



## **Hydroponic Greenhouse Crop Optimization**

**Miguel João Bordalo Fernandes**

Thesis to obtain the Master of Science Degree in  
**Electrical and Computer Engineering**

Supervisor(s): Prof. Bertinho Manuel D' Andrade da Costa  
Prof. João Manuel Lage de Miranda Lemos

### **Examination Committee**

Chairperson: Prof. João Fernando Cardoso Silva Sequeira  
Supervisor: Prof. Bertinho Manuel D' Andrade da Costa  
Member of the Committee: Prof. Alexandre José Malheiro Bernardino

**November 2017**



## Resumo

À medida que a população mundial continua a aumentar, há necessidade premente de melhorar a eficiência da exploração agrícola, particularmente em locais onde a agricultura depende da disponibilidade do solo.

Uma forma de superar esta questão é intensificar a produção, aumentando a velocidade de crescimento da safra e a capacidade de produção por metro quadrado. Os métodos de tecnologia e automação podem facilitar a gestão do sistema, permitindo a redução do esforço humano. Para além disso, o princípio do controlo ótimo pode melhorar a eficiência do processo de automação.

Com o intuito de controlar as condições de crescimento de culturas, foi projetada e construída uma estufa autónoma sem solo, equipada com um conjunto de dispositivos electrónicos. Pretende-se maximizar a massa final da colheita, minimizando os custos de controlo e o tempo de cultura. Foi desenvolvido um software de processamento de imagem para calcular o crescimento da produção agrícola, que foi modelado utilizando a função logística não-linear. O modelo de crescimento da produção desenvolvido permitiu a aplicação de controlo ótimo com base no princípio mínimo de Pontryagin para determinar a solução economicamente mais rentável.

Este projeto confirma as vantagens económicas da automatização dos processos agrícolas de culturas hidropónicas.

**Palavras-chave:** hidroponia, estufa, automação, controlo óptimo, função logística, planta, plantação, IoT



## Abstract

As the world population continues to increase, there is an urgent need for improvements on the efficiency of crop growth, particularly in places where agriculture practices are limited to soil availability.

An approach to overcome this issue is to intensify the production by increasing the crop growth speed and the production capacity per square meter. Technology and automation methods could improve the system management, allowing a reduction of human effort. Additionally, optimal control theory can boost the automation process efficiency.

An autonomous soilless greenhouse was designed and built providing a set of hardware devices in order to control crop growth conditions. It aimed to maximize the crop growth final weight while minimizing control costs and the cultivation period. Image processing software was used to estimate crop growth, modeled using the non-linear logistic function. The developed plant growth model allowed to apply optimal control based on Pontryagin's minimum principle to determine the most profitable solution.

This project confirms the advantages of using autonomous soilless technologies as a profitable and more manageable way of crop growth.

**Keywords:** hydroponics, greenhouse, automation, optimal control, logistic model, plant, crop, IoT



# Contents

Resumo . . . . .	iii
Abstract . . . . .	v
List of Tables . . . . .	xi
List of Figures . . . . .	xiii
Nomenclature . . . . .	xix
Glossary . . . . .	xxi
<b>1 Introduction</b>	<b>1</b>
1.1 World food demanding . . . . .	1
1.2 An automated soilless greenhouse . . . . .	1
1.3 Document outline . . . . .	2
<b>2 Hydroponic greenhouses</b>	<b>3</b>
2.1 The appearance of hydroponics . . . . .	3
2.2 Greenhouses and hydroponics at present . . . . .	4
2.3 Hydroponics . . . . .	5
2.3.1 Methodological approaches . . . . .	5
2.3.2 Growing mediums . . . . .	9
2.3.3 Plant nutrition . . . . .	10
2.3.4 Hydroponic monitoring . . . . .	11
2.4 Greenhouses . . . . .	12
2.4.1 Greenhouse architectures . . . . .	12
2.4.2 Greenhouse monitoring . . . . .	14
2.5 Plant growth models . . . . .	19
2.6 Crop optimal control . . . . .	21
2.6.1 Solve optimal problem numerically . . . . .	23
2.7 Modeling . . . . .	23
2.8 Linear Control . . . . .	26
2.8.1 Proportional Integral Derivative . . . . .	26
2.8.2 Anti-Windup . . . . .	27
2.8.3 Feedforward . . . . .	27

2.8.4 LQR . . . . .	28
<b>3 Experimental system</b>	<b>29</b>
3.1 Controller framework . . . . .	30
3.1.1 Controller overview . . . . .	31
3.1.2 Hardware implementation . . . . .	31
3.1.3 Software architecture . . . . .	32
3.1.4 Data Base . . . . .	35
3.1.5 Security . . . . .	36
3.1.6 Sensors and actuators setup and calibration . . . . .	36
<b>4 Growth environment</b>	<b>41</b>
4.1 Modeling greenhouse temperature . . . . .	41
4.1.1 Parameters identification . . . . .	43
4.1.2 Model validation . . . . .	48
4.2 Control greenhouse temperature . . . . .	49
4.3 Modeling greenhouse humidity . . . . .	52
4.3.1 Parameters identification . . . . .	53
4.3.2 Model Validation . . . . .	54
4.4 Humidity and temperature controller . . . . .	55
4.5 Light supplement . . . . .	58
4.6 Other environmental variables . . . . .	59
<b>5 Optimizing crop growth</b>	<b>61</b>
5.1 Logistic plant model . . . . .	61
5.2 Optimal control of plant growth . . . . .	64
5.2.1 Nutrients optimal control . . . . .	64
5.2.2 Nutrients and light optimal control . . . . .	67
5.2.3 Optimal harvest day . . . . .	69
5.2.4 Adjustment of the cost function . . . . .	69
5.3 Results and validation . . . . .	72
5.3.1 Control Sample . . . . .	74
5.4 Economic perspective . . . . .	74
<b>6 Conclusions</b>	<b>75</b>
6.1 Achievements . . . . .	75
6.2 Learned lessons . . . . .	76
6.3 Future Work . . . . .	76
<b>Bibliography</b>	<b>79</b>
<b>A Experimental datasets of plant growth</b>	<b>83</b>

<b>B Economic Perspective</b>	<b>85</b>
<b>C Hardware Devices</b>	<b>87</b>
C.1 Microcontroller . . . . .	87
C.2 Tiny computer . . . . .	88
C.3 Sensors . . . . .	88
C.4 Actuators . . . . .	92



# List of Tables

2.1	Ideal environmental temperature for different crop [2] . . . . .	15
B.1	Calculus to compute $\epsilon_1$ , $\epsilon_2$ and $\epsilon_{gh}$ parameters used in cost function (5.21). These allow to interpret the cost function in <i>euros</i> . . . . .	85
B.2	List of expenses to build and automate the greenhouse. . . . .	86



# List of Figures

2.1 Examples of nowadays hydroponic greenhouse applications. . . . .	4
2.2 Semi Hydroponic system . . . . .	6
2.3 The Nutrient Film Technique . . . . .	7
2.4 Comparison between NFT common PVC tube and proper rounded tube . . . . .	7
2.5 Deep water technique . . . . .	7
2.6 Aeroponics . . . . .	8
2.7 Aquaponics . . . . .	9
2.8 Hydroponics root support mediums . . . . .	10
2.9 The effect of pH . . . . .	11
2.10 Greenhouses most common design shapes. Illustration from <a href="#">globalspec.com</a> . . . . .	13
2.11 Polycarbonate sheets are a solid but flexible and easy to use material for greenhouses covers. Photo from <a href="#">How to Select and Buy Polycarbonate Sheet ebay post</a> . . . . .	14
2.12 General plant spectrum absorption. Plants absorption is bigger in reds and blues. Illustration from [18]. . . . .	16
2.13 Number of daily light hours according to day of the year and latitude in the globe. Illustration from <a href="#">Wikipedia Daytime</a> . . . . .	17
2.14 Generalized patterns of the growth rate (broken line) and generated biomass (solid line) in the three growth phases as a function of time. Graphic from [28]. . . . .	19
2.15 Continuous time approximations. From [41] . . . . .	26
2.16 Example of PI implementation with anti-windup enhancement. . . . .	27
3.1 Designed greenhouse and the hydroponic system to fit inside it . . . . .	30
3.2 Greenhouse plant system depicting the sensing and acting devices . . . . .	31
3.3 System architecture . . . . .	32
3.4 Software framework architecture describing the connectivity between different layers. . . . .	33
3.5 CSS user interface allowing to monitor and control greenhouse live conditions. . . . .	34
3.6 Archive Engine System Overview . . . . .	36
3.7 Linear calibration of TSL2561 lux sensor. . . . .	37
3.8 Linear calibration for DHT22 humidity sensors. . . . .	38
3.9 Linear mapping of the top window aperture according to the servo motor angle position. . . . .	39

3.10 Sample image after image processing. The pipeline identifies different plants and its contours, allowing thus to compute the corresponding LAI. . . . .	40
4.1 Temperature inside and outside the greenhouse after turning off an external heater. In this experiment the fan and the window aperture were forced to zero, and there was no solar radiation. It allows to compute the decay constant time characteristic of the greenhouse temperature evolution. Note however that $\hat{\theta}_{T1}$ time constant can not be computed directly from this graph but rather from the differences of these curves. . . . .	44
4.2 Prediction of the next sample $k + 1$ inside temperature based on the current state $k$ using the discrete model on equation (4.10) and the estimators computed based on experience of Figure 4.1 . . . . .	45
4.3 Discrete and continuous time inside temperature simulation using only the initial condition. The Figure proofs a correct transformation between the discrete and continuous time model identification. There is a slightly difference between the simulated and experience data. This can be explained as in the real greenhouse there are elements that store energy (like the greenhouse walls) that will release it over time, explaining thus the slower response of the real system compared to the simulation. . . . .	45
4.4 Experiment that took place to fit the term related to the influence of sunlight on the greenhouse inside temperature. The picture also presents the simulation based on the experiment results. The simulation predicts the temperature $T_{in}(t + 1)$ based on the sample at current time $t$ . . . . .	46
4.5 Influence of the window aperture ( $\Theta$ ) on the greenhouse inside temperature when a heater is operating inside at a constant power. The heater allows to maintain a measurably difference of temperatures. Only the green dots were considered for the linearization. . . . .	47
4.6 Influence of the fan speed on the greenhouse inside temperature. The experiment was performed were a heater was turned on inside the greenhouse in order to maintain a measurably difference of temperatures. The greenhouse aperture angle was maintained at its maximum aperture. . . . .	47
4.7 Influence of the mist maker on the greenhouse inside temperature (blue line). The remaining actuators were turned off and there was no sunlight effect. It is possible to notice the delay on the temperature after acting. The green line shows the corresponding discrete model to the same input data. . . . .	48
4.8 Two days performance of the continuous and discrete obtained models. All the curves represent experimental data except for $T_{inDis}$ and $T_{inCon}$ . $T_{inCon}$ is the inside predicted temperature using the continuous model based only on the initial condition of the inside temperature at time $t = 0$ . $T_{inDis}$ predicts the next inside temperature based on the last available experimental sample $T_{inExp}(k)$ . The root mean square error between the continuous and discrete simulation with the experimental data is $RMSE_{con} = 0.75^\circ$ and $RMSE_{dis} = 0.0283^\circ$ respectively. . . . .	49

4.9 Simulated PI temperature controller. The fan proportional term is fed by the difference from the aperture window before and after it's saturation so that it only actuates when the window is saturated. . . . .	50
4.10 Simulation of the temperature controller described in Figure 4.9. Within the controller acting limits the system stabilizes around the desired setpoint ( $t=[12;20]$ ). However, the actuators can only lower the temperature less than $2^{\circ}\text{C}$ in respect to the open loop response due to saturation limits ( $t=[20;24]$ ). . . . .	51
4.11 Experimental versus simulated controller results with the same controller gains. Because the model does not follow exactly the reality neither will the actuators. The experimental results, within the time highlighted in grey, revealed a good track of the setpoint when the actuators are not saturated. . . . .	51
4.12 Performance of the inside humidity discrete model obtained after manually tunning the actuators parameters. $H_{inDis}$ predicts the next inside humidity based on the last available experimental sample $H_{inExp}(k)$ according to equation (4.23). The root mean square error of the difference from these two signals is $RMSE_{H_{discrete}} = 3.03\%$ . . . . .	55
4.13 Greenhouse temperature and humidity controller. The temperature is controlled with a simple PI. The humidity controller has a feedforward term because it is more sensitive to the ventilation actuators controlled by the temperature PI. . . . .	56
4.14 Humidity controller detailed view. It is split in positive and negative control that feed two different actuators . . . . .	57
4.15 Simulated response to the greenhouse humidity and temperature controller represented in Figure 4.13. The controller reveals to successfully approach the control variables to its respecting setpoints. This is only not true when the actuators are saturated. For example around 17-20h. . . . .	57
4.16 Performance of the Humidity and Temperature controller (during a normal day). The temperature follows well the setpoint within its actuators limits. The setpoint is automatically changed according to the outside temperature to avoid the controller to operate in saturation conditions. The humidity controller is more oscillatory mainly due to the fan non linearities . . . . .	58
4.17 Detailed view of Figure 4.16 from 10 to 15 hours. . . . .	58
 5.1 Example of fitting crop growth experimental data to logistic function using <code>fminunc</code> function of Matlab. . . . .	63
5.2 Fitting to linear equation (5.1b) using experimental dataset of appendix A.1. Dots represent experimental data and the plane represents the corresponding fitting. . . . .	63

5.3	Crop growth optimal control (subscript <i>opt</i> ) with the nutritive solution concentration as the only input variable. Other non-optimal solutions (subscript <i>no</i> ) are presented, as well as the corresponding cost function ( $J$ ). The last non-optimal solution actually shows a higher cost function, suggesting thus that the numerical algorithm converges to a sub-optimal solution. . . . .	66
5.4	Cost function evaluated at different time for the bang-bang control of the nutritive solution concentration. The optimal value of the switch is given by $t_s^*$ . Moreover, the wide concavity of the curve suggests a robust controller. . . . .	66
5.5	Hamiltonian for the two input variables optimal control. As it is a linear plane, the combination of optimal inputs $(u_1^*, u_2^*)$ that maximizes the Hamiltonian is in the boundary of the input limits given by $\mathcal{U}$ . . . . .	68
5.6	Optimal control (subscript <i>opt</i> ) with the nutritive solution concentration ( $u_1$ ), and the amount of daily light received ( $u_2$ ) as the input variables. Other non-optimal solutions (subscript <i>no</i> ) are present as well as the corresponding cost function ( $J$ ). . . . .	68
5.7	Cost function evaluated when switching the control variables at different time combinations $(t_{s1}, t_{s2})$ . The optimal combination is given at $(t_{s1}^*, t_{s2}^*)$ . Again, the wide concavity of the surface suggest a robust controller. . . . .	69
5.8	Cost function ( <i>euros</i> ) of the optimal solution for different experiments time durations. Computing the harvest time that leads to the maximum profit, allows to relax the final time fixed constrain of Pontryagin's principle. . . . .	70
5.9	Optimal solution for quadratic cost function. This result allows to have continuous control inputs in contrast to the bang-bang solution. Note that, despite the cost function value is not positive, it does not mean that there is no actual profit once this value is not in euros. . . . .	72
5.10	Final experiment to validate the fitted plant growth model as well as the optimal controller solution proposed. The root mean square error between the experimental growth curves compared to simulation is around 10% or 9g. . . . .	73
A.1	Blue dots represent the experimental dataset allowing to model plant growth. In green the simulated model fit. In light blue the maximum growth level computed only with the respective experiment data. In red the maximum growth level computed by the global model. M is the maximum growth level and K the rate of growth. N, L, T and H are the average values of the nutritive solution concentration, the amount of light received daily, the temperature and the humidity inside the greenhouse, respectively. Finally, RMSE stands for the root mean square error of the difference between the experimental and the simulated data. . . . .	84
C.1	Arduino Mega characteristics . . . . .	87
C.2	Raspberry Pi v3 characteristics . . . . .	88
C.3	Spectral Responsibility comparison between a PAR sensor and a Lux sensor . . . . .	89
C.4	TSL2561 Luminosity Sensor characteristics . . . . .	89

C.5 RHT03 — DHT22 Air humidity and temperature sensor characteristics . . . . .	90
C.6 DS18B20 waterproof temperature sensor characteristics . . . . .	90
C.7 Waterproof conductivity probe designed at IST university. . . . .	90
C.8 Analog SEN0161 pH meter sensor characteristics . . . . .	91
C.9 Analog MG811 Carbon dioxide sensor characteristics . . . . .	91
C.10 Raspberry pi Camera characteristics . . . . .	92
C.11 Yate Loon 140mm Medium Speed Fan (D14SM-12) characteristics . . . . .	93
C.12 Fan circuit schematic allowing to have a $V_{cc} = 12$ Volt PWM controlled with Arduino input $V_i = 5$ V. . . . .	93
C.13 HS-322HD servo motor characteristics . . . . .	93
C.14 Mist Maker characteristics . . . . .	94
C.15 FIT0200 water pump characteristics . . . . .	95
C.16 Air pump characteristics . . . . .	95



# Nomenclature

## Greek symbols

- $\mathcal{M}$  Plant maximum biomass achieved.  
 $\mathcal{U}$  Space of admissible controls.  
 $\Theta$  Angle of the roof aperture window.

## Roman symbols

- $c_{air}$  Air heat capacity.  
 $H$  Average humidity over the plant grow cycle.  
 $H$  Heater actuator.  
 $J$  Cost function value.  
 $K$  Plant growth rate.  
 $L$  Average DLI value over the plant grow cycle.  
 $L$  Ligth received by the plants.  
 $M$  Mist maker actuator.  
 $N$  Average nutritive solution concentration over the plant grow cycle.  
 $T$  Average temperature over the plant grow cycle.  
 $t$  Time.  
 $T_{in}$  Greenhouse inside temperature.  
 $T_{out}$  Greenhouse outside temperature.  
 $u$  Control inputs.  
 $V_{fan}$  Speed of the ventilation fan.  
 $w$  Plant biomass.

## Subscripts

- 0 Initial.
- ap* Window aperture.
- exp* Experiences.
- f* Final.
- gh* Greenhouse.
- in* Inside the greenhouse.
- out* Outside the greenhouse.

### **Superscripts**

- \* Optimal solution.
- T Transpose.

# Glossary

<b>CSS</b>	Control System Studio
<b>DB</b>	Data Base
<b>DLI</b>	Daily Light Integral
<b>DWC</b>	Deep Water Culture
<b>EPICS</b>	Experimental Physics and Industrial Control System
<b>GUI</b>	Graphic User Interface
<b>LAI</b>	Leaf Area Index
<b>LQR</b>	Linear Quadratic Regulator
<b>LSQ</b>	Least-Squares Method
<b>MIMO</b>	Multiple Input Multiple Output
<b>NFT</b>	Nutrient Film Technique
<b>PAR</b>	Photosynthetically Active Radiation
<b>PID</b>	Proportional Integral Derivative
<b>PPFD</b>	Photosynthetic Photon Flux Density
<b>PV</b>	Process Variable
<b>Pi</b>	Raspberry Pi
<b>RMSE</b>	Root Mean Square Error



# **Chapter 1**

## **Introduction**

### **1.1 World food demanding**

Human beings are unconditionally dependent on plant growth as a food production source. As the world population continues to increase, so does the need for improvements on the efficiency of crop growth. On the other hand, the concentration of human population around urban areas tends to limit the agriculture practice close to where people live. Nevertheless, there has been an increase of public awareness on the source of food and a growing desire to eat fresh and locally.

Taking these issues into account, an engineering problem arises: how to make the growth of plants more efficient and possible in different and challenging environments, such as where soil is limited.

Greenhouses are transparent constructions that allow to grow plants in a closed environment, usually to increase the inside temperature by action of the sun. Hydroponics is the method of growing plants without soil. This way, the plants' roots will get the nutrients and oxygen from a water oxygenated solution with dissolved mineral nutrients. Greenhouses and hydroponics are not a novelty, but there is a lot of room for improvement with the advance of technology.

By using sensors and actuators, autonomous crop growth can be done [1], reducing thus the burden of time-consuming and physically challenging agriculture tasks. It also allows to increase production efficiency operating in an optimal way by reducing acting costs and maximizing crop growth.

Growing plants in extreme conditions, where natural crop growth would not be possible at all, is another interesting possibility. Places with extremely adverse weather conditions - even in space - are motivational examples.

The author of this document, as a person who is passionate about nature, felt tremendously motivated by this possibility to apply engineering to plant growth.

### **1.2 An automated soilless greenhouse**

The main goal of this work is to develop an efficient automated soilless greenhouse. A feature allowing users to remotely control and monitor crop growth and environmental conditions should be available.

Regarding this, four main objectives should be accomplished:

1. Build a physical smart greenhouse with all required sensors, actuators and computers.
2. Fit experimental data to environmental and crop growth models.
3. Develop the optimal control software to efficiently reduce acting costs while increasing crop growth.
4. Connect the greenhouse to the Internet and implement a graphical user interface.

This project could be extended by applying the optimal controller developed to large-scale greenhouses. Furthermore, these scalings could give us, not only an insight on the engineering adequacy, but will also allow testing its economic viability. Ultimately, these conclusions could be an answer to the present increase of the world food demand.

### **1.3 Document outline**

After a short introduction, chapter 2 is dedicated to the fundamentals of planning and materializing this project. Along with a review on how plants grow in greenhouses using hydroponics technologies, some insights on plant models and optimal control are provided. Chapter 3 details the experimental system design developed through this work, allowing to evaluate the performance of the controller. Chapter 4 explores the approach on how to control the temperature and humidity environmental variables, and the corresponding results on the physical system. Later, chapter 5 clarifies the process of modeling plant growth and how it allowed to control it in an optimal way, also providing the final results as well as an economic insight on the greenhouse profitability and industrial application. Finally, chapter 6 concludes critically, resuming the main accomplishments and possible future work.

# **Chapter 2**

## **Hydroponic greenhouses**

### **2.1 The appearance of hydroponics**

During ancient times, people faced many challenges while gardening. Being a curious animal and facing dependency on crops, man wanted to find out what made plants grow. The soil was a mysterious material that somehow provided the right conditions for plants to grow. Plagues and pestilence often reduced or even destroyed the yields of plants that societies heavily depended on for their well-being. When crops failed, societies suffered famine and death. Such crop failures led to wars and deaths. This points out that humans are dependent on crops for their survival.

If man knew more about the causes of crop failures, he could try to prevent them. This became the basis of agriculture — to find out the reasons for plants to thrive so that man could cultivate plants under favorable conditions, which would lead to abundant production [1].

Rapidly ancient civilizations became aware that water was essential for any agricultural practices, so populations gathered in areas that had an abundant source of water that could be used for growing plants. The first records of hydroponics refer to the Egyptian. Several hundred years BC, growing plants in water was already a reality. Also, a form of hydroponics was established with the hanging gardens of Babylon, the floating gardens of the Aztecs in Mexico and the Chinese.

In the early times, scientists discovered that adding minerals to water in plant grow without soil would favor their growth. This became “nutriculture,” where plant roots were immersed in a water solution containing salts. From 1925 to 1935, laboratory-scale nutriculture was expanded to commercial scale production of crops, known today as hydroponics.

Hydroponics had a huge boost during World War II. The U.S. army developed a strong interest in using this method to produce fresh vegetables in remote installations [2].

Since before 1966, many mechanical and electronic devices started to be introduced assisting in commercial greenhouse operation [2]. Automatic light control, ventilation, humidity, carbon dioxide control and automatic watering was already a reality [1].

Hydroponic greenhouse growing is now worldwide. Some of the largest vegetable greenhouse production regions include Holland, Spain, England, Canada, United States, Mexico, Turkey, China, Aus-

tralia, and Middle Eastern countries. Holland has more than 10,000 hectares of greenhouse production, which includes ornamentals and flowers. Canada has about 1,000 hectares of greenhouse hydroponic vegetable production and the United States 600 hectares. China is rapidly expanding its greenhouse production with approximately 1200 hectares [1] presently. Advanced automatic control is already implemented in some of these systems like the one presented in Figure 2.1a.

Not only commercial use of greenhouses with hydroponics has grown. With the increased interest in home hydroponics, a vast number of designs are marketed for all types of crops. Systems being established on rooftops of buildings in the centers of cosmopolitan cities (figure 2.1b), or industrial circular hydroponic structures, are some innovative ideas. The possibility for people to grow their own food in main cities is a reality. Alternatively, even sending small systems to space is being a target of study by NASA [3].



(a) Gandpa Dome hydroponic farm in Japan. The plants are seeded in the center of the circular nutrient bath and are collected in the outside edge when ready. Photo from [youtube video](#)

(b) Portuguese Agro-Designer João Henriques (Biovivos) using rooftops to produce local micro-greens (photo by Luís Barra in an entrevue for Visão newspaper)

Figure 2.1: Examples of nowadays hydroponic greenhouse applications.

## 2.2 Greenhouses and hydroponics at present

Greenhouses and hydroponics have been widely used for several decades to increase crop growth. In Jako's 1966 review of greenhouse food production [2], besides these two techniques, other strategies of environmental control were being tested. Carbon dioxide started to be object of study on plants control, in addition to humidity, light and temperature. These variables were controlled through basic mechanical and electronic devices such as timers and fans.

More recently, though, food production benefits from smarter and more complex and efficient technology. Countries such as the US, The Netherlands, and China have been investing heavily in crop growth automation technology which brings much higher yields.

Born in the MIT University (USA), the Open Agriculture "OpenAg" open source project called "food computer" [4] have collaborators, entitled as nerd farms, all over the world. This is a very controversial project and is similar to the one being developed in this thesis project. Nerd Farmers can download the building material list along with instructions to assemble themselves a small greenhouse filled with sensors and electronics. The framework including the software is available for all new nerd farmers that want to build their own food computer. With this broad and evolving community progresses, new ideas

are constantly shared. These researchers are also working on bigger scales projects like the "Food server" and the "Food Data center".

As mentioned, The Netherlands is also a country that does a lot of research and implements smart systems in their food production. Van Straten, Van Ooteghem and Van Henten are some names of authors researching deeply about optimal control of crop growth in greenhouses. The referred documents [5],[6], [7], [8] have complex and detailed crop growth and variable models.

NASA is also developing systems that allow fresh food to grow in micro gravity environment of space. NASA's plant experiment, called Veg-01, already proved the viability of the process [3].

In Portugal, there are several businesses using hydroponics and greenhouses. For example, GroHo [9] is a dedicated hydroponic company focusing on the city user. Biovivos [10] grows micro greens, i.e. plant sprouts, in Lisbon buildings rooftops, using small greenhouses and some automation processes. Another interesting Portuguese company is CoolFarm [11] which focus on control and sensing. CoolFarm does image processing using a fisheye camera. It allows following crop growth and to detect crop diseases in real time.

Finally, a good deal of information about greenhouses and hydroponics is available for every kind of reader. Books such as [1, 12] summarize these fields pretty well.

## 2.3 Hydroponics

The word "hydroponics" is derived from two Greek words hydro ("water") and ponos ("labor") — "water working".

Hydroponics is the method of growing plants without soil. Instead, the plants' roots will get the nutrients and oxygen from a water solution oxygenated with dissolved mineral nutrients. Hydroponics allows the plant to uptake its food with tiny effort as opposed to soil where the roots must search out the nutrients and extract them. Controlling and answering plants needs is easier and more accurate with this method of crop growth.

In hydroponics it is important to control the nutrients' concentration in the water solution as well as the value of Ph at the plant roots. Oxygen is essential for the plant to perform respiration at the root level. It is also necessary to ensure plant support. There are various hydroponics approaches that address these problems differently.

### 2.3.1 Methodological approaches

In this section some of the most common hydroponic technologies are listed and explained in order to give an insight on available methods and the ones that best suit this work.

#### Semi hydroponic system or drip system

The principle behind hydroponic drip systems is the simplest and the one closest to traditional gardening. Plants are supported by an inert material instead of soil. Then, vital nutrients are added to a tank of water

to create a balanced nutrient reservoir, which is kept separate from the plants. This water is then pumped throughout a network of tubes and is released to the plants individually, see Figure 2.2.

When choosing the growing medium in this method it is crucial to check its aeration characteristics. It is also important to take into account the medium water holding capacity combined with good drainage.

This technique is more common whenever plants have extensive and a large root system. Examples of this are tomatoes, peppers or orchids flower plants. For simple and less expensive solutions this is an easy go one.

Some disadvantages are a smaller control over the nutritive solution over the root temperature, and humidity of the substrate. The possibility of clogs is also bigger.

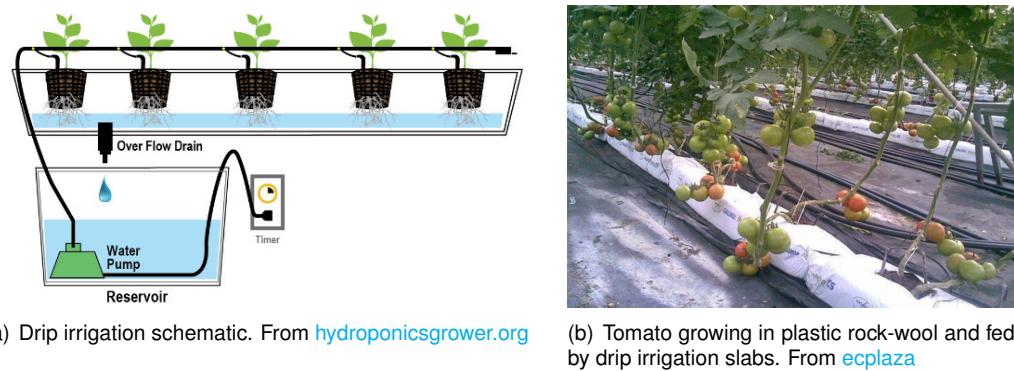


Figure 2.2: Semi Hydroponic system

### Nutrient Film Technique

The principle of the Nutrient Film Technique (NFT) system is to keep a constant flow of a thin layer of nutrient solution through the plant's roots in a grow tray. This is a recirculation method returning the solution from the grow tray back to the nutrient reservoir below. The grow tray is slightly tilted  $2 - 4^\circ$  so that water moves with gravity. Then water is pumped back to the upper part of the grow tray, see Figure 2.3.

The flow of the nutrient solution should be between 1 and 2 liters per minute providing the adequate oxygenation. Because roots can grow along the channel, they can block water passage. Therefore, it is not suitable for large root plants. Furthermore, the channel length cannot be too long as well, because the last plants will be deprived of oxygen and nutrients, making it more suitable for short size implementation and home systems. Efficiency, on the other hand, is excellent because the closed channel limits evaporation. Maintenance is easier once the reservoir is independent from the system and plants can be easily removed from it.

Initially the NFT system was selected for this work mainly for technical easy accessibility and maintenance, permitting the crop to be inside the greenhouse while the reservoir is outside. Furthermore, the control over the nutrient in the root zone is more accurate. However, there were some problems of water leakage. Moreover, NFT system would not allow to simultaneously grow plants with different nutritive solution concentration. Therefore, in a second approach the Deep Water culture method, explained later, was used.

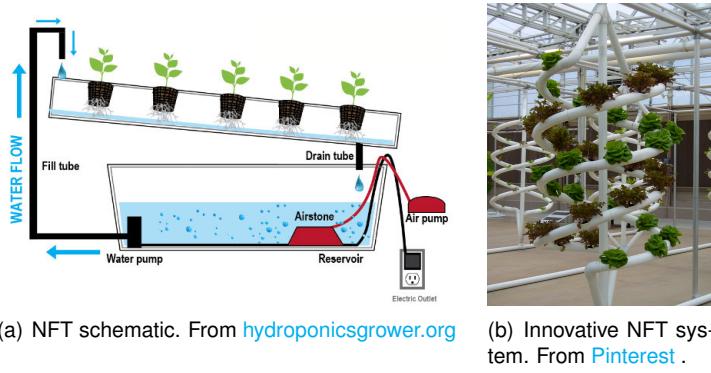


Figure 2.3: The Nutrient Film Technique

When using NFT systems choosing the pipe is crucial. Although it is not mandatory to use the NFT rounded commercial tubes, they present some advantages over the popular PVC tubes. The NFT rounded pipes present some stretch marks that produce a more uniform flow of nutrient solution increasing the contact area with the plant roots and also increasing oxygenation. These are made of polypropylene white outside and dark inside to reduce brightness in the roots which is bad for the plant, see Figure 2.4.

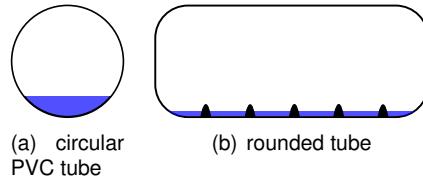


Figure 2.4: Comparison between NFT common PVC tube and proper rounded tube

### Deep Water Culture

In Deep Water Culture(DWC) method, also known as floating or RAFT, plants are supported by baskets which fit into foam sheets directly upon the reservoir of water with nutrients. The nutrient solution is circulated and aerated with an air pump from below to maintain high level of oxygen and avoid stagnation, see Figure 2.5.

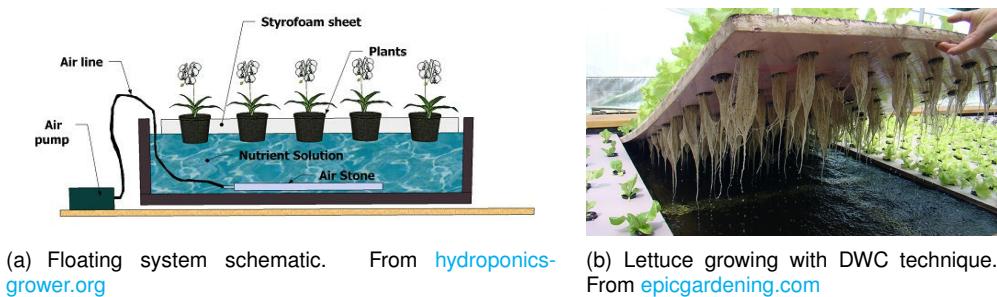


Figure 2.5: Deep water technique

DWC is more economic for bigger implementations with large quantities of short size and light crops.

From the economic perspective this method is also energy efficient, as an air pump consumes less energy than water ones. This system is also suitable for simple DIY home projects. Finally, there is also the possibility to prepare several reservoirs with different water solution within the same system. Because of these advantages, DWC was adopted as a second approach for this project. For long roots plants DWC is not appropriated, although this is not the case of lettuce.

## Aeroponics

From all the hydroponic systems aeroponics is the most technologically advanced. Plants are supported in the top of the reservoir with their roots suspended in the air. Roots are misted every few minutes with nutrient solution, see Figure 2.6.

This means that plants' roots are far more oxygenated than in any other system, and this helps the plants achieve faster growth rates. Plants will also adjust to their feeding methods and will grow more roots to enable them to absorb more nutrients from the mist. In this method it is essential to maintain a saturated air humidity to prevent plant dehydration. The greatest advantage of aeroponics is to allow the growth of roots and tuberous vegetables, like potatoes or carrots.

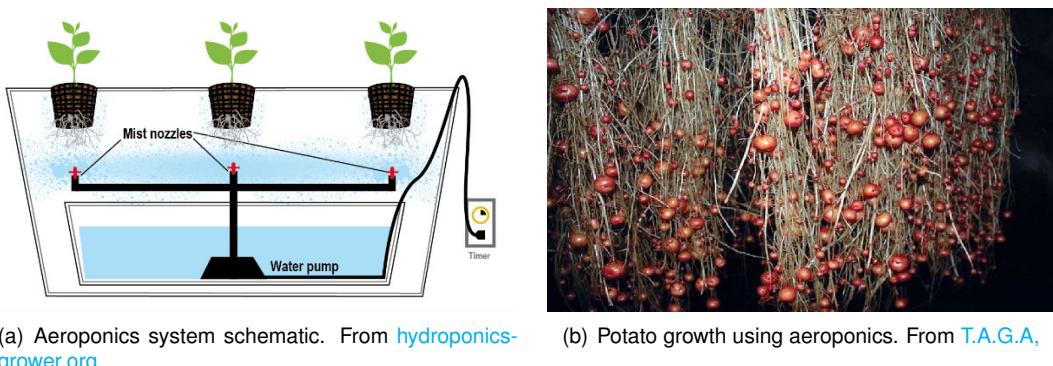
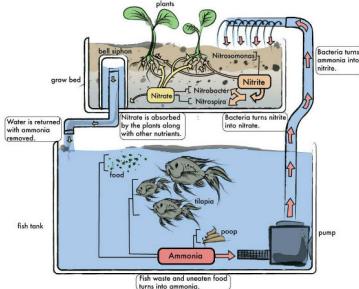


Figure 2.6: Aeroponics

## Aquaponics

Aquaponics relies on using fishes in symbioses with plants. This arrangement creates an ecosystem that benefits both plants and the aquatic life swimming below. The fish are essentially living in the nutrient reservoir. Their excrement releases ammonia into the water. Ammonia is toxic to fish, but it is a natural fertilizer for plants. When bacteria metabolize the ammonia in the water, it will turn into nitrate, one of the nutrients that are essential for healthy plant growth. This water is then pumped into the grow tray where it is fed to the plants.

As plants absorb the nutrients, they remove the toxins from the water. This water is then drained straight back into the nutrient reservoir, where the fish live. Aquaponics is a bolder method because maintaining this balance is not so easy.



(a) Aquaponics schematic. From [hydroponicsgrower.org](http://hydroponicsgrower.org)

Figure 2.7: Aquaponics

### 2.3.2 Growing mediums

For some hydroponic systems, plants roots have to get some kind of supported medium to grow. There are countless solutions for this, from inert to biodegradable, from natural to synthetic. The most popular ones will be presented here. A suitable medium has to have proper physical characteristics like density and porosity ability to retain and drain water. Being stable, not salty, and having a neutral pH are some of the chemical requirements.

1. **LECA** stands for Lightweight Expanded Clay Aggregate and is an extremely coarse growing medium. It is made of expanded clay pellets that hold water by virtue of its porosity and surface area. These mediums are pH neutral and reusable. This is one of the most used medium in home hydroponics solutions. This medium is more suitable for plants with ticker roots.
2. **Rockwool** is made from basaltic rock (solidified lava) that is liquefied at  $1500^{\circ}C$  and then extruded into filaments. It is slightly alkaline but inert and does not decompose. With 95% pore spaces [13], it has good water-holding capacity. The pH between 7 and 8.5 must be adjusted before seeding or planting by saturation with an acid nutrient solution to reach an optimum pH. Rockwool for hydroponics is not the same as the one used as insulating material, it is a more dense and compact version of it. Although it is a very popular and standard medium used in hydroponics, there has been a decline in its use mainly due to environmental concerns and because it can be irritating to the skin and lungs.
3. **Coco Coir** is compressed and dried coconut palm skin. Its porosity and aeration are good. This substrate is now becoming the principal one used in large greenhouse operations because it is considered as “sustainable” agricultural technology. The reason for that is that coconut skin is a renewable and biodegradable product. This also means that it is not an inert medium therefore it can have some high levels of phosphorus and low levels of calcium and magnesium. It also loses volume with time.
4. **Perlite** comes from volcanic rocks. It is very light-weight and it is sterile due to a process of heating. A coarse particle size between 2 mm and 3 mm is best for hydroponics. It contains no nutrients and has a pH of 6.0–8.0.

5. **Peat** is partially decomposed swamp vegetation. In general it is acid and has high capacity of water retention.



Figure 2.8: Hydroponics root support mediums

### 2.3.3 Plant nutrition

One of the basic needs for plant growth, either in soil or hydroponics, is the availability of required nutrients at its roots. Minerals that are required for plant growth and development are termed essential. From those, carbon (C), hydrogen (H), oxygen (O) and (N) nitrogen are called organic. With the exception of nitrogen, all the minerals come from air and water. The remaining essential elements come from the soil or nutrient solution in case of hydroponic culture. These include nitrogen (N), phosphorus (P), potassium (K), sulfur (S), calcium (Ca), and magnesium (Mg), which are required in relatively large amounts and therefore designated as macro elements. The others, needed in very small amounts, are termed micro elements. These include iron (Fe), manganese (Mn), zinc (Zn), copper (Cu), boron (B), molybdenum (Mo), and chlorine (Cl). Nickel (Ni) is now believed to be also an essential element.

Plants are composed by 90% to 95% of its weight in the organic elements, C,H and O. However, for them to grow, the other elements are essential combined with favorable environmental conditions.

In hydroponics, all these elements are dissolved in water accordingly with some nutrient recipes. These can change with different plant needs. If any of these elements is in deficiency or excess, disorders will occur in the plants. These disorders will be expressed as symptoms. Symptoms ( displaying specific colors or deformities) will give a clue that plants are under stress and the nutrients must be corrected to avoid loss in production.

Nutrients concentration and water quality should be controlled. It is thus important to check if the water already has some elements or not, and its pH level. The best solution, although not very common, is to use distilled water if possible.

For optimal growth to take place, the nutrient concentration and pH must be consistently balanced over time to insure that plants have what they need, when they need it.

### 2.3.4 Hydroponic monitoring

To control an hydroponic medium some intrinsic variables should be tracked. Water conductivity, ph, and temperature are the main crucial variables to take into account.

#### Water conductivity

The easiest way to keep on top of the nutrient solution is to take a measurement of Parts Per Million (PPM) and Total Dissolved Solids (TDS). This measurements are related with Electrical Conductivity (EC) of a solution and can be used as its proxy. There are various methods to measure PPM, but the most accurate and straightforward method is by doing a digital measurement of the water impedance, which is the most suitable approach for automatic systems.

The total concentration of the nutrient solution elements should be between 1000 and 1500 *ppm* to facilitate the uptake by the plant roots. Specific conductivity readings of these concentrations would correspond to 1.5 and 3.5 *mS/cm*. Typical favorable values for lettuce are about 2 – 3 *mS/cm* [1].

#### Water Ph

The pH level of the soil or hydroponic solution determines on how much of the essential elements the plant will be able to absorb (Figure 2.9). The pH is a measure of the acidity or alkalinity. If the pH solution is less than seven it is acidic, seven is neutral, and greater than seven is alkaline or basic. Most plants prefer a pH between 6.0 and 7.0 for optimum nutrient uptake regardless of whether it is the soil solution or a nutrient solution. Specific crops require different optimum pH ranges. For example, lettuce likes a pH between 5.5 and 5.8, whereas tomatoes, peppers, and cucumbers prefer a pH from 6.0 to 6.4.

There are several ways of measuring pH solution. In home solutions using tapes that get colored according to ph levels are very popular due to its cheap and simple application. There are electrical sensors able to measure water pH level. However, this tend to be quite expensive with a short lifetime and requiring periodic calibrations.

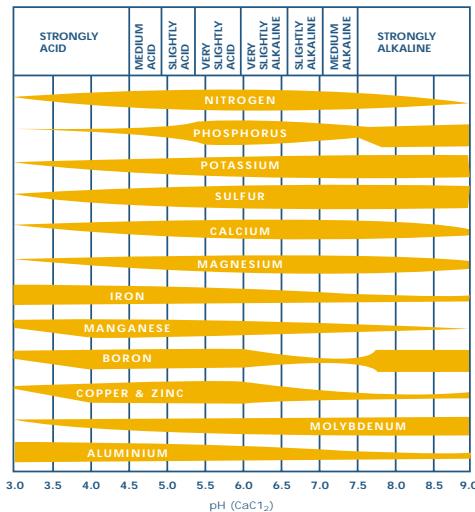


Figure 2.9: The effect of pH on the availability of plant nutrient uptake. Illustration from Roberto [12].

## Water temperature

The nutrient solution temperature should never exceed  $30^{\circ}\text{C}$ . The ideal value should be in a range from  $16^{\circ}\text{C}$  to  $24^{\circ}\text{C}$  whenever in Summer and from  $10^{\circ}\text{C}$  to  $16^{\circ}\text{C}$  in Winter. Temperatures outside this range can cause damages to the plants, decreasing its capability of absorbing nutrients.

## 2.4 Greenhouses

Crop growth is highly influenced by environment characteristics and dynamics. By growing plants in a sheltered ambient, greenhouse production offers a means for moving towards a greater degree of environmental control. The dimensions can vary from mere protection from rain, to complete environmental control. Therefore, greenhouse production exceeds all other forms of agriculture in the degree of control over plant environment. The major influential factors are temperature, light, and air composition. At the same time, it provides more protection against undesired diseases and plagues.

This section outlines and traces the development of the major ways to modify plant environment in greenhouses.

### 2.4.1 Greenhouse architectures

There are different styles of shapes to design a greenhouse, some take into account light angle, wind conditions or air temperature and air flow. A list of different shapes is presented below, see also Figure 2.10.

1. **Lean-to** is placed against an existing wall typically facing south. This type of shape is more common for hobby or small greenhouses.
2. **Traditional/Even-span/Gable** this the most common configuration and the one that closer looks like a traditional house. Its main advantages are the design simplicity and the straight lines. It also allows circulating air passively by adding an aperture at the top of the house.
3. **Sawtooth** is an interesting solution to increase passive ventilation. Thus, more suitable for hotter places.
4. **Dome** has a very stable internal environment and an excellent light distribution. It is also more robust against the wind. Unfortunately, it is expensive to build.
5. **Tunel** has a simple curved structure. However, it has poor spacing and layout considerations. It also requires active ventilation.

For its simplicity and easy to build design, the traditional greenhouse shape was adopted. The advantages of straight lines allowed an easier handling of the building materials. This shape also permits passive and active ventilation, increasing the control capabilities.

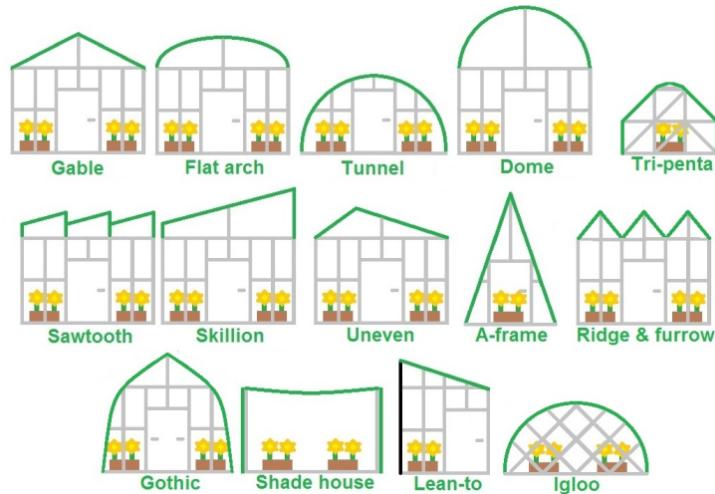


Figure 2.10: Greenhouses most common design shapes. Illustration from [globalspec.com](http://globalspec.com).

### Cover and structure materials

The major shift in the post-World War II period has been the adoption and widespread use of plastics as a covering. In the early years, plastic did not offer a qualitative improvement over glass concerning environmental control. Plastic covered houses are now in wide use throughout the world and represent the largest share of new greenhouse construction. The plastic has several advantages, it is easy to work with, to transport, and is also cheaper. Although, over time it can have some downsides that wisely have been overtaken. Single layers of plastic covering are not as effective as glass in reducing heat loss. Furthermore, single-layer coverings also collect a large amount of condensation underneath reducing light transmission and sometimes leading to disease problems. By adding a second layer of plastic helps reducing heat losses and condensation. A summary of the nowadays most commons options is below:

1. **Glass** was the only choice until the 1950s[14]. Price, dangerous installation and transportation are some of its downsides. But it still has some advantages over plastics like being best at evapotranspiration.
2. **Polyethylene (plastic)** is a very common material because it is the cheapest one. On the other hand, it is not the best insulator, it is also weak and it is not the most durable as it is very susceptible to ultraviolet light. By increasing its thickness it is possible to reduce the downside impact but the price will raise rapidly.
3. **Polycarbonate** is a relatively new solution. It is very robust and flexible at the same time, a good insulator, a good light diffuser and it also allows to add UV protection. Cost is one of its downsides when compared to plastic. Although, with a little bigger investment, it is a great solution over plastic. Very common in harsh weather conditions like strong winds and cold climates.
4. **Solexx** is an American brand of special polycarbonate that improves some of its quality, like having a special outside treatment to be an even better insulator and a better diffuser of light. Therefore it is one of the most expensive materials.



Figure 2.11: Polycarbonate sheets are a solid but flexible and easy to use material for greenhouses covers. Photo from [How to Select and Buy Polycarbonate Sheet ebay post](#).

When designing the structure, several features have to be taken into account, mainly, cost and materials ease of use. For big infrastructures, materials such as aluminum and steel are strong and solid. Aluminum is more malleable, it is lighter and does not rust unlike the steel, although the price is more expensive per kilogram.

Wood or PVC are other materials that can be used for smaller backyard house frames. The decision of choosing one of these materials to work with depends a lot on the user preference. They are both ease of use, being relatively durable and light. While treated wood can be more expensive, untreated one can be subject to rod.

For this project untreated wood was chosen as the material to work with because of its cheap price and because it is easier to find tools to work with. Regarding the cover material the polycarbonate is a great solution to go with. Once it is a good isolator it will allow to better control the inside temperature. Its robustness will allow to have structures, for example electronics, screwed directly. Its diffusing light qualities are also inviting to work with. Because this project greenhouse is of small dimensions, the cost on polycarbonate sheets will not get huge.

## 2.4.2 Greenhouse monitoring

Inside a greenhouse temperature, humidity, carbon dioxide ( $CO_2$ ), light incidence, root involving conditions and crop weight are some variables with great significance concerning plant growth.

### Temperature

Temperature exerts a significant influence on the rate of photosynthesis. Generally, the higher the temperature, assuming carbon dioxide and light are abundant, the faster photosynthesis takes place. While the greenhouse structure slightly reduces incident light, it compensates by helping retaining radiant heat.

However, there are some limits to the process as different plants have different optimum temperatures. University of Arizona researchers [2] have divided vegetables into three main optimum growth temperature categories for their work in northern Mexico each are presented in Table 2.1 .

### Heating

The main original purpose of the greenhouse is to raise, in a controlled way, the temperature of plant-growing environment. The air, soil and other objects inside the house absorb this heat and re-radiate a fraction of it.

Category	Temperature °C		Crop
	Day	Night	
Warm	27-38	24-27	cantaloupe, cucumber, squash watermelon
Moderate	24-27	16-18	eggplant, onion, pepper, tomato
Cool	21-27	10-16	beans, beets, broccoli, cabbage, carrots, cauliflower, lettuce, radish, spinach

Table 2.1: Ideal environmental temperature for different crop [2]

One solution to maintain the temperature, mainly overnight, is exactly to increase the greenhouse mass thermal capacity, also called a heat sinks strategy. Materials like water and concrete can be used for these purposes.

To go one step beyond standard thermal mass, a heat exchanger can be incorporated to circulate air or a liquid through the source of mass. For example, by burring a network of pipes in the ground, a bigger thermal mass energy can be stored outside the greenhouse.

Another solution is the use of rotting manure that will free energy in its process or by circulating steam or hot water.

Heating the house with electric power is yet another solution, although it tends to get very expensive in large implementations. In small solutions as the one implemented, the electrical solution is easy and more accurate when actuating.

## Cooling

Plants themselves have a built-in cooling system in the form of transpiration by releasing water vapor. About 50 percent of the radiant energy received by the plant is lost this way [2]. The cooling process may be increased by the use of passive or forced ventilation, by sprinkling the outside of the glass with water or using heat exchangers. Ventilation is by far the most common method, once it is cheaper and easier to control.

Both active and passive ventilation were adopted in this project implementation. The greenhouse has a controlled aperture on the roof and one air opening at the bottom with a fan attached to it. Therefore, allowing air to flow with the thermal gradient or by active ventilation the air flux can be controlled.

Greenhouse temperature control can get expensive. More recent studies aim to control the greenhouses with sustainable energies instead of fossil energy. For example, the work done by Van Ooteghem [8] extends a conventional greenhouse with an improved roof cover, ventilation with heat recovery, a heat pump, a heat exchanger and an aquifer.

## Light

Light is essential for photosynthesis. In this process, sunlight is converted into sugars to provide fuel for plant growth. Photosynthesis is more pronounced in red (600-680nm) and blue and violet (380-480nm) wavelengths of light spectrum, reflecting green light, that is why plants appear green, see Fig. 2.12. Because of this specific spectrum of energy absorption, Photosynthetically Active Radiation (PAR) refers to radiation with wavelengths between 400 and 700 nm. A PAR sensor with a flat receiver measures

Photosynthetic Photon Flux Density (PPFD). PPFD is defined as the number of photons in the 400–700 nm wavelength interval arriving per unit time on a unit area of a flat receiver, most commonly expressed in micro-mol per square meter per second ( $\mu\text{mol} \cdot \text{m}^{-2} \cdot \text{sec}^{-1}$ ) units [15]. This sensor tends to be very expensive, therefore another solution is to go with cheaper light sensors that measure a broader wavelength interval, and then do an approximation conversion. For example, lux sensors are very popular and way less costly. Lux is the SI unit of luminance measuring the intensity of light that hits or passes through a surface, as perceived by the human eye. It is possible to do a rough conversion from Lux to PPFD by knowing the light source [16] [17]. For sunlight the conversion is done as follows:

$$1 \text{ PPF}(\mu\text{mol} \cdot \text{m}^{-2} \cdot \text{sec}^{-1}) = 0.0185 \text{ Lux} \quad (2.1)$$

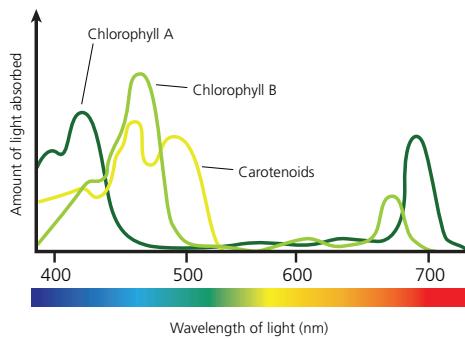


Figure 2.12: General plant spectrum absorption. Plants absorption is bigger in reds and blues. Illustration from [18].

On the one hand, the amount of natural light available is influenced by both the length of the day and the intensity of the light. Day length, in turn, is a function of season and latitude. Earth's day length as a function of latitude and of the day of the year is pictured in Figure 2.13. Portugal latitude is of almost  $40^\circ$ , therefore, it has from 9 to 15 hour of daylight over the year. On the other hand, weather conditions will obviously influence light intensity, mostly due to clouds presence.

Plants also need dark periods, but regardless of the latitude or external conditions, plants are more apt to face a problem of less light rather than too much of it. Most plants grow best when exposed to 16-18 hours of light per day at a radiation level of  $400 \mu\text{mol} \cdot \text{m}^{-2} \cdot \text{s}^{-1}$ . Generally, if the photoperiod is extended or decreased and the irradiance varied proportionately so that the  $\text{irradiance} \times \text{time}$  product is a constant value, growth rates will be similar [12, 19]. This product is the amount of light available per square foot per day in the PAR spectrum defined as the Daily light integral (DLI) expressed in  $\text{mol} \cdot \text{m}^{-2} \cdot \text{day}^{-1}$  [20]. Lower-light crops like lettuce require a DLI of  $12\text{-}14 \text{ mol} \cdot \text{m}^{-2} \cdot \text{day}^{-1}$  for maximum growth rates, and higher-light crops such as tomatoes require at least an average DLI of  $22\text{-}30 \text{ mol} \cdot \text{m}^{-2} \cdot \text{day}^{-1}$  to reach light saturation at maturity.

In order to overcome the lack of light, artificial lighting can be used. The use of artificial light to influence plant growth has been studied in England and France since late 1870 [2]. Although results showed an improvement on crop growth, it was not profitable for food growers. Nowadays, LED light usage reduces lighting costs. Also with this type of luminaries it is possible to light in specific wavelengths

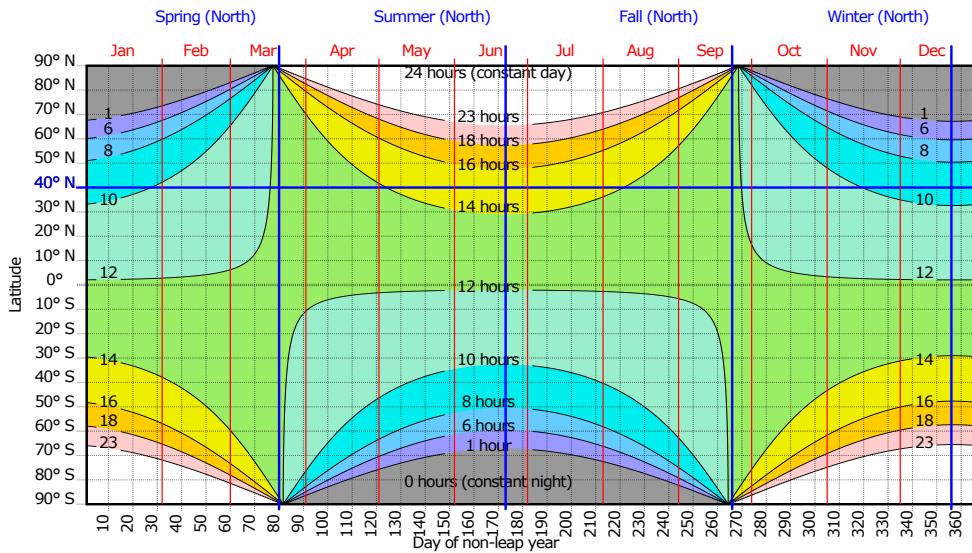


Figure 2.13: Number of daily light hours according to day of the year and latitude in the globe. Illustration from [Wikipedia Daytime](#)

reducing, even more, lighting costs [21]. Lamps intensity is commonly measured in power (watts) per square meter. For a fully artificial light dependent system a general rule of thumb is  $200\text{-}500 \text{ w/m}^2$  [12].

In a greenhouse, the rooftop angle will affect the amount of light that goes through it and the one reflected. The rooftop angle should be perpendicular to sunlight to maximize light absorption. In Portugal, in Winter the lower angle of the sun is about  $28^\circ$  and the maximum in Summer is of  $74^\circ$ . Therefore, this project greenhouse should have an average optimum angle around 45 degrees.

## Humidity

The relative humidity in a greenhouse is usually higher than in open air. While plants produce water vapor in the course of transpiration, relative humidity in greenhouses builds up to higher levels than in the field because of the heavier plant populations and reduced air movement. Humidity is generally controlled by ventilation, but it can be increased by misting devices and reduced by use of heating. An Ohio study has suggested that the optimum humidity level for tomatoes is about 70 percent [2]. However, even higher values presents no problem in the presence of ample light. The effect of high humidity on the spread of disease is also an important question.

## Carbon Dioxide

The gaseous composition of the atmosphere has a significant influence on plant growth. It has long been recognized that greenhouses need to be ventilated to provide adequate carbon dioxide.

Carbon dioxide ( $\text{CO}_2$ ) along with water, is one of the two major ingredients in the process of photosynthesis. Below-normal levels of  $\text{CO}_2$ , often found in unventilated greenhouses, can reduce the rate of photosynthesis while above-normal levels can increase photosynthetic activity.

During the 1960's, there was an upsurge of interest in using  $\text{CO}_2$  in greenhouse operations. The expansion first took place in The Netherlands when a grower increased the rate of  $\text{CO}_2$  during daytime

and obtained exceptional quality and weight. Some of these experiments counted on an increase growth acceleration by 20 to 30 percent for lettuce [2].

Typical good values for lettuce are around 750ppm, close to twice the normal air concentration.

## Root zone

Plants absorb through their roots all of the water, most of the nutrients, and some of the oxygen they use. Hence, plants require root mediums with adequate moisture, fertility, and aeration.

Using soil, this is done by watering, fertilizing and trusting the soil ability of air retention. In some facilities, the fertilization is done via the watering system. On the other hand, a hydroponic system is a soil-free crop grow method where plants grow in an inert substrate. This solution allows the enhancement of crop growth and level of control, and this is the main reason why it was adopted.

## Crop growth

For plant growth evaluation, plant weight is an important and commonly used indicator. Traditional plant weight measurements are destructive and laborious. In order to measure and record the plant weight during crop production, some non invasive methods have been developed. Van Henten and Bontsema in [22] demonstrated that crop volume could be related with its weight by means of image processing.

Motivated by this possibility, Jung et al. 2015 predicted fresh weights of hydroponic lettuce using different image processing methods. The simplest method of relating the number of pixels in images of leaf areas to actual fresh weights of lettuce showed to have errors smaller than 5g.

Moreover, Lin et al. in Lin et al. 2012 developed a stereo vision system image processing algorithm to monitor plant growth of Boston lettuce on an hydroponic system. With the third dimension information this software would be able to calculate other, and more precisely, geometric features such as projected leaf area, plant height and plant volume.

The same authors Lin et al. 2016 continued this work combining it with a different technology composed on an electrical weight measurement device based on load cell. They used this sensor to automatically calibrate the image processing algorithm. In this paper it is stated that the load cell revealed a result with an average of 6% measurement error. On the other hand, the estimation of plant weight using the image information revealed to be even better with a error below 1% across the whole growth period.

By all means, the stereo image processing vision solution is more appealing for following plant growth. But for the sake of simplicity, in this work the simpler procedure of counting projected leaf area image pixels, relating that with the plant weight, was opted.

[Plant Image Analysis](#) web site [26] is an online database for plant image analysis software tools that resumes the big majority of software available for this purpose. From this web page the [Image Harvest](#) and [PlantCV](#)[27] softwares were the ones that best suited this project needs. Both based on OpenCv computer vision open source library, PlantCV showed to have better documentation and a stronger support community.

## 2.5 Plant growth models

To control plant growth is fundamental to have a model that relates the size or the mass of the plants to the conditions it is subject to.

Within the majority of crops life cycle, three growth phases can be distinguished: an early, exponential phase, a linear growth phase, and a phase of stabilization, see Fig. 2.14.

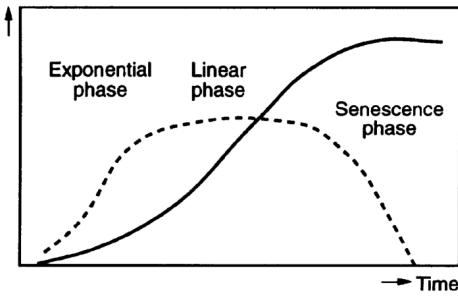


Figure 2.14: Generalized patterns of the growth rate (broken line) and generated biomass (solid line) in the three growth phases as a function of time. Graphic from [28].

During the first, exponential, growth phase, most of the space around the plants has not yet been occupied. Each new leaf that is formed contributes to more light being intercepted so that growth increases even more. There is no mutual shading yet and the contribution of the new leaf is identical to that of the existing ones.

Later on, the leaves will gradually start to overshadow each other, and above a LAI (Leaf Area Index) of  $3m^2(\text{leaf})m^{-2}(\text{ground})$ , new leaf area hardly results in any increase in light being intercepted. This does not mean that the LAI stops increasing. When water and nutrients are optimally supplied, LAI can easily exceed a value of  $6m^2.m^{-2}$ , but such a high value does not contribute further to the production of biomass. Instead, the growth remains linear.

The final growth phase that of the senescence starts when light interception decreases again, and the LAI has decreased to below  $3m^2.m^{-2}$ . This usually means that the process of translocation of the nutrients from the leaf to the storage organs (grains, tubers, etc.) has already started [28].

Some models from background research will now be presented.

### A simple approach

The paper Goudiaan and Laar [28] suggests a quite simple plant growth model by splitting its three characteristic phases in different equations, an initial exponential, followed by a linear and then a constant function. Then they suggest a combination of the exponential and linear phases characterized by an expolinear growth equation as:

$$w = \frac{c_m}{r_m} \ln(1 + e^{r_m(t-t_b)}) \quad (2.2)$$

where  $w$  is the biomass ( $g.m^{-2}$ ),  $c_m$  is the maximum growth rate ( $g.m^{-2}.d^{-1}$ ),  $r_m$  is the relative growth rate of the biomass ( $d^{-1}$ ),  $t$  the time in days, and  $t_b$  the start moment of the linear phase.

They also suggest that the maximum growth rate  $c_m$  could be related linearly with the incident radiation flux of photosynthetically active radiation (PAR). On the other hand, the constant  $r_m$  depends on the type of crop also being proportional to  $c_m$ .

### Logistic Function

This is a first order differential equation that is often used to model the growth of a species, namely, the number of elements of a given population. Therefore also known as the logistic law of population growth [29].

Let  $p(t)$  denote the population of a given species at time  $t$ , the logistic function is

$$\frac{dp(t)}{dt} = ap(t) - bp^2(t) \quad (2.3)$$

where  $a$  and  $b$  are constants. It is possible to obtain an analytical expression for  $p(t)$ , as

$$p(t) = \frac{ap_0}{bp_0 + (a - bp_0)e^{-a(t-t_0)}}. \quad (2.4)$$

The deduction can be seen in reference [29].

Note that, constant  $b$ , in general, will be very small compared to  $a$ , so that if  $p$  is not too large, the term  $-bp^2$  will be negligible compared to  $ap$  and the population would grow exponentially. However, when  $p$  is very large, the term  $-bp^2$  is no longer negligible, and thus serves to slow down the rapid rate of increase of the population. Moreover, looking at equation (2.4) for  $t \rightarrow \infty$ ,  $p(t) \rightarrow \frac{a}{b}$ .

Thus, this function has the shape presented in Figure 2.14 , in other words similar to plant growth. Indeed, this function extends the simple approach of the previous section by allowing to have a single expression for the entire plant lifetime.

Supported by the paper [30] it is possible to successfully model some plant species with this function. As it will be seen in the experimental section, the lettuce growth is suitable to be modeled with the logistic function. Therefore, due to its simplicity but at the same time allowing to model the growth during the entire plant lifetime, this was the chosen model to work with.

Therefore, equation (2.3) will be rewritten for plant growth as

$$\frac{dw(t)}{dt} = K(\mathcal{M} - w(t))w(t) \quad (2.5)$$

where  $w$  denotes the plant biomass,  $K$  and  $\mathcal{M}$  are constants defining the plant growth rate and its maximum mass achieved, respectively.

### Van Henten Model

One of the most important works for optimizing the dry matter production in greenhouses is the one done by van Henten and van Straten [5, 7].

In this work, lettuce crop production process is described by a four state variable dynamic model.

The model describes the evolution in time of the dry matter content of the lettuce crop,  $W$  [ $kg.m^{-2}$ ], the carbon dioxide concentration in the greenhouse,  $C$  [ $kg.m^{-3}$ ], the air temperature in the greenhouse,  $T$  [ $^{\circ}C$ ] and the humidity content of the greenhouse air,  $C_{H_2O}$  [ $kg.m^{-3}$ ]. Although this model predicts very well the crop growth the equations that define this model are nonlinear and quite complex, taking into account a lot of variables. For sake of simplicity they will not be presented or listed.

### **Van Henten simplified**

In the paper E.Leal-Enríquez and Bonilla-Estrada [31] it is proposed a simplification of the van Henten's model. This simplification is done in two steps: firstly they separate the van Henten's model into two independent models: one for the day period and the other one for the night period. Then, they express explicitly the relative growth rate in terms of the environmental variables. This approximation allows to avoid the high non-linearities present in the van Henten's model. Along with Henten's model this paper focus on lettuce crop.

This reduced model is based on photosynthetic and respiration process, it integrates the lettuce dry matter, [ $kg.m^{-2}$ ], with the greenhouse climate behavior: the  $CO_2$  concentration, [ $kg.m^{-3}$ ], and the air temperature, [ $^{\circ}C$ ], inside the greenhouse. This state space variables depend on the ventilation flux throughout the windows, the energy supply by heating the system, the  $CO_2$  supply rate, the outdoor  $CO_2$  concentration, the outdoor air temperature and the Photosynthetically Active Radiation (PAR).

### **NICOLET Model**

The Nitrate Control in Lettuce (NiCoLet) model accurately simulates lettuce crop growth and nitrate uptake on a long-term basis. The NICOLET model possesses two state variables: vacuole carbon content and structure carbon content. Crop growth can be estimated because it is directly related to carbon content.

Thus, measuring carbon dioxide, temperature and radiation taken inside a greenhouse used as inputs to the model, it gives out fresh and dry matter of lettuce aerial part.

Works done in Mexico Juárez-Maldonado et al. [32] [33] tested this model in soilless system. Results have shown that data obtained from experiences were accurately predicted with this model. Moreover, the paper Mathieu et al. [34] analysis hypothesis of predicting crop growth in short-therm ( hours) for fault detection in hydroponic systems.

The appealing part of this model to this project, is that it was already tested in hydroponic systems showing a great degree of accuracy in the results. Nevertheless, its complexity is still considerable.

## **2.6 Crop optimal control**

Greenhouse climate management can be significantly and efficiently improved when implementing advanced controllers using optimal control theory [22]. In The Netherlands, many studies have been

developed on this subject. Authors as Van Straten, Van Henten [5, 7, 35], also Van Ooteghem [36, 8, 6], and even Bontsema [37] have written books, thesis and scientific papers about it.

A goal is formulated in the form of a cost function describing the energy cost and the crop yield.

Having a model for the crop growth written in state space form

$$\dot{w} = f(t, w, u, v) \quad (2.6)$$

with initial condition

$$w(t_0) = w_0. \quad (2.7)$$

The goal will be to maximize crop weight, while minimizing the associated cost. This can be written mathematically as maximizing the cost function

$$J(u) = \Psi[w(t_f)] + \int_{t_0}^{t_f} \mathcal{L}(w, u) dt. \quad (2.8)$$

Where  $t$  is time,  $w$  is the plant weight,  $u$  are control inputs, that takes values in the space of admissible controls  $\mathcal{U}$ ,  $t_f$  the terminal time, and  $f$  is a non-linear function. Moreover,  $\Psi[w(t_f)]$  represents what is pretended to be maximized: the final crop weight, while the function  $\mathcal{L}$  is given by the sum of the penalties of actuation. Note that maximizing  $J$  is the same as minimizing its symmetric. In sum one wants to solve the problem

$$\begin{aligned} & \underset{w}{\text{maximize}} \quad J(u) \\ & \text{s.t.} \quad \dot{w} = f(w, u), \\ & \quad w(t_0) = w_0, \\ & \quad u(t) \in \mathcal{U}, \\ & \quad t \in [t_0, t_f]. \end{aligned} \quad (2.9)$$

Pontryagin's principle gives a necessary condition for the solution of (2.9). Let  $\mathcal{H}$  be the Hamiltonian, defined as

$$\mathcal{H}(\lambda, w, u) = \lambda^T f(w, u) + \mathcal{L}(w, u), \quad (2.10)$$

additionally, the derivative of the co-state equation defined as

$$-\dot{\lambda}^T(t) = \lambda^T(t) f_w(w(t), u(t)) + L_w(w(t), u(t)), \quad (2.11)$$

with the boundary condition

$$-\lambda^T(t_f) = \Psi_w[w(t_f)], \quad (2.12)$$

where the subscript applied to a function represents the partial derivative with respect to the subscript.

Pontryagin's principle states that, for every  $t$ , the Hamiltonian is maximum for the optimal control input  $u^*(t)$ .

In the case where the maximum of the Hamiltonian is achieved in the interior of  $\mathcal{U}$ , the maximum is

satisfied in one of the solutions of the equation

$$\frac{\partial \mathcal{H}}{\partial u} = 0. \quad (2.13)$$

If the optimal is in the frontier of  $\mathcal{U}$  than the previous equation cannot be used to find it. However, in this case an analytical approach is often possible.

Pontryagin's principle allows thus to translate a maximization problem in respect to a function in a problem of maximizing in respect to a variable  $u(t)$ , for every  $t$ .

The success of using optimal control seems to be very sensible to the accuracy of the models used as well as the tuning of the cost function. Expensive computation costs also seems a problem that can be solved with numeric simplifications.

### 2.6.1 Solve optimal problem numerically

In the big majority of problems addressed using Pontryagin's principle, solving it analytically is not easy. Moreover, sometimes the state and co-state functions are coupled.

However, there are several numeric approaches to solve it. Below it is presented an algorithm that addresses the problem using the gradient method.

Initially, for an arbitrary values of inputs  $u$ , the state equation is integrated to find out  $x$  for all time  $t$ . Afterwards,  $\lambda$  is also computed integrating the co-state equation backwards using the final co-estate conditions. Finally, the maximum of the Hamiltonian must be found in respect to  $u$  for all time  $t$ . This process is repeated using the obtained  $u$  until the optimal solution converges. The pseudo code algorithm is presented in algorithm 1:

---

**Algorithm 1** Numerically solve optimal control problem using gradient method

---

- 1:  $N \leftarrow$  number of time samples to consider
  - 2:  $u \leftarrow$  arbitrary vector of size  $N$
  - 3: **for** certain number of iterations **do**
  - 4:   compute  $x(t) \in \mathbb{R}^N$  integrating equation (2.6) using initial condition (2.7)
  - 5:   compute  $\lambda(t) \in \mathbb{R}^N$  integrating backwards equation (2.11) using final condition (2.12)
  - 6:   **for**  $i = 0 : N$  **do**
  - 7:     find optimal  $u^*(i)$  that maximizes the Hamiltonian  $\mathcal{H}$  defined in equation (2.10)
- 

A different method is the *shooting method* where the initial condition of the co-estate is adjusted so that the final condition is met.

## 2.7 Modeling

Models are a key component in control design. They allow to simulate and test control approaches without using the real plant system. In addition, the plant model can also be used in real time to feedback control the real plant system. These models can be obtained by applying physical and geometric laws. Sometimes this identification leads to non-linear models. Fortunately this can be summed to linearizations around an equilibrium point, for example using Taylor series. Another solution is to come up with

a model based on experimental data, where one starts with simple first order system and increases the order as needed.

### Linear model identification

Many dynamical systems can be modeled accurately by linear differential equations. The general form of a linear state space system is

$$\dot{x} = Ax + Bu, \quad y = Cx + Du, \quad (2.14)$$

where  $A \in \mathbb{R}^{n \times n}$ ,  $B \in \mathbb{R}^{n \times m}$ ,  $C \in \mathbb{R}^{p \times n}$ ,  $D \in \mathbb{R}^{p \times m}$ . Equation (2.14) is a system of linear, first order, differential equations with input  $u \in \mathbb{R}^m$ , state  $x \in \mathbb{R}^n$  and output  $y \in \mathbb{R}^p$ . The state space representation is often used in modern control theory as it offers more versatility [38].

Linear models parameters, that are valid around an operating region, can be estimated using identification techniques based on plant data. The greenhouse, as an open-loop stable plant system, will allow to identify the temperature and humidity models parameters using this technique.

### Least-Squares Method

The Least-Square is a statistical method of fitting data to a an unknown linear model by observing its input-output data pairs. This method allows to identify, for example, the parameters  $a_i$  and  $b_j$  of a linear differential equation as:

$$y(t) + a_1y(t-1) + \dots + a_ny(t-n) = b_1u(t-1) + \dots + b_mu(t-m-1) \quad (2.15)$$

Based on a set of input  $u(t)$  and output  $y(t)$  experimental samples, and defining the regressor,  $\varphi$ , as

$$\varphi(t-1) = [-y(t-1) \dots -y(t-n) \ u(t-1) \dots u(t-m-1)]^T \quad (2.16)$$

and the vector of parameters to estimate,  $\theta$ , as

$$\theta = [a_1 \dots a_n \ b_1 \dots b_m]^T \quad (2.17)$$

The model (2.15) can be written in matrix form as

$$Y = \Phi\theta + \varepsilon \quad (2.18)$$

considering a white noise occurrence in each measurement of  $\varepsilon(n)$ . The estimated parameters  $\hat{\theta}$  can be computed through the relation

$$\hat{\theta} = (\Phi^T\Phi)^{-1}\Phi^T Y \quad (2.19)$$

existing the inverse of  $\Phi^T\Phi$ . Moreover  $\Phi^T\Phi$  has to be positive-definite matrix. This happens if the samples are rich enough, which in turn depend on the system input [39].

## Non linear model identification

For some systems nonlinearities cannot be ignored, especially if one cares about the global behavior of the system (and not just around a fixed point).

In these cases a possible approach is to select a known curve that resembles the experimental data and try to fit the modeled data to experimental one. One approach is to use a solver algorithm that minimizes a cost function for various combinations of the model's parameters to estimate. This cost function can be defined as

$$J = \sum_{t=1}^N [y(t) - f(t, \hat{\theta})]^2 \quad (2.20)$$

where  $y(t) \in R^n$  is the observed output at time  $t$  that goes from zero to  $N$  number of observations, and  $f(t, \hat{\theta})$  the function to approximate evaluated at the same time sample and at a certain combination of estimators  $\hat{\theta}$ . With this minimization technique, if a proper cost function is defined, it is thus possible to fit non linear models to experimental data.

This technique will be used to discover the plant grow non-linear model in section (5.1)

## Continuous and Discrete Time

A model can be classified accordingly to the time domain in which they operate as continuous or discrete. As discrete, they can also be classified as having a constant or varying sampling time. Nowadays, the first is the most common type of discrete control.

Continuous time models are directly obtained from the application of basic physical principles, and amount to a set of ordinary differential equations. These can be converted to first order with the structure of the state model

$$\dot{x} = f(x, u) \quad (2.21)$$

where  $x \in R^N$  is the state vector,  $u \in R$  is the manipulated variable.

Although it is possible to do continuous control with analog electronics, almost all control systems today are implemented in digital computers. Unlike analog electronics, digital computers cannot integrate. Therefore, in order to solve a differential equation in a computer, it must be approximated being reduced to an algebraic equation.

One particularly simple way to make a digital computer approximate the real time solution of differential equation is to use **Euler's method**, also know as **forward difference** :

$$\frac{dx(t)}{dt} = \frac{x(t + T_s) - x(t)}{T_s} \quad (2.22)$$

where  $T_s$  denotes the sampling period. [40]

This tool allows not only to design continuous controllers and later convert them to discrete, but also to translate continuous models to discrete and vice-versa.

There are other approximations like the **forward difference** or the **Tustin method** that have a different geometric approach of the integral, see Fig. 2.15.

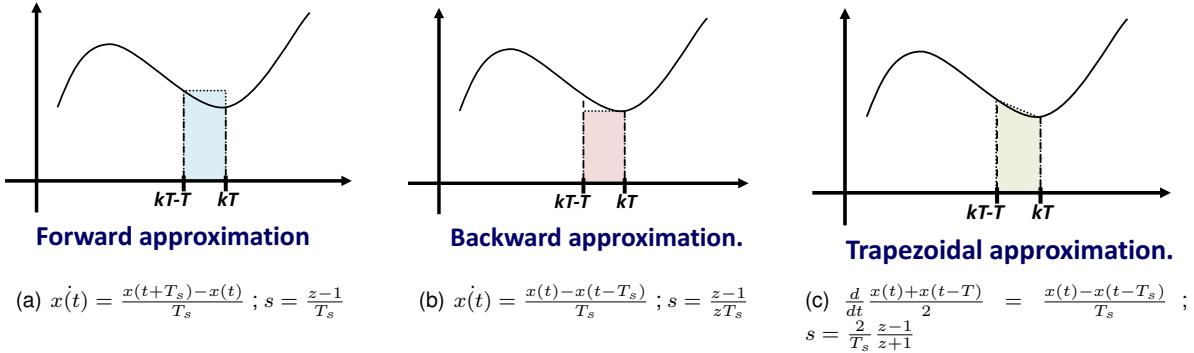


Figure 2.15: Continuous time approximations. From [41]

## 2.8 Linear Control

Linear control theory applies to systems made of devices which obey the superposition principle. Roughly this means that the output is proportional to the input. Modeling the system by a set of differential equations and doing linear control over it, means that these systems are amenable to powerful frequency domain mathematical techniques of great generality, providing design techniques for most systems of interest.

Linear control systems often use linear negative feedback to produce a control signal mathematically based on other variables, in order to maintain the controlled process within an acceptable operating range.

### 2.8.1 Proportional Integral Derivative

Proportional Integral Derivative (PID) control is by far the most common way of implementing feedback in engineering systems. It appears in simple devices and in large factories with thousands of controllers. PID's popularity is due to its easy understanding, relative easy tuning and simple implementation. Nevertheless, PID controllers do not rely on a model of the system to be controlled, but only on the measured process variable. On the other hand, sometimes it can lead to non optimum solutions, or even not stabilizing the system [42, 43].

The PID controller consists of three terms, that take the error  $e(t) = r(t) - y(t)$  as an input, and produce the control input to the plant

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}, \quad (2.23)$$

where  $r(t)$  is the desired setpoint and  $y(t)$  refers to the measured system output.

The control action is thus the sum of three terms: proportional feedback, the integral term, and derivative action. The proportional term (P) compensates for the current error, and outputs a response proportional to it. The integral term (I) takes into account the past errors, and is advantageous to compensate for steady state errors. Finally, the differential (D) term produces an anticipatory response, based on the current rate of change of the output.

Not all the three terms are required to perform control, and often a P, PI or PD is enough to honor the performance requirements of the controller. The design of the weights can be made manually or with more systematic techniques, such as Ziegler–Nichols's method [44].

Equation (2.23) control law represents an idealized controller. It is a useful abstraction for understanding the PID controller, but several modifications must be made in order to obtain a controller that is practically useful. It is the case of avoiding integral windup effect when the actuators saturate, and the need of low-pass filtering of the derivative effect. Moreover, using feed forward terms can improve system control [41].

## 2.8.2 Anti-Windup

**Integral windup**, also known as **reset windup**, refers to the situation, in controllers with integral action, where a large change on the setpoint occurs and the integral terms accumulates a significant error during the convergence, thus overshooting and continuing to increase as this accumulated error is unbound. This issue often arises whenever the controller leads to actuators' saturation. In other words, the controller tries to lead the system to the desired setpoint but it is limited to the actuators physical limits. Undesirably, the integral term continues to integrate the error.

This is one of the most important modifications to the PID controller, in case of integral term implementation. There are some solutions to work this around. One is to bound the integral term, where the right limits must be computed. Another one is to reset the integral term, although this can generate discontinuities in the command. Finally, when the controller structure allows the separation of the integral action, a simple negative feedback allows to discharge the integral during saturation. This last method is known as **anti-windup**. There is the need to tune a gain that multiplies the difference from the input signal before and after saturation. This should reduce the integral fast enough, typically set from 10 to 100 times the integral gain [41].

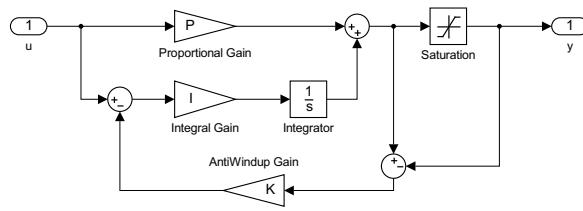


Figure 2.16: Example of PI implementation with anti-windup enhancement.

## 2.8.3 Feedforward

In feedback control the output of the system is measured and the controller will react accordingly to it. In opposition, with feedforward control, the system responds in a pre-defined way without responding to how the output reacts. This mechanism is often associated to open-loop control. Although it can provide a fast and roughly correct control, it does not respond to external disturbances, neither rejects steady state error.

Knowing a priori how the system responds, it is possible to predict the signal to feed the plant input to achieve a desired response. In simple linear controllers the feedforward term is a signal composed by a gain multiplied to the setpoint. This signal is then fed to the plant input. In simple approaches the feedforward term is commonly the inverse of the plant static gain.

As this technique can be complemented with feedback control, it is another modification that can enhance PID controllers.

## 2.8.4 LQR

An alternative to select the closed loop eigenvalue locations, or using the Ziegler-Nichols rule to choose the closed loop feedback gains, is to attempt to optimize a cost function. This can be particularly useful in helping balance the performance of the system with the magnitude of the inputs required to achieve a certain level of performance. The infinite horizon, linear quadratic regulator (LQR) problem is one of the most common optimal control problems. The goal is to find the gain vector  $K$  of the state space feedback law

$$u = -Kx \quad (2.24)$$

that applied to the multi-input linear system

$$\dot{x} = Ax + Bu \quad (2.25)$$

minimizes the quadratic cost function

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (2.26)$$

where  $Q \succeq 0$  and  $R \succ 0$  are symmetric, positive (semi-) definite matrices of the appropriate dimension, which allow to balance the rate of convergence of the solutions with the cost of the control.

The solution to the LQR problem is given by

$$K = R^{-1} B^T P \quad (2.27)$$

where  $P \in \mathbb{R}^n$  is a positive definite, symmetric matrix that satisfies the Riccati equation,

$$PA + A^T P - PBR^{-1}B^T P + Q = 0 \quad (2.28)$$

from which  $P$  can be calculated. One of the advantages of the LQR feedback controller is to be able to handle (MIMO) systems.

If the state  $x$  is not measurable physically, or simply is not available, it is possible to predict it using a Linear Quadratic Estimator (LQE), also known as the Kalman Filter. The resulting signal is called the observer. The combination of these techniques, the LQR and LQE, supported by the superposition theorem, is called the Linear Quadratic Gaussian (LQG).

# Chapter 3

## Experimental system

To actually test and evaluate the performance of the optimal crop growth control algorithm a real hydroponic greenhouse was developed. Moreover, environmental and plant models were formulated based on real greenhouse data. The goal was to build a small system that is practical and cheap, able to hold multiple plants so that there is some experimental redundancy.

Due to the fact that there was no greenhouse off the shelf that answered all the desired needs, an adequate one was developed from scratch. The design is based on the traditional greenhouse shape, prioritizing simplicity, and low investment.

Because the greenhouse stayed inside a closed room with light access, it does not have to be the most robust ever concerning weather conditions. Therefore untreated wood was the chosen structure material, due to its low cost and being easy to work with.

Polycarbonate was used as covering, due to its good insulating properties, good diffusing light, being easy to work with and allowing to hold electronics or other objects directly. This is not the cheapest cover material, although, because of the system reduced size, it was not very costly.

Part of the roof was extended for protection as the other became a passive aperture controlled by a servo motor. The passive aperture allows the system to have a smooth air flow. Close to the bottom of the roof, a fan enables to force air flow, actively, when needed. The ceiling angle was chosen to  $45^\circ$  to maximize light exposure. See Figure 3.1a.

Lettuce was the selected crop since this type of crop has a fast development and easy greenhouse management, also because it is a very well documented hydroponic crop.

Inside the greenhouse, a small hydroponic system was installed.

Initially the NFT hydroponic system was chosen. It seemed a simple solution where the nutrient solution reservoir would not have to be inside the greenhouse, saving space for plant growth. Also, from research, this system seemed to suit small solutions like this.

The NFT hydroponic system developed consists of three NFT tubes, a central smaller one for when the crop is still small and two other ones for when the crop is already big enough to go to standard size tubes. The distance from the holes in the tubes is 25 cm according to lettuce growing suggested space. The NFT tubes are also supported with a slightly adjustable angle for water flow ( $2 - 4^\circ$ ). The nutrient

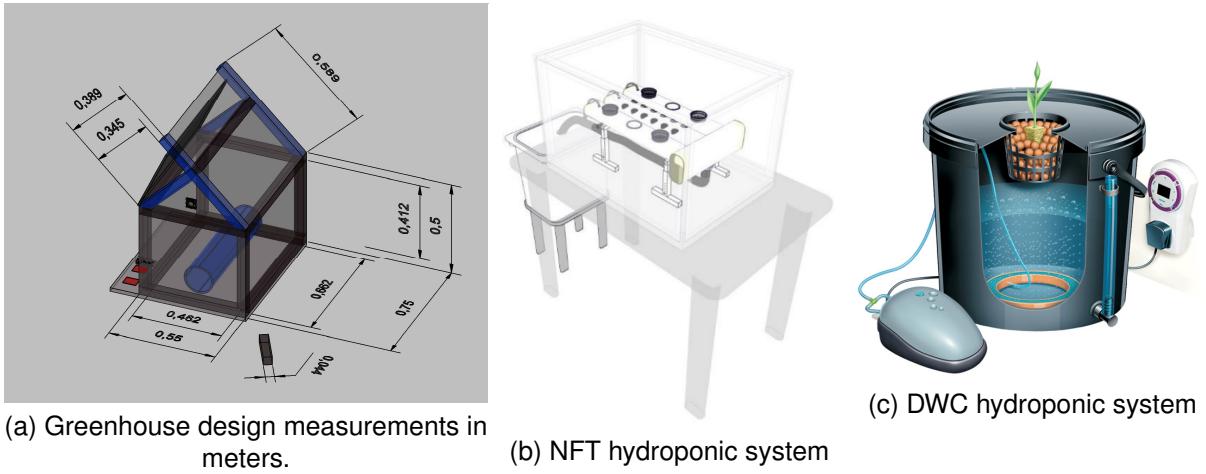


Figure 3.1: Designed greenhouse and the hydroponic system to fit inside it

solution is saved in a reservoir at a lower height than the greenhouse. Using pipes and a water pump the solution circulates. See Figure 3.1b.

This approach was used for the first experiment. Initially it worked well, but soon some undesired problems emerged. There were some leaks on the tubes and it was not possible to move away the plants when they were overlapping in the camera pictures, making image process impossible. Also it did not allow to have plants growing simultaneously with different nutritive concentration. Therefore another approach was developed.

In Deep Water Culture (DWC) hydroponic method, if separate containers are used, it is possible to simultaneously grow plants with different nutritive solutions. This would allow to collect more data in the same period of time, allowing to have a bigger dataset of experiments and, therefore, achieving a better model for plant growth.

This second system, described in section 2.3.1, was pretty simple to build. Composed by four recycled yogurt plastic cases of one litter, it was only necessary to drill a hole on top for the plant support and another to pass a tube that oxygenates the water. Therefore, it also needs an air-pump that will distribute air for all the plastic cases that contain the nutritive solution. Changing the hydroponic approach also allowed to overcome the leaking problems. Figure 3.1c illustrates this system profile.

### 3.1 Controller framework

The controller is composed by several hardware devices connected by a physical layer. It consists on a set of sensors and actuators connected directly, or through a relay, to an Arduino microcontroller, which in turn is connected to a Raspberry Pi mini computer. The microcontroller is only responsible for interacting with the sensors and actuators, where no control is done. On the other hand, the computer does all the computation, including image processing from a camera directly connected to it. Moreover, the small computer allows remote access throughout a Graphic User Interface (GUI). This two hardwares are connected via serial communication between them, whereas the computer is also connected to the Internet.

### 3.1.1 Controller overview

As seen in the previous subsections 2.3 and 2.4 there are many variables influencing crop growth. The main ones that should be taken into account are (i) light, (ii) greenhouse temperature, (iii) carbon dioxide concentration, (iv) relative humidity, (v) nutrient solution concentration (EC), and finally (vi) Ph.

Figure 3.2 diagram summarizes the sensing and actuation in the greenhouse that were taken into account allowing crop growth control. To track some of the variables listed above light, temperature and humidity sensors were used. To actually measure the growth of plants a top-view camera was installed. On the other hand, to influence the inside greenhouse conditions, a fan and a servo to control the window aperture were installed. An artificial light allowed to supplement available sun light, whereas a mist maker actuator allowed to increase the greenhouse inside relative humidity.

Nevertheless, some of the initially planned configuration was subject to change. Heating was not necessary because the experiments took place during Summer time. Moreover, because the nutritive solution sensors revealed to be quite expensive, neither the pH nor EC levels were measured. Finally, during the project period there was not enough time to further develop the control of  $CO_2$ , therefore neither the sensor was used nor the injector built.

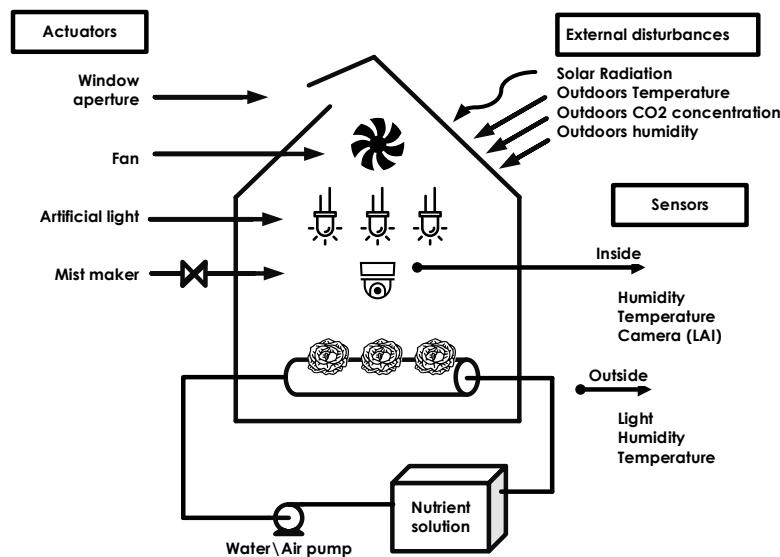


Figure 3.2: Greenhouse plant system depicting the sensing and acting devices

### 3.1.2 Hardware implementation

The sensors and actuators were connected to the Arduino except for the camera that is attached to the Raspberry Pi. The artificial light, the air pump and the mist maker actuator had to be attached to an Arduino shield with relays for their higher power supply voltage. Sensors and actuators can be connected to Arduino's digital ports, analog (which have an ADC or a DAC), or PWM (~). The Pi and the Arduino are connected with an USB cable via Serial communication. The Pi card memory saves experimental data. External users and the programmer are able to connect to the Pi through the World Wide Web.

Wide Web. Figure 3.3 summarizes this configuration. A more in depth hardware list followed with some explanations are presented in appendix C.

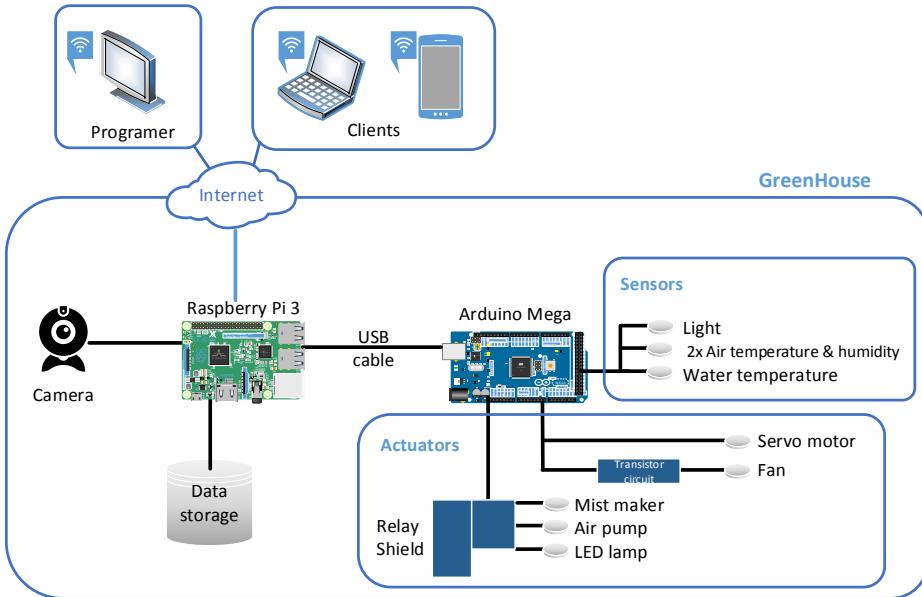


Figure 3.3: System architecture

### 3.1.3 Software architecture

The Arduino Mega is only responsible to interact with the sensors and actuators doing neither computation nor control. Its implementation runs a simple C code including several libraries that allow specific sensor and actuators communication. It also allows the communication with the Raspberry Pi throughout a very simple set\get protocol.

On the other hand, the Pi does all the computation. It uses Experimental Physics and Industrial Control System (EPICS) [45] software to monitor and control the crop and environmental variables. The Pi also serves as a database server storing the variables log. Moreover, it runs the image processing software. Finally, the Pi permits users to connect to a Graphic User Interface (GUI) developed with Control System Studio (CSS)[46] software that works well with EPICS.

#### Experimental Physics and Industrial Control System - EPICS

EPICS is an open source free software for distributed control systems. It is a collection of client/server software to build distributed real-time control systems. Acting as a middleware, it highly reduces the burden of coding. Instead of programming, the user just has to select the plugins he needs to build its control system and configure it. For example, for this project the Stream Device plugin was installed to allow the communication with Arduino. The Archiver was another used plugin. There are also some available libraries or secondary software able to interact with EPICS. For example, the Matlab library was tested, allowing to remotely get and put values from EPICS's database. Similarly, the python library

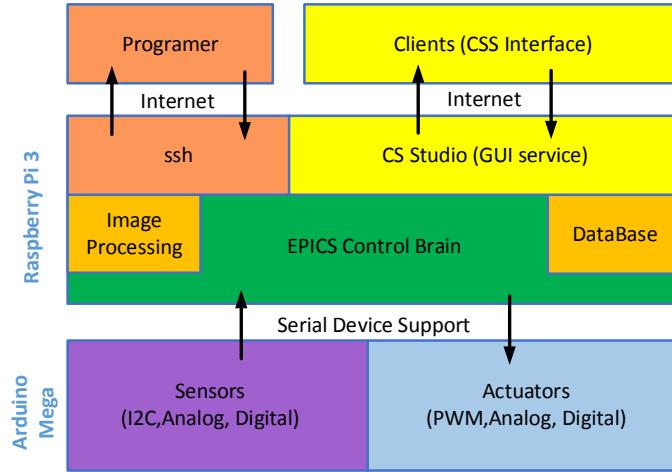


Figure 3.4: Software framework architecture describing the connectivity between different layers.

was also useful, allowing the image processing software to interact with EPICS. The great advantage of EPICS is the simplicity to extend the control system. In this project, for example, the possibility to control several greenhouses in parallel could be easily extended using this software. These characteristics justify its common use on big plant systems, like ITER tokamak [47].

Some pre-built functionalities come with EPICS by default, for example data processing, variable alarms and automatic readings smoothing. Finally, the management over the network is also configured by default with EPICS.

This software is transversal to different operating systems. However, it was originally designed for linux, and because Raspberry Pi is designed for linux as well, the installation process [48] took place in this platform.

Besides addressing well this engineering problem, EPICS also has some disadvantages. It is a considerable old software, and although there is a very active community, only minor updates take place. This way, the installation process of the main software and its plugins is quite complex, where the configuration is totally based on command line style.

Conversely, as a very well tested, bug free, reliable and open source software it answered well the needs of this project.

### Control System Studio

Control System Studio (CSS) [46] is an Eclipse-based collection of tools to monitor and operate large scale control systems. So it can run in any platform and it is also possible to extend its capacities.

Along with BOY (Best OPI, Yet ) extension, CSS allows to build an interacting dashboard without the need to code. With the philosophy "what you see is what you get", the design is done in a friendly graphical editor with a drag and drop approach. Simultaneously, it allows to run predefined scripts for a more friendly interaction, for example, if a certain variable exceeds its upper/down limit, a pop alarm can be defined. External scripts developed in python or Javascript can also be integrated.

CSS also enables to access EPICS database allowing to show saved records.

Once again, although this software reduces a lot the need to code, its design capabilities are a bit old-fashioned.

Figure 3.5 presents the user interface developed for this project.

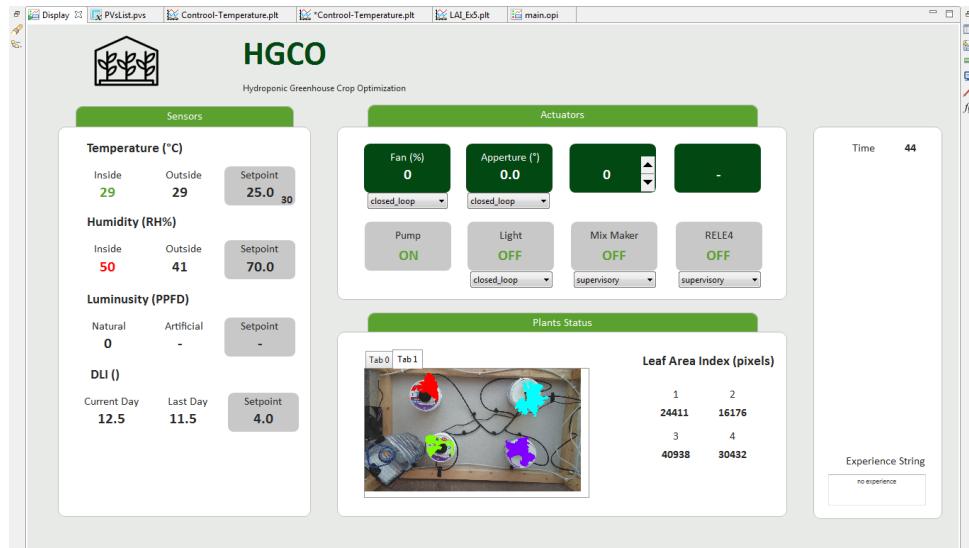


Figure 3.5: CSS user interface allowing to monitor and control greenhouse live conditions.

## Communication Protocol

The Arduino microcontroller and the Raspberry Pi computer communicate via Serial (RS-232). Requiring thus the existence of a communication protocol. This is the simplest it could be: In case of a get message, EPICS sends to Arduino a request message composed of a single character. Then the Arduino responds with the respective answer as a float number. In case of a set message, EPICS sends a character followed by a space and a float number. Again, Arduino will interpret the character and understand that it is followed by a float number and should set the respective variable accordingly to it.

## Process Variables

EPICS stores instances of variables called Process Variables (PV). These can be input or output values as well as intermediate calculations that will allow to define other variable value. Some of its properties allow to:

- Get data from other records or from hardware
- Perform calculations
- Check if values are in range and raise alarms
- Push data to other records or to hardware
- Activate or disable other records
- Wait for hardware signals (interrupts)

- Specifying client monitoring properties

An example for the temperature PV defined in EPICS is as:

```

1 #Temperature of inside DHT22 Sensor
2 record(ai, "$(user):TempIn"){
3     field(DTYP, "stream")      #defines that should get variable from arduino
4     field(INP, "@arduino.proto.getTemp()_$(PORT)")  #specifies protocol function
5     field(DESC, "Scaled_Temperature")
6     field(SCAN, "2_second")    #scan periodicity
7     field(LOPR, "0")          #specifies low operating range
8     field(HOPR, "50")         #specifies high operating range
9     field(ADEL, "0.1")        #specifies precision to display
10    field(EGU, "Celsius")     #describes the engineering units to display
11    #Alarms                  # specify alarms limits
12    field(LOLO, "0")
13    field(LOW, "15")
14    field(HIGH, "35")
15    field(HIHI, "40")
16 }
```

A great advantage defining these parameters is that, they can not only be useful within the EPICS development, they can also be used in the CSS user interface, facilitating the interactive properties. For example, if one wants the interface to show a pop up message in case of a alarm limit exceed the limits do not have to be redefined in CSS.

### 3.1.4 Data Base

By default, EPICS does not archive any data of the process variables (PVs). Although there is a plugin called Archive Engine that allows you to do it. It takes PV's data samples from EPICS and saves them in the data storage database. Thus, before using this tool, it is necessary to create a proper table structure. A MySql database was configured within the Pi, but Oracle or PostgreSQL are also supported. Not only the samples are saved in the database but also in the configuration file.

CSS GUI is also capable of connecting to the database allowing the user to see past stores samples, for example showing graphs with the evolution of PVs.

An overall summary of the archive is presented in Figure 3.6

Unfortunately, archive engine software has not been compiled yet for the Raspberry Pi and thus it had to be installed in another Linux computer. It is important to emphasize that this was the only application installed in a secondary computer. The MySql DB was installed in the Pi computer.

### Sample Modes

The archive engine supports several sample modes, i.e. ways in which it decides what samples should be written to the archive data store. The most relevant modes are *sample mode* and *monitored with threshold*. In the first one, Pv records are stored periodically, whereas in the second samples are stored

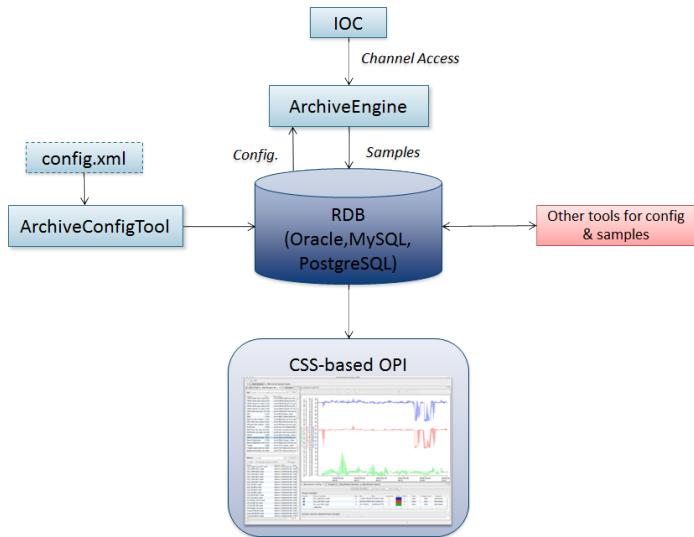


Figure 3.6: Archive Engine System Overview

only when they differ from the last written sample by at least some configurable margin. Although this margin can be set in the archive engine config file, it can also be set in the EPICS PV definition with the ADEL tag.

Finally, the archive engine provides a web base front end easily allowing to follow the database overall condition and the archive engine status. For better a management of the MySql DB, phpMyAdmin plug-in was installed. This also offers a web-based front end for an easier DB configuration and management.

### 3.1.5 Security

In order to remotely develop the project and to have live external access to the greenhouse the Pi was connected to the Worl Wide Web with a public IP and no firewall protecting it. This means total exposure for hacker attacks. Therefore, at this state, it is very important to refer to the system security.

To avoid any unwanted attacks, it is important to only allow certain computers to change and access the system status. EPICS and CSS already provide this tool. So, the access to these tools was restricted to certain computer IPs. The same was done to the MySQL DB.

### 3.1.6 Sensors and actuators setup and calibration

Only some of the many variables influencing crop growth were monitored and controlled. Captured light, greenhouse temperature and relative humidity where directly monitored by sensors. On the other hand, a fan and a window opening allowed to control the inside temperature and relative humidity, while an artificial light allowed to increase the amount of light supplied to the plants. Although part of the setup and calibration of these hardware devices is presented next, its selection choice and other additional details are presented in annex C.

## Light Sensor

Ideally, to work with plants a PAR sensor would be the most appropriate. However, they are very expensive and not suitable for small projects. The affordable, yet precise, *TSL2561* lux sensor was selected. This comes already calibrated providing a trustable reading. Indeed such statement was confirmed using the *Mastech MS6610* luxemeter. However, the sensor easily saturated when exposed to direct sunlight. Adjusting its gain and integration time via software was possible and helped, but it was not enough. Therefore, a home made plastic cap was used as a filter to produce an adequate attenuation. At this point the sensor required to be calibrated. Thus, an experiment took place measuring the sensor and luxemeter value at different light conditions. According to Figure 3.7 the calibration is linear. The original trendline could lead to small negative measurements, therefore it was forced to cross the origin.

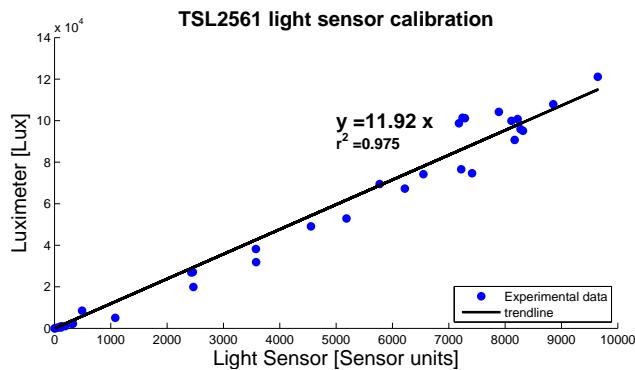


Figure 3.7: Linear calibration of TSL2561 lux sensor.

Note that this calibration still provides a reading in lux units, although it is important to measure PAR units. Because this sensor is facing the sun, and according to section 2.4.2, knowing the light source is possible to relate its measurement with PPFD units, according with

$$light(PPFD) = 0.0185 \times 11.92 \times SensorInput, \quad (3.1)$$

where 11.92 already corresponds to the linearization computed with the data on Figure 3.7.

## Temperature and relative humidity sensor

There are a lot of temperature and humidity sensors for home cheap applications. While temperature sensors are usually quite reliable, humidity readings are slightly less accurate. The *DHT22* sensor is able to measure both temperature and relative humidity in air, claiming a 2% humidity precision. The manufacturer also claims that these come already calibrated from factory. In fact, when submitting two of these sensors at the same conditions along with a mercury thermometer (*ZEAL*) the temperature measurement of the sensors had the maximum difference from each other of 1 Celsius, and a maximum difference from the reference thermometer of 1.5 Celsius. Therefore, it was decided not to do any calibration to the temperature readings. To confirm the sensors humidity values, a similar experience was performed, this time with two mercury thermometers, where one had its end submerged in water. This technique allows to use the dry and wet measurements of the thermometers and, by looking at

a table, infer about the relative humidity. Although this experience was not so successful as the first one because the error of reading the table can be more than 10 percent. Also, submitted to the same conditions, the various *DHT22* sensors showed a difference of more than 5% between them.

Therefore, it was decided to do another experiment this time with an electrical meter: *EXTECH SD800* datalogger, claiming an accuracy of  $\pm 0.8^{\circ}C$  and  $\pm 4RH\%$ . Comparing the temperature readings from the sensors and the meter again, it showed a bigger contrast - maximum of  $3^{\circ}$  Celsius. However, the humidity readings were a lot more satisfactory then before. Therefore, a calibration was performed as the Figure 3.8 suggests.

$$RelativeHumidity1(\%RH) = 0.992 * SensorInput + 5.886 \quad (3.2)$$

$$RelativeHumidity3(\%RH) = 0.948 * SensorInput + 2.259 \quad (3.3)$$

After this calibration the relative humidity of the two sensors at the same conditions would not differ more than 3% from each other and not more than 5% from the reference meter.

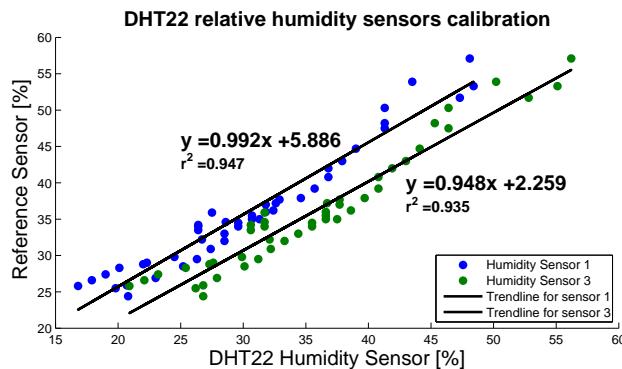


Figure 3.8: Linear calibration for DHT22 humidity sensors.

### Window aperture actuator

The top window is opened by means of the servo motor attached to it, allowing a passive air flow control. An experiment was performed measuring the angle of the window aperture for different servo positions. According to Figure 3.9, it was possible to linearly relate this entities in certain regions.

### Fan actuator

When passive air flow control is not enough, a fan allows to force air ventilation. As explained in appendix C, the selected fan has a very good air flow capacity, providing a smother and wider control ability. Because the fan has a 12 V supply, in the annex, it is also detailed how the fan was connected to the Arduino throughout a transistor, allowing to control the fan speed with the Arduino PWM controller.

Experimentally, the fan revealed to have a mechanical static friction. Therefore, until around 20% duty cycle it does not operate. This discontinuity revealed to be a problem when controlling the inside

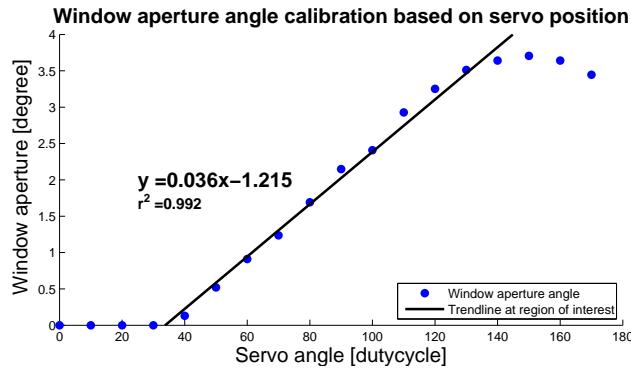


Figure 3.9: Linear mapping of the top window aperture according to the servo motor angle position.

humidity as it will be developed in section 4.4. It is assumed that for higher values its speed grows linearly with the operating dutycycle.

### LED artificial light actuator

Daily light integral (DLI) corresponds to the total amount of light received by plants per day. Integrating over a full day the PPFD light measured every second and multiplied by  $10^{-6}$  gives us DLI. Computing this parameter every day and storing it, allows to track the effect of light in plant growth.

In order to achieve high DLI values a LED light allows to complement the sun light supply. As a reinforcement light, it does not have to be very powerful. The greenhouse light sensor revealed that the plants would capture around 80 PAR from the LED (this was not a rigorous measurement - see appendix C.4 for details).

To control the LED light, once it has a 120 V power supply, it was connected to the relay which is operated by the Arduino. Therefore, this is a simple on-off actuator.

### Camera - crop growth monitoring

The three most interesting techniques that could be used to compute the crop weight by means of image processing are:(i) using a single top view camera, (ii) using a top view and lateral view camera, and (iii) using a stereo vision camera (two side-by-side). Because the simple process of counting the leaf 2D area of a top view camera revealed to be quite satisfactory and, for the sake of simplicity, this method was adopted.

Based on section 2.4.2 research, it was decided to use the open source python library [PlantCV](#), able to run on the Raspberry Pi.

The Pi camera takes top view photos of the plants inside the greenhouse. Therefore, the first step should be to differentiate the multiple plants from the photos' background. This is done by the first implemented image process pipeline, i.e. a set of image processing functions, which distinguishes the different plants based on their green color in contrast with the background color. This pipeline is available at PlantCV documentation at [plantcv.readthedocs.io](http://plantcv.readthedocs.io). Then, from the different split images, the documentation provides another pipeline allowing to find plant contours which also count the corresponding number

of pixels. A sample image, revealing the successful python implementation, is presented in Figure 3.10

As explained in section 2.4.2 it is possible to relate the LAI with the plant weight. However, this calibration would require a scale with good precision and a couple of destructive measurements would have to occur. Therefore, no calibration was performed.

However, because the number of pixels is not a very intuitive and readable scale, the following normalization was considered:

$$100 \text{ grams} = 2.5 \times 10^5 \text{ pixels}$$



Figure 3.10: Sample image after image processing. The pipeline identifies different plants and its contours, allowing thus to compute the corresponding LAI.

# Chapter 4

## Growth environment

In order to control some of the environmental variables that affect plant growth it is important to develop models of those variables. Supported by these models, linear control is used to regulate the greenhouse inside temperature and humidity. Whereas, a very simple on-off controller is implemented to control the amount of light the plant receives daily. Due to budget and time limitations, the nutritive solution was not controlled automatically.

### 4.1 Modeling greenhouse temperature

**Temperature** is a measure of the average kinetic energy of particles in a certain environment. On the other hand, **heat** is the transfer of thermal energy due to a temperature difference between the system and its surroundings. The transfer or dispersion of heat can occur by means of three main mechanisms: conduction, convection and radiation. In this greenhouse, it will also be considered ventilation, which represents a form of forced convection as well as electrical heating.

From Fourier heat conduction law it is known that  $\frac{dQ}{dt} = \frac{kA\Delta T}{D}$  where  $Q$  is the conduction heat ,  $k$  the thermal conductivity of the material,  $A$  is the cross-section surface area ,  $\Delta T$  and  $D$  is the temperature difference and distance between the ends of the material respectively. Thus, the conduction heat loss throughout the walls of the greenhouse is written as

$$\frac{dQ_{loss}}{dt} = \frac{kA(T_{in} - T_{out})}{D} = \frac{T_{in} - T_{out}}{R_{eq}} [W] \quad (4.1)$$

where  $R_{eq}$  represents the total thermal resistance of the house expressed in  $^{\circ}\text{C}/\text{W}$ .

From the derivative of the heat capacity relation

$$Q = mc\Delta T \quad (4.2)$$

the convection passive heat loss trough the roof aperture of the house can be written as

$$\frac{dQ_{roof}}{dt} = \frac{dm_{roof}}{dt} c_{air}(T_{in} - T_{out}) \approx K_{roof}\Theta_{roof}(T_{in} - T_{out}) [W] \quad (4.3)$$

where  $\frac{dm_{roof}}{dt} \approx K_{roof}\Theta_{roof}$  denotes the amount of mass moved by air convection allowed with a roof aperture of angle  $\Theta_{roof}$  where  $K_{roof}$  have the dimensions  $Kg/\text{o}$ .

The forced ventilation follows the same equation as the passive ventilation

$$\frac{dQ_{fan}}{dt} = \frac{dm_{fan}}{dt}c_{air}(T_{in} - T_{out}) \approx K_{fan}V_{fan}(T_{in} - T_{out}) [W] \quad (4.4)$$

but now the amount of mass moved is proportional to the speed of the fan  $V_{fan}$  as  $\frac{dm_{fan}}{dt} \approx K_{fan}V_{fan}$ ,  $K_{fan}$  can be computed accordingly to the fan characteristics.

The radiation heat received by the sun is proportional to the solar irradiance  $L(t)$  with units joule per second per square meter. In reality  $K_{rad}$  is not constant, instead it depends on the angle of the sun along the year. However it will be assumed as constant for the sake of simplicity.

$$\frac{dQ_{rad}}{dt} = K_{rad}L(t) [W] \quad (4.5)$$

Finally, the Joule heat flux is

$$\frac{Q_H}{dt} = H(t) = RI^2 [W]. \quad (4.6)$$

Therefore, the inside house temperature comes as

$$\frac{dT_{in}}{dt} = \frac{1}{m_{house}c_{air}}\left(\frac{dQ_H}{dt} + \frac{dQ_{rad}}{dt} - \frac{dQ_{loss}}{dt} - \frac{dQ_{fan}}{dt} - \frac{dQ_{roof}}{dt}\right) \quad (4.7)$$

where  $m_{house}$  is the air mass in kilograms of the inside greenhouse, and  $c_{air}$  the air heat capacity in  $J.\text{oC.Kg}^{-1}$ .

This last model has some non-linear relations namely of the fan and the window aperture actuators because this actuators parameters are being multiplied with the state variable to be modeled ( $T_{in}$ ).

However, for the sake of simplicity, and also because we are dealing with relative slow control, a linear continuous model is suggested based on the previous one as

$$\frac{dT_{in}}{dt} = -K_{loss}(T_{in} - T_{out}) + K_{sun}L(t) - K_{roof}\Theta(t) - K_{fan}V_{fan}(t) + K_HH(t) - K_MM(t) \quad (4.8)$$

where the house mass and air heat capacity constants are implicit, taken in consideration within the  $K$  constants. This leads to the arising of  $K_H$ , which corresponds to  $\frac{1}{m_{house}c_{air}}$  and of  $K_{loss} = 1/R_{eq}$ . Moreover, note that the model has another additional term  $K_MM(t)$ . The mist maker actuator  $M(t)$ , responsible to increase air humidity, was added, considering that its acting will decrease the greenhouse temperature linearly.

Now all the  $K$  coefficients must be computed experimentally, once they will depend on the physical location of the greenhouse and its building structural characteristics. Note, that  $K_{roof}$  and  $K_{fan}$  are not constants. For example, whenever there is no difference of temperature from the inside and outside of the greenhouse, acting on the fan or in the windows aperture, should not change the temperature inside the greenhouse. Moreover, if the difference of temperatures is negative, namely, when the outside

temperature is bigger than the inside, acting on the ventilation should have a reverse effect.

An acceptable approximation is to say that these coefficients vary linearly with the temperature's difference as

$$K_{roof}(T_{in}, T_{out}) = \begin{cases} K'_{roof} & \text{if } T_{in} - T_{out} > t_{sat} \\ -K'_{roof} & \text{if } T_{in} - T_{out} < t_{sat} \\ K'_{roof} \frac{T_{in}-T_{out}}{t_{sat}} & \text{if } |T_{in} - T_{out}| \leq t_{sat} \end{cases}$$

$$K_{fan}(T_{in}, T_{out}) = \begin{cases} K'_{fan} & \text{if } T_{in} - T_{out} > t_{sat} \\ -K'_{fan} & \text{if } T_{in} - T_{out} < t_{sat} \\ K'_{fan} \frac{T_{in}-T_{out}}{t_{sat}} & \text{if } |T_{in} - T_{out}| \leq t_{sat} \end{cases}$$

where  $t_{sat}$  represents the average difference from the inside and outside temperature of the experience that took place when computing  $K_{roof}$  and  $K_{fan}$  coefficients.

#### 4.1.1 Parameters identification

As explained in section 2.7, using Euler's method, the linear continuous differential equation (4.8) can be rewritten in discrete time as

$$T_{in}(k+1) = T_{in}(k) + \theta_{T1}[T_{in}(k) - T_{out}(k)] + \theta_{T2}L(k) + \theta_{T3}\Theta(k) + \theta_{T4}V_{fan}(k) + \theta_{T5}M(k) + \theta_{T6}H(k) \quad (4.9)$$

where,

$$\theta_{Ti} = \begin{bmatrix} -K_{loss}T_s \\ K_{sun}T_s \\ -K_{roof}T_s \\ -K_{fan}T_s \\ -K_M T_s \\ K_H T_s \end{bmatrix}$$

In this section a series of experiments took place to estimate the constants  $\theta_{Ti}, i \in \{1, 2, \dots, 6\}$  of the discrete model which in turn allow to compute the same constants in continuous time. With these constants it will be possible to get a model of the temperature not only in discrete but also in continuous time. Achieving this will be fundamental when designing the controller. Note that the last constant was not estimated once the heater will not be used to control the greenhouse, because this project took place during Summer time.

### Thermal losses term : $K_{loss}$

If one turns off the fan, closes the window aperture, and puts the greenhouse away from external radiation equation (4.9) becomes a first order equation depending only on  $\theta_{T1}$  as

$$T_{in}(k+1) = T_{in}(k) + \theta_{T1}[T_{in}(k) - T_{out}(k)]. \quad (4.10)$$

An experiment on this conditions was performed. The initial inside temperature was raised through an external heater that was turned off before time zero. Therefore, by analyzing the decay of the inside temperature, represented in Figure 4.1, it is possible to estimate  $\theta_{T1}$  using the LSQ method as described in section 2.7.

Consider that  $Y = T_{in}(k) - T_{in}(k-1) \in \mathbb{R}^N$  is the inside temperature experimental data column vector with  $N$  samples, and  $\Phi = [T_{in}(k-1) - T_{out}(k-1)] \in \mathbb{R}^N$  is a matrix composed by a column of the inside temperature and another with the difference of temperatures one sample delayed in respect to  $T_{in}(k)$ . Writing the LSQ linear relation  $Y = \Phi\theta_T$ , the estimative of the linear constant parameter  $\hat{\theta}_T$  come as

$$\hat{\theta}_T = (\Phi^T \Phi)^{-1} \Phi^T Y, \quad (4.11)$$

which from equation (4.10) is considered as

$$\hat{\theta}_T = \hat{\theta}_{T1}. \quad (4.12)$$

The results from the previous experience calculus was  $\hat{\theta}_{T1} = -0.023$  for a sampling period of  $T_s = 60s$ .

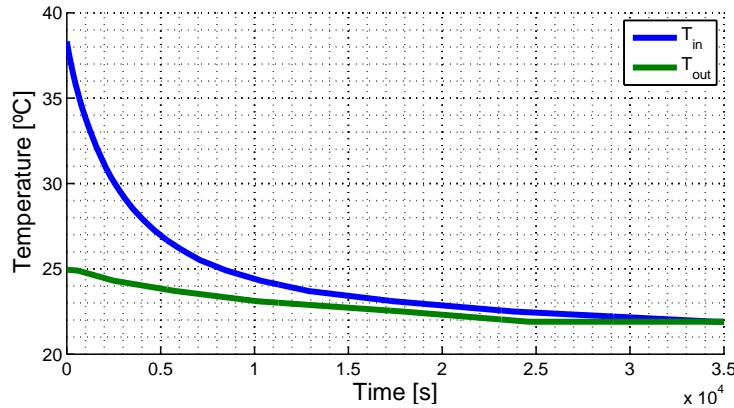


Figure 4.1: Temperature inside and outside the greenhouse after turning off an external heater. In this experiment the fan and the window aperture were forced to zero, and there was no solar radiation. It allows to compute the decay constant time characteristic of the greenhouse temperature evolution. Note however that  $\hat{\theta}_{T1}$  time constant can not be computed directly from this graph but rather from the differences of these curves.

The estimative  $\hat{\theta}_{T1}$  is a constant in discrete time, so one can predict the next inside temperature sample based on the current state of the greenhouse. This is confirmed in Figure 4.2.

From equation (4.9),  $K_{loss}$  constant of the continuous differential equation is given by  $K_{loss} = -\frac{\hat{\theta}_{T1}}{T_s} =$

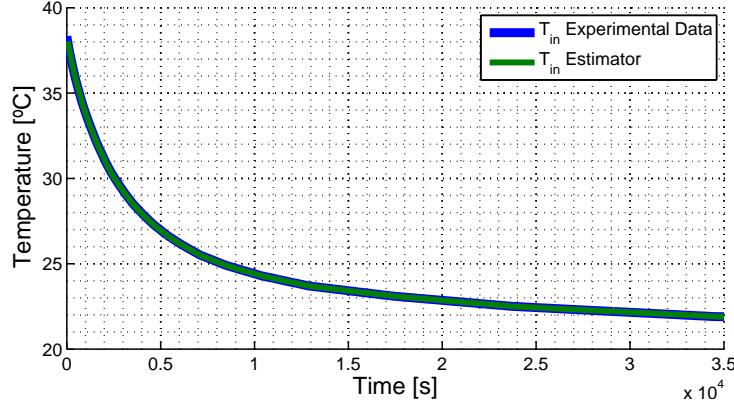


Figure 4.2: Prediction of the next sample  $k + 1$  inside temperature based on the current state  $k$  using the discrete model on equation (4.10) and the estimators computed based on experience of Figure 4.1

$$3.8 \times 10^{-4}$$

Now it is possible to simulate the inside temperature evolution in continuous time, based only on the initial condition as shown in Figure 4.3.

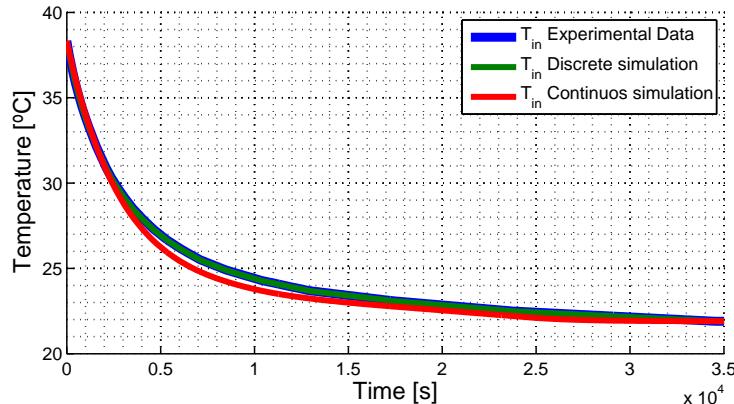


Figure 4.3: Discrete and continuous time inside temperature simulation using only the initial condition. The Figure proofs a correct transformation between the discrete and continuous time model identification. There is a slightly difference between the simulated and experience data. This can be explained as in the real greenhouse there are elements that store energy (like the greenhouse walls) that will release it over time, explaining thus the slower response of the real system compared to the simulation.

#### **Sunlight term: $K_{sun}$**

The next experiment performed intends to determine the constant that reflects the sun influence on the greenhouse. The discrete equation, where the whole system acting is turned off and there is only the presence of solar radiation, can be considered as

$$T_{in}(k + 1) = T_{in}(k) + \theta_{T1}[T_{in}(k) - T_{out}(k)] + \theta_{T2}L(k). \quad (4.13)$$

An experiment under these conditions is presented in Figure 4.4.

Knowing  $\hat{\theta}_{T1}$  estimator, equation (4.13) can be written in the linear format  $Y = \Phi\theta_T$  where,  $Y =$

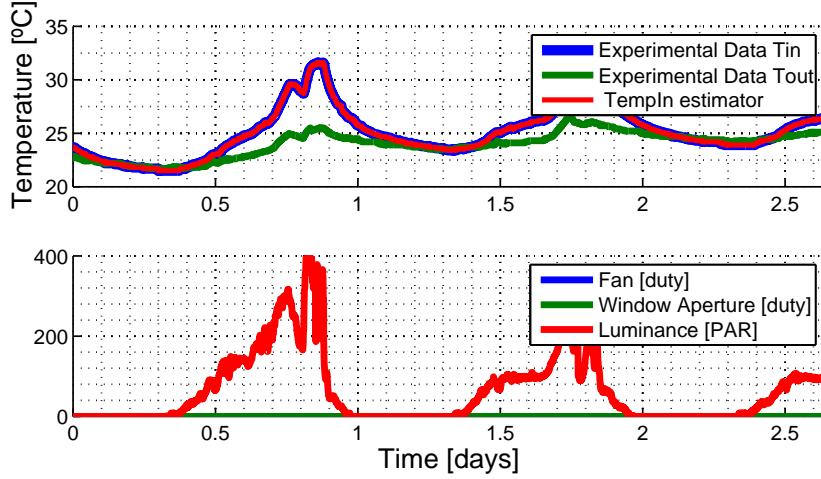


Figure 4.4: Experiment that took place to fit the term related to the influence of sunlight on the greenhouse inside temperature. The picture also presents the simulation based on the experiment results. The simulation predicts the temperature  $T_{in}(t + 1)$  based on the sample at current time  $t$ .

$T_{in}(k + 1) - T_{in}(k) - \hat{\theta}_{T1}[T_{in}(k) - T_{out}(k)] \in \mathbb{R}^N$ ,  $\Phi = L(k) \in \mathbb{R}^N$ , and  $\theta_T = \theta_{T2}$ . Following the same thinking as before the estimative is  $\hat{\theta}_{T2} = (\Phi^T \Phi)^{-1} \Phi^T Y$ . The experimental data of Figure 4.4 gives  $\hat{\theta}_{T2} = 4.2 \times 10^{-4}$  whereas now  $K_{sun} = \frac{\hat{\theta}_{T2}}{T_s} = 7.1 \times 10^{-6}$ .

The same Figure 4.4 shows the discrete simulation based on the results so far.

The following two terms ( $K_{roof}$  and  $K_{fan}$ ) were computed based on steady state experiments. However, it is not straightforward to project an experiment controlling the sunlight intensity, that is why  $K_{sun}$  was not computed with a steady state experiment.

#### Window aperture term: $K_{roof}$

An experiment took place in order to understand how the window aperture actuator ( $\Theta$ ) influences the internal temperature of the greenhouse. The fan was turned off and there was no presence of sunlight. Moreover, a heater ( $H$ ) was working inside the greenhouse at a constant power. The difference of the inside and outside temperatures were registered for different window aperture angles every time the inside temperature achieved steady state. The corresponding discrete equation comes as,

$$T_{in}(k + 1) = T_{in}(k) + \theta_{T1}[T_{in}(k) - T_{out}(k)] + \theta_{T3}\Theta(k) + \theta_{T6}H. \quad (4.14)$$

The result is presented in Figure 4.5.

Although the response is not really linear, it was approximated using the measurements marked in green. Once more, this was done using the LSQ method. The only difference is that, because the steady state is considered, one can state that  $T_{in}(k - 1) = T_{in}(k)$ . Therefore, equation (4.14) can be rewritten in the linear format  $Y = \Phi\theta_t$  where,  $Y = -\hat{\theta}_{T1}[T_{in}(k) - T_{out}(k)] \in \mathbb{R}^N$ ,  $\Phi = [1 \ \Theta(k)]^T \in \mathbb{R}^{N \times 2}$ , and  $\theta_T = [\theta_{T6}H \ \theta_{T3}]^T \in \mathbb{R}^2$ .

Note that the first element of the estimator is not an independent constant. This is related to the

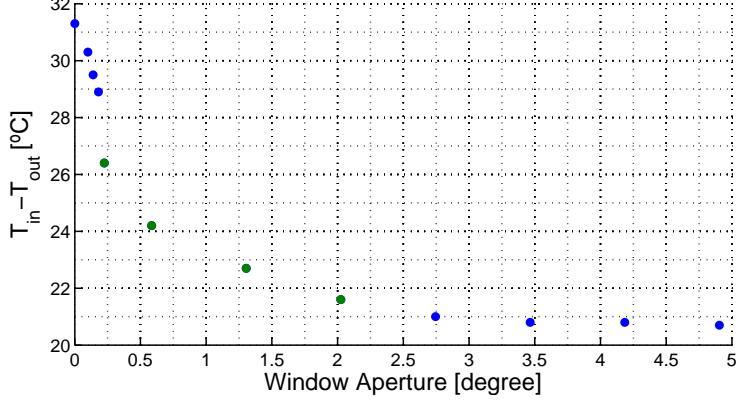


Figure 4.5: Influence of the window aperture ( $\Theta$ ) on the greenhouse inside temperature when a heater is operating inside at a constant power. The heater allows to maintain a measurably difference of temperatures. Only the green dots were considered for the linearization.

heater, because there was no intention of computing the used heater influence on the greenhouse. As before, the estimate comes as  $\hat{\theta}_T = (\Phi^T \Phi)^{-1} \Phi^T Y$ . Using the green data of Figure 4.5 gives  $\hat{\theta}_{T3} = -4.7 \times 10^{-2}$  whereas  $K_{roof} = -\frac{\hat{\theta}_{T3}}{T_s} = 7.8 \times 10^{-4}$ .

#### Fan term: $K_{fan}$

The term regarding the fan actuator ( $V_{fan}$ ) was also computed throughout an experiment where the steady state of the difference of temperatures was registered for different fan speeds. Again, a heater was turned on at a constant power to maintain a difference of temperatures. Moreover, the window aperture was fixed to its maximum aperture angle  $\Theta_{max}$ , once it makes more sense to actuate using the fan only whenever the window aperture is fully opened. The corresponding discrete equation is,

$$T_{in}(k+1) = T_{in}(k) + \theta_{T1}[T_{in}(k) - T_{out}(k)] + \theta_{T3}\Theta_{max} + \theta_{T4}V_{fan}(k) + \theta_{T6}H. \quad (4.15)$$

The results are presented in Figure 4.6.

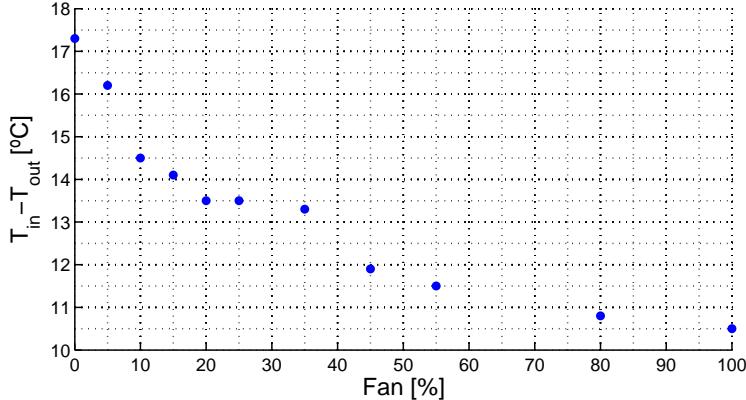


Figure 4.6: Influence of the fan speed on the greenhouse inside temperature. The experiment was performed were a heater was turned on inside the greenhouse in order to maintain a measurably difference of temperatures. The greenhouse aperture angle was maintained at its maximum aperture.

This time the fan actuator seemed to have a more linear relation with the temperature. Again, the LSQ method was applied. In steady state, equation (4.15) can be rewritten as  $Y = \Phi\theta_T$  where,  $Y = -\hat{\theta}_{T1}[T_{in}(k) - T_{out}(k)] - \hat{\theta}_{T3}\Theta_{max} \in \mathbb{R}^N$ ,  $\Phi = [1 V_{fan}(k)] \in \mathbb{R}^{N \times 2}$ , and  $\theta_T = [\theta_{T6}H \ \theta_{T4}]^T \in \mathbb{R}^{N \times 2}$ . The result was  $\hat{\theta}_{T4} = -1.4 \times 10^{-3}$  and therefore  $K_{fan} = -\frac{\hat{\theta}_{T4}}{T_s} = 2.3 \times 10^{-5}$ .

#### Mist maker term: $K_M$

Finally, the last term concerns the mist maker effect on the temperature. Another simple experiment took place where there was no acting besides the mist maker and there was no sunlight effect. The corresponding discrete equation is,

$$T_{in}(k+1) = T_{in}(k) + \theta_{T1}[T_{in}(k) - T_{out}(k)] + \theta_{T5}M. \quad (4.16)$$

Applying the LSQ method, where  $Y = \Phi\theta_T = T_{in}(k+1) - T_{in}(k) - \hat{\theta}_{T1}[T_{in}(k) - T_{out}(k)] \in \mathbb{R}^N$ ,  $\Phi = M(k)^T \in \mathbb{R}^N$ , and  $\theta_T = \theta_{T5} \in \mathbb{R}^N$ . Figure 4.7 presents the experimental data yielding  $\hat{\theta}_{T5} = -0.13$  and therefore  $K_M = -\frac{\hat{\theta}_{T5}}{T_s} = 2.3 \times 10^{-3}$ . The same Figure shows the discrete model response to the same mist maker input data.

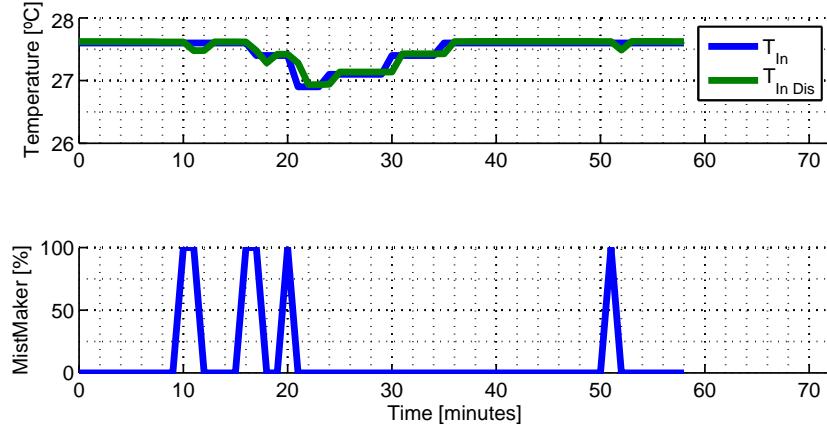


Figure 4.7: Influence of the mist maker on the greenhouse inside temperature (blue line). The remaining actuators were turned off and there was no sunlight effect. It is possible to notice the delay on the temperature after acting. The green line shows the corresponding discrete model to the same input data.

#### 4.1.2 Model validation

In order to confirm the discrete and continuous model, an experiment took place where the inside temperature was foreseen and compared to the experimental one. The continuous model predicts the inside temperature based only on the initial inside temperature, the outside temperature, external luminance, roof aperture angle and the fan speed. The discrete time model simulated, not only depends on this variables, but also on the last temperature sample obtained experimentally ( $T_{in}(k-1)$ ). The performance of the model is presented in Figure 4.8.

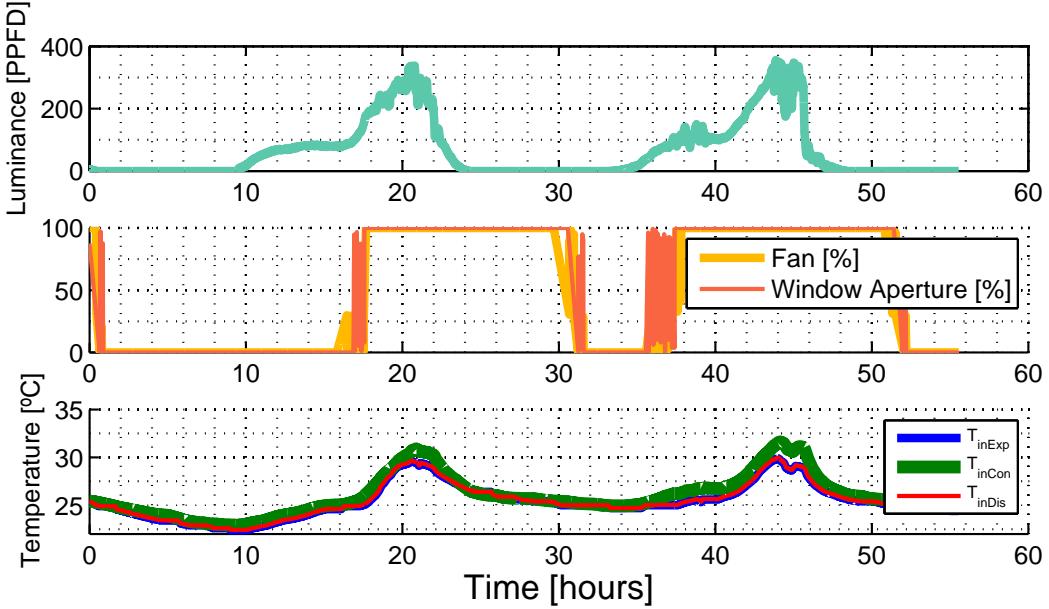


Figure 4.8: Two days performance of the continuous and discrete obtained models. All the curves represent experimental data except for  $T_{inDis}$  and  $T_{inCon}$ .  $T_{inCon}$  is the inside predicted temperature using the continuous model based only on the initial condition of the inside temperature at time  $t = 0$ .  $T_{inDis}$  predicts the next inside temperature based on the last available experimental sample  $T_{inExp}(k)$ . The root mean square error between the continuous and discrete simulation with the experimental data is  $RMSE_{con} = 0.75^\circ$  and  $RMSE_{dis} = 0.0283^\circ$  respectively.

All things considered, the obtained model revealed to properly predict the inside temperature of the greenhouse within a very good accuracy. Moreover, the discrete model error is on average below the temperature sensors quantification level. It is now possible to move on to design a controller for the temperature of the greenhouse.

## 4.2 Control greenhouse temperature

A linear model for the greenhouse temperature is now known. It is thus possible to design a controller in simulated environment. The goal is to actuate on the window aperture and fan speed, accordingly to the greenhouse conditions, so that the desired temperature setpoint is achieved. Despite the ability of the mist maker actuator to decrease the inside temperature, it is not substantial, and therefore it was not considered as a manipulated variable.

In order to follow the temperature setpoint with null steady-state error, it was decided to use a Proportional Integral (PI) controller for the window aperture, which is simple and easy to implement. This main loop includes an anti-windup mechanism, which consists on subtracting a term to the integral whenever the actuator is saturated, represented in Figure 4.9a. When projecting the PI gains the Ziegler Nichols frequency response method was used, yielding the values  $K_{p_{ap}} = -75$ ,  $K_{i_{ap}} = -0.58$  for a control period of 10 seconds. Initially, the controller was dimensioned to operate at one minute periodicity, as done on the modeling experiments. Although this would perfectly work for temperature control, as it will be discussed further on, a faster control period would benefit the humidity control. Despite these gains

contributed for a good response, it revealed to be slightly oscillatory, in particular when the humidity controller was added. Moreover, these gains revealed to be quite high as the actuators would easily saturate. Therefore, the original gains were reduced 10 times. This is,  $K_{p_{ap}} = -7.5$ ,  $K_{i_{ap}} = -0.058$ .

Whenever the window aperture gets saturated it is no longer possible to increase the airflow in order to control the temperature. Therefore, at these circumstances the fan actuator is triggered. The fan speed is proportional to the difference from the window aperture signal before and after its saturation limits. This will ensure that the fan is only operating when the aperture window is saturated. This secondary control loop is presented in Figure 4.9b. It is composed by a single proportional term dimensioned such that the fan capacity to reduce the greenhouse temperature is proportional to the window aperture, this is  $K_{p_{fan}} = \frac{K_{roof}}{K_{fan}} = 33$ .

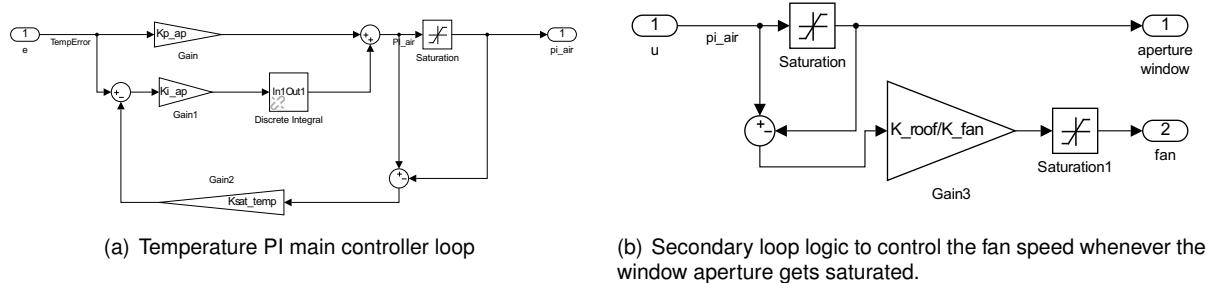


Figure 4.9: Simulated PI temperature controller. The fan proportional term is fed by the difference from the aperture window before and after its saturation so that it only actuates when the window is saturated.

A simulation is presented in Figure 4.10 and an experimental result in Figure 4.11. Above all, the controller was successfully implemented within its actuating limits. However, the temperature actuators revealed to be limited to the greenhouse external conditions as they tend to get easily saturated.

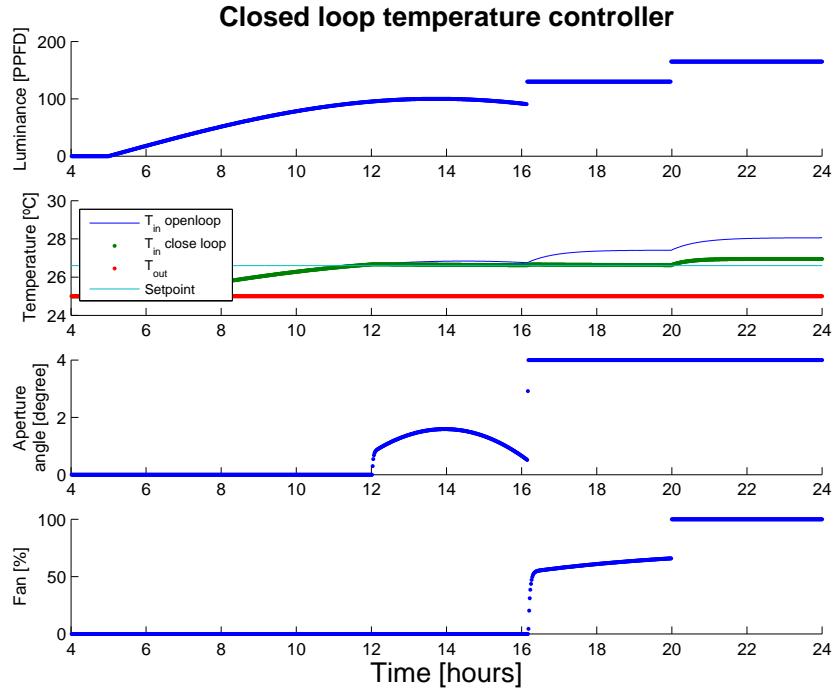


Figure 4.10: Simulation of the temperature controller described in Figure 4.9. Within the controller acting limits the system stabilizes around the desired setpoint ( $t=[12;20]$ ). However, the actuators can only lower the temperature less than  $2^{\circ}\text{C}$  in respect to the open loop response due to saturation limits ( $t=[20;24]$ ).

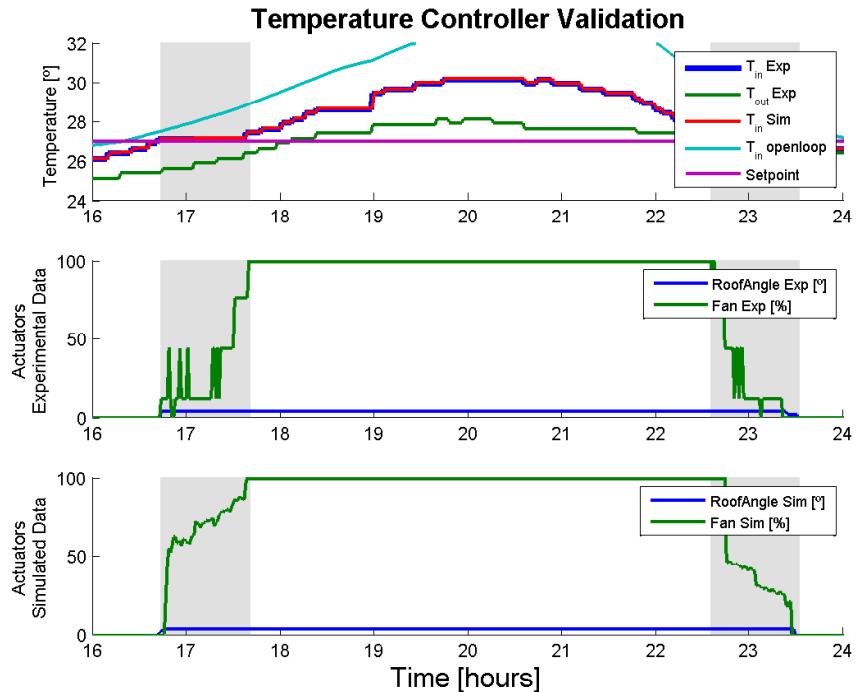


Figure 4.11: Experimental versus simulated controller results with the same controller gains. Because the model does not follow exactly the reality neither will the actuators. The experimental results, within the time highlighted in grey, revealed a good track of the setpoint when the actuators are not saturated.

### 4.3 Modeling greenhouse humidity

**Humidity** is the amount of water vapor present in air. In addition, the amount of water vapor that is needed to achieve saturation increases with temperature. Thus, as the temperature of a parcel of gas containing water vapor decreases, it will eventually reach a point without adding or losing water mass. This leads to distinct notions of humidity. **Absolute humidity** is the amount of water content in air expressed in grams per cubic meter. **Relative humidity**, expressed as a percent, measures the current absolute humidity relative to the saturation humidity for a given temperature [49].

A physical approach of the greenhouse relative humidity interactions is more complex compared to the temperature due to the humidity temperature dependence. Previous studies [50? , 51] showed that a more heuristic approach would still lead to accurate models. Here, a first order linear equation is proposed.

Part of the water transfer in the system is due to water losses throughout the walls borders and the air openings. Suggesting the first term of the model as

$$\frac{dH_{in}}{dt} = -K_{H1}(H_{in} - H_{out}), \quad (4.17)$$

where the vapor water flow is proportional to the gradient of partial pressures inside and outside.

Moreover, the greenhouse inside humidity is highly affected by the passive and active ventilation. This follows a similar relation as the one before, where the water flow should now be proportional to the ability for these actuators to move air mass. For the window aperture ( $\Theta$ ) as

$$\frac{dH_{in}}{dt} = -K_{H3}(H_{in} - H_{out})\Theta(t) \quad (4.18)$$

and for the fan actuator ( $V_{fan}$ )

$$\frac{dH_{in}}{dt} = -K_{H4}(H_{in} - H_{out})V_{fan}(t). \quad (4.19)$$

However, similarly to the temperature model, these two terms would lead to a non-linear differential equation, because the state  $H_{in}$  is multiplying the controls  $\Theta(t)$  or  $V_{fan}(t)$ . Therefore, the humidity terms will be omitted in the model but included in the respective linear coefficient, explained ahead.

As said before, and according to the Psychrometric chart, temperature and RH are interrelated. This is not a linear relation, although this model assumes a fixed operation point and a linearization around it. This term is thus

$$\frac{dH_{in}}{dt} = -K_{H6}T_{in}(t). \quad (4.20)$$

Regarding the mist maker actuator ( $M$ ), it allows to directly increase the greenhouse inside humidity . The mist maker ability to increase humidity is also considered to be proportional to its operation capacity as

$$\frac{dH_{in}}{dt} = +K_{H5}M(t). \quad (4.21)$$

Finally, the crop respiration will also contribute to increase the inside humidity proportionally to the total plant mass, in other terms, to the sum of Leaf Area Index LAI ( $\Sigma LAI$ ).

It is then reasonable to describe the inside relative humidity as

$$\frac{dH_{in}}{dt} = -K_{H1}(H_{in} - H_{out}) - K_{H3}\Theta(t) - K_{H4}V_{fan}(t) + K_{H5}M(t) - K_{H6}T_{in}(t) + K_{H7}\Sigma LAI(t). \quad (4.22)$$

### 4.3.1 Parameters identification

In this section the linear continuous differential equation (4.22) will be subject to some changes, be rewritten in discrete time, and its linear parameters will be computed experimentally.

The task of finding the humidity model parameters have proved, not surprisingly, harder to accomplish. Two similar approaches based on LSQ method were tested. The first one, similar to what was done with the temperature model, was to try to isolate each term effect on the humidity, for example by turning off the actuators one at a time. However, it was not possible to omit some parcels, like the temperature dependency. It was also virtually impossible to achieve steady state responses as done before. Therefore, the second approach was to do a single but richer experiment (due to its time extension it was omitted). The results should be as good as more excitation the system is subject to. Finally, in order to achieve a high level of accuracy a group of these rich experiments spaced in time were combined.

A single and longer experience can have the advantage of estimating better parameters. However, one loses the ability to analyze each term's non linearities or approximations precision.

Experimentally though, the model (4.22) revealed to be more accurate if the inside and outside humidity were considered separately (see comparison with (4.24)). Moreover, the term depending on the plants transpiration was suppressed because it changes quite slowly with time and a very long experience would have to take place. Applying the Euler's approximation to (4.24) the discrete model is thus as

$$H_{in}(k+1) = \theta_{H1}H_{in}(k) + \theta_{H2}H_{out}(k) + \theta_{H3}\Theta(k) + \theta_{H4}V_{fan}(k) + \theta_{H5}M(k) + \theta_{H6}T_{in}(k) \quad (4.23)$$

where

$$\theta_{Hi} = \begin{bmatrix} 1 - K_{H1}T_s \\ K_{H2}T_s \\ -K_{H3}T_s \\ -K_{H4}T_s \\ K_{H5}T_s \\ -K_{H6}T_s \end{bmatrix}, i \in \{1, 2, \dots, 6\}.$$

Applying the LSQ method, where  $Y = H_{in}(k) \in \mathbb{R}^N$ ,  $\Phi = [H_{in}(k-1) \ H_{out}(k-1) \ \Theta(k-1) \ V_{fan}(k-1) \ M(k-1) \ T_{in}(k-1)] \in \mathbb{R}^{N \times 6}$  yields  $\hat{\theta}_{Hi} = (\Phi^T \Phi)^{-1} \Phi^T Y \in \mathbb{R}^6$ .

The results from the experiment were

$$\hat{\theta}_{Hi} = [0.87 \ 0.20 \ -1.60 \ -0.22 \ 20 \ -0.04]^T$$

for a sample period of 10 *seconds* and therefore,

$$K_{Hi} = [0.0021 \ 0.0034 \ 0.026 \ 0.0037 \ 0.33 \ 0.0007].$$

Finally, the continuous model was complemented to expand its work region to a broader range of values. The ventilation terms were modified so that they would be smaller when the inside humidity is close to the outside (similar to the modification in the temperature model). In fact, in these conditions the ability for the ventilation to change the humidity reduces. Besides, adding the reverse effect on the ventilation acting whenever the outside humidity is bigger than the inside. The final continuous model, where the constants are already known, comes as

$$\frac{dH_{in}}{dt} = -K_{H1}H_{in} + K_{H2}H_{out} - K_{H3}\Theta(t) - K_{H4}V_{fan}(t) + K_{H5}M(t) - K_{H6}T_{in}(t) \quad (4.24)$$

where,

$$K_{H3}(H_{in}, H_{out}) = \begin{cases} K'_{H3} & \text{if } H_{in} - H_{out} > h_{sat} \\ -K'_{H3} & \text{if } H_{in} - H_{out} < h_{sat} \\ K'_{H3} \frac{H_{in} - H_{out}}{h_{sat}} & \text{if } |H_{in} - H_{out}| \leq h_{sat} \end{cases}$$

$$K_{H4}(H_{in}, H_{out}) = \begin{cases} K'_{H4} & \text{if } H_{in} - H_{out} > h_{sat} \\ -K'_{H4} & \text{if } H_{in} - H_{out} < h_{sat} \\ K'_{H4} \frac{H_{in} - H_{out}}{h_{sat}} & \text{if } |H_{in} - H_{out}| \leq h_{sat} \end{cases}$$

and  $h_{sat} = 25$  represents the average difference from the inside and outside humidity of the experience that took place.

### 4.3.2 Model Validation

The task of finding a mathematical expression for the greenhouse inside humidity revealed to be harder compared to the inside temperature. After computing the linear discrete model first order equation explained before in section 4.3.1, it was implemented in the physical plant system. This allowed to compare the live real system and the discrete model response. Therefore, allowing to tune some of the model parameters at runtime. Only the actuators parameters were tuned, i.e. the ventilation and the mist maker. These modifications allowed to slightly improve the humidity model.

It is important to note that the model revealed to be less accurate when both the ventilation and mist maker actuators are operating simultaneously.

An overall performance of the discrete model is presented in Figure 4.12.

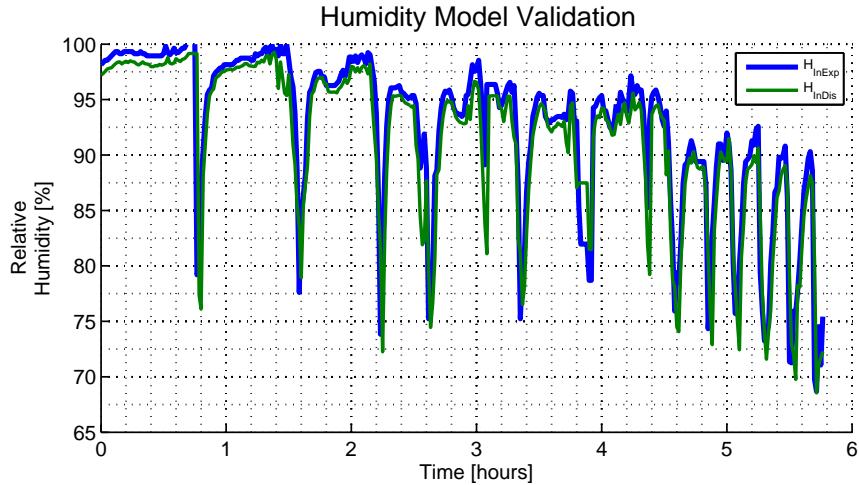


Figure 4.12: Performance of the inside humidity discrete model obtained after manually tuning the actuators parameters.  $H_{inDis}$  predicts the next inside humidity based on the last available experimental sample  $H_{inExp}(k)$  according to equation (4.23). The root mean square error of the difference from these two signals is  $RMSE_{H_{discrete}} = 3.03\%$ .

All things considered, the model obtained for the inside humidity revealed to be less accurate compared to the inside temperature. This time the discrete model has an average error around 3%, this is, 3 times the quantification accuracy of the sensor. However, as it will be seen in section 4.4 it will be good enough helping to design a controller off-line.

## 4.4 Humidity and temperature controller

At this stage, with the continuous linear model for the inside humidity of the greenhouse, it would be possible to develop an independent controller as done before with the temperature. However, according to the models, it is noticeable that one affects the other, and the challenge of controlling both will be greater. Indeed, one can understand, intuitively, that by decreasing the greenhouse temperature by ventilation means it will consequently force the inside humidity to approach the outside humidity. Therefore, the design of the humidity and temperature controller should be done simultaneously.

The problem of controlling multiple input multiple output systems (MIMO) like this, is met by the LQR control theory described in 2.8.4. Moreover, this controller has an optimal control approach allowing to tune according to the desired effort that one wants to put on the actuators. The LQR controller was developed in simulated environment and experimentally as well, furthermore including integral effect. However, this approach revealed to have a big disadvantage. The LQR controller tries to decrease the sum of the error of all state variables. In this particular case, this revealed to be a disadvantage when it is not possible to actuate on one of the variables. For example, when the temperature is lower than the setpoint the system cannot actuate to decrease this error due to saturation limitations. Consequently the LQR control will not actuate on the mist maker as it would increase this error even more. Therefore, limiting the humidity control. In the author's point of view, it is considered to be more important to try

to regulate the greenhouse humidity successfully even if that is not possible to the temperature, rather than trying to balance both. This opinion is based on this particular system physical limits of actuation.

Therefore, the next step was to try to combine the existing PI temperature controller with another PI for the humidity. Firstly, the humidity PI gains were tuned using Ziegler Nichols' method and the anti-windup method implemented as in section 4.2. The PI gains also suffered a decrease in modulus such that the response would be softer as it revealed to be too drastic in the real system. Secondly, there was also the inclusion of a feed forward term regarding the effect of the ventilation input computed by the temperature controller. This helped to reduce the unwanted effect of disturbing the humidity control caused by the temperature controller. The feedforward modification was only applied to the humidity control because the temperature inside the greenhouse does not change as fast as the humidity. Finally, the humidity PI control was split in positive or negative control. In other words, if the inside humidity is lower than the desired the actuation is done by the mist maker, in the opposite case it will actuate on the ventilation. In this last case, the control ventilation signal is added to the existing one computed by the temperature controller.

In a first approach, the controller was designed to operate at a periodicity of a minute. However, in the real system, this lead to an unsatisfactory oscillatory result. Therefore, this time it was reduced to a ten second interval resulting on a considerable improvement.

A general view of the greenhouse controller is presented in Figure 4.13 and a more detailed view of the humidity controller in Figure 4.14.

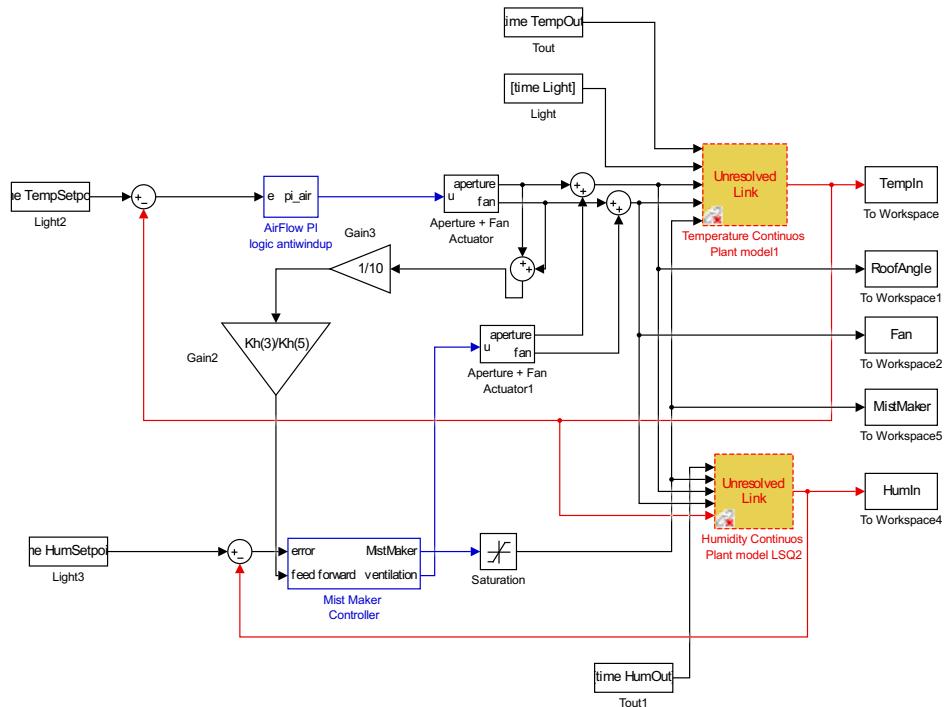


Figure 4.13: Greenhouse temperature and humidity controller. The temperature is controlled with a simple PI. The humidity controller has a feedforward term because it is more sensitive to the ventilation actuators controlled by the temperature PI.

Moreover, the simulation while designing the controller is present in Figure 4.15 and the performance

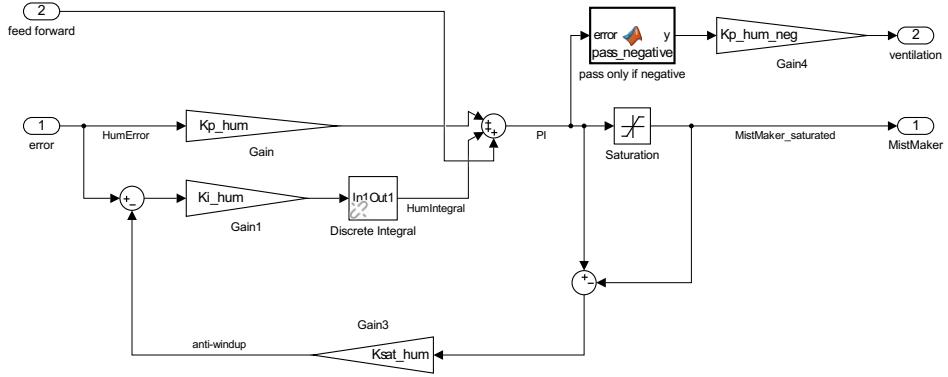


Figure 4.14: Humidity controller detailed view. It is split in positive and negative control that feed two different actuators

on the real system in Figure 4.16 and 4.17.

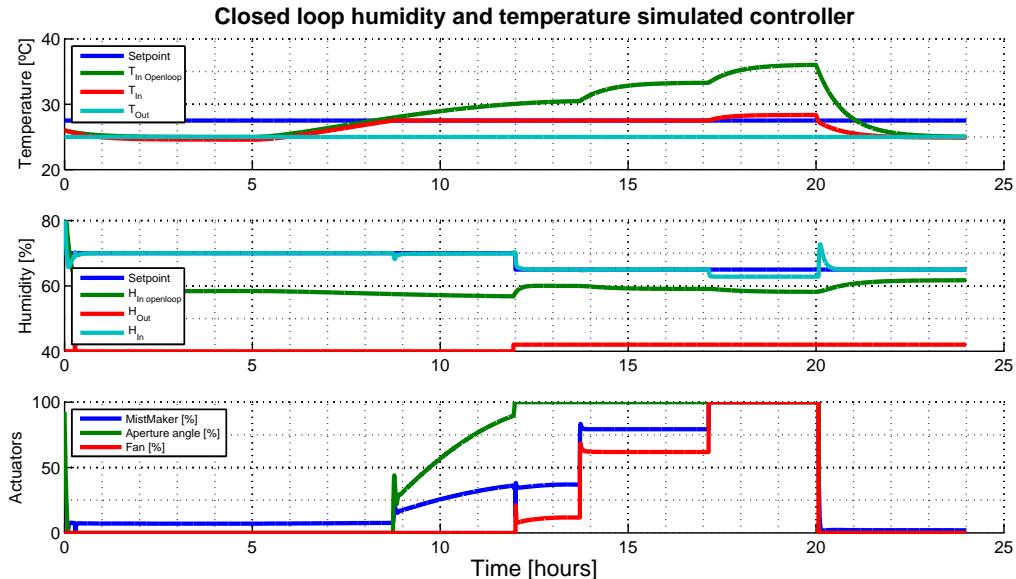


Figure 4.15: Simulated response to the greenhouse humidity and temperature controller represented in Figure 4.13. The controller reveals to successfully approach the control variables to its respecting setpoints. This is only not true when the actuators are saturated. For example around 17-20h.

The overall performance of the controller implemented in the real system revealed to be a little different than in simulation. Despite some oscillation, the greenhouse inside temperature was successfully controlled within a maximum error of  $0.2^{\circ}\text{C}$  within the actuators limits ( see Figure 4.16 from around 10-20h). Notice that the temperature setpoint changes accordingly to the outside temperature. Actually, the setpoint can not be smaller than the outside temperature plus  $0.5^{\circ}$ . This allows to avoid the temperature controller to saturate in conditions where the control is not possible.

Further more, the humidity control revealed to be more oscillatory, with maximum errors of around 10% when is hotter in the day (around 14h-17) and of 5% in the remaining period.

These oscillations can be justified manly due to actuators limitations. The fan revealed to have a dead band from 0% to 20% (see for example Figure 4.17 at time 10-11h the humidity suddenly drops

several times when the fan reaches around 20%), being this the most likely cause. Moreover, the mist maker is an on-off controller, and it was approximated linearly to an dutycycle actuator, although, it can only switch within steps of 10%. Finally, as stated before, there are periods in this system when it is just not possible to actuate due to external conditions. For example, if one desires a temperature setpoint lower than the outside environment, there is no possible actuation.

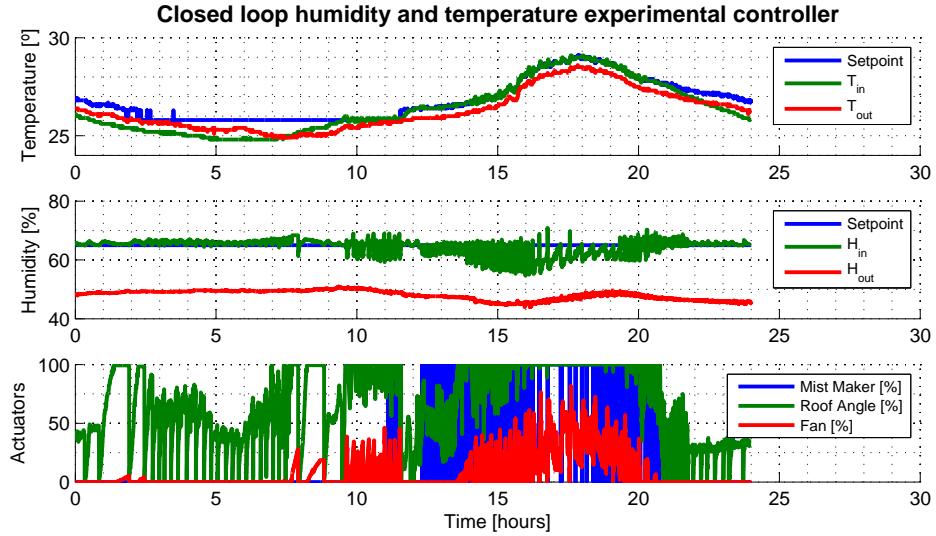


Figure 4.16: Performance of the Humidity and Temperature controller (during a normal day). The temperature follows well the setpoint within its actuators limits. The setpoint is automatically changed according to the outside temperature to avoid the controller to operate in saturation conditions. The humidity controller is more oscillatory mainly due to the fan non linearities

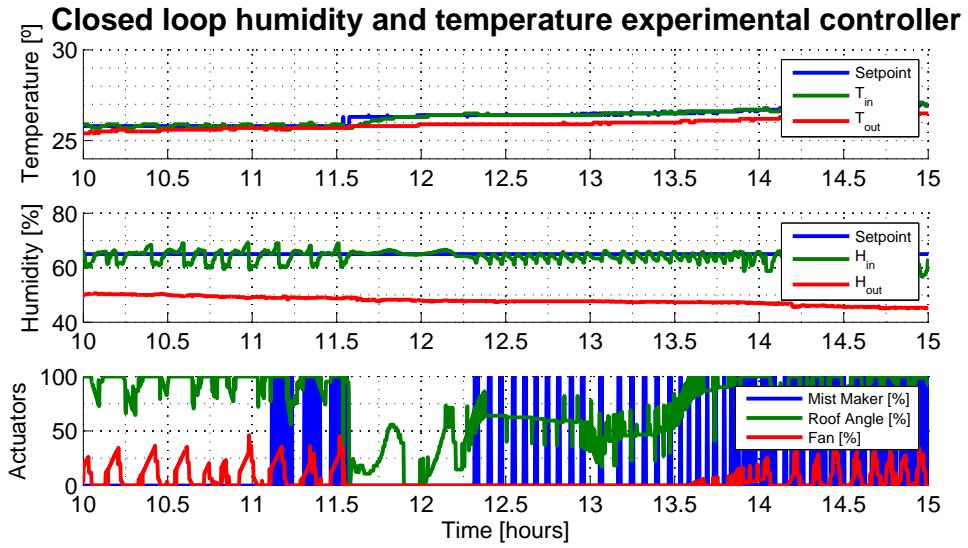


Figure 4.17: Detailed view of Figure 4.16 from 10 to 15 hours.

## 4.5 Light supplement

In order to achieve a higher level of light received by plants, an artificial LED light is employed.

It is turned on when the value of Daily Light Integral (DLI) is under the desired setpoint and the time of the day exceeds a certain hour (15 h), and finally, if the actual light level is also inferior to a certain threshold (80 PAR).

In short, this condition makes sure that the desired daily amount of light is achieved, using as much sunlight as possible.

## 4.6 Other environmental variables

There are other important variables influencing crop growth. However some were not controlled or it was done manually.

Because there was neither a pH or Ec meter available these were not monitored or controlled automatically. However, the nutritive solution concentration was manually controlled by renewing it every 7 days. It is thus assumed that the nutritive concentration maintains itself constant during this periods.

Finally, due to time and budget limitations no efforts were put on carbon dioxide control.



# Chapter 5

## Optimizing crop growth

In order to compute the solution that leads to an optimal crop growth, where the biomass is maximized and the acting costs minimized, optimal control theory was used. For that, it is important to have a model of the controlled variable. Therefore, the first step was to fit experimental data to a plant growth model. The next step was to simulate an optimal controller that, by minimizing a cost function, provides the environmental setpoints that should be followed in order to obtain an optimal solution. Finally, the optimal controller solution was experimentally tested to confirm simulation results validating the controller as well as the plant growth model fitting.

It was then possible to make a short economic analysis as well as a potential industrial application feasibility.

### 5.1 Logistic plant model

From the different possible mathematical expressions to describe the growth of a plant, the logistic non-linear function was chosen based on the experimental data and due to its simplicity. Indeed, this is a simple first order differential non-linear equation with a sigmoid shape expressed as

$$\frac{dw}{dt} = K(\mathcal{M} - w)w \quad (5.1a)$$

$$\mathcal{M}(N, L) = \alpha N + \beta L \quad (5.1b)$$

$$K(L, T, H) = \gamma L + \delta T + \eta H, \quad (5.1c)$$

where  $w$  denotes the mass of the plant,  $\mathcal{M}$  and  $K$  are constants influencing the growth speed and the maximum mass achieved, respectively. Further,  $K$  and  $\mathcal{M}$  are given by functions depending on the growing conditions that the plant will be subject to. A linear suggestion for these functions are (5.1b) and (5.1c). Where  $N$  represents the nutritive average solution concentration over the plant growth cycle, often expressed in ( $mS/cm$ ). Although, because there was no conductivity sensor available,  $N$  will be measured according to the nutrient dilution factor ( $ml/l$ ).  $L$  represents the average daily light received by the plant (DLI-Daily Light Integral). Finally,  $T$  and  $H$  are the average inside temperature ( $^{\circ}C$ ) and relative

humidity (%) in the greenhouse, respectively. It is important to stress that  $N$ ,  $L$ ,  $T$  and  $H$  represent the average value during the entire growing process. The model assumes that these values are kept to small deviation around the average value. Moreover, these should be bounded to  $0 < N < N_{max}$ ,  $0 < L < L_{max}$ ,  $T_{min} < T < T_{max}$  and  $H_{min} < H < H_{max}$  for a more accurate model. These bounds are deducted both from experimental data and from bibliography.

## Fitting the model

To discover the model described by equations (5.1) several experiments took place at different growth conditions. The richer the experiences the better the fitting. The first experience, which correspond to the first four graphics on appendix A.1, were done at similiar growth conditions. The following four graphics represent the second experience where each plant was subject to different nutritive solution concentrations. In the final experience, the plants were also subject to different nutritive concentration, whereas the amount of light was inferior.

The measure of plant biomass is considered to be proportional to the number of pixels computed by the image processing software which, in turn, is assumed to measure the plant LAI - Leaf Area Index. This software is called periodically at every hour whenever there is enough light for a good picture.

Only with the biomass measurements one can already fit the non-linear function of equation (5.1a) to experimental data. As explained in section 2.7, it is possible to compute the non-linear model parameters  $K$  and  $\mathcal{M}$  by minimizing the cost function  $J$  that represents the error of the fitting to the experimental data, this is

$$\min_w J = \sum_{t=1}^{N_s} [y(t) - w(t, \hat{K}, \hat{\mathcal{M}})]^2 \quad (5.2)$$

where  $y(t) \in \mathbb{R}^{N_s}$  is the experimental data at time  $t$  with  $N_s$  samples, and  $w(t, \hat{\mathcal{M}}, \hat{K}) \in \mathbb{R}^{N_s}$  is the plant mass at time  $t$  defined by the fitted model with the estimated parameters  $\hat{\mathcal{M}}$  and  $\hat{K}$ .

As an unconstrained optimization problem, it is possible to use simple software like the `fminunc` function of Matlab to find the parameters that minimize this function. Figure 5.1 presents an example on estimating these parameters.

Knowing the estimators  $\hat{K}$  and  $\hat{\mathcal{M}}$  that best fit each experiment it is possible to apply the LSQ method over a group of experiments ( $N_{exp}$ ) to find out the linear parameters of equations (5.1b) and (5.1c). Which in turn will allow to relate the crop biomass with the growing conditions it is subject to.

The first LSQ allows to compute the constants  $\alpha$  and  $\beta$  of equation (5.1b), where  $Y = \mathcal{M}_{exp} \in \mathbb{R}^{N_{exp}}$ ,  $\Phi = [N \ L] \in \mathbb{R}^{N_{exp} \times 2}$  and  $\theta_{\mathcal{M}} = [\alpha \ \beta]^T \in \mathbb{R}^2$ .

The data presented in appendix A.1 yielded

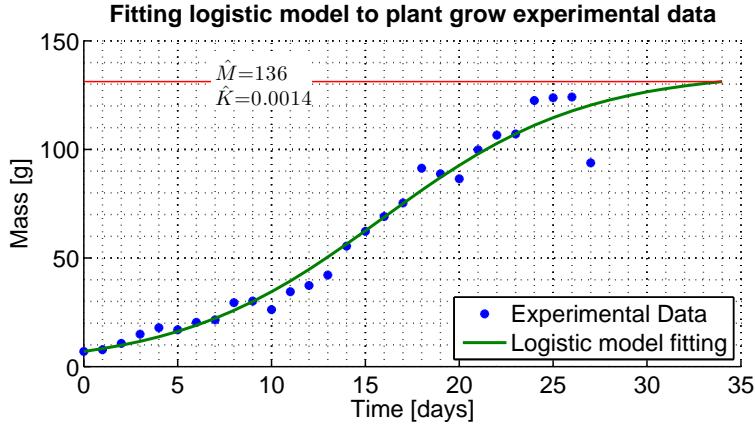


Figure 5.1: Example of fitting crop growth experimental data to logistic function using `fminunc` function of Matlab.

$$\begin{bmatrix} \hat{\alpha} \\ \hat{\beta} \\ N_{max} \\ L_{max} \end{bmatrix} = \begin{bmatrix} 11.065 \\ 13.082 \\ 2 \\ 14 \end{bmatrix} \quad (5.3)$$

Figure 5.2 shows the LSQ experimental data fitting to equation (5.1b).

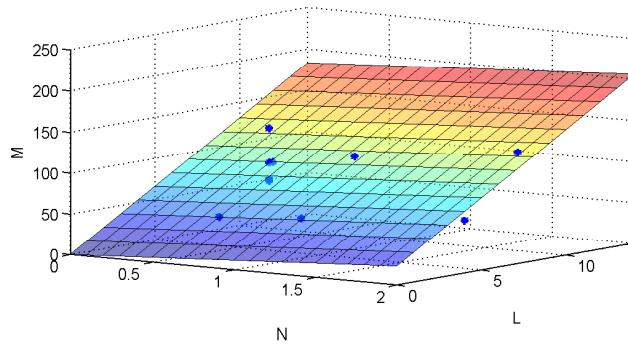


Figure 5.2: Fitting to linear equation (5.1b) using experimental dataset of appendix A.1. Dots represent experimental data and the plane represents the corresponding fitting.

$L_{max}$  parameter was obtained from bibliography, meaning that even if the lettuce plants receive more than this value of DLI, it will not contribute for further growing. Moreover,  $N_{max}$  was suggested from the manufacturer to be the maximum nutrient concentration level. Indeed it was confirmed experimentally that its increase would not contribute for further growing.

The second LSQ allows to compute the parameters  $\gamma$ ,  $\delta$  and  $\eta$  of the second linear equation (5.1c), where  $Y = K_{exp} \in \mathbb{R}^{N_{exp}}$ ,  $\Phi = [L \ T \ H] \in \mathbb{R}^{N_{exp} \times 3}$  and  $\theta_K = [\gamma \ \delta \ \eta]^T \in \mathbb{R}^3$ .

The same data as before of appendix A.1 yielded

$$\begin{bmatrix} \hat{\gamma} \\ \hat{\delta} \\ \hat{\eta} \\ T_{min} \\ T_{max} \\ H_{min} \\ H_{max} \end{bmatrix} = \begin{bmatrix} -7.44e-04 \\ 0.0014 \\ -4.78e-04 \\ 15 \\ 30 \\ 60 \\ 80 \end{bmatrix}. \quad (5.4)$$

Once again, the operating range where the model should be considered for the temperature and humidity, are the optimal values for lettuce growing (see research in section 2.4.2).

With the estimated parameters in this section it is now possible to predict crop growth biomass according to environmental conditions. In appendix A.1 the continuous curves represent the model fit to the growth of several lettuce, subject to different grow conditions.

This model will allow you to study how it is possible to change the growth conditions with the goal of maximizing the final biomass achieved while simultaneously minimizing the effort doing it. This problem finally addresses the optimal control theory presented next.

## 5.2 Optimal control of plant growth

Knowing a model for the plant growth, it is now possible to design an optimal control strategy to improve the efficiency of growing plants in the controlled greenhouse.

This section only addresses the design in simulation environment, whereas section 5.3 shows the results of the implemented controller.

As a first approach, the concentration of the nutritive solution  $N$  is the only system input considered. Afterwards, the amount of light that plants receive daily is also taken in consideration. Controlling both these inputs allows to fully control the final biomass of the plant, defined by parameter  $\mathcal{M}$  of equation (5.1). This mean that, the only concern is the maximum value of the plant mass at the end of the experience, giving little importance to its rate of growth  $K$ , moreover considering it as constant.

### 5.2.1 Nutrients optimal control

For a first approach to optimal control, a simpler version of the crop growth model described by equation (5.1) will be considered as

$$\frac{dw}{dt} = K(\alpha u + \beta L - w)w. \quad (5.5)$$

Not only the rate of growth will be considered as constant ( $K = 0.0015$ ), but also as the amount of light plants receive daily ( $L = 7$ ). These are values for a common lettuce growth with an average

temperature and humidity within the favorable bounds described in equation (5.4), and without providing artificial light. The only input, described by  $u$ , is the concentration of the nutritive solution.

At this stage a simple cost function can be defined as

$$J(u) = w(t_f) + \int_{t_0}^{t_f} -u(t) dt. \quad (5.6)$$

According to section 2.6 formulation, the goal is to maximize the final biomass of each plant and its derivative is

$$\Psi[w(t_f)] = w(t_f) \quad \Psi_w[w(t_f)] = 1, \quad (5.7)$$

the sum of the penalties of actuation and its derivative are

$$\mathcal{L} = -u \quad \mathcal{L}_w = 0, \quad (5.8)$$

the function that models the plant growth its derivative and its initial condition are

$$\dot{w} = f(w, u) = K(\alpha u + \beta L - w) \quad f_w = -2Kw + K(\alpha u + \beta L) \quad w(0) = w_0. \quad (5.9)$$

From Pontryagin's principle, the co-state derivative and its final condition are

$$\dot{\lambda} = -\lambda f_w + \mathcal{L}_w = 2\lambda Kx - \lambda K(\alpha u + \beta L) \quad \lambda(t_f) = \Psi_w[w(t_f)] = 1, \quad (5.10)$$

and the Hamiltonian is

$$\mathcal{H} = \lambda^T f(w, u) + \mathcal{L}(w, u) = (\lambda K\alpha w - 1)u + \text{Independent terms of } u. \quad (5.11)$$

Studying the Hamiltonian it is easy to see that it has a linear dependency in respect to the input  $u$ , and thus its maximum can be computed splitting the problem in two conditions

$$\mathcal{H}_{max} = \begin{cases} u_{max} & \text{if } \lambda K\alpha w - 1 > 0 \\ u_{min} & \text{if } \lambda K\alpha w - 1 < 0 \end{cases} \quad (5.12)$$

where  $u_{min}$  and  $u_{max}$  are the limits of acting of the interval  $\mathcal{U}$ .

At this stage, using the algorithm explained in section 2.6 and considering a time increment of one day, it is possible to find out what is the optimal control  $u^*(t)$  at each time  $t$ , that lead to a maximum of the cost function. Figure 5.3 shows the simulation response for this problem.

When the solution for the maximum of the Hamiltonian is a condition as the one obtained, one states that we are dealing with bang-bang control, because there are only two possible values for the optimal input.

As seen in Figure 5.3, the numerical approach can converge to sub-optimal solutions. However,

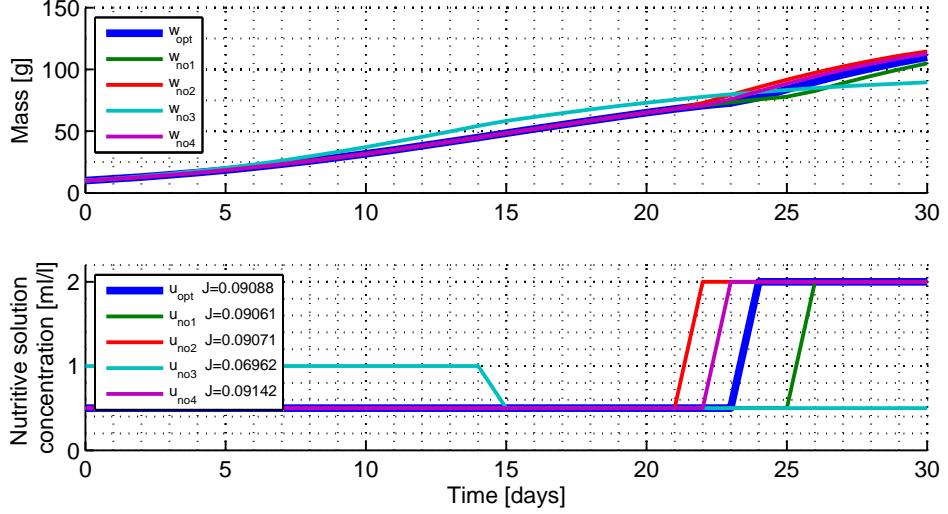


Figure 5.3: Crop growth optimal control (subscript  $opt$ ) with the nutritive solution concentration as the only input variable. Other non-optimal solutions (subscript  $no$ ) are presented, as well as the corresponding cost function ( $J$ ). The last non-optimal solution actually shows a higher cost function, suggesting thus that the numerical algorithm converges to a sub-optimal solution.

the bang-bang control solution allows to do an interesting analysis. By evaluating the value of the cost function ( $J$ ) where the bang-bang control is switched at different times ( $t_s$ ) it is possible to directly compute the optimal control without using Pontryagin's principle (assuming we already know that bang-bang control leads to optimal solution). Finding the maximum of Figure 5.4 curve, allows to compute a more reliable optimal solution. Indeed, the optimal switch time  $t_s^*$  of Figure 5.4 corresponds to the best solution ( $w_{no4}$ ) of Figure 5.3.

Furthermore, this same Figure 5.3 allows to infer about the optimal controller robustness. Because the cost does not change drastically with the time change around the optimal value it is considered a robust solution. In other words, if one affects the time control switch around its optimal value it is guaranteed that the final profit will not change drastically.

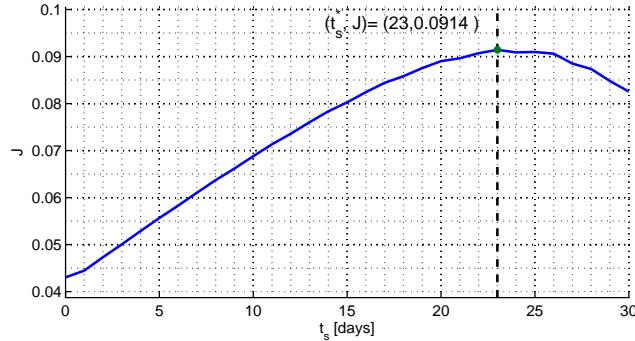


Figure 5.4: Cost function evaluated at different time for the bang-bang control of the nutritive solution concentration. The optimal value of the switch is given by  $t_s^*$ . Moreover, the wide concavity of the curve suggests a robust controller.

### 5.2.2 Nutrients and light optimal control

In order to be able to fully control the crop biomass at harvest time, as described in the model in equation (5.1a) and (5.1b), it is necessary not only to control the nutritive solution concentration at root level, as well as the amount of light the plant receives on average per day. Having this in mind another model will be considered, now as

$$\frac{dw}{dt} = K(\alpha u_1 + \beta u_2 - w)w, \quad (5.13)$$

where the rate of growth  $K$  is still considered as constant, but now there are two control inputs.  $u_1$  represents the control of the nutritive solution , whereas,  $u_2$  is the average amount of light the plant receives daily.

Therefore, the cost function can now be described as

$$J(u_1, u_2) = w(t_f) + \int_{t_0}^{t_f} -u_1(t) - u_2(t)dt, \quad (5.14)$$

And from formulation of section 2.6, one gets

$$\Psi[w(t_f)] = w(t_f) \quad \Psi_w[w(t_f)] = 1, \quad (5.15)$$

$$\mathcal{L} = -u_1 - u_2 \quad \mathcal{L}_w = 0, \quad (5.16)$$

$$\dot{w} = f(w, u_1, u_2) = K(\alpha u_1 + \beta u_2 - w)w \quad f_w = -2Kw + K(\alpha u_1 + \beta u_2) \quad w(0) = w_0. \quad (5.17)$$

Again, from Pontryagin's principle, the co-state derivative and its final condition are

$$\dot{\lambda} = -\lambda f_w + \mathcal{L}_w = 2\lambda Kx - \lambda K(\alpha u_1 + \beta u_2) \quad \lambda(t_f) = \Psi_w[w(t_f)] = 1, \quad (5.18)$$

and the Hamiltonian is

$$\mathcal{H} = (\lambda K\alpha w - 1)u_1 + (\lambda K\beta w - 1)u_2 + \text{Independent terms of } u. \quad (5.19)$$

Once again, the Hamiltonian, represented in Figure 5.5, depends linearly on the inputs. Although, this time it is represented as a 3D surface instead of a curve. It would be also possible to come up with a function with four conditions that return the optimal combinations of inputs as done in equation (5.12). However, a simpler and more reliable solution is to evaluate the Hamiltonian for the four possible combinations and return the one that leads to a maximum. In other words, to compute the optimal control inputs one wants to solve the problem, assuming that  $\lambda$  and  $w$  are known,

$$\begin{aligned}
& \underset{u_1, u_2}{\text{maximize}} \quad \mathcal{H}(\lambda, w, u_1, u_2) \\
& \text{s.t.} \quad (u_1, u_2) \in \{(u_{1min}, u_{2min}), (u_{1min}, u_{2max}), (u_{1max}, u_{2min}), (u_{1max}, u_{2max})\}
\end{aligned} \tag{5.20}$$

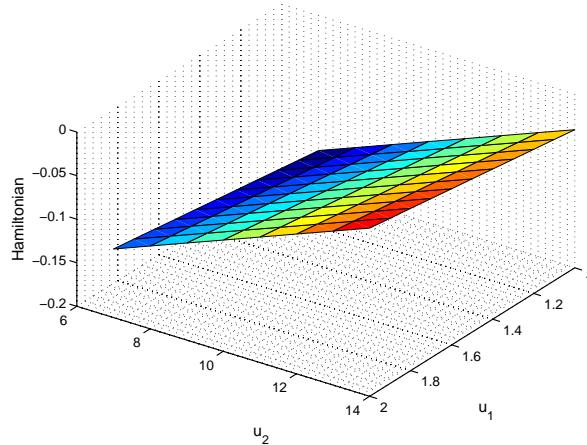


Figure 5.5: Hamiltonian for the two input variables optimal control. As it is a linear plane, the combination of optimal inputs  $(u_1^*, u_2^*)$  that maximizes the Hamiltonian is in the boundary of the input limits given by  $\mathcal{U}$ .

The simulation response that shows the optimal input of nutrients and light is presented in Figure 5.6. Again, this is a bang-bang optimal control solution similar to controlling only one input.

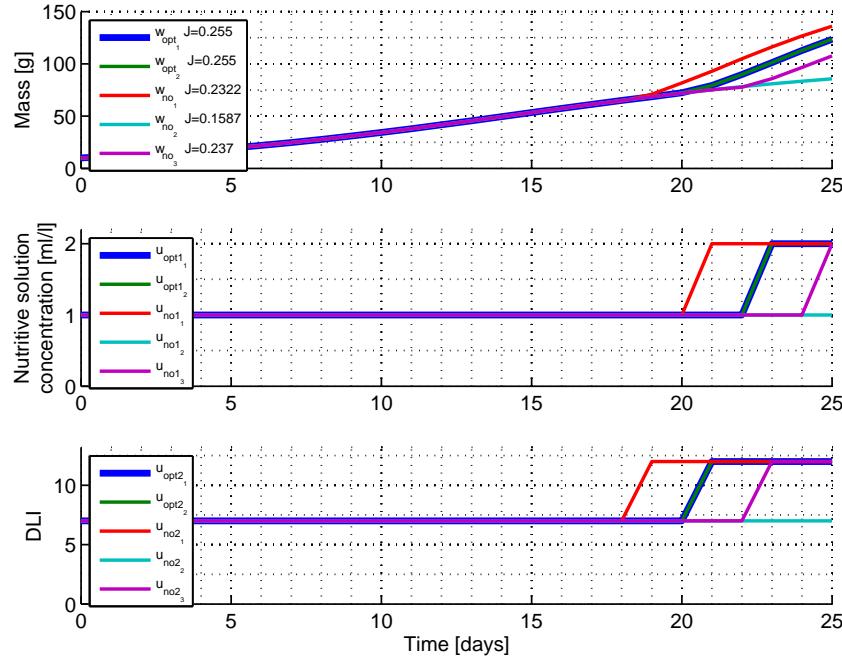


Figure 5.6: Optimal control (subscript  $opt$ ) with the nutritive solution concentration ( $u_1$ ), and the amount of daily light received ( $u_2$ ) as the input variables. Other non-optimal solutions (subscript  $no$ ) are present as well as the corresponding cost function ( $J$ ).

Once again, because the optimal control is known to be of the bang-bang type, it is possible to plot a 3d surface, represented in Figure 5.7, with the cost of switching the control at different time combinations within the experience. This allows to have a better precision on when this changes should take place. Based on these results, another optimal solution ( $w_{opt2}$ ) is presented in Figure 5.6. This time, the optimal control solution gave the same result as searching for the maximum value of Figure 5.7.

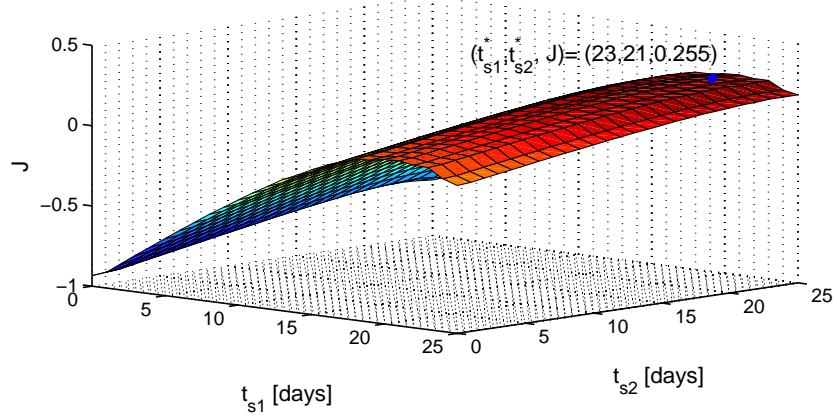


Figure 5.7: Cost function evaluated when switching the control variables at different time combinations ( $t_{s1}, t_{s2}$ ). The optimal combination is given at  $(t_{s1}^*, t_{s2}^*)$ . Again, the wide concavity of the surface suggest a robust controller.

In conclusion the numeric optimal control solution can give a sub-optimal result for the problem. However, once the problem is robust it does not pose a dramatic problem. Nevertheless, because one is facing a bang-bang control, it is possible to compute a mesh for different solutions and search for the optimal solution with higher precision.

### 5.2.3 Optimal harvest day

Growing plants for longer or shorter time periods will achieve distinct final biomass, which will reflect on different profit values.

Therefore, the optimal problem should ideally take the experiment duration into consideration. However, the selected method uses Pontryagin's principle which requires a fixed value to the harvest time ( $t_f$ ). Although there are other control approaches that allow to relax this constrain, there is a way of taking it in consideration with the method developed so far.

By solving the same optimal problem for different harvest days ( $t_f$ ) it is possible to come up with the one that achieves a higher profit value. Figure 5.8 shows what is the day when the plants should be harvested, which leads to a higher profit value. The best solution was already considered in Figure 5.6 simulation.

### 5.2.4 Adjustment of the cost function

Several modifications can be done to the cost function providing other useful solutions or economic insights. For example, it is possible to enhance the current cost function so that it actually returns a

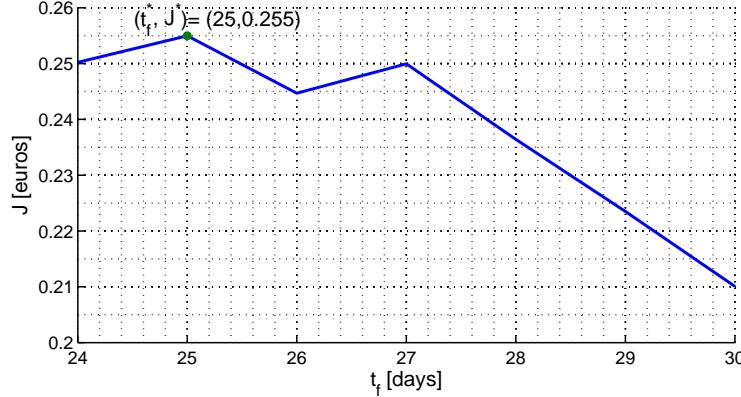


Figure 5.8: Cost function (*euros*) of the optimal solution for different experiments time durations. Computing the harvest time that leads to the maximum profit, allows to relax the final time fixed constrain of Pontryagin's principle.

measurable profit value in euros. Changing the cost function using quadratic terms gives a smoother control solution, which can also be an alluring result.

### Cost function units

It is possible to complete the current cost function of equation (5.14) so that it returns a more useful value, i.e. translating it into a measurable profit.

By adding a term  $\epsilon_w$  multiplying the final biomass with units  $\frac{\text{euros}}{\text{g}}$ , a term  $\epsilon_1$  multiplying the nutritive solution concentration with units  $\frac{\text{euros}}{\text{ml/L}}$  and finally a term  $\epsilon_2$  multiplying the amount of light received by the plant with units  $\frac{\text{euros}}{\text{DLI}}$  the cost function of equation (5.14) will result in a profit value measured in *euros*. Moreover, one can also consider to grow more than one plant, which will hopefully increase the yield profit. For this, another therm  $n_w$ , representing the number of plants, will be multiplying the final biomass. However, having more plants will consequently cause some of the inputs to have a greater cost. This is the case of the nutritive solution maintenance. Therefore it will also be multiplied by  $n_w$ . Finally, one can also add a constant term regarding fixed costs operating the greenhouse daily to the integral term of the cost function, such as the power consumption of the hardware. This constant is named  $\epsilon_{gh}$  having  $\frac{\text{euros}}{\text{day}}$  as units.

Therefore, the cost function in *euros* units comes as

$$J(u_1, u_2) = n_w \epsilon_w w(t_f) + \int_{t_0}^{t_f} -n_w \epsilon_1 u_1(t) - \epsilon_2 (u_2(t) - u_{2min}) - \epsilon_{gh} dt \quad [\text{euros}]. \quad (5.21)$$

Note that the artificial light input was reduced with a constant factor of  $u_{2min}$ . It corresponds to the average daily natural light value received by the sun, that, as it is considered free, should not be considered as a cost. In other words, the optimal problem is built so that the artificial light should always illuminate at a minimum level of  $u_{2min}$ . However, removing the respective cost from the cost function, it means that illuminating at that value is free, which in turn represents the amount of light received by the sun.

Section's 5.2.2 simulations already took into account these inclusions, considering the values as

$$\begin{bmatrix} n_w \\ \epsilon_w \\ \epsilon_1 \\ \epsilon_2 \\ \epsilon_{gh} \\ u_{2min} \end{bmatrix} = \begin{bmatrix} 8 [\text{plants}] \\ 1.5/1000 [\text{euros/g}] \\ 0.0021 [\text{euros/N}] \\ 0.0144 [\text{euros/DLI}] \\ 0.018 [\text{euros/day}] \\ 7 [\text{DLI}] \end{bmatrix}. \quad (5.22)$$

A more detailed explanation on how these parameters were computed is present in appendix B.1

The number of plants  $n_w$  was considered as 8 because it is actually possible to grow this amount of plants in the physical space available in the greenhouse. However, due to image processing limitations, was only possible to grow 4.

If one wants to know the profit per plant just has to compute  $\frac{J^*}{n_w}$ .

### Quadratic cost function

Considering quadratic costs on the control variables of the cost function as

$$J(u_1, u_2) = \epsilon_w w(t_f) + \int_{t_0}^{t_f} -\epsilon_1 u_1^2(t) - \epsilon_2 u_2^2(t) dt, \quad (5.23)$$

allows to have a continuous optimal control solution for the inputs in comparison to the bang-bang control. This can reveal to be an advantage when abrupt stimulus harm the system. One could state that the plants growth would benefit more if its environmental conditions wont change brusquely, although the bang-bang control, if bounded, does not seem to be harmful. The bang-bang control has the advantage to be easier to implement in the real system software. Moreover, because there is no automatic way of controlling the nutrient solution, its way easier to manual control it using the bang-bang solution. Therefore, the quadratic cost approach was not further developed. Figure 5.9 reveals the optimal solution for a quadratic cost function. Note that, the quadratic cost function changes the Hamiltonian to

$$\mathcal{H} = (\lambda K \alpha w) u_1 - \frac{\epsilon_1}{2} u_1^2 + (\lambda K \beta w) u_2 - \frac{\epsilon_2}{2} u_2^2 + \text{Independent terms of } u. \quad (5.24)$$

This implies that the Hamiltonian is now a curved plane and to maximize it one has to derive it in respect to each control variable and making the derivative equal to zero, this is

$$\frac{\partial \mathcal{H}}{\partial u} = 0, \quad (5.25)$$

which implies that

$$\begin{bmatrix} u_1^* \\ u_2^* \end{bmatrix} = \begin{bmatrix} \frac{w\lambda K\alpha}{\epsilon_1} \\ \frac{w\lambda K\beta}{\epsilon_2} \end{bmatrix}. \quad (5.26)$$

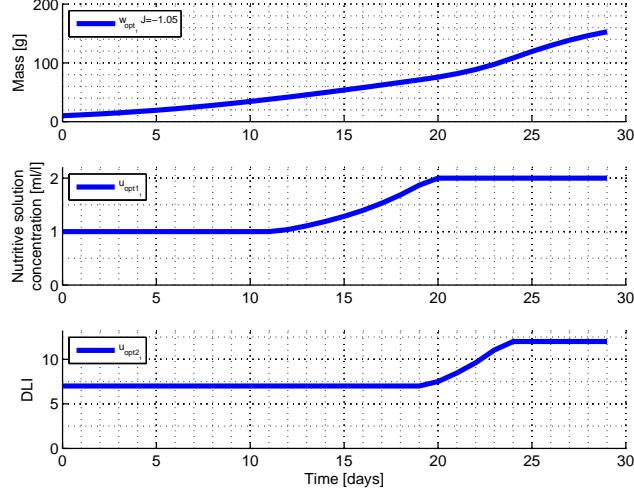


Figure 5.9: Optimal solution for quadratic cost function. This result allows to have continuous control inputs in contrast to the bang-bang solution. Note that, despite the cost function value is not positive, it does not mean that there is no actual profit once this value is not in euros.

### Optimal control of more variables

It would be very interesting to obtain optimal solutions taking also in account the input variables that shape the crop growth rate. This is defined by  $K$  in equations (5.1a) and (5.1c). However, this would lead to non-convex Hamiltonian functions due to the occurrence of crossed multiplication terms of input variables. Non-convex problems are hard to solve, therefore this problem was not further developed.

## 5.3 Results and validation

A final experience was performed to validate the optimal solution proposed in section 5.2.2 with the optimal harvest day computed in section 5.2.3. In this experience all four plants' growth conditions were the same, following the simulation of Figure 5.6. The nutritive solution was kept at 1 ml/l up to day 22 and raised to 2 ml/l until the harvest time on day 25. On the other hand, the amount of light was kept to 7 DLI up to day 20 and raised to 12 DLI until the end of the experience. Moreover, the humidity and temperature were kept at ideal average values of 65% and 27°, respectively.

Figure 5.10 overlaps the simulated optimal result of Figure 5.6 with the four plants experimental growth. The Figure allows to conclude that the four plants grew in similar conditions having thus similar growth rates. Furthermore, the most important experimental evidence is the similar growth of the four plants compared to the optimal solution simulated curve. The error between the curves is

initially very small, increasing with time. It is important to remember that the image processing software also gets more inaccurate as the plants size increase, being this a possible justification for an increase of the error. The root mean square error between each experimental curve and the simulation is [14.5136 8.8183 4.5022 8.5963]  $g$  respecting Figure 5.10 order, where the average of these values is 9.107 $g$ . These are errors of around 10% compared to its harvest weight, which is considered to be a quite satisfactory result.

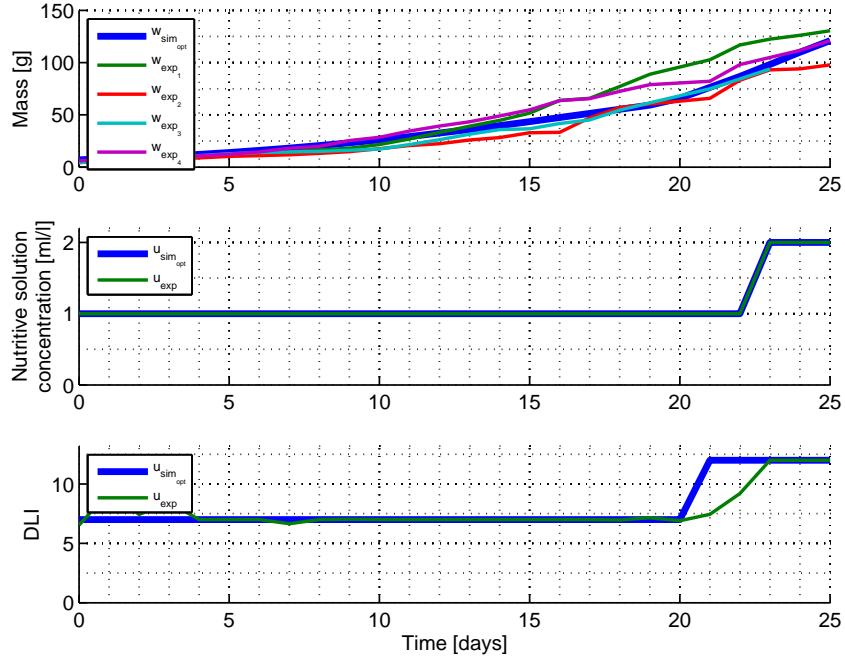


Figure 5.10: Final experiment to validate the fitted plant growth model as well as the optimal controller solution proposed. The root mean square error between the experimental growth curves compared to simulation is around 10% or 9 $g$ .

Note that the daily light integral (DLI) control of the experience did not follow precisely the desired values, as it can be seen in Figure 5.10. In fact, this controller has some limitations. Sometimes the sunlight captured is bigger than the desired setpoint (although this is free light). Other times the desired setpoint is so big, compared to the captured sunlight, that for the LED to deliver that amount of light daily, it would have to be turned on for so long that the plant would not have night periods (which was set to a minimum of 6 hours). Therefore, not achieving the desired value.

Overall, this validation experience brought the confirmation of a successful fitting of the plant growth model, as well as, the possibility to apply the proposed optimal control algorithm in real plants. For further validation of the developed work, it would be important to also compare the growth curve of non optimal solutions. For example, experimenting the least profit curve of Figure 5.6 (curve two in bright blue) would allow even better conclusions. However, this experiments are very time consuming, therefore it was not possible to perform them.

### 5.3.1 Control Sample

To have a comparison with the traditional growing technique, a parallel experience took place. This consisted of simultaneously grow another lettuce in enriched soil. The plant was daily watered with common sense, whereas the remaining conditions were similar to the external greenhouse conditions during the validation optimal control experiment. Its growth curve was not monitored, once this plant was not inside the greenhouse. However, the final biomass result could not be more conclusive. This plant only achieved a mass of around 20g at harvest time, compared to 120g of the optimal control. It is then possible to confirm the advantageous of growing plants on a controlled environment in hydroponics.

## 5.4 Economic perspective

The initial investment to build the prototype greenhouse structure was around 80 euros. This comprises mainly the cost on wood and on the polycarbonate cover. The DWC hydroponic system (cheaper than the NFT one) costed less than 30 euros. As the individual plant container was recycled, adding new plants has a marginal fixed cost of around only one euro. The investment in sensors was of 40 euros and in actuators around 112 euros. Finally, the total cost on other electronics was around 90 euros. Therefore, the total cost was around 350 euros. A more detailed list on these expenses is presented in the appendix B.2

The built greenhouse has a production area of around  $0.3 \text{ m}^2$ . Considering that each lettuce requires a  $0.04 \text{ m}^2$  area to grow, it is possible to grow eight plants in this system. For this amount of plants the optimal solution, presented in the previous section, yield a profit of *0.255 euros* considering that the mass of lettuce would be sold at *1.5 euros/kg*. Therefore, the profit per plant is slightly bigger than *0.03 euros* per plant. If each lettuce has 100g then the profit is around  $\frac{0.03}{0.15} \times 100 = 20\%$ .

Considering that by increasing the greenhouse production area up to 1 square meter would not change the production cost per plant on equation (5.21) - this means for example that no extra light is needed and that the fixed costs of operating the greenhouse are the same. Then it would be possible to grow up to 25 plants. Running the optimal algorithm to this amount would yield a profit of *0.12 euros* per plant. Now the profit percentage is greater and around  $\frac{0.12}{0.15} \times 100 = 80\%$ .

Further increase of growth area would lead to a proportional investment on the cost of lighting and therefore no increase in profit per plant would occur. Indeed, duplicating the cost of light supply and considering the growth of 50 plants, the optimal algorithm would also output the same profit per plant of 80%. Although an increase of area bigger than a square meter does not seem to bring a higher profit per plant, it probably has an important role considering other business expenses, including the amortization of the initial investment.

# **Chapter 6**

## **Conclusions**

Growing plants in a controlled environment with hydroponic technology was proven to be much faster compared to traditional cultivation methods.

It was possible to use simple linear first order equations and PI controllers to successfully model and control the greenhouse environment. Moreover, the logistic function allowed to model lettuce growth which was used with optimal control to define the acting that leads to the most profitable solution. It was also possible to compute the optimal harvest moment for each solution.

The greenhouse was built with simple materials and noncommercial cheap hardware proved to be profitable with a production area of  $0.3\ m^2$ . Furthermore, with small structural investment, increasing this area to  $1\ m^2$  would achieve the best possible profit per square meter, around 80%.

Experimental Physics and Industrial Control System (EPICS) allowed to develop a reliable control system, although its configuration and installation is not the easiest one, for example extra plugins or required third party software force the dispersion when configuring the system. As an old software it revealed to have some other disadvantages, for example as the archiver software was not compiled for the Pi it had to be installed in another Linux computer. Its installation and configuration as well as the learning curve for developing applications is also considerably difficult. However, this was the only open source software that responded to the desired need without the hard burden of coding everything from scratch. The possibility to easily develop a GUI using Control System Studio (CSS) revealed to be very satisfactory, only falling short if a web interface is required.

### **6.1 Achievements**

An automated soilless greenhouse was built. It was possible to remotely connect to the greenhouse throughout a graphical interface, where manual and autonomous control and monitoring features are available. The greenhouse experimental data allowed to fit models to the inside temperature and humidity. With these models, linear control was successfully implemented to regulate the temperature and humidity values to the desired setpoints. It was possible to follow plants' growth through simple image processing means.

## **Greenhouse Extra Features**

- Database automatic backup
- Automatic control system reset in case of a power failure
- Simple DIY design
- Easy extensibility to distributed system applications

## **6.2 Learned lessons**

Some experimental setbacks had to be overcome during the project development, such as the following:

- The NFT hydroponic system was not quite practical for small greenhouses like the one built. Moreover, water leaks occurred. The alternative DWC hydroponic system was implemented, furthermore with a lower investment.
- As a quite cheap hardware, the Raspberry Pi computer revealed to be unstable because unrecoverable shutdowns would occur. Fortunately, it already comes with watchdog technology that allowed to overcome this problem. Moreover, after some months running, the SD card got damaged without prior notice. It is deduced that this was caused either by solar incidence or by high temperature levels.
- The used sensors, as inexpensive hardware, proved not to be very accurate and with a short longevity. For example, the inside humidity and temperature sensor got damaged after five months.

## **6.3 Future Work**

During the development of this work several divergent lines of research or possible analysis were identified. However, given the time limitations for its development, they have not been addressed.

The temperature and humidity controllers could use optimal control for higher profitable results. These controllers could also be enhanced so that they do not get saturated so easily. A solution could be to introduce different types of actuators, for example introducing a heat pump to better control the greenhouse temperature. On the other hand, the implemented light controller was not very precise or efficient. A smart approach could be to use predictive control to better combine the amount of light supplied by the sun and the need to supply artificial light. Moreover, this controller could allow a fixed nocturne period.

Due to time limitation the  $CO_2$  monitoring and control was not developed. However, from literature this is a factor that can heavily increase greenhouse plantations' profitability. Due to the same reason, neither the pH or EC monitoring or control subject was performed, which is also an important study in greenhouse cultivation.

It would also be interesting to enhance the optimal controller, so that it takes in consideration, not only the final biomass achieved at harvest time, but also contemplate the plant growth rate.

EPICS control system software allows to develop distributed control systems. This potentiality was not explored and could be very interesting for systems with various greenhouses operating simultaneously.

Finally, it would be of a major interest to further develop the image processing software. This is a complex field that could be strongly explored. Moreover, a set of experiments could allow to calibrate the leaf area measurements converting it to weight units. Focusing on other plant species could also be a potential image processing future work also allowing to test the plant model extensibility.



# Bibliography

- [1] H. M. Resh. *Hydroponics for the HOME GROWER*. Press, CRC, 2015. ISBN 9781482239263.
- [2] I. E. K. Jako. *A Global review of Greenhouse food Production*. Controller enviroment agriculture, 1966.
- [3] Sample Leafy Greens Grown on Space Station, 2015. URL [https://www.nasa.gov/mission\\_pages/station/research/news/meals\\_ready\\_to\\_eat](https://www.nasa.gov/mission_pages/station/research/news/meals_ready_to_eat).
- [4] MIT Open Agriculture open source project, 2017. URL <http://openag.media.mit.edu/>.
- [5] G. Van Straten and E. J. Van Henten. Optimal greenhouse cultivation control: Survey and perspectives. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 3(PART 1), 2010. ISSN 14746670.
- [6] R. J. C. Van Ooteghem. Optimal control design for a solar greenhouse. *IFAC Proceedings Volumes*, 3(1), 2010. ISSN 14746670.
- [7] G. V. Straten, G. V. Willigenburg, E. V. Henten, and R. V. Ooteghem. *Optimal Control of Greenhouse*. CRC Press, 2011. ISBN 9781420059632.
- [8] R. J. C. Van Ooteghem. *Optimal control design for a solar greenhouse*. PhD thesis, Wageningen Universiteit, Wageningen, Nederland, 2007.
- [9] GroHo. Programa, 2017.
- [10] BioVivos webpage, 2017. URL <http://www.biovivos.pt/>.
- [11] CoolFarm webpage, 2017. URL <https://www.cool-farm.com/>.
- [12] B. K. Roberto. *How To Hydroponics*. Futuregarden, 2003. ISBN 0967202604. URL <http://futuregarden.com>.
- [13] characteristics\_of\_rockwool, 2017. URL [http://www.canna-uk.com/characteristics\\_of\\_rockwool](http://www.canna-uk.com/characteristics_of_rockwool).
- [14] F. Rodríguez, M. Berenguel, J. L. Guzmán, and A. Ramírez-arias. *Modeling and Control of Greenhouse Crop Growth-chapter 3*., 2014. ISBN 9783319111339. doi: 10.1007/978-3-319-11134-6.

- [15] Photosynthetically Active Radiation (PAR), 2017. URL [http://www.biospherical.com/index.php?option=com\\_content&view=article&id=49:par-introduction&catid=31&Itemid=46](http://www.biospherical.com/index.php?option=com_content&view=article&id=49:par-introduction&catid=31&Itemid=46).
- [16] Conversion - PPF to Lux, 2017. URL <http://www.apogeeinstruments.co.uk/conversion-ppf-to-lux/>.
- [17] Dana Riddle. Product Review: Lighting for Reef Aquaria: Tips on Taking Light Measurements, 2017. URL <http://www.advancedaquarist.com/2008/2/review>.
- [18] Heliospectra. Which regions of the electromagnetic spectrum do plants use to drive photosynthesis? *Heliospectra*, pages 1–7, 2012. URL [https://www.heliospectra.com/sites/default/files/general/Whatlightdoplantsneed{\\_}5.pdf](https://www.heliospectra.com/sites/default/files/general/Whatlightdoplantsneed{_}5.pdf).
- [19] R. Langhans and T. Tibbitts. *Plant Growth Chamber Handbook - Chapter1-Radiation.*., 1997. URL [http://www.controlledenvironments.org/Growth{\\_}Chamber{\\_}Handbook/Plant{\\_}Growth{\\_}Chamber{\\_}Handbook.htm](http://www.controlledenvironments.org/Growth{_}Chamber{_}Handbook/Plant{_}Growth{_}Chamber{_}Handbook.htm).
- [20] maximumyield, 2017. URL <https://www.maximumyield.com/hydroponic-illumination-the-daily-light-integral/2/1450>.
- [21] P. J. Aphalo and T. Lehto. Effects of light quality on growth and N accumulation in birch seedlings. *Tree physiology*, 17(2):125–32, 1997. ISSN 1758-4469. doi: 10.1093/treephys/17.2.125. URL <http://www.ncbi.nlm.nih.gov/pubmed/14759882>.
- [22] E. J. Van Henten and J. Bontsema. Non-destructive Crop Measurements by Image Processing for Crop Growth Control. *J.agric.Engng Res*, 1995.
- [23] D.-h. Jung, S. H. Park, X. Z. Han, and H.-j. Kim. Image Processing Methods for Measurement of Lettuce Fresh Weight. *Journal of Biosystems Engineering*, 40(1):89–93, 2015.
- [24] T.-t. Lin, T.-c. Lai, T.-y. Liu, Y.-h. Yeh, C.-c. Liu, and W.-c. Chung. An Automatic Vision-Based Plant Growth Measurement System for Leafy Vegetables. *Department of Bio-Industrial Mechatronics Engineering*, 2012.
- [25] T.-t. Lin, W.-t. Chen, Y.-h. F. Yeh, and T.-y. Liu. An Automated and Continuous Plant Weight Measurement System for Plant Factory. *Department of Bio-Industrial Mechatronics Engineering*, 7 (March):1–9, 2016. doi: 10.3389/fpls.2016.00392.
- [26] G. Lobet, X. Draye, and C. Périlleux. An online database for plant image analysis software tools. *Biomed Central*, 2013.
- [27] N. Fahlgren, M. Feldman, M. A. Gehan, M. S. Wilson, C. Shyu, D. W. Bryant, S. T. Hill, C. J. Mcentee, S. N. Warnasooriya, I. Kumar, T. Ficor, S. Turnipseed, K. B. Gilbert, T. P. Brutnell, J. C. Carrington, T. C. Mockler, and I. Baxter. A Versatile Phenotyping System and Analytics Platform Reveals Diverse Temporal Responses to Water Availability in Setaria. *Molecular Plant*, 8(10):1520–1535, 2015. ISSN 1674-2052. URL <http://dx.doi.org/10.1016/j.molp.2015.06.005>.

- [28] I. Goudriaan and H. V. Laar. *Modeling Potential crop Growth processes*. Production ecology, 1994. ISBN 9780792332206.
- [29] M. Braun. *Differential Equations and Their Applications*. New York: Springer, 2 edition, 1975.
- [30] C. E. T. Paine, T. R. Marthews, D. R. Vogt, D. Purves, M. Rees, A. Hector, and L. A. Turnbull. How to fit nonlinear plant growth models and calculate growth rates : an update for ecologists. *Methods in Ecology and Evolution*, pages 245–256, 2012. doi: 10.1111/j.2041-210X.2011.00155.x.
- [31] E. Leal-Enríquez and M. Bonilla-Estrada. Modelling the greenhouse lettuce crop by means of the daily interaction of two independent models. *IEE Conference on Decision and Control*, pages 4667–4672, 2010.
- [32] A. Juárez-Maldonado, K. De-Alba-Romenus, M. I. M. Ramírez-Sosa, A. Benavides-Mendoza, and V. Robledo-Torres. An experimental validation of NICOLET B3 mathematical model for lettuce growth in the southeast region of Coahuila Mexico by dynamic simulation. *Program and Abstract Book - 2010 7th International Conference on Electrical Engineering, Computing Science and Automatic Control, CCE 2010*, pages 128–133, 2010. doi: 10.1109/ICEEE.2010.5608663.
- [33] A. Juarez Maldonado, K. De Alba Romenus, A. Morales, and M. I. Ramirez-Sosa Moran. Dynamic behavior analysis of the {NICOLET} B3 model. *World Automation Congress ({WAC})*, 2012, pages 1–6, 2012. ISSN 21544824. URL <http://encompass.library.cornell.edu/cgi-bin/checkIP.cgi?access=gateway{ }standard{ }26url=http://ieeexplore.ieee.org/servlet/opac?punumber=6311121>.
- [34] J. Mathieu, R. Linker, L. Levine, L. Albright, A. J. Both, R. Spanswick, R. Wheeler, E. Wheeler, D. DeVilliers, and R. Langhans. Evaluation of the Nicolet Model for Simulation of Short-term Hydroponic Lettuce Growth and Nitrate Uptake. *Biosystems Engineering*, 95(3):323–337, 2006. ISSN 15375110. doi: 10.1016/j.biosystemseng.2006.07.006.
- [35] G. Van Straten. Optimal greenhouse cultivation control: Quo vadis? *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 1(PART 1):11–16, 2013. ISSN 14746670. doi: 10.3182/20130327-3-JP-3017.00006. URL <http://dx.doi.org/10.3182/20130327-3-JP-3017.00006>.
- [36] R. van Ooteghem, L. J.D. Stigter, V. Willigenburg, and G. van Straten. Optimal Control of a solar Greenhouse. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2002.
- [37] E. V. Henten, J. Bontsema, and G. V. Straten. Improving the cost effectiveness of greenhouse climate control. *Computers and Electronics in Agriculture*, 10(3):203–214, 1994. ISSN 01681699. doi: 10.1016/0168-1699(94)90041-8.
- [38] J. M. Lemos. Introdução ao Controlo Óptimo, 2012.
- [39] J. M. Lemos. Equações do Modelo de Estado de Sistemas Lineares Contínuos, 2016.
- [40] G. F. Franklin, D. Powell, and M. Workman. *Digital Control of Dynamic Systems*. ADDISON-WESLEY, 3th edition, 1998.

- [41] A. J. M. Bernardino. Distributed Real-Time Control Systems Slide lectures, 2016.
- [42] S. A. Balula. *Nonlinear control of an inverted pendulum*. Master thesis, Instituto Superior Técnico, 2016.
- [43] K. Johan and R. M. Murray. Feedback Systems An Introduction for Scientists and Engineers University of California Santa Barbara. *Princeton university press*, 2007.
- [44] J. Ziegler and N. Nichols. Optimum Settings for Automatic Controllers, 1942.
- [45] EPICS (Experimental Physics and Industrial Control System), 2017. URL <http://www.aps.anl.gov/epics/>.
- [46] CSS (Control System Studio), 2017. URL <http://controlsystemstudio.org/>.
- [47] iter newsline, 2017. URL <https://www.iter.org/newsline/95/1305>.
- [48] Smolloy. smolloy.com, 2017. URL <http://www.smolloy.com/professional/microcontrollers-mini-computers-and-more/>.
- [49] Humidity, 2017. URL <https://en.wikipedia.org/wiki/Humidity>.
- [50] E. Fitz-rodríguez, C. Kubota, G. A. Giacomelli, M. E. Tignor, S. B. Wilson, and M. Mcmahon. Dynamic modeling and simulation of greenhouse environments under several scenarios : A web-based application. *Computers and Electronics in Agriculture*, 70:105–116, 2010. doi: 10.1016/j.compag.2009.09.010.
- [51] R. Shamshiri, W. Ishak, and W. Ismail. A Review of Greenhouse Climate Control and Automation Systems in Tropical Regions. *Agricultural Science and Applications*, 2(3):176–183, 2013.
- [52] nft systems, 2017. URL [http://www.homehydrosystems.com/hydroponic-systems/nft{\\_}systems.html](http://www.homehydrosystems.com/hydroponic-systems/nft{_}systems.html).

## **Appendix A**

### **Experimental datasets of plant growth**

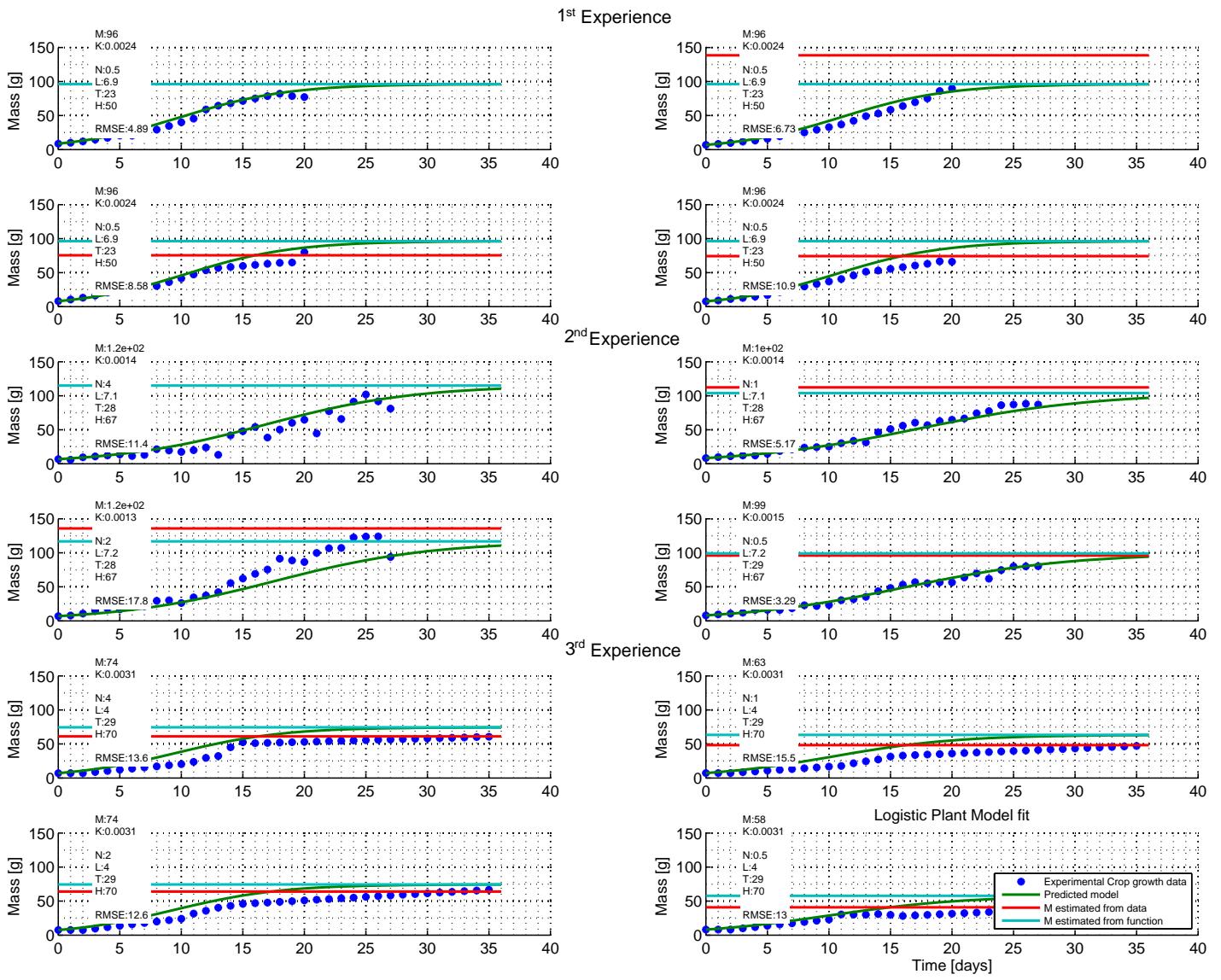


Figure A.1: Blue dots represent the experimental dataset allowing to model plant growth. In green the simulated model fit. In light blue the maximum growth level computed only with the respective experiment data. In red the maximum growth level computed by the global model.  $M$  is the maximum growth level and  $K$  the rate of growth.  $N$ ,  $L$ ,  $T$  and  $H$  are the average values of the nutritive solution concentration, the amount of light received daily, the temperature and the humidity inside the greenhouse, respectively. Finally, RMSE stands for the root mean square error of the difference between the experimental and the simulated data.

## Appendix B

# Economic Perspective

	A	B	C	D	E
1	Description	Value	units	Formula	Source
2	Electricity price	0,16	eur/KWh	0,16	domestic bill
3	Hours in a day	24	h/day	24	-
4		0,001	K/10 <sup>3</sup>	=10 <sup>-3</sup>	-
5					
6	nutrients ( $\epsilon_w$ )	2/1000	eur/g	2/1000	pingo doce supermarket during summer time
7					
8	nutrients ( $\epsilon_1$ )	0,00214	eur/(ml/l).day	=D10/D9	
9	Replacement periodicity	7	days	7	-
10	Concentrated solution cost	0,015	eur/ml	=15/1000	provider alojadamaria.com
11	Diluted solution cost	0,015	eur/(ml/l)	=15/1000	-
12					
13	Artificial light cost ( $\epsilon_2$ )	0,0144	eur/DLI	=D14*D4*D2/D16	
14	Power	26	W	26	provider datasheet
15	Plant effective light intake	80	PAR	80	manual measure with greenhouse sensor
16	Ability to increase DLI per hour	0,288	DLI/h	=D15*10 <sup>-6</sup> *3600	
17					
18	Greenhouse Constant Cost ( $\epsilon_{gh}$ )	0,022	eur /day	=D23*D3*D4*D2	
19	Fan	0,24	W	=0,2*0,1*12	manual measure with a clamp meter
20	Raspberry	2,5	W	=0,5*5	raspberry datasheet
21	Arduino	0,5	W	=0,1*5	manual measure with a clamp meter
22	AirPump	2,5	W	2,5	manual measure with a clamp meter
23	Sum	5,74	W	=SUM(D19:D22)	-

Table B.1: Calculus to compute  $\epsilon_1$ ,  $\epsilon_2$  and  $\epsilon_{gh}$  parameters used in cost function (5.21). These allow to interpret the cost function in *euros*.

Material	#	price /un	Total price	company	link
Greenhouse Structure					
Ripa aplainada 44X44X2400 MM	4	5,29 €	21,16 €	Leroy	<a href="http://www.leroymerlin.pt">http://www.leroymerlin.pt</a>
Policarbonato	2	20,99 €	41,98 €	Leroy	<a href="http://www.leroymerlin.pt">http://www.leroymerlin.pt</a>
Dobradiça AÇO INOXIDÁVEL 60X50MM	2	1,99 €	3,98 €	Leroy	<a href="http://www.leroymerlin.pt">http://www.leroymerlin.pt</a>
30 anilhasFUNIL AÇO D6MM	1	2,19 €	2,19 €	Leroy	<a href="http://www.leroymerlin.pt">http://www.leroymerlin.pt</a>
30 ParafusosPOZIDRIV 3.5X25	2	1,49 €	2,98 €	Leroy	<a href="http://www.leroymerlin.pt">http://www.leroymerlin.pt</a>
40 ParafusosCABEÇA REDONDA 4X30	1	1,79 €	1,79 €	Leroy	<a href="http://www.leroymerlin.pt">http://www.leroymerlin.pt</a>
Esquadro angular40X40MM BICROMADO	20	0,39 €	7,80 €	Leroy	<a href="http://www.leroymerlin.pt">http://www.leroymerlin.pt</a>
<b>Total</b>			<b>81,88 €</b>		
DWC Hidroponic system					
Caixa de yogurte grego 1L (ou equivalente)	4	- €	- €		
Bomba de Ar	1	17,45 €	17,45 €	Urbicult	<a href="https://www.urbicult.pt/bc">https://www.urbicult.pt/bc</a>
Pedra Difusora 2.5cm	8	0,85 €	6,80 €	Urbicult	<a href="https://www.urbicult.pt/bc">https://www.urbicult.pt/bc</a>
Tubo silicone 6mm	3	0,90 €	2,70 €	Urbicult	<a href="https://www.urbicult.pt/tu">https://www.urbicult.pt/tu</a>
<b>Total</b>			<b>26,95 €</b>		
Sensors					
Sensor de Luminosidade	1	6,50 €	6,50 €	SparkFun	<a href="http://www.botnroll.com/p">http://www.botnroll.com/p</a>
Humidity and Temperature Sensor - RHT03   DHT22	2	11,60 €	23,20 €	SparkFun	<a href="http://www.botnroll.com/p">http://www.botnroll.com/p</a>
Temperature Sensor - Waterproof (DS18B20)	1	10,70 €	10,70 €		<a href="http://www.botnroll.com/e">http://www.botnroll.com/e</a>
<b>Total</b>			<b>40,40 €</b>		
Actuators					
TLED 26W 6500K Crescimento	1	65,00 €	65,00 €	Urbicult	<a href="https://www.urbicult.pt/ilu">https://www.urbicult.pt/ilu</a>
Yate Loon 140mm Medium Speed Fan	1	6,00 €	6,00 €		<a href="https://www.amazon.com/">https://www.amazon.com/</a>
Kit Humidificador / Nebulizador	1	30,00 €	30,00 €	Urbicult	<a href="https://www.urbicult.pt/hu">https://www.urbicult.pt/hu</a>
Window Apperture - Servo	1	11,00 €	11,00 €	Hitec's	<a href="https://www.servocity.com">https://www.servocity.com</a>
<b>Total</b>			<b>112,00 €</b>		
Electronics					
flat cable pi camera (2m)	1	3,95 €	3,95 €	Adafruit	<a href="https://www.adafruit.com/">https://www.adafruit.com/</a>
Conectores - macho	10	0,17 €	1,70 €	Mauser	<a href="https://mauser.pt/catalog/">https://mauser.pt/catalog/</a>
ficha femea	100	0,02 €	2,25 €	Mauser	<a href="https://mauser.pt/catalog/">https://mauser.pt/catalog/</a>
pino	100	0,02 €	1,77 €	Mauser	<a href="https://mauser.pt/catalog/">https://mauser.pt/catalog/</a>
raspberry pi 3	1	30,00 €	30,00 €		
raspberry pi camera	1	25,00 €	25,00 €		
Arduino Mega	1	15,00 €	15,00 €		
Arduino relay shield	1	7,00 €	7,00 €		
<b>Total</b>			<b>86,67 €</b>		
<b>Total</b>			<b>347,90 €</b>		

Table B.2: List of expenses to build and automate the greenhouse.

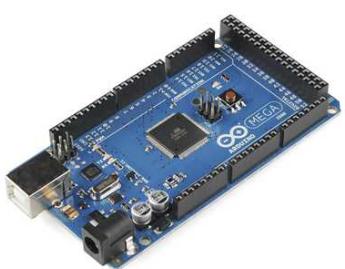
# Appendix C

## Hardware Devices

This appendix lists the hardware devices used in this project. When relevant, some details on the setup and selection preference are given. Some of them were not actually used either because they would not fit the requirements, did not work, or simply there was no opportunity to work with them.

### C.1 Microcontroller

The Arduino Mega is a microcontroller board based on the ATmega1280 chip. The Mega version of Arduino makes sure there are enough input and output pins available for this implementation. The microcontroller was used to read sensors measurements and control the actuators. The main advantage of this microcontroller is that many of the manufacturers of the sensors in use already offer some beginners example code, confirming that these electronics work with such microcontroller. In opposite to the tiny computer Raspberry Pi, also used, Arduino offers both analog and digital handling capability with integrated Analog to digital (ADC), and Digital to Analog Converters (DAC). Arduino also provides Pulse Width Modulation (PWM) technology that will be used, for example, to control the servo motor. Finally, it also works with I2C protocol sensors. See Figure C.1



Model	D14SM – 12
Microcontroller	ATmega1280
Operating Voltage	5V
Input Voltage (recommended)	7 to 12V
Input Voltage (limits)	6 to 20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40mA
DC Current for 3.3V Pin	50mA
Flash Memory	128KB
SRAM	8KB
EEPROM	4KB
Clock Speed	16MHz

Figure C.1: Arduino Mega characteristics

Except for the camera, all the sensors and actuators were connected to the Arduino Mega. To extend

Arduino capabilities, external shields can be attached to it. For example, a shield with relays was used to control the LED light, that has a 220v power supply, greater than the maximum five volts of the Arduino.

## C.2 Tiny computer

The Arduino microcontroller was connected via serial USB cable to a Raspberry Pi v3 - a tiny affordable computer. The Pi was the brain of the system, where all the control happened. Thus, the Pi processes the data from the Arduino and sends the commands back so that Arduino can act. The camera and image processing were also dealt within the Pi. It also had the task of saving the processed information. Finally, the Raspberry Pi was connected to the Internet, allowing external users to connect using a web-based graphical user interface. See figure C.2



---

Raspberry Pi 3

---

A 1.2GHz 64-bit quad-core ARMv8 CPU  
802.11n Wireless LAN  
Bluetooth 4.1  
Bluetooth Low Energy (BLE)  
1GB RAM  
4 USB ports  
40 GPIO pins  
Full HDMI port  
Ethernet port  
Combined 3.5mm audio jack and composite video  
Camera interface (CSI)  
Display interface (DSI)  
Micro SD card slot (now push-pull rather than push-push)  
VideoCore IV 3D graphics core

---

Figure C.2: Raspberry Pi v3 characteristics

## C.3 Sensors

The following devices listed are responsible for sensing the variables that were taken into account in the controller.

### Light sensor

As described in section 2.4.2, plant light absorption is in the 400nm to 700nm spectral region named PAR. In order to measure energy received in this spectrum, PPFD is defined as the number of photons in the 400–700 nm wavelength interval arriving per unit time on a unit area, most commonly in the units micro-mol per square meter per second ( $\mu\text{Mol} \cdot \text{m}^{-2} \cdot \text{sec}^{-1}$ ).

There are sensors that measure PPFD but are very expensive. Another solution is to go with cheaper light sensors that measure a broader wavelength interval, and then do an approximation conversion. (see figure C.3)

Lux sensors are very popular and way less expensive. Lux is the SI unit of luminance measuring the intensity of light that hits or passes through a surface, as perceived by the human eye. It is possible to do a rough conversion from Lux to PPFD by knowing the light source [16] [17]. For sunlight the conversion is done as follows:

$$1PPF(\mu M\text{ol} \cdot \text{m}^{-2} \cdot \text{sec}^{-1}) = 0.0185Lux \quad (\text{C.1})$$

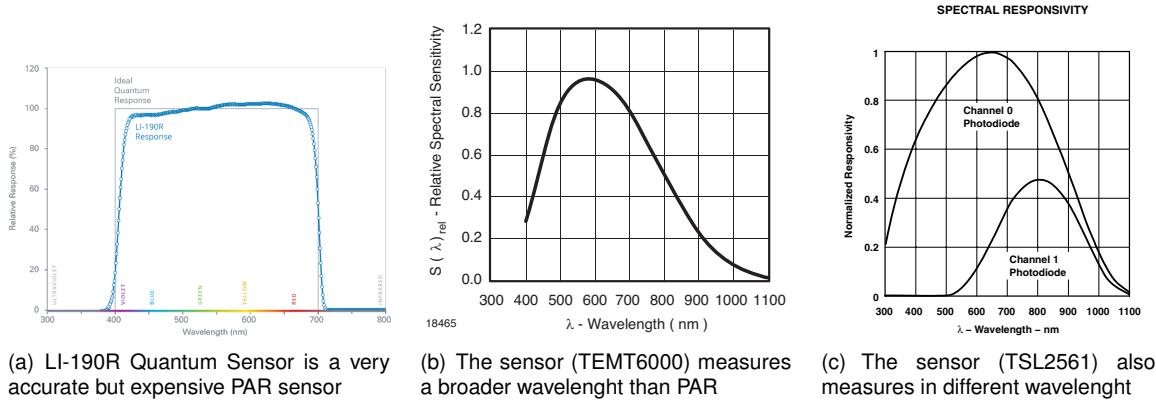
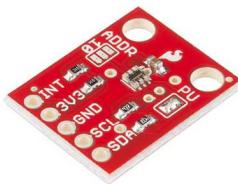


Figure C.3: Spectral Responsivity comparison between a PAR sensor and a Lux sensor

The digital sensor TSL2561 Luminosity sensor (see figure C.4) was provided to work with. It supports I2C protocol, which the Arduino also does. This sensor incorporates both infrared and visible light sensors to better approximate the response of the human eye. Although its supply is of 3.3V, Arduino also has such power supply.

As explained in section 3.1.6 targeting TSL2561 directly to sun light would saturated it, therefore there was a need to cover it in a plastic bottle cap.



Model	<i>TSL2561</i>
Output	Digital Luminance ( <i>Lux</i> )
Dynamic range	0.1 to 40,000 <i>Lux</i>
Supply voltage	3.3V
Communication Protocol	I <sup>2</sup> C

Figure C.4: TSL2561 Luminosity Sensor characteristics

## Air Temperature & Humidity

The RHT03 (also known by DHT-22) (see figure C.5) is a low-cost humidity and temperature sensor with a single wire digital interface. It was connected to one of Arduino's input digital pins. The sensor is calibrated and does not require extra components. One sensor is used to track the humidity and temperature inside and other outside the greenhouse.

## Water temperature

The DS18B20 Digital Waterproof Thermometer provides 9 to 12-bit (configurable definition) temperature readings. Because each DS18B20 contains a unique silicon serial number, multiple DS18B20s can exist



Model	<i>RHT03 DHT22</i>
Output	Digital temperature ( $^{\circ}C$ ) and relative humidity (%)
Power supply	3.3 to 6V
Maximum current	1.5mA
Humidity	0 to 100% $\pm$ 2%
Temperature	-34 to 80 $\pm$ 0.5 $^{\circ}C$
Collecting period	2sec

Figure C.5: RHT03 — DHT22 Air humidity and temperature sensor characteristics

on the same 1-Wire bus. This sensor allows to control the nutrient solution temperature, see figure C.6



Model	<i>DS18B20</i>
Output	Digital temperature ( $^{\circ}C$ )
Power supply	3.3 to 5.5V
Temperature	-55 to 125 $\pm$ 0.5 $^{\circ}C$
Collecting period	1sec

Figure C.6: DS18B20 waterproof temperature sensor characteristics

### Water conductivity (EC)

Measuring water conductivity revealed to be a hard task. Moreover, pH and EC sensors are quite expensive.

Nevertheless, there was an attempt to use a soil probe that was developed at IST university, see Fig. C.7. Because it is specially designed to measure moisture in soil it revealed not to have the ability to measure the EC conductivity in water. It was actually possible to measure different conductivities until a certain point, although, reaching the operating point for hydroponic use it would saturate.

As expected, the experiment confirmed that, changing the water temperature would change its conductivity. This would have to be a factor to take into account if proceeding with its use.

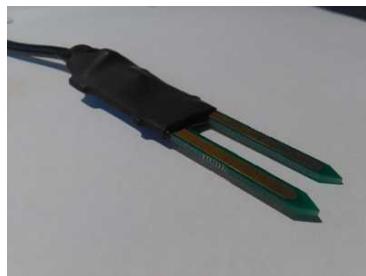


Figure C.7: Waterproof conductivity probe designed at IST university.

### Water pH

Without being discouraged about the EC probe, there was an attempt to buy a cheap pH probe online, see Fig. C.8. However it revealed not to work at all.

This pH probe is composed by a single cell battery with a very high resistance, where the voltage produced is proportional to the Log of the hydrogen ion concentration. When the concentration is greater on either side of the probe, the ion flow will induce a slight voltage between the probe's electrodes. This voltage can swing both positive or negative which will indicate either an acid or a base solution.

The types of sensor have to be calibrated regularly using a standard pH solution. Generally, this period is about half a year.



Model	<i>SEN0161</i>
Output	Analog voltage
Supply voltage	5V
Range	0 to 14pH
Accuracy	$\pm 0.1\text{pH}(25^\circ\text{C})$
Operating Temperature	0 to 60
Collecting period	< 1min

Figure C.8: Analog SEN0161 pH meter sensor characteristics

## CO<sub>2</sub>

The analog MG811 Carbon dioxide sensor (see Fig. C.9) has an output voltage proportional in logarithmic scale with the CO<sub>2</sub> ppm concentration in air. This sensor has to be heated before a correct measure can be done. Such a process can take from half an hour to 48 hours. The 5V power input should be boosted to 6V for this process. Therefore an external power supply 7 ~ 12V > 500mA is necessary to feed Arduino. In order to calibrate the sensor, a CO<sub>2</sub> calibrated meter is needed.

Unfortunately, there was no time to develop the control of the greenhouse CO<sub>2</sub> concentration in air, therefore this sensor was not used.

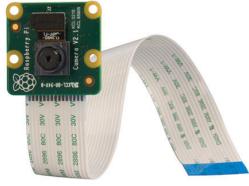


Model	<i>MG811</i>
Output	Analog voltage 0 to 5V 350 to 10 <sup>4</sup> ppm Co2
Power supply	5V > 500mA recommended
Operating Temperature	-20 to 50°C
Collecting period	after 0.5 ~ 48h (heating process)

Figure C.9: Analog MG811 Carbon dioxide sensor characteristics

## Camera

To estimate crop growth image processing was used. A camera (see Fig. C.10) was attached on the top inside of the rooftop facing the crop. Because of the computational resource needed in image processing, the camera was connected directly to the Raspberry Pi which was able to handle it.



Model	Raspberry Pi Camera Module v2
Still resolution	8 Megapixels
Video modes	1080p30, 720p60, 640x480p60/90

Figure C.10: Raspberry pi Camera characteristics

## C.4 Actuators

The following listed devices are responsible for actuating on some of the variables that were taken in account by the controller.

### Fan

Generally speaking it is best to exchange the entire contents of the growing area about once an hour during daylight hours[12]. A fan is characterized by the volume it can exchange per time unit. In America, the common units are CFM - cubic feet per minute which can easily be translated in cubic meters per hour. The conversion is as follows  $1CFM = 1.70m^3/h$ .

Therefore, for a system with less than  $1m^3$  like this  $1CFM$  fan would be enough, although it would have to work most of the time and at full speed. Because there was already one fan available (see Fig. C.11), the one presented was used. Its characteristic is of  $62CFM = 105m^3/h$  that is more than enough for this system, meaning that it can work at a low speed and in shorter time periods.

This fan was powered with an  $12V$  power supply. One solution to control the speed of two wire fans like these, is by inputting a Pulse With Modulation (PWM) signal. The Arduino has PWM outputs but its power supply is only of  $5V$ . Therefore, a very simple transistor circuit was designed.

The transistor (Bd137) works as a digital switch, thus its collector and emitter were connected in series with the fan as shown in figure C.12. The current flowing through the fan depends on the transistor gain  $h_{FE}$  according to:  $I_C = h_{FE}I_B$ . The fan maximum input current is  $0.7A$  and the transistor gain for currents around this value can fluctuate from 25 to 250. Considering the worst case, the base current should be at least

$$I_B = \frac{I_C}{h_{FE}} = \frac{0.7}{25} = 28mA.$$

From Kirchhoff's voltage law  $V_i = R_B I_B + V_{BE}$  the minimum base resistance  $R_B$ , that allows the fan maximum input current, comes as

$$R_B = \frac{V_i - V_{BE}}{I_B} = \frac{5 - 0.5}{28 \times 10^{-3}} \approx 160\Omega.$$

It is important to note that the maximum frequency of this transistor is  $190MHz$ . This is not a problem since is quite higher compared to the used PWM frequency of Arduino.



Model	D14SM - 12
Dimensions	140x140x20mm
Air Flow Max	62.0CFM
Noise Level Max	29dBA
Fan Speed:	1400RPM ± 10%
Rated Voltage:	12VDC
Operation Voltage	6.5 to 13.8VDC
Input Max current	0.70Amps
Operation Temp	-10°C to + 65°C
RPM monitoring	No

Figure C.11: Yate Loon 140mm Medium Speed Fan (D14SM-12) characteristics

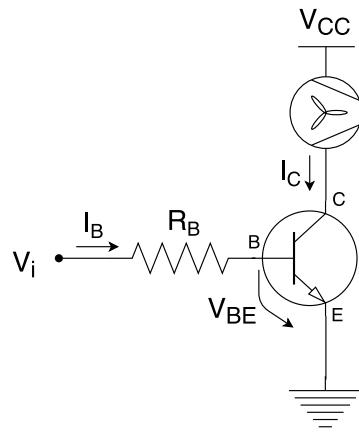


Figure C.12: Fan circuit schematic allowing to have a  $V_{cc} = 12$  Volt PWM controlled with Arduino input  $V_i = 5$  V.

## Servo Motor

The window aperture actuator was added in order for the system to have a natural and smooth air circulation, avoiding the constant use of forced ventilation, saving energy and having a more steady evolution of the greenhouse variables. It influences the air temperature inside the greenhouse along with humidity and gas concentration like carbon dioxide.

To control the aperture angle  $\alpha$  a servo motor was used. There is an arm attaching the rooftop with the servo, letting the top roof to open at a certain fixed angle (see figure C.13).



Model	hs-322HD
Voltage Range	4.8V to 6.0V
Stall Torque (4.8V)	$3.0\text{kg.cm}^{-1}$
Max PWM Signal Range	553 to 2450 $\mu\text{sec}$
Pulse Amplitude	3 to 5V
Operating Temperature	-20 to + 60°C
Current Drain - idle	(4.8V)7.4mA

Figure C.13: HS-322HD servo motor characteristics

## **LED artificial light**

Once the greenhouse is subject to sun light, there is no need for a very powerful artificial light. From the offer available in Portuguese hydroponic stores, the *TLED 26W 6500K Crescimento* light was chosen.

According to the manufacture this is a good solution for reinforcing existing light. Indeed, confirming with the light sensor at a distance of around *40cm* it delivers around 80 PAR. Note that the greenhouse sensor measures Lux and its measurement is subject to a translation ir PAR assuming the source of light is the sun. Therefore this is not the correct way to discover how much light in the PAR spectrum the LED light is able to provide to the plants. However, there was no other solution available with more precision. In conclusion, the amount of light provided by the artificial light is not known with rigor.

The light temperature could be *6500K* or *2100K*. The first was chosen because for the initial growth period and vegetation stage plants tend to benefit more from colder lights. On the other hand, during flowering plants prefer warmer colors.

LED light also allows to avoid green light which is not so much used by plants, allowing to improve lighting efficiency.

## **Ultrasonic Mist Maker**

In order to increase relative humidity inside the greenhouse two common solutions are possible. For big applications the use of small water sprinklers with a pipe system is used, but this requires a pressure of at least *2 bar*, and there are leakage concerns. For small applications like this, an ultra sonic mist maker device is an excellent choice. This device sends, at water surface, ultrasonic waves that makes the water particles to split into the air. It does not produce water vapor as it does not heat the water. Therefore, it is very advantageous as it does not contribute to increase the temperature, it actually helps reducing it due to the later evaporation process. A DIY system using the mist maker,a small fan, and a recycled recipient was built for a more effective result, this is represented in Figure C.14



Model	Kit Humidificador / Nebulizador from urbicult
Operating Voltage	12 V

Figure C.14: Mist Maker characteristics

## **Water Pump**

A water pump was used to circulate the nutrient solution from the tank to the tubes where the plants are sustained. This was only used for the first implemented hydroponic system. The recommended flow rate for a NFT system is typically between 1 to 2 liters per minute for each grow tube [52].

Therefore if the FIT0200 water pump is used at a unevenness of 40 cm with a power supply of 3.5V, it will be able to flow 1.6 L/m which is in the recommended water flow, see Figure C.15



Model	FIT0200
Power supply	3.5V to 12.0V, 65mA to 500mA
Pumping head	40 to 220cm
Capacity	1.6 to 5.8L/m
Power range	0.5 to 5W

Figure C.15: FIT0200 water pump characteristics

### Air Pump

For the second hydroponic system, an air pump was required. Along with a set of connected air pipes, the air is spread over the multiple water containers. At the end of each container tube, there is a diffuser stone that produces even and small air bubbles inside the water solution. Moreover, each tube has its individual valve allowing to control the amount of air that goes through each stone.



Model	Air Professional 150
Power supply	220 V
Power consumption	2.5 W
Air flow	150 l/h

Figure C.16: Air pump characteristics

