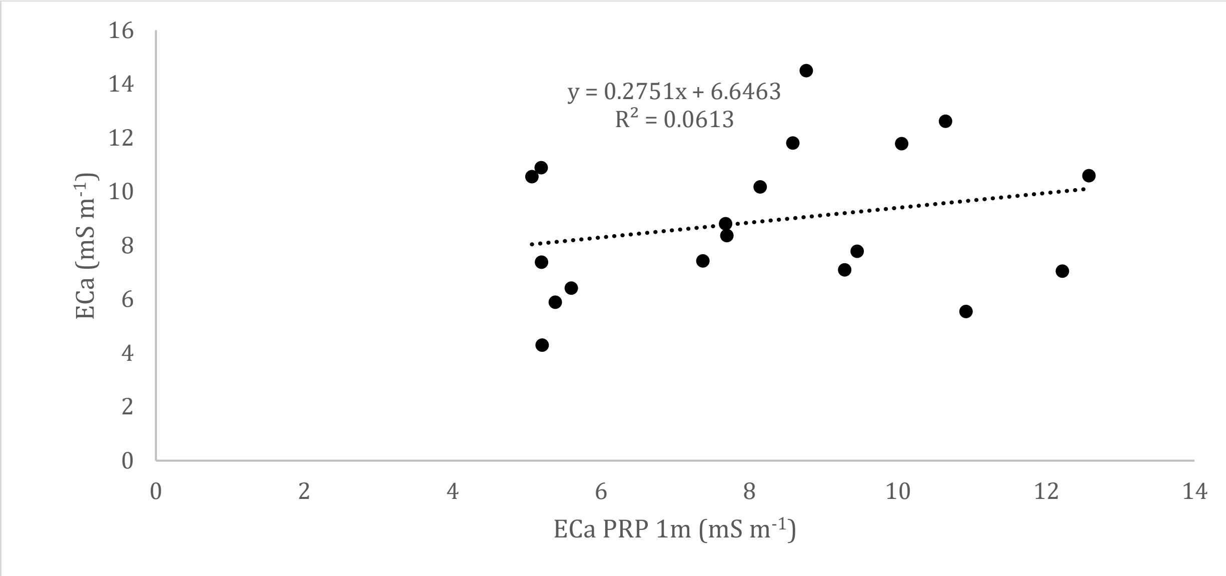


Figure 4.19. Sensor EC_a vs Dualem EC_a from 2014 with points with insufficient penetration removed

4.7 Overall System Performance

Since the sensor made use of pneumatics to keep the sensor raised, air would leak over a length of time and if air pressure was not restored to the system, the sensor would slowly begin to descend. This would prove a danger to the system because driving while the sensor is on the ground could cause significant damage. A human user must visually confirm that the sensor is raised before displacing the vehicle. This problem can be addressed with an autonomous vehicle platform by checking safety interlocks, such as a limit switch, that closes when the sensor is raised and requires a minimum pressure in pistons before allowing the system to be displaced. The BearCub-24 conveniently features a C-Bus interface for implements that can be used for this purpose.

The small-profiled and standardized design of the sensor array lends itself to being compatible with the autonomous vehicle platform provided by Ztractor. The sensor requires access to a 2" hitch receiver that can support 45 kg, a serial communication interface and a 12VDC power supply that can support 20 A peak draw. The Ztractor Bearcub-24 is a versatile and robust machine that comes in stock with a class III hitch receiver, a C-bus communication interface and a 12VDC port that links to an electric vehicle battery.

4.8 Future improvements

Overall, the sensor worked to reliably perform its sequence and log readings from its sensors for softer soil types; however, some suggestions are recommended. The system currently uses serial processing since it is widely used on personal computers. However, the standard for interfacing implements in agricultural machinery is ISOBUS (Haapala and Nurkka, 2006).

One issue was the probe sealing for the CO_2 concentration measurement and the vacuum pump measurement for air permeability. Since it was difficult to have a single, solid probe that was compatible with both sensor measurements, it may be worth making a sensor probe that changes shape to facilitate either measurement.

Since it was sized to provide just over 2 MPa of penetration pressure, it was not sized to penetrate soils of high clay content when dry. This resulted in shallow penetration for some measurements. The reduced penetration seemed to have the biggest effect on the moisture sensor since it was calibrated to take measurements when its probes were fully in the soil. Increasing the bore size of the pistons would increase the penetrating force without demanding a higher-pressure circuit or significantly increasing the size of the system. This extra force may also warrant more weight to keep the system from lifting the vehicle platform itself.

While sensor factors such as final penetration depth influenced measurements such as moisture readings, the measurements themselves are hypothesized to be related based on prior literature. More in-depth statistical analysis such as principal component analysis (PCA) are methods used in sensor fusion to single out confounding factors and yield more accurate results.

Measurements can also vary highly in locations close to each other due to random variations in the soil. Testing the system on a field with greater spatial variation would increase the relative variation caused by sample location and parameters over the random variation. This would make relationships between measured parameters easier to see.

Chapter 5 Conclusions

Obtaining high-quality data on soil properties is necessary for the evaluation of soil health and for guiding best management practices. This project aimed to use robotics to increase the scalability of *in situ* invasive proximal soil sensing to facilitate the mapping of rooting-depth soil biological properties.

A sensor system was developed to be coupled to an autonomous vehicle to measure multiple parameters simultaneously. The prototype successfully measured penetration resistance, soil-metal friction, CO_2 concentration, air permeability, and θ_{vol} while only needing to stop for 60 s. Since the system is mountable on a standard hitch receiver, it can be a versatile tool for farmers looking for a convenient method to gather soil health data. The system is also robust and designed to handle uneven terrain, rocks, and ground cover, which can prove useful off the field and help researchers in natural resource sciences.

However, while the correlations between prototype measurements and reference measurements were significant but not strong, this is likely due to the relatively high random variation of soils compared to the spatial variation in the field. Few reference measurements were chosen, and some sensor measurements did not have direct analogs.

In the future, a more involved field test with more samples in fields with higher spatial variation can be conducted to increase the range of responses. Sensors should also be validated against more direct analogs (e.g., CO_2 concentration measurement vs. GC, and air permeability measurement vs. porosity). This would increase the visibility of relationships between the sensor readings and the reference measurements and the sensor readings with each other. Further comparisons can be made with other established soil health indicators (e.g., CO_2 flux, flush of CO_2). Since the system is capable of multiple simultaneous sensor measurements, statistical modeling, and sensor fusion would also help to improve the prediction of reference measurements. Additionally, comparisons with established soil health indicators such as biodiversity and flush of

CO_2 would also help to link the system with the current wealth of knowledge on soil health and guide the future development of this project.

Some design improvements would call for a probe that can create a tighter seal for porosity measurement and an ISOBUS interface to work with agricultural machinery. Increasing the clock speed of the microcontroller would result in a higher temporal resolution, which would be especially beneficial for the soil-metal friction measurements. Moreover, a larger piston bore size would allow for more reliable penetration of harder soils.

Automation is increasingly benefiting agriculture through proximal soil sensing by facilitating the invasive measurements of soil health indicators. This project was a step along this path by providing a compact and ergonomic autonomous prototype.

References

4Farm. (2020). Aplicativo Para Agricultura de Precisão. Accessed 11 March 2024 at. Accessed at https://4farm.me/

A&L Canada Laboratories, Inc. (2024). Welcome to A&L Canada. London, ON. Accessed at https://alcanada.com/

Adamchuk, V. I., & Viscarra Rossel, R. A. V. (2010). Development of on-the-go proximal soil sensor systems. *Proximal Soil Sensing :*, 15–28. doi:10.1007/978-90-481-8859-8_2

Adamchuk, V. I., Hummel, J. W., Morgan, M. T., & Upadhyaya, S. K. (2004). On-the-go soil sensors for precision agriculture. *Computers and Electronics in Agriculture*, *44*(1), 71–91. doi:10.1016/j.compag.2004.03.002

Adamchuk, V. I., Viscarra Rossel, R. A., Sudduth, K. A., & Lammers, P. S. (2011). *Sensor fusion for precision agriculture* (I. Sensor, Ed.). doi:10.5772/19983

Adamo, F., Andria, G., Attivissimo, F., & Giaquinto, N. (2004). An acoustic method for soil moisture measurement. *IEEE Transactions on Instrumentation and Measurement*, *53*(4), 891–898. doi:10.1109/TIM.2004.831126

Aerospace.org. (2024). Brief history of GPS. El Segundo, CA: Aerospace Corporation. Accessed 11 March 2024 at https://aerospace.org/article/brief-history-gps

Al-Hamed, S., Wahby, M., Aboukarima, A., & Ahmed, K. (2014). Development of a computer program using visual basic for predicting performance parameters of tillage implements. *Misr Journal of Agricultural Engineering*, *31*(3), 1157–1190. doi:10.21608/mjae.2014.99147

Allen, D. E., Singh, B. P., & Dalal, R. C. (2011). Soil health indicators under climate change: a review of current knowledge. *Soil health and climate change*, 25-45. doi:10.1007/978-3-642-20256-8_2

AMS, Inc. (2024). 12" Original Gas Vapor Probe Base Only. American Falls, ID. Accessed at https://www.ams-samplers.com/gvp-base-section-only/

Arias, M. E., González-Pérez, J. A., González-Vila, F. J., & Ball, A. S. (2005). Soil health: A new challenge for microbiologists and chemists. *International Microbiology*, *8*(1), 13–21. Accessed at http://hdl.handle.net/10261/2130

ASABE. (2019). *ASABE S313.3. FEB1999 (R2018) Cone Penetration Testing of Soils*. Joseph, Mich.: American Society of Agricultural and Biological Engineers.

Barrile, V., Simonetti, S., Citroni, R., Fotia, A., & Bilotta, G. (2022). Experimenting agriculture 4.0 with sensors: A data fusion approach between remote sensing, UAVs and Self-Driving Tractors*. Sensors*, *22*(20), 7910. doi:10.3390/s22207910

Bekku, Y., Koizumi, H., Oikawa, T., & Iwaki, H. (1997). Examination of four methods for measuring soil respiration. *Applied Soil Ecology*, *5*(3), 247–254. https://doi.org/10.1016/S0929-1393(96)00131-X

Bengough, A. G., & Mullins, C. E. (1990). Mechanical impedance to root growth: a review of experimental techniques and root growth responses. *Journal of Soil Science*, *41*(3), 341–358. https://doi.org/10.1111/j.1365-2389.1990.tb00070.x

Campbell, J. B., & Wynne, R. H. (2011). *Introduction to Remote Sensing* (4th ed.). New York: Guilford Press.

Cardoso, E. J. B. N., Vasconcellos, R. L. F., Bini, D., Miyauchi, M. Y. H., Santos, C. A. D., Alves, P. R. L., & Nogueira, M. A. (2013). Soil health: looking for suitable indicators. What should be considered to assess the effects of use and management on soil health? *Scientia Agricola*, *70*, 274–289. doi:10.1590/S0103-90162013000400009

Cardoso, Elke Jurandy Bran Nogueira, Vasconcellos, R. L. F., Bini, D., Miyauchi, M. Y. H., dos Santos, C. A., Alves, P. R. L., … Nogueira, M. A. (2013). Soil health: looking for suitable indicators. What should be considered to assess the effects of use and management on soil health? *Scientia Agricola*, *70*, 274–289. https://doi.org/10.1590/S0103-90162013000400009

Carlson, P. A. D., Adamchuk, V. I., Kvezereli, B., & Madramootoo, C. (2020). Development of an integrated sensor system for automated on-the-spot measurement of physical soil properties.

In *2020 ASABE Annual International Virtual Meeting (p. 1). American Society of Agricultural and Biological Engineers*. doi:10.13031/aim.202001444

Choi, T. Y., Choi, B. S., & Seo, K. H. (2010). Position and compliance control of a pneumatic muscle actuated manipulator for enhanced safety. *IEEE Transactions on Control Systems Technology*, *19*(4), 832–842. doi:10.1109/TCST.2010.2052362

CO2Meter. (2022). K-30 CO2 Sensor Module. Ormond Beach, FL: Co2meter.com. Accessed 11 March 2024 at. Accessed at https://www.co2meter.com/products/k-30-co2-sensor-module

Cropper, M., & Griffiths, C. (1994). The interaction of population growth and environmental quality. *The American Economic Review*, *84*(2), 250–254. Accessed at https://www.jstor.org/stable/2117838

Danielson, R. E., & Sutherland, P. L. (1986). Porosity. *P.*, *In: Methods of Soil Analysis: Part 1 Physical and Mineralogical Methods, 2nd ed. (A. Klute, ed.). Madison, WI: American Society of Agronomy, Inc. and Soil Science Society of America, Inc*, 443–461. doi:/10.2136/sssabookser5.1.2ed.c18

Davidson, D. T. (1965). Penetrometer measurements. *Methods of Soil Analysis: Part*, *1 Physical and Mineralogical Properties, Including Statistics of Measurement and Sampling, 9*, 472–484. doi:10.2134/agronmonogr9.1.c37

Debbagh, M. (2019). Development of a Low-cost Wireless Sensor Network for Passive in situ Measurement of Soil Greenhouse Gas Emissions. Master of Science Thesis. *Dept. of Bioresource Engineering, McGill University.* Accessed at https://escholarship.mcgill.ca/concern/theses/3j333666s

Dias Carlson, P. (2021). Development of an Integrated Sensor System for Automated on-the-spot Measurement of Physical Soil Properties. Master of Science Thesis. *Dept. of Bioresource Engineering, McGill University.* Accessed at https://escholarship.mcgill.ca/concern/theses/n296x3897

Doran, J. W., & Zeiss, M. R. (2000). Soil health and sustainability: managing the biotic component of soil quality. *Applied Soil Ecology*, *15*(1), 3-11. doi: 10.1016/S0929-1393(00)00067-6

Dow, A. I., & James, D. W. (1973). Intensive soil sampling: A principle of soil fertility management in intensive irrigation. *Washington Agric. Exp. Sta. Bull.*, *781*.

Erbach, D. C. (1987). Measurement of soil bulk density and moisture. *Trans. ASAE*, *30*(4), 922–931. doi:10.13031/2013.30500

Erktan, A., Or, D., & Scheu, S. (2020). The physical structure of soil: determinant and consequence of trophic interactions. *Soil Biology and Biochemistry*, *148 : 107876*.

Farahani, H. J., Khosla, R., & Buchleiter, G. W. (2022). Field EC Mapping. A New Tool to Make Better Decisions -- 0.568. *Accessed* March 2024. Accessed at https://extension.colostate.edu/topic-areas/agriculture/field-ec-mapping-a-new-tool-to-make-better-decisions-0-568/

Galassi, T. (2015). OSHA Procedures for Safe Weight Limits when Manually Lifting. Washington. *D. C. : U. S. Dept. of Labor, Occupational Safety and Health Administration. Accessed* March 2024 at https://www.osha.gov/laws-regs/standardinterpretations/2013-06-04-0#:

Gale, E. S., Sullivan, D. M., Cogger, C. G., Bary, A. I., Hemphill, D. D., & Myhre, E. A. (2006). Estimating plant-available nitrogen release from manures, composts, and specialty products. *Journal of Environmental Quality*, 35(6), 2321-2332.

Guigue, J., Mathieu, O., Lévêque, J., Mounier, S., Laffont, R., Maron, P. A., & Lucas, Y. (2014). A comparison of extraction procedures for water-extractable organic matter in soils. *European Journal of Soil Science*, *65*(4), 520–530.

Haapala, H. E. S., Pesonen, L., & Nurkka, P. (2006). *Usability as a Challenge in Precision Agriculture--case study: an ISOBUS VT. Agricultural Engineering International 8: MES 05 001*.

Hague, T., Marchant, J. A., & Tillett, N. D. (1997). *A system for plant scale husbandry* (I. Proceedings, Ed.). Oxford, UK: BIOS Scientific Publishers.

Håkansson, I., & Lipiec, J. (2000). A review of the usefulness of relative bulk density values in studies of soil structure and compaction. *Soil and Tillage Research*, *53*(2), 95–91. doi:10.1016/S0167-1987

Haney, R. L., Haney, E. B., Smith, D. R., Harmel, R. D., & White, M. J. (2018). The soil health tool---Theory and initial broad-scale application. *Applied Soil Ecology*, *125*, 162–168. doi:10.1016/j.apsoil.2017.07.035

Hardt, E., Pereira-Silva, E. F. L., Dos Santos, R. F., Tamashiro, J. Y., Ragazzi, S., & da S. Lins, D. B. (2013). *The influence of natural and anthropogenic landscapes on edge effects.'' Landscape and Urban Planning 120: 59-69*. Accessed at http://dx.doi.org/10.1016/j.landurbplan.2013.08.014

Hashimoto, S., & H., K. (2006). Relationships between soil CO2 concentration and CO2 production, temperature, water content, and gas diffusivity: implications for field studies through sensitivity analyses. *Journal of Forest Research*, *11*, 41–50. doi:10.1007/s10310-005-0185-4

Hobbs, P. R. (2007). Conservation agriculture: what is it and why is it important for future sustainable food production? *The Journal of Agricultural Science*, *145*(2), 127. doi:10.1017/S0021859607006892

Huang, C., Bian, C., Wang, L., Zhou, W., Li, Y., & Li, B. (2022). *Development and validation of a method for determining d-limonene and its oxidation products in vegetables and soil using GC--MS. Microchemical Journal 179: 107470*. Accessed at http://dx.doi.org/10.1016/j.microc.2022.107470

Jabro, J. D., Stevens, W. B., & Iversen, W. M. (2020). Comparing two methods for measuring soil bulk density and moisture content. *Open Journal of Soil Science*, *10*(6), 233–243. doi:10.4236/ojss.2020.106012

Keen, B. A., & Haines, W. B. (1925). Studies in soil cultivation. I. The evolution of a reliable dynamometer technique for use in soil cultivation experiments. *The Journal of Agricultural Science*, *15*(3), 375–386. doi:10.1017/S002185960000681X

Koolen, A. J., & Vaandrager, P. (1984). Relationships between soil mechanical properties. *Journal of Agricultural Engineering Research*, *29*(4), 90086–90086. doi:10.1016/0021-8634

Kuzyakov, Y., Friedel, J. K., & Stahr, K. (2000). Review of mechanisms and quantification of priming effects. *Soil Biology and Biochemistry*, 32(11-12), 1485-1498.

Laishram, J., Saxena, K. G., Maikhuri, R. K., & Rao, K. S. (2012). Soil quality and soil health: A review. *International Journal of Ecology and Environmental Sciences*, *38*(1), 19–37.

Lauber, C. L., Ramirez, K. S., Aanderud, Z., Lennon, J., & Fierer, N. (2013). Temporal variability in soil microbial communities across land-use types. *ISME Journal*, *7*(8), 1641–1650. doi:10.1038/ismej.2013.50

Lehmann, J., Bossio, D. A., Kögel-Knabner, I., & Rillig, M. C. (2020). The concept and future prospects of soil health. *Nature Reviews. Earth & Environment*, *1*(10), 544–553. doi:10.1038/s43017-020-0080-8

Li, Y., Cui, S., Chang, S. X., & Zhang, Q. (2019). Liming effects on soil pH and crop yield depend on lime material type, application method and rate, and crop species: a global meta-analysis. *Journal of Soils and Sediments*, *19*, 1393–1406. doi:10.1007/s11368-018-2120-2

Lin, H., Wheeler, D., Bell, J., & Wilding, L. (2005). Assessment of soil spatial variability at multiple scales. *Ecological Modelling*, *182*(3–4), 271–290. doi:10.1016/j.ecolmodel.2004.04.006

Lloyd, J., & Taylor, J. A. (1994). On the temperature dependence of soil respiration. *Functional Ecology :*, 315–323.

Ma, X., Geng, G., Zhang, H., Bian, C., Chen, H. Y. H., Jiang, D., & Xu, X. (2021). Global negative effects of nutrient enrichment on arbuscular mycorrhizal fungi, plant diversity and ecosystem multifunctionality. *New Phytologist*, *229*(5), 2957–2969. doi:10.1111/nph.17077

McKenzie, R. H. (2010). Agricultural soil compaction: Causes and management. *Agri-Facts*. Alberta Agriculture and Rural Development. https://www1.agric.gov.ab.ca/$department/deptdocs.nsf/all/agdex13331/$file/510-1.pdf

Min, K., Berhe, A. A., Khoi, C. M., van Asperen, H., Gillabel, J., & Six., J. (2020). Differential effects of wetting and drying on soil CO 2 concentration and flux in near-surface vs. deep soil layers. *Biogeochemistry*, *148*, 255–269.

Minasny, B., & McBratney, A. B. (2016). Digital soil mapping: A brief history and some lessons. *Geoderma*, *264(B)*, 301–311. doi:10.1016/j.geoderma.2015.07.017

Mulla, D., & Khosla, R. (2016). Historical evolution and recent advances in precision farming. In R. Lal & B. A. Stewart (Eds.), *In: Soil-Specific Farming. Precision Agriculture* (pp. 1–35). Boca Raton, FL: CRC Press.

Navarro, N., Chateau, C., Amiotte-Suchet, P., & Lucas, Y. (2014). A comparison of extraction procedures for water-extractable organic matter in soils. *European Journal of Soil Science*, *65*(4), 520–530.

Nawaz, M. F., Bourrie, G., & Trolard, F. (2013). Soil compaction impact and modelling. A review. *Agronomy for Sustainable Development*, *33*(2013), 291–309. doi:10.1007/s13593-011-0071-8

Or, D. (2001). Who Invented the Tensiometer? *Soil Science Society of America Journal*, *65*(1), 1–3. doi:10.2136/sssaj2001.6511

Orchard, V. A., & Cook, F. J. (1983). Relationship between soil respiration and soil moisture. *Soil Biology and Biochemistry*, *15*(4), 447–453.

Perumpral, J. V. (1987). Cone penetrometer applications---A review. *Transactions of the ASAE*, *30*(4), 939–0944. doi:10.13031/2013.30503

Pett-Ridge, J., Nuccio, E., & Mcfarlane, K. (2018). Deeply rooted: Evaluating plant rooting depth as a means for enhanced soil carbon sequestration. Paper LLNL-CONF-750320. In I. Conference (Ed.), *on Negative CO2 Emissions (Gothenburg, Sweden, May, 2018), 13 pp* (pp. 21–24). Accessed at https://www.osti.gov/servlets/purl/1467801

Piper, P. M., Vogel, J. S., Fahrenkrug, M. T., McNamee, S. J., Pham, Q. N., & Lewin, G. C. (2015, April). Designing an autonomous soil monitoring robot. In 2015 *Systems and Information Engineering Design Symposium* (pp. 137-141). IEEE. doi:10.1109/SIEDS.2015.7116962

Raper, R. L., & Erbach, D. C. (1987). Bulk density measurement variability with core samplers. *Trans. ASAE*, *30*(4), 878–881. doi:10.13031/2013.30491

Rengel, Z. (1996). Tansley review No. *89. Uptake of Aluminium by Plant Cells. New Phytologist*, *134*(3), 389–406. doi:10.1111/j.1469-8137.1996.tb04356.x

Reumont, F. (2017). Development of a Portable Instrumentation System for in situ Assessment of Soil Respiration. Master of Science. *Dept. of Bioresource Engineering, McGill University.* Accessed at https://escholarship.mcgill.ca/concern/theses/sx61dp980

Rhoades, J. D., & Corwin, D. L. (1990). Soil electrical conductivity: effects of soil properties and application to soil salinity appraisal. *Communications in Soil Science and Plant Analysis*, *21*(11–12), 837–860. doi:10.1080/00103629009368274

Rochette, P., Desjardins, R. L., & Pattey, E. (1991). Spatial and temporal variability of soil respiration in agricultural fields. *Canadian Journal of Soil Science*, *71*(2), 189–196. doi:10.4141/cjss91-018

Rooney, D. J., & Lowery, B. (2000). A profile cone penetrometer for mapping soil horizons. *Soil Science Society of America Journal*, *64*(6), 2136–2139. doi:10.2136/sssaj2000.6462136x

Sadler, J., Evans, R., Buchleiter, G., King, B., & Camp, C. (2000). Venturing into precision agriculture. *Irrigation Journal*, *50*(3), 15–17.

Sanglerat, G. (2012). *The penetrometer and soil exploration*. Elsevier.

Sharma, S. K., Ramesh, A., Sharma, M. P., Joshi, O. P., Govaerts, B., Steenwerth, K. L., & Karlen, D. L. (2011). *Microbial community structure and diversity as indicators for evaluating soil quality* (B. A. In: Biodiversity & C. A. (E Lichtfouse, Eds.). doi:10.1007/978-90-481-9513-8_11

Spectrum Technologies. (2024a). SC 900 Soil Compaction Meter. FieldScout. Accessed 11 March 2024. https://www.specmeters.com/soil-and-water/soil-compaction/fieldscout-sc-900-meter/sc900/

Spectrum Technologies. (2024b). FieldScout TDR 350 Soil Moisture Meter.. Accessed 11 March 2024 at https://www.specmeters.com/soil-and-water/soil-moisture/fieldscout-tdr-meters/fieldscout-tdr-350-soil-moisture-meter-case-included-/

Spohn, M. (2015). Microbial respiration per unit microbial biomass depends on litter layer carbon-to-nitrogen ratio. *Biogeosciences*, *12*(3), 817–823.

Stafford, J. V. (2000). Implementing precision agriculture in the 21st century. *Journal of Agricultural Engineering Research*, *76*(3), 267–275. doi:10.1006/jaer.2000.0577

Veum, K. S., Sudduth, K. A., Kremer, R. J., & Kitchen, N. R. (2017). Sensor data fusion for soil health assessment. *Geoderma*, *305*, 53–61. doi:10.1016/j.geoderma.2017.05.031

Viscarra Rossel, R. V., Adamchuk, V. I., Sudduth, K. A., McKenzie, N. J., & Lobsey, C. (2011). Proximal soil sensing: An effective approach for soil measurements in space and time. *Advances in Agronomy*, *113*, 243–291. doi:10.1016/B978-0-12-386473-4.00005-1

Ward Laboratories, Inc. (2024). Accessed at https://www.wardlab.com/

Waters, T. R., Putz-Anderson, V., & Garg, A. (1994). Applications manual for the revised NIOSH lifting equation. Cincinnati, OH: U. *S. Department of Health and Human Services, Centers for Disease Control and Prevention, National Institute for Occupational Safety and Health, DHHS (NIOSH) Publication No.*, 94–110. doi:10.26616/NIOSHPUB94110revised092021

Weida, G., Whalley, W. R., Tian, Z., Liu, J., & Ren, T. (2016). A simple model to predict soil penetrometer resistance as a function of density, drying and depth in the field. *Soil and Tillage Research*, *155*, 190–198. doi:10.1016/j.still.2015.08.004

Yasuda, T., Yonemura, S., & Tani, A. (2012). Comparison of the characteristics of small commercial NDIR CO2 sensor models and development of a portable CO2 measurement device. *Sensors*, *12*(3), 3641–3655. doi:10.3390/s120303641

Yildirim, S., Birrell, S. J., & Hummel, J. W. (2006). Laboratory evaluation of an electro-pneumatic sampling method for real-time soil sensing. *Trans. ASABE*, *49*(4), 845–850. doi:10.13031/2013.21724

Zhang, N., Wang, M., & Wang, N. (2002). Precision agriculture—a worldwide overview. *Computers and electronics in agriculture*, *36*(2-3), 113-132. doi:10.1016/S0168-1699(02)00096-0

Zhang, X., & Cai, X. (2011). Climate change impacts on global agricultural land availability. *Environmental Research Letters*, *6*(1), 014014. doi:10.1088/1748-9326/6/1/014014

ZTractor, Inc. (2024). The world's 1st Autonomous Electric Tractor. Palo Alto, CA. Accessed at https://ztractor.com/

Appendix A: Arduino Mega script

```
#include <SDI12.h>
#include "HX711.h"
#include "CytronMotorDriver.h"

#define DOUT1  5
#define CLK1  4
#define DOUT  3
#define CLK  2
#define DATA_PIN 13

HX711 scale;
HX711 scale1;
SDI12 moistSensor(DATA_PIN);
CytronMD motor(PWM_DIR, 7, 6);
CytronMD vacuum2(PWM_DIR, 11, 8);
char[12] mode;

// unsigned long cleared=0;

unsigned long lastlowtime = millis();
int lowestCnt = 0; int stopped=0; int lastlow[5] = {1023, 1023, 1023, 1023, 1023};
int stopcnt=0; int linpotprev; int linpot; int torqueCnt=0;
float torque=0; int start=0; int valMultiplier=1;
int pressurePin = A0; int pressurePin2 = A3; int linpotPin=A1;
int anglePin=A2; int pressure = 0; const int trigPin = 9;
const int echoPin = 10; float duration, distance, distance2; float calibration_factor = -20000;
float calibration_factor1 = -20000; const unsigned int MAX_MESSAGE_LENGTH = 24;
byte readCO2[] = {0xFE, 0X44, 0X00, 0X08, 0X02, 0X9F, 0X25}; //Command packet to read Co2 (see
app note)
byte response[] = {0,0,0,0,0,0,0}; int lastmsgtime=0; int lastrx;
int stopCount=0; unsigned long itemstart; unsigned long itemstart1;
unsigned long lastCO2; unsigned long valCO2 =1; char down;
int mainValve0=41; int syringe=42; int directionValve1=43;
int blowoutValve=44; int CW4=45; int CCW5=46;
int inletValve=47; int ventPin=48; int on = 12;
int gnd = 24; float moistLoad; char reason[]="";
char stopreason[]="";  int delayTime; unsigned long timelast = millis();
int SDI12Step=0; char CS655[MAX_MESSAGE_LENGTH]; unsigned int message_pos1 = 0;
int lastlowest=1023;

struct LastTimes {
  unsigned long lastCO2;
  unsigned long lastMoist;
  unsigned long cleared;
  int noavail;
```

```
  int SDI12Step;
};
int sort_desc(const void *cmp1, const void *cmp2)
{
  int a = *((int *)cmp1);
  int b = *((int *)cmp2);
  return a > b ? -1 : (a < b ? 1 : 0);
}
void printValues(LastTimes& times, unsigned long delayTime, char Message[], int iteration=0);

unsigned long getValue(byte packet[])
  {
   int high = packet[3]; //high byte for value is 4th byte in packet in the packet
   int low = packet[4]; //low byte for value is 5th byte in the packet
   unsigned long val = high*256 + low; //Combine high byte and low byte with this formula to get value
   return val* valMultiplier;
  }


void sendRequest(byte packet[])
  {
    while(!Serial1.available())
    {
      // Serial.println("waiting for Software.serial port availability");
      Serial1.write(readCO2,7);
      delay(50);
    }
  int timeout=0; //set a timeout counter
  while(Serial1.available() < 7 ) //Wait to get a 7 byte response
    {
      timeout++;
      if(timeout > 10) //if it takes too long there was probably an error
      {
        while(Serial1.available()) //flush whatever we have
        Serial1.read();
        break; //exit and try again
      }
      delay(50);
    }
    for (int i=0; i < 7; i++)
    {
      response[i] = Serial1.read();
    }
}

void readSDI12() {
  CS655[MAX_MESSAGE_LENGTH];
  message_pos1 = 0;
  while (moistSensor.available() > 0) {
```

```
    char inByte = moistSensor.read();
    if (inByte == '\r' && (message_pos1 < MAX_MESSAGE_LENGTH - 1)) {
      CS655[message_pos1] = 'r';
      message_pos1++;
    }
    else if (inByte != '\n' &&  (message_pos1 < MAX_MESSAGE_LENGTH - 1)) {
      CS655[message_pos1] = inByte;
      message_pos1++;
    } else {
      CS655[message_pos1] = '\0';
      message_pos1 = 0;
    }
  }
}

void printValues(LastTimes& times, unsigned long delayTime, char Message[], int iteration=0) {
  itemstart=millis();
  while (millis()-itemstart < delayTime && (torqueCnt < 5 || mode!="Torque") && (stopCnt<5 ||
mode!="Lowering")) {
    scale1.set_scale(calibration_factor1);
    moistLoad=(scale1.get_units());
    Serial.print(millis());Serial.print(",");Serial.print(analogRead(A0)); Serial.print(",");
Serial.print(analogRead(A4)); Serial.print(",");Serial.print(analogRead(A1));
    Serial.print(","); Serial.print(analogRead(A2)); Serial.print(","); Serial.print(analogRead(A3)); delay(5);
    scale.set_scale(calibration_factor); Serial.print(","); Serial.print(scale.get_units(), 1);
     Serial.print(","); Serial.print(moistLoad); Serial.print(",");
    if (millis()-times.lastCO2 > 2000) {sendRequest(readCO2); valCO2 = getValue(response);
times.lastCO2=millis();}
    Serial.print(valCO2);Serial.print(",");
     if (torque >  50 || torque < -30) {
      torqueCnt++;
    }
    else {
      torqueCnt=0;
    }
    if (times.SDI12Step == 4) {
    if (moistSensor.available() > times.noavail) {
      times.noavail = moistSensor.available();
      times.cleared = millis();
    }
    if (millis() - times.cleared > 800) {
      times.noavail = 0;
      readSDI12();
      times.SDI12Step = 0;
    }
  }
  if (times.SDI12Step == 3) {
    moistSensor.sendCommand("0D!");
    times.SDI12Step = 4;
```

```
  }
  if (times.SDI12Step == 2) {
    while (moistSensor.available()) {
      moistSensor.read();
      times.cleared = millis();
    }
    if (millis() - times.cleared > 3000) {
      times.SDI12Step = 3;
    }
  }
  if (times.SDI12Step == 1 && moistSensor.available()) { times.SDI12Step = 2; }
  if (millis() - times.lastMoist > 6500) {
    while (moistSensor.available()) {
      // delay(5);
      moistSensor.read();
    }
    moistSensor.sendCommand("0M!");
    times.lastMoist = millis();
    times.cleared = millis();
    times.SDI12Step = 1;
  }
    Serial.print(CS655);Serial.print(","); Serial.print(millis()-timelast); Serial.print(","); Serial.print(millis()-
itemstart);Serial.print(",");Serial.print(Message);Serial.println(iteration);
    timelast=millis();
      if (lastlowest < 1000) {
        if (lastlowest-distance2>5) {lastlow[lowestCnt]=distance2;lowestCnt++;}
        if (lowestCnt > 4) {
          lastlowtime=millis();
          int lastlow_length = sizeof(lastlow) / sizeof(lastlow[0]);
          qsort(lastlow, lastlow_length, sizeof(lastlow[0]), sort_desc);
          lastlowest=lastlow[2]; //find median
          lowestCnt=0;

        }
        if (millis()-lastlowtime > 1500) {
          stopped=1;
          Serial.println("//stopped");
        }
        else {
          stopped=0;
        }
      }
      if (distance2<160 || moistLoad>35 || stopped==1) {
        stopCount=stopCount+1;
      }
      else {
        stopCount=0;
      }
```

```
    if (moistLoad>42) {
    digitalWrite(directionValve1, HIGH);
    delay(60);
    Serial.println("//overload");
    digitalWrite(directionValve1, LOW);
  }

  }
}


LastTimes lastTimes;

void setup() {
  Serial.begin(115200);
  while(!Serial) {
    delay(5);
  }
  Serial.println("//Serial started");
  Serial1.begin(9600);
    while(!Serial1) {
    delay(5);
  }
  Serial.println("//CO2 Connected");
  moistSensor.begin();
  delay(500);
  Serial.println("//SDI12 connected");

  lastTimes.lastCO2=millis(); lastTimes.lastMoist=millis(); lastTimes.cleared=0;
  lastTimes.noavail=0; lastTimes.SDI12Step=0;
  pinMode(53, OUTPUT); digitalWrite(53, HIGH);
  pinMode(24, OUTPUT); digitalWrite(24, LOW);
  pinMode(52, OUTPUT); digitalWrite(52, HIGH);
  pinMode(10, OUTPUT); digitalWrite(10, HIGH);
  scale.begin(DOUT, CLK);
  scale.set_scale();
  scale.tare(); //Reset the scale to 0
  scale1.begin(DOUT1, CLK1);
  scale1.set_scale();
  scale1.tare();
  Serial.println("//Scale connected");
  pinMode(LED_BUILTIN, OUTPUT); pinMode(mainValve0, OUTPUT); pinMode(directionValve1,
OUTPUT);
  pinMode(CW4, OUTPUT); pinMode(CCW5, OUTPUT); pinMode(on, OUTPUT);
  pinMode(syringe, OUTPUT); pinMode(trigPin, OUTPUT); pinMode(blowoutValve, OUTPUT);
  pinMode(inletValve, OUTPUT); pinMode(ventPin, OUTPUT);

  digitalWrite(mainValve0, LOW);
  digitalWrite(directionValve1, HIGH); digitalWrite(blowoutValve, HIGH);
```

```
  digitalWrite(CW4, HIGH); digitalWrite(CCW5, HIGH);
  digitalWrite(syringe, HIGH); digitalWrite(ventPin,HIGH);
  digitalWrite(inletValve, HIGH);
  Serial.println("//starting");
  delay(5000);
  digitalWrite(blowoutValve, LOW);
  delay(1000);
  digitalWrite(blowoutValve, HIGH);
  digitalWrite(CCW5, LOW);
  delay(1250);
  digitalWrite(CCW5, HIGH);
  digitalWrite(inletValve, HIGH);
  digitalWrite(mainValve0, HIGH);
  Serial.println("//starting2");
  // lastCO2=millis();
}

void loop() {

  static char message[MAX_MESSAGE_LENGTH];
  static unsigned int message_pos = 0;

  if (Serial.available()>0){
    while (Serial.available() > 0){
        char inByte = Serial.read();
        if ( inByte != '\n' && (message_pos < MAX_MESSAGE_LENGTH - 1) )
        {
          message[message_pos] = inByte;
          message_pos++;
        }
        else
        {
          message[message_pos] = '\0';
        message_pos = 0;
        lastrx=millis();
        Serial.println(message);
        start=1;
        }
      }
  }
  if (start==1) {
    scale.set_scale();
    scale.tare();
    delay(100);

    digitalWrite(ventPin, HIGH);
    delay(400);
    digitalWrite(blowoutValve, LOW);
    itemstart = millis();
```

```
  // lastCO2 = millis();

Serial.println("//Start clean");
printValues(lastTimes, 1000, "start clean");
  digitalWrite(inletValve, HIGH);
  digitalWrite(blowoutValve, HIGH);
  digitalWrite(mainValve0, LOW);
  digitalWrite(directionValve1, LOW);
  printValues(lastTimes, 250, "delay to dislodge");
  digitalWrite(mainValve0, HIGH);
Serial.println("//stopblowing and dislodge");
printValues(lastTimes, 2000, "stop blowing and dislodge");

  digitalWrite(mainValve0, LOW);
  stopCount=0;
  lowestCnt=0;
  stopped=0;
  lastlowest=1023;
  for (int i = 0; i <= 4; i++) {
    lastlow[i]=i;
  }
  Serial.println("//descending");
  itemstart=millis();
  itemstart1=millis();
  //45000
  mode="Lowering";
  printValues(lastTimes, 15000, "going down");
  mode="Default";
  digitalWrite(mainValve0, HIGH);

      printValues(lastTimes, 1000, "wait");

  scale.set_scale();
  scale.tare();
  digitalWrite(CW4, LOW);
  digitalWrite(CCW5, HIGH);
  Serial.println("//turn 1");
  mode="Torque";
  printValues(lastTimes, 1250, "calibrate 1");

  digitalWrite(CW4, HIGH);
  digitalWrite(CCW5, LOW);
  Serial.println("//motor2");
  Serial.println("//turn 2");
  printValues(lastTimes, 1250, "calibrate 2");
  digitalWrite(CW4, HIGH);
  digitalWrite(CCW5, HIGH);
  mode="Default";
  scale.set_scale();
```

```
  scale.tare();
  printValues(lastTimes, 100, "delay");
  Serial.println("//turning motors");
  mode="Torque";
  for (int i = 0; i <= 5; i++) {
    digitalWrite(CW4, LOW);
    digitalWrite(CCW5, HIGH);
    Serial.println("//turn 1");
    printValues(lastTimes, 1250, "CW ", i);

    digitalWrite(CW4, HIGH);
    digitalWrite(CCW5, LOW);
    Serial.println("//motor2");
    Serial.println("//turn 2");
    printValues(lastTimes, 1250, "CCW ", i);
  }
  mode="Default";
  digitalWrite(CW4, HIGH);
  digitalWrite(CCW5, HIGH);

  printValues(lastTimes, 1000, "wait before draw with syringe");

  Serial.println("//drawing air..");
  digitalWrite(ventPin,LOW);
  printValues(lastTimes, 200, "delay for draw");
  digitalWrite(syringe, LOW);
  printValues(lastTimes, 7000, "drawfirst");

  digitalWrite(ventPin, HIGH);
  digitalWrite(syringe, HIGH);
  printValues(lastTimes, 3500, "drawfirst2");

digitalWrite(ventPin, LOW);
printValues(lastTimes, 250, "delay for draw2");
  digitalWrite(syringe, LOW);
  Serial.println("//reading CO2");
  // printValues(lastTimes, 60000, "reading CO2");
  // reason="distance: "+int(distance2)+"Load: "+int(moistLoad);
  if (distance2<150) {
    printValues(lastTimes, 7000, "bottomed out");
  }
  else if (moistLoad>35) {
    printValues(lastTimes, 7000, "moistureoverload");
  }
  else {
    printValues(lastTimes, 7000, "reading CO2");
  }

  digitalWrite(ventPin, HIGH);
```

```
  //12 seconds total

  printValues(lastTimes, 1000, "wait");

digitalWrite(inletValve, LOW);
motor.setSpeed(255);
printValues(lastTimes, 2000, "12v");

digitalWrite(inletValve, HIGH);
motor.setSpeed(0);
printValues(lastTimes, 1100, "stop");

 digitalWrite(blowoutValve, LOW);
  printValues(lastTimes, 800, "high pressure");

  digitalWrite(blowoutValve, HIGH);
  printValues(lastTimes, 1000, "stopped blowing");


  printValues(lastTimes, 250, "delay");
  digitalWrite(mainValve0, LOW);
  digitalWrite(directionValve1, HIGH);
  Serial.println("//raising");
  goUpWait(lastTimes, 2500, "raising");

  scale.set_scale();
  scale.tare();
  digitalWrite(CW4, LOW);
  digitalWrite(CCW5, HIGH);
  Serial.println("//turn 1");
  printTorqueI(lastTimes, 1750, "break 1");

  digitalWrite(CW4, HIGH);
  digitalWrite(CCW5, LOW);
  Serial.println("//motor2");
  Serial.println("//turn 2");
  printTorqueI(lastTimes, 1750, "break 2");
  digitalWrite(CW4, HIGH);
  digitalWrite(CCW5, HIGH);

  goUpWait(lastTimes, 6000, "raising");

  Serial.println("//cleaning");
  digitalWrite(blowoutValve, LOW);
  printValues(lastTimes, 800, "cleaning");

  digitalWrite(blowoutValve, HIGH);
  printValues(lastTimes, 1000, "stopped blowing");
  printValues(lastTimes, 60000, "wait for values");
```

```
    for (int i=0; i <= 5; i++) {
      digitalWrite(syringe, LOW);
      digitalWrite(ventPin, LOW);
      Serial.println("//ventdraw1");
      printValues(lastTimes, 10500, "draw ", i);

      digitalWrite(ventPin, HIGH);
      digitalWrite(syringe, HIGH);
      Serial.println("//ventrelease1");
      printValues(lastTimes, 5000, "vent ", i);

      digitalWrite(ventPin, HIGH);
      printValues(lastTimes, 500, "delay for solenoid ", i);
      digitalWrite(blowoutValve, LOW);
      printValues(lastTimes, 800, "clean ", i);
      digitalWrite(blowoutValve, HIGH);
      digitalWrite(inletValve, HIGH);
      printValues(lastTimes, 1200, "wait ", i);
    }

    digitalWrite(ventPin, HIGH);
    printValues(lastTimes, 40000, "done");
    digitalWrite(mainValve0, HIGH);
    start=0;
    Serial.flush();
    Serial.println("//end");
  }

  delay(5);
}
```

Appendix B: Data parsing python script

```
import re
import numpy as np
import os
import csv
import itertools
from itertools import islice
from collections import deque
from bisect import bisect_left,insort
import numpy
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
from statistics import mean
excel = True
```

```
csvMode =True
bore=0.016
output=[]
filname='F30_10-19-2023.csv'
outfil='parsed'+filname
data=[]
yes=False
yes2=0
comments=[]

def median(s):
    sp = [nz for nz in s if nz!=0]
    Mnow = len(sp)
    if Mnow == 0:
        return 0
    else:
        return np.median(sp)
```

```
def RunningMedian(seq, M):
    seq = iter(seq)
    s = []
    s = [item for item in islice(seq,M)]
    d = deque(s)
    s.sort()
    medians = [median(s)]
    for item in seq:
        old = d.popleft()
        d.append(item)
        del s[bisect_left(s, old)]
        insort(s, item)
        medians.append(median(s))
    return medians

def nattylogfunc(x, A):
        y = 16.2-3.5*np.log(A*x)
        return y

with open(filname, 'r') as f:
        turning=0
        potDown=0
        stopcount=0
        porMes=0
        first=1
        twist=[]
        twists=[]
        Down=[]
        Downs=[]
        Downlow=[]
        Downfits=[]
        porosity=[]
        porosities=[]
        oops=[]
        lastdown=[0,20]
        lastdown[1]=20
        lastmark=20
        for line in f.readlines():
                print(line)
                if yes==True:
                        if line[0].isnumeric():
                                sampleN=line
                        else:
                                sampleN=""
                if yes2==2:
                        comment=str(line)[:-1]
                        yes2=0
                if yes2==1:
                        yes2=2
                if 'starting2' in line:
                        yes=True
                        yes2=2
                if 'end' in line and not 's' in line:
                        yes2=1

                if line[0].isnumeric() and len(line.split(','))>7:
```

```
split=line.split(',')
if 'r' in split[9] and len(split[9].replace('-','+').split('+'))==4:
	removeR=split[9].replace('r','')
	positive=removeR.replace('-','+-')
	split8=positive.split('+')
	moisture=[split8[1],split8[2],split8[3]]
else:
	moisture=['','','']
split[0]=int(split[0])/1000
split[1]=int(split[1])-100
split[2]=int(split[2])
split[3]=float(split[3])/1024*20
split[4]=(int(split[4])-160)/2
split[5]=(int(split[5])-100)*bore*bore*1000
newline=[]
newline.extend(split[0:9])
newline.extend(moisture)
newline.extend(split[10:11])
newline.append(split[12].replace('\n','').replace('\r',''))
output.append(newline)
print("Potdown"+str(potDown))
print("lastmark"+str(lastmark))
print(newline[3])
print((lastmark-newline[3])/int(newline[12]))
if turning==0 and "calibrate 2" in newline[13]:
	turning=1
if turning==1 and not "calibrate 2" in newline[13]:
	turning=2
	turntime=float(split[0])
if turning == 2:
	twist.append([float(newline[0])-
turntime,float(newline[6]),int(newline[4]),float(newline[3]),int(newline[12]),newline[13]])
if turning ==2 and "draw" in newline[13]:
	turning=3
if turning ==3 and "vent" in newline[13]:
	turning=0

if potDown==0 and newline[3] < 16 and newline[3] > 10:
	potDown=1
	Downtime=float(newline[0])
if potDown==1 or potDown==5:
	if "going down" not in newline[13]:
		potDown=2
		print("hey")
		Downlow.append(float(newline[3]))
if potDown==1:
	if first==1:
		first=0
	Down.append([float(newline[0])-
Downtime,float(newline[3]),float(newline[5]),float(newline[6]),newline[13]])
	if (lastmark-float(newline[3]))/int(newline[12])<0.001:
		stopcount+=1
	else:
		lastmark=float(newline[3])
		stopcount=0
	if stopcount>6:
```

```
					potDown=5
					oops.append([newline[13],comment])
					print(comment)

				lastdown=[-1*float(newline[12]),float(newline[3]),newline[13]]

				if potDown==2 and "done" in newline[13]:
					pass

				if porMes==0:
					if "reading" in newline[13] or "bottom" in newline[13] or "overload" in 
newline[13]:
						porMes=1
				if porMes==1 and "reading" not in newline[13] and "bottom" not in newline[13] 
and "overload" not in newline[13]:
					porMes=2
					porosityStart=float(newline[0])
				if porMes==2 and "raising" in newline[13]:
					porMes=3
				if porMes==2:
					porosity.append([float(newline[0])-
porosityStart,int(newline[1]),int(newline[2]),newline[13]])
			if 'end' in line and not 's' in line and excel==True:
				lastmark=20
				stopcount=0
				first=1
				data.append(output)
				comments.append(comment)
				porosities.append(porosity)
				porosity=[]
				porMes=0
				output=[]
				twists.append(twist)
				twist=[]
				Downs.append(Down)
				DownT=[list(i) for i in zip(*Down)]
				print(Down)
				try:
					fit = numpy.poly1d(numpy.polyfit(DownT[0], DownT[1], 2))
					parameters, covariance = curve_fit(nattylogfunc, DownT[0], DownT[1])


					Downfits.append(fit.c.tolist())

				except:
					Downfits.append([0,0])
					print(DownT)
				if potDown==0:
					Downlow.append(0)
				Down=[]
				potDown=0


pointN=0
for point in data:
	pointT=[list(i) for i in zip(*point)]
```

```
        pointT[6]=RunningMedian([float(x) for x in pointT[6]],3)
        pointT[7]=RunningMedian([float(x) for x in pointT[7]],5)
        data[pointN]=[list(i) for i in zip(*pointT)]
        pointN+=1

if excel==False:
    with open(outfil, 'w', newline='') as out:
        writer=csv.writer(out, delimiter=',')
        writer.writerow(["osTime (s)","porosity (kPa)","Current","linPot (cm)","angle (deg)","downLoad (N)","torque (N*m)","moistLoad (N)","CO2 (ppm)","VWC (m3/m3)","EC (dS/m)","temp (C)","execTime (ms)","taskMessage","Twist Time (s)","Torque"])
        writer.writerows(output)
elif excel==True:
    import jpype
    import asposecells
    jpype.startJVM()
    from asposecells.api import Workbook, FileFormatType
    workbook = Workbook(FileFormatType.XLSX)

    from string import ascii_uppercase as aup
    aup2=[*aup]
    for char in aup:
        for char2 in aup:
            aup2.append(char+char2)
    aup=aup2
    for l, point in enumerate(data):
        i=l+1
        worksheets = workbook.getWorksheets()
        print(comments[i-1])
        worksheets.add(comments[i-1]+"_"+str(i))
        for x, key in enumerate(["osTime (s)","porosity (kPa)","Current","linPot (cm)","angle (deg)","downLoad (N)","torque (N*m)","moistLoad (N)","CO2 (ppm)","VWC (m3/m3)","EC (dS/m)","temp (C)","execTime (ms)","taskMessage"]):
            column=str(aup[x])
            try:

    workbook.getWorksheets().get(i).getCells().get(column+"1").putValue(key)
            except:
                pass
        for y, instance in enumerate(point):
            row=str(y+2)
            for x, sensor in enumerate(instance):
                column=str(aup[x])

                if x==13:
                    try:

    workbook.getWorksheets().get(i).getCells().get(column+row).putValue(sensor)
                    except:
                        pass
                elif sensor != 0:
                    if x==0:
                        print(sensor)
                    try:

    workbook.getWorksheets().get(i).getCells().get(column+row).putValue(float(sensor))
```

```
                                        except:

        workbook.getWorksheets().get(i).getCells().get(column+row).putValue("")
                                else:

        workbook.getWorksheets().get(i).getCells().get(column+row).putValue("")
        print("Adding twists")
        for l, point in enumerate(twists):
                i=l+1
                pointT=[list(i) for i in zip(*point)]
                pointT[1]=RunningMedian([x for x in pointT[1]],3)
                print(pointT[1])
                pointT[1]=[float(x) for x in pointT[1]]
                print(type(pointT[1][4]))
                point=[list(i) for i in zip(*pointT)]
                print(point)
                print(type(point[1][1]))
                for y, instance in enumerate(point):
                        row=str(y+2)
                        for x2, key in enumerate(["Twist Time
(s)","Torque","Angle","linPot","execTime","Comment2"]):
                                column=str(aup[x2+x+1])

        workbook.getWorksheets().get(i).getCells().get(column+"1").putValue(key)
                        for x2, value in enumerate(instance):
                                column=str(aup[x2+x+1])
                                try:

        workbook.getWorksheets().get(i).getCells().get(column+row).putValue(value)
                                except Exception as e:
                                        print(e)
                                        print(value)
                                        pass
        print("Adding downs")
        for l, point in enumerate(Downs):
                i=l+1

                for y, instance in enumerate(point):
                        row=str(y+2)
                        for x2, key in
enumerate(["Downtime","DownPot","downLoad","moistLoad","Comment3","Lowest"]):
                                column=str(aup[x2+x+7])

        workbook.getWorksheets().get(i).getCells().get(column+"1").putValue(key)
                        for x2, value in enumerate(instance):
                                column=str(aup[x2+x+7])
                                try:

        workbook.getWorksheets().get(i).getCells().get(column+row).putValue(value)
                                except Exception as e:
                                        print(e)
                                        print(value)
                                        pass
        print("Adding Porosity")
        for l, point in enumerate(porosities):
                i=l+1
```

```
			for y, instance in enumerate(point):
				row=str(y+2)
				for x2, key in enumerate(["PoreTime","Porosity","Current","Comment4"]):
					column=str(aup[x2+x+13])

		workbook.getWorksheets().get(i).getCells().get(column+"1").putValue(key)
				for x2, value in enumerate(instance):
					column=str(aup[x2+x+13])
					try:

		workbook.getWorksheets().get(i).getCells().get(column+row).putValue(value)
					except Exception as e:
						print(e)
						print(value)
						pass
		print("Saving xlsx...")
		workbook.save(filname+"_thingies.xlsx")

if csvMode==True:
		print("Saving CSV...")
		i=0
		with open(filname+"_thingies.csv", "w", newline='') as f:
			fwriter=csv.writer(f, delimiter=',')
			fwriter.writerow(["Point","CO2	Max",	"MoistMax","ECMax","lowestdepth","Pressure
max","vacuum
max","currentmax","torquemax","loadcellmax","Downfit1","Downfit2","Downlow","slope@5","slope@7.5","
slope@10","slope@12.5"])
			for point in data:
				CO2Max=0
				MoistMax=0
				lowestdepth=20
				pressuremax=0
				vacuummax=0
				torquemax=0
				loadcellmax=0
				lowpresmax=0
				ECMax=0
				for line in point:
					if int(line[8])>int(CO2Max):
						CO2Max=int(line[8])
					try:
						if float(line[9])>float(MoistMax):
							MoistMax=float(line[9])
					except:
						pass
					try:
						print(float(line[10]))
						if float(line[10])>float(ECMax) and float(line[10])<1:
							ECMax=float(line[10])

					except:
						pass
					if float(line[3])<float(lowestdepth):
						lowestdepth=float(line[3])
					if int(line[1])>int(pressuremax):
						pressuremax=int(line[1])
```

```
				if "12v" in line[13] and int(line[1])>int(vacuummax):
					vacuummax=int(line[1])
				if "CW" in line[13] and int(line[6])>float(torquemax):
					torquemax=float(line[6])
				if float(line[5])>loadcellmax:
					loadcellmax=float(line[5])
				if float(line[1])>lowpresmax and "12v" in line[13]:
					lowpresmax=float(line[1])

			try:
				root=np.roots([Downfits[i][0],Downfits[i][1],Downfits[i][2]-15]).tolist()
				slope050=2*root[1]*Downfits[i][0]+Downfits[i][1]
			except:
				slope050=0
				print(Downfits[i])
			try:
				root=np.roots([Downfits[i][0],Downfits[i][1],Downfits[i][2]-12.5]).tolist()
				slope075=2*root[1]*Downfits[i][0]+Downfits[i][1]
			except:
				slope075=0
			try:
				root=np.roots([Downfits[i][0],Downfits[i][1],Downfits[i][2]-10]).tolist()
				slope100=2*root[1]*Downfits[i][0]+Downfits[i][1]
			except:
				slope100=0
			try:
				root=np.roots([Downfits[i][0],Downfits[i][1],Downfits[i][2]-7.5]).tolist()
				slope125=2*root[1]*Downfits[i][0]+Downfits[i][1]
			except:
				slope125=0


			porositiesT=[list(k) for k in zip(*porosities[i])]
			try:
				currentmax=mean([h for h in porositiesT[2] if h > 10])
			except:
				currentmax=0

	fwriter.writerow([comments[i],CO2Max,MoistMax,ECMax,lowestdepth,pressuremax,vacuummax,currentmax,torquemax,loadcellmax,Downfits[i][0],Downfits[i][1],Downlow[i],slope050,slope075,slope100,slope125])

			i+=1
```