

# HYGROTERMIC CONTROL OF THE MICROCLIMATE AROUND BUILDINGS

Master Thesis

Christian Kongsgaard Nielsen

March 2018



---

## 1.1 - PREFACE

Before you lie the thesis “Hygrotermic Control of the Microclimate Around Buildings”, it presents and deploys two developed models to investigate the effect of vegetation on the microclimate.

It has been written to graduate from the Master program: Architectural Engineering at the Technical University of Denmark (DTU). The research and writing were conducted during my last semester, from October 2017 to March 2018.

My research questions were formulated in corporation with my supervisor Kristoffer Negendahl. The research has been difficult at times, but with extensive investigation it has been possible to answer the question identified in the beginning.

I would like to thank my supervisor for excellent guidance and support during this process. I also wish to thank all the respondents, without whose cooperation I would not have been able to conduct this study. I would also like to thank friends and family that have helped me, given me feedback and helped with editing.

I hope you enjoy your reading.

Christian Kongsgaard Nielsen

---

Copenhagen, 19<sup>th</sup> of March 2018

## 1.2 - ABSTRACT

The urban heat island (UHI) is a well-documented phenomenon, that increases the temperature within a city. In hotter climate that leads to uncomfortable outdoor spaces. Research shows that vegetation and trees can help decrease the temperature (Shashua-Bar et al. 2009). Current tools, for investigating the effect of vegetation on the microclimate in the urban space, are not well suited in an integrated design process. The objective of the thesis is to develop such a method, which evaluates the influence of vegetation on thermal comfort in the microclimate and apply it to a case study in Abu Dhabi.

For that purpose, two models were developed: The Soil Model and The Atmosphere Model. The Soil Model computes the evapotranspiration from the ground. It is implemented as a wrapper around the Catchment Modelling Framework (Kraft et al. 2011). The Atmosphere Model is a simple air volume model, that turns the evapotranspiration into a change in air temperature and relative humidity. The two models are written in Python and implemented in the parametric design tool Grasshopper. It has been demonstrated how they can function in an analytic workflow with Ladybug Tools to compute outdoor thermal comfort. The thermal comfort metric Universal Thermal Climate Index (UTCI) was used to assess the thermal comfort.

A case study in Abu Dhabi, United Arab Emirates, was conducted. The case study was 20m times 22m site, with a wadi running through and five trees. The thermal comfort was evaluated with the proposed method and compared to an assessment method that neglects evapotranspiration. The results showed that an average reduction of 3.7°C UTCI was present, when evapotranspiration was included. During the daytime of June 1<sup>st</sup>, the reduction would reach 15°C UTCI.

From this study it could be concluded that evapotranspiration has a substantial impact on the thermal comfort in the microclimate and that it is possible include such an analysis in an integrated design process.

---

## 1.3 - RESUMÉ

Den urbane varmeø (UHI) er et vel dokumenteret fænomen, som øger temperaturen i en by. I varmere klimaer giver det ukomfortable uderum. Forskning viser at vegetation og træer kan hjælpe med at nedbringe temperaturen (Shashua-Bar et al. 2009). Nuværende metoder til at undersøge effekten af vegetation på mikroklimaet i uderum, er fungerer ikke i en integreret design proces. Målet med dette speciale er at udvikle en metode, der kan evaluere planters indflydelse på den termiske komfort i uderummets mikroklima og anvender det på et casestudie i Abu Dhabi.

Til det formål er to modeller blevet udviklet: en jord model og en atmosfære model. Jordmodellen beregner evapotranspirationen fra jorden. Den er blevet implementeret som en "wrapper" omkring Catchment Modelling Framework (Kraft et al. 2011). Atmosfærermodellen er en simpel luft volumen model, som veksler evapotranspiration om til en ændring i luft temperaturen og den relative luftfugtighed. De to modeller er skrevet Python og er blevet implementeret i det parametriske design værktøj Grasshopper. Det er blevet demonstreret at de kan fungere i en analytisk arbejdsgang med Ladybug Tools til at beregne den termiske komfort i uderummet. Til at vurdere den termiske komfort er det universale termiske klima indeks (UTCI) blevet brugt.

Et casestudie i Abu Dhabi, De Forende Arabiske Emirater, blev foretaget. Casestudiet var et område på 20m gange 22m, med en wadi løbende igennem og fem træer. Den termiske komfort blev evalueret med den foreslænde metode og en metode der negligerer evapotranspiration. Resultaterne viste at en gennemsnitlig reduktion på 3.7°C UTCI kunne opnås, når evapotranspiration var inkluderet. I løbet af dagen op den første juni kunne reduktion stige til 15°C UTCI.

Ud fra dette studie kan der konkluderes at evapotranspiration har en væsentlig effekt på den termiske komfort i mikroklimaet og at det er muligt inkludere en analyse af det i Grasshopper.

---

# 1.4 - TABLE OF CONTENTS

<b>1.1 - PREFACE</b>	I
<b>1.2 - ABSTRACT</b>	II
<b>1.3 - RESUMÉ</b>	III
<b><u>2 - INTRODUCTION</u></b>	<b>1</b>
<b>2.1 - INTRODUCTION TO TOPIC</b>	<b>2</b>
Motivation	2
Research Question	3
<b><u>3 - THEORY</u></b>	<b>4</b>
<b>3.1 - ATMOSPHERE MODEL</b>	<b>5</b>
Introduction	5
Model Description	5
Model Expressions	6
<b>3.2 - HYDROLOGY &amp; SOIL MODEL</b>	<b>9</b>
Introduction	9
Basic Concepts of Hydrology	9
Soil Model Data	16
Deciduous Trees	17
<b>3.3 - MEAN RADIANT TEMPERATURE</b>	<b>19</b>
Soil Temperature	19
Sky Temperature	19
<b>3.4 - COMFORT METRICS</b>	<b>21</b>
Introduction	21
Standard Effective Temperature	21
Physiologically Equivalent Temperature	22
Universal Thermal Climate Index	24
Comfort Metric Comparison	26
Conclusion	29

---

<b>3.5 - SOFTWARE</b>	<b>30</b>
Applied Software	30
Software Connections	30
<b>4 - DEVELOPED MODELS</b>	<b>32</b>
<b>4.1 - CONCEPTUAL APPROACH</b>	<b>33</b>
Developed Models	36
<b>4.2 - LIVESTOCK</b>	<b>37</b>
Background	38
Documentation	39
Grasshopper Components	39
GHComponent	40
Data Formats	41
Data Flow	41
Grasshopper to CPython Link	42
SSH Tunnel	45
<b>4.3 - ATMOSPHERE MODEL</b>	<b>47</b>
Description	47
Grasshopper Components	51
Script setup	51
<b>4.4 - SOIL MODEL</b>	<b>53</b>
Description	53
Grasshopper Components	53
Script Setup	53
<b>5 - CASE STUDY</b>	<b>56</b>
<b>5.1 - ABU DHABI CASE</b>	<b>57</b>
Case Description	57
<b>5.2 - WEATHER ANALYSIS</b>	<b>59</b>
Abu Dhabi	59
Copenhagen	63
<b>5.3 - THEORETICAL LIMIT</b>	<b>69</b>

---

<b>5.4 - WADI ANALYSIS</b>	<b>73</b>
Initial Model	73
Model with Sprinklers	77
Parameter Variations of Ground Properties	82
Wadi in Copenhagen	84
Wadi with Water	86
Optimized Model	88
<b>5.5 - AIR CONDITION ANALYSIS</b>	<b>90</b>
Optimized Model	90
Optimized Model Without Wind	92
<b>5.6 - MRT ANALYSIS</b>	<b>95</b>
Script	95
Ground Temperature	97
Water Filled Wadi	98
Canopy Temperatures	100
<b>5.7 - COMFORT ANALYSIS</b>	<b>101</b>
Comfort with Current Methods	101
Comfort Optimized Model	105
Comfort Comparison Between Models	109
Comfort Model Without Wind	112
<b>6 - DISCUSSION AND CONCLUSION</b>	<b>118</b>
<b>6.1 - DISCUSSION</b>	<b>119</b>
Atmosphere Model	119
Soil Model	120
Comfort Computation	122
Case Study Results	122
Model Validation	125
<b>6.2 - CONCLUSION</b>	<b>126</b>

---

**7 - POSTFACE** **127**

**7.1 - BIBLIOGRAPHY** **128**

---

**8 - APPENDIX** **I**

Appendix A II

Appendix B II

Appendix C II

## 2 - INTRODUCTION

---

## 2.1 - INTRODUCTION TO TOPIC

" For decades the human dimension has been an overlooked and haphazardly addressed urban planning topic, while many other issues, such as accommodating the rocketing rise in car traffic, have come more strongly into focus. In addition, dominant planning ideologies – modernism in particular – have specifically put a low priority on public space, pedestrianism and the role of city space as a meeting place for urban dwellers. Finally, market forces and related architectural trends have gradually shifted focus from the interrelations and common spaces of the city to individual buildings, which in the process have become increasingly more isolated, introverted and dismissive."

(Gehl 2001)

### Motivation

Cities are a place where people live, work, meet, trade, play and in general interact with each other. These interactions either require an outdoor space or transport through such a space. These spaces have often been neglected in the past to given way for cars and massive superstructures. The world is becoming more and more urbanized: in 2014, 73% of the population of Europe lived in cities and 54% of the world's population did the same, in 2050 that number is projected to be 66% (UNDESA 2014). There is a need to address some of the issues Gehl presents in his statement.

One of the most important qualities for human activities within the city are if the weather is good or not. The weather can be subdivided into three climate levels: macroclimate, local climate and microclimate. Macroclimate is the surrounding climate related to the location; local climate is the climate surrounding cities and the microclimate is climate experienced in streets, squares and parks in the city (Gehl 2001). The macroclimate is determined by the geography and location on earth. The local and micro- climate are influenced by the macroclimate but are also highly depended on the fabrics and geometry of the city (Oke 1981) and these things can be engineered! The Urban Heat Island (UHI) is a well-known and documented example of this (Roth 2013). The phenomenon describes the local increase in temperature within a city compared to the rural surroundings. The increase in temperature is among others caused by the difference in surface coverings between urban and rural and especially the lack of vegetation in the urban context.

Even though the UHI is a local climate phenomenon caused and influencing the city as a whole, it is made up of the different microclimates in the city. Where it is hard and complex to make citywide changes to alter the local climate, the microclimate can easier be influenced. Surface coverings, building and street geometry can be formed so they provide the right amount of shade, blockage from the wind and relative humidity such that they create a comfortable frame for the activities that are conducted in the space –

---

whether that is walking, sitting and drinking coffee, playing or exercising.

A lot of research have gone into describing the effects of surface coverings and geometries, while the vegetation aspect has often been neglected. Vegetation have a strong influence on the microclimate as well; grass, bushes and trees alter the humidity and provides cleaner air. bushes and trees block the wind and finally trees provide shade for people in the outdoor space. Besides the influence on microclimate, vegetation can also be used as storm water management (Københavns Kommune 2015) and it makes humans more productive and happy (Nieuwenhuis et al. 2014). The missing focus on vegetation's influence on the microclimate is what this thesis will try to compensate for.

A way to evaluate the human perception of the microclimate is through their thermal comfort levels. Thermal comfort metrics takes a range of microclimate factors into account to give prediction of what the experienced temperature may be in a space and if that is comfortable or not for a person. It is therefore an obvious way to measure the effects vegetation may have on the microclimate while still keeping the human aspect and it will be used in this thesis as a way to do that.

## Research Question

The objective of this thesis to develop a method which evaluates the influence of vegetation on thermal comfort in the microclimate. This will be done through the following:

An account of significant theories and methods concerning the subject, with a special attention to methods of evaluating outdoor comfort.

By developing a method which considers the hydrothermal effects from vegetation on air temperature and relative humidity.

By investigating and evaluating the effects of vegetation on the outdoor human comfort and the microclimate through a case study with the developed method.

The research questions should not be seen as parallel but as serial steps to answer the overall question. The method and model to investigate and evaluate the effects of vegetation in 3) is developed in 2) and the methods and theories used to develop that is described in 1).

The main challenge and where most of the work hours have been placed is to develop the method and model of 2). Computer code is not well suited for a report like this, therefore a webpage with the documentation is created, and it can be found on <https://ocni-dtu.github.io/> or in Appendix B and C.

This report will thoroughly cover 1) and 3), but also give a presentation of 2).

# 3 - THEORY

# 3.1 - ATMOSPHERE MODEL

## Introduction

To calculate a local air temperature and relative humidity a simple atmospheric model has been developed. The model computes a new temperature and relative humidity for a column of air above each ground surface cell, defined in the soil model.

## Model Description

The model assumes that at the top of the air column the measured temperature and relative humidity values from the weather file are present and at half of the air column's height the mean temperature and relative humidity are present. For each time step the air column is initialized with the temperature and relative humidity from the weather file. It does therefore not remember the previous temperature and relative humidity it had. Furthermore, it is assumed that air columns do not exchange any heat or vapour between each other. The influence of the wind is modelled as if the wind expands the air volume above the surface cell.

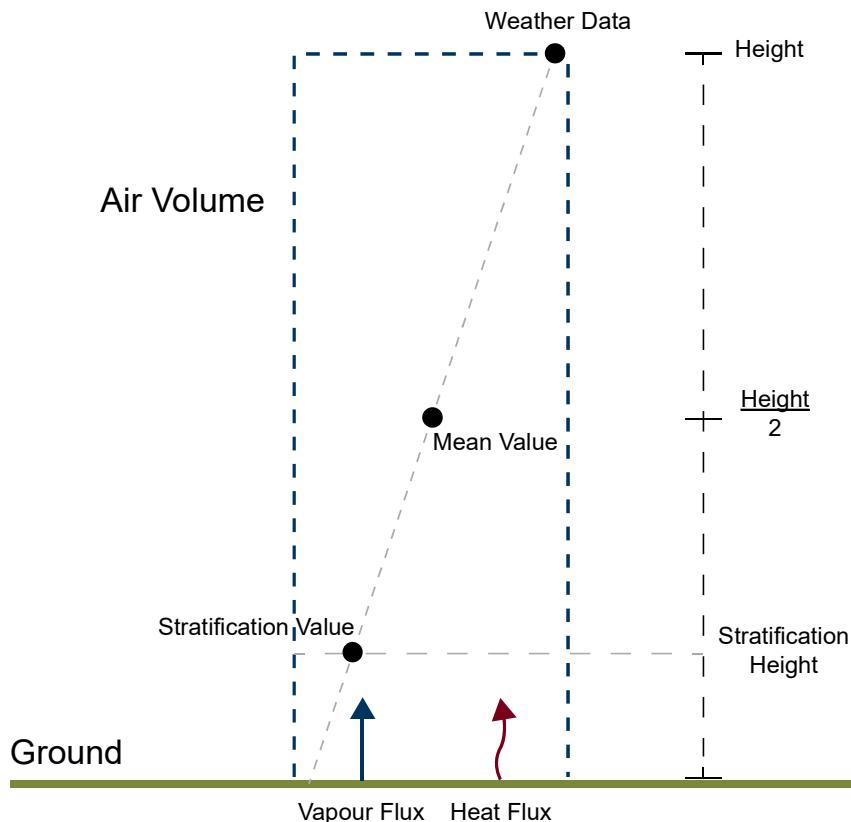


Figure 3.1 - Illustration of the Atmosphere model's air column.

It is assumed that the temperature and relative humidity in the column is equal to the temperature and relative humidity from the weather file at the top of the column. It is further assumed that at 0.5 \* height of the column that the temperature and relative humidity is equal to the same mean values. These two assumptions give the option to create a linear stratification so that it is possible to calculate a temperature and relative humidity for an arbitrary height within the column.

These assumptions all comes down to: weather file temperature, relative humidity and wind speed, air column height, surface cell area, vapour and heat fluxes from the ground. The vapour and heat fluxes are transferred from the soil model.

## Model Expressions

The formulas of the model can be found in (Peuhkuri and Rode 2016) unless stated otherwise. The model uses the following expressions:

$$T_{mean} = \frac{E_{air} + E_{heat}}{V_{air} + \rho_{air} + Cp_{air}} \quad (1)$$

Where  $T_{mean}$  is the mean temperature of the air column in K.

$E_{heat}$  is the added latent heat to the air column from the heat flux in J. (Manickathan et al. 2018) express  $E_{heat}$  as:

$$E_{heat} = -2.5 \cdot 10^6 \cdot M_{vp} \quad (2)$$

$E_{air}$  is the energy in the air column in J, computed by:

$$E_{air} = V_{air} \cdot Cp_{air} \cdot \rho_{air} \quad (3)$$

Where  $V_{air}$  is the air volume in  $m^3$ .

$Cp_{air}$  is the specific heat capacity in  $J/kgK$

$\rho_{air}$  is the density of the air in  $kg/m^3$ , computed by:

$$\rho_{air} = 1.29 \cdot \frac{273.15}{T_{air}} \quad (4)$$

Where  $T_{air}$  is the air temperature above the air column in K.

The mean relative humidity is computed by:

$$RH_{mean} = \frac{P_{v,mean}}{P_{v,sat}(T_{air})} \quad (5)$$

Where  $RH_{mean}$  is the mean relative humidity in the air column and is unitless.

$p_{v,sat}$  is the saturated vapour pressure in Pa at a given temperature,  $T_{air}$  in °C and it is calculated as:

$$p_{v,sat}(T_{air}) = 288.68 \cdot \left(1.098 + \frac{T_{air}}{100}\right)^{8.02} \quad (6)$$

$p_{v,mean}$  is the mean vapour pressure in the air column in Pa.  
This can be expressed as:

$$p_{v,mean} = \frac{(M_v + M_{vp}) \cdot R_v \cdot T_{air}}{V_{air}} \quad (7)$$

Where  $M_v$  is the mass of vapour in the air in kg. It is computed by:

$$M_v = \rho_v \cdot V_{air} \quad (8)$$

$M_{vp}$  is mass of vapour production in kg.

Where  $R_v$  is the vapour gas constant and equals 461.5 J/kgK

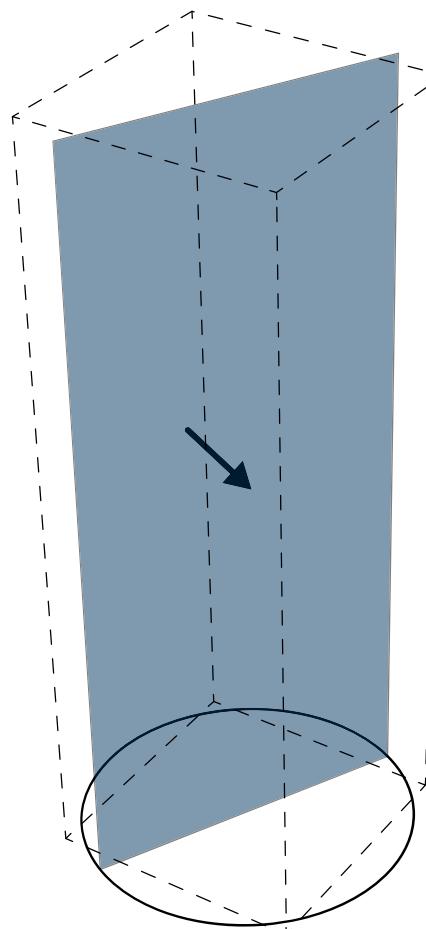


Figure 3.2 - Visualization of the wind flux calculation. The blue surface is the cross-sectional area, determined by the diameter of the circle and the height of the air volume. The dashed cube is the actual air volume. The arrow is the wind speed vector.

$\rho_v$  is the density of the vapour. It can be calculated by:

$$\rho_v = \frac{P_{v,air}}{R_v \cdot T_{mean}} \quad (9)$$

To adjust for the wind speed, the air flux through the volume needs to be calculated. The wind speed adjusted volume -  $V_{adjusted}$  in m<sup>3</sup> – can be formulated as:

$$V_{adjusted} = V_{air} + W_F \quad (10)$$

Where  $W_F$  is the wind flux in m<sup>3</sup>/h. It can be computed by multiplying the wind speed –  $W_s$  in m/s – with the vertical cross-sectional area of the volume –  $A_{cs}$  in m<sup>2</sup>:

$$W_F = W_s \cdot A_{cs} \quad (11)$$

As the only information known about the volume is the height and the foot-print area, it is not possible to get the exact cross-sectional area, so an assumption must be made. It is assumed that using the diameter from a circle, with the same area as the mesh face, multiplied by the height the height of the volume is a close enough approximation for the wind flux.

The EPW reports wind speeds at 10m over the ground, in order to adjust that the wind speeds decrease the closer you get to the ground, the wind profile power law is used (Wikipedia 2017):

$$u = u_r \cdot \left( \frac{z}{z_r} \right)^{\frac{1}{7}} \quad (12)$$

An average of the wind speed from 0 – 10m will then be used as  $W_s$

---

## 3.2 - HYDROLOGY & SOIL MODEL

### Introduction

The soil model is a boundary condition to the atmospheric model. The basic reason for developing this model is to calculate a vapour flux from the ground to the air. This is modelled in the soil model as evapotranspiration as it releases both a vapour flux. Most of the processes in the soil model is hydrology, therefore a brief introduction to hydrology will be given later in this section. For modelling the hydrological processes, the Catchment Modelling Framework will be used.

### Catchment Modelling Framework (CMF)

The Catchment Modelling Framework, from here on called CMF, is a programming library for hydrological models. CMF is modular and lets the user create and solve a wide range of hydrological Finite Volume Models, from lumped models, over 1D to 3D models.

CMF is written in C++ to obtain high performance and made into a Python library to be more accessible. CMF is an open-source tool, released under GPLv3. (Philipp Kraft et al. 2011)

### Grasshopper - CMF integration

The objective of using CMF is to rely on an already existing tool and let that tool handle the hydrological modelling. By using Python to make custom components in Grasshopper it is possible to connect the geometry from Grasshopper with CMF to provide a basis for the hydrological modelling.

### Basic Concepts of Hydrology

Water is one of the key ingredients in most natural processes. The abundance of water in the atmosphere and oceans gives it a strong influence on the global climate by water's high capacity for storing energy and the substantial amounts of energy, which goes into the phase-changes of water. Hydrology is the study of water and its movement on and below earth's surface (Hornberger 1998).

## Catchment

A catchment is the basic hydrological element of investigation. A catchment is defined as an area, where all the water flowing through it, ultimately ends up running through a single point e.g. a stream or river. A simple equation for conservation of mass (water) can be written as:

$$\frac{dV}{dt} = P - R_s - R_g - ET \quad (13)$$

Where V is the water stored in the catchment in  $m^3$ .

P is the precipitation rate in  $m^3/h$ .

$R_s$  is the surface run-off rate in  $m^3/h$ .

$R_g$  is the groundwater run-off in  $m^3/h$

ET is the evapotranspiration in  $m^3/h$  (Hornberger 1998).

A more detailed description of the processes in a catchment can be sketched as in Figure 3.3. CMF models are able to handle the processes depicted in that figure.

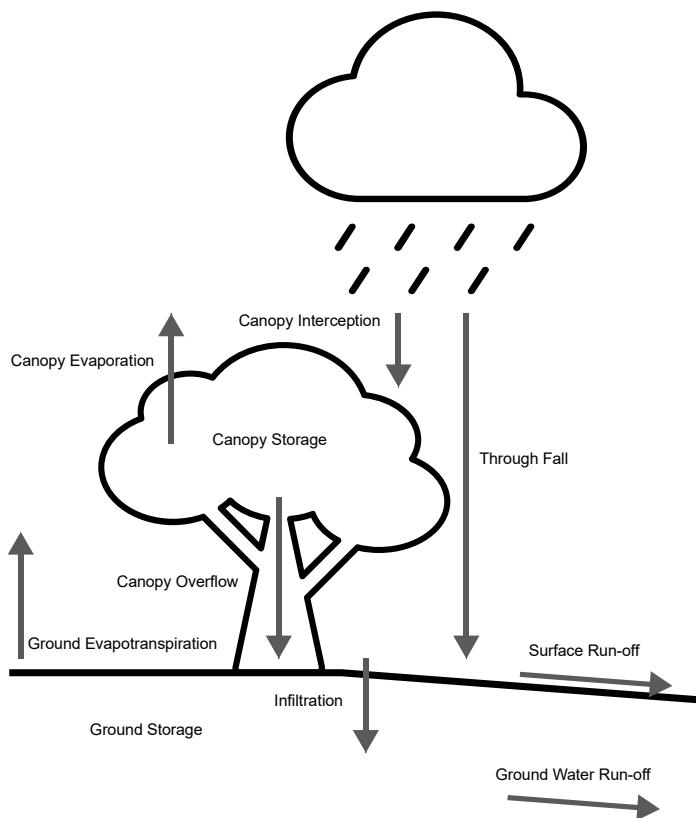


Figure 3.3 - Hydrological processes in a catchment.

## Evapotranspiration

Evapotranspiration is the processes of how liquid water at or near the ground become atmospheric water vapour. It consists of two parts the evaporation of water from wet surfaces and transpiration of water vapour through plants. It is normally not possible to separate the two and therefore the process is considered as one (Hornberger 1998).

There are two types of evapotranspiration to be considered:

Potential evapotranspiration ( $ET_p$ ), which is the maximum rate of evapotranspiration under an unlimited supply of water.

Actual evapotranspiration ( $ET_A$ ), which is a function of soil moisture deficit (SMD) and available water content (AWC) their relationship can be described in the following way (Eslamian 2014):

$$ET_A = ET_p \cdot \left(1 - \frac{SMD}{AWC}\right) \quad (14)$$

The explicit calculation of  $ET_A$  is rather complicated and depend on numerous factors – such as relative humidity, radiation, temperature and wind speed and will not be described in detail here.

CMF has implemented several methods to calculate actual evapotranspiration. Two of the noteworthy are FAO Penman-Monteith (Allen et al. 1998) and Shutterworth-Wallance (Federer 2017).

The FAO Penman-Monteith method will be used in this thesis. It is derived from the original Penman-Monteith equation from 1948. The Food and Agriculture Organization of the United Nations (FOA) have modified the original equation to address the shortcomings of it and make it applicable worldwide.

## Canopy interception

Where tree cover exists up to 50% of the precipitation can be intercepted by the tree canopy and thereby preventing it from hitting the ground. It is therefore an essential element to consider. The canopy interception is reliant on two sets of characteristics; characteristics related to the vegetation, such as canopy shape, size and moisture storage capacity and the leaves' size and shape and characteristics related to the rainfall, such as intensity and frequency. Rain will be stored in the canopy until its capacity is reached, where after it overflows and becomes through fall or stemflow (Eslamian 2014).

CMF uses the Rutter-Morton Interception method to calculate the canopy interception (Meuser 1990).

## Surface run-off

There are four ways for precipitation to enter a stream and exit the catchment: Direct precipitation into a stream channel, surface run-off, shallow subsurface stormflow and groundwater flow (Hornberger 1998). The direct precipitation into a stream, is not very interesting in relation to this topic, as it does not affect the ground.

Surface run-off is the water that flows on top of the ground and discharges into the stream that way. It goes without saying that surface run-off only occurs if the water cannot infiltrate the ground, because the precipitation rate exceeds the infiltration rate. This process is much more relevant in relation to this topic as urban and build-up areas often have a lower infiltration rate, because of e.g. pavement etc. In CMF surface run-off is implemented with two models; Kinematic Surface Run-off or the Diffusive Run-off:

$$q_{srf} = A_{cross} \cdot d_{eff}^{\frac{2}{3}} \cdot \frac{\sqrt{S}}{n} \quad (15)$$

Where  $q_{srf}$  is the surface run-off flow in  $\text{m}^3/\text{day}$

$d_{eff}$  is the effective water flow depth in m. The effective flow depth can be expressed as:

$$d_{eff} = \begin{cases} \frac{V}{A} - d_{puddle} & \text{if } \frac{V}{A} > d_{puddle} \\ 0 & \text{if } \frac{V}{A} \leq d_{puddle} \end{cases} \quad (16)$$

$V$  is the volume of the surface water in  $\text{m}^3$ .

$A$  is the area of the cell in  $\text{m}^2$ .

$d_{puddle}$  is the average depth of water needed to start the run-off.

$A_{cross}$  is the effective cross-sectional area between the two neighboring cells. It is given as:

$$A_{cross} = d_{eff} \cdot w \quad (17)$$

Where  $w$  is the shared boundary between the two cells in m.

$n$  is the Manning roughness.

$S$  is the slope between the center of the surface water and the neighbor cell. For the Kinematic Model,  $S$  is defined as:

$$S = \left\| \frac{\Delta z}{d} \right\| \quad (18)$$

Where  $\Delta z$  is the difference in heights between the two cells in m.

d is the distance between the two cells in m.

For the Diffusive Run-Off Model, S is given as:

$$S = \left\| \frac{\Delta h}{d} \right\| \quad (22)$$

Where  $\Delta h$  is the difference in potential in the surface water of the two cells.

As precipitation infiltrates the ground surface it continues to move downward. Normally soil's ability to conduct water decreases with depth. In some cases, this means that water "pile-up" in local areas of the ground, this causes the effect that some of the water will move lateral towards the stream, and this is what is known as shallow subsurface stormflow.

The water that is not part of the subsurface flow will eventually enter the groundwater basin.

## Infiltration

Infiltration of precipitation through the top soil layer are implemented in CMF with different methods; one of them are the Green-Ampt equation, another is called Simple Infiltration (Philip Kraft 2017). Simple Infiltration method will be used as it is easier for the ODE solver to solve and gives less numerical problems than the Green-Ampt. Simple Infiltration can be expressed with:

$$q_{inf} = (1 - e_{sat}(W, W_0)) \cdot q_{inf,pot} \quad (19)$$

Where  $q_{inf}$  is the effective infiltration from the ground surface to the first soil layer.  $q_{inf}$  is in  $m^3/day$ .

$e_{sat}$  is the saturation excess, which is defined between 0 and 1 as not saturated and fully saturated respectively.  $e_{sat}$  can be calculated as:

$$e_{sat}(W, W_0) = \frac{1}{1 + e^{-0.2 \cdot (W-W_0) \cdot (1-W_0)}} \quad (20)$$

Where W is the wetness of the soil layer and it is unit less.

$W_0$  is a parameter signifying the wetness where 50% of the layer is saturated.

$q_{inf,pot}$  is the potential infiltration, determined by the incoming fluxes and it is limited by the saturated conductivity in the soil layer. It is given by:

$$q_{inf,pot} = \min(q_{in}, K_{sat} \cdot A) \quad (21)$$

Where  $q_{in}$  is the sum of incoming fluxes in  $m^3/day$ .

$K_s$  is the saturated conductivity in  $m/day$ .

A is surface area of the cell in  $m^2$ .

## Retention Curve

To calculate the diffusivity of the soil layers CMF uses the Van Genuchten-Mualem Retention Curve:

$$K(W) = K_{sat} \cdot \sqrt{W} \cdot \left(1 - \left(1 - W^{\frac{1}{m}}\right)^m\right)^2 \quad (23)$$

$$W(\theta) = \frac{\theta - \theta_r}{\theta_s - \theta_r} \quad (27)$$

$$m = 1 - \frac{1}{n} \quad (28)$$

$$\psi(W) = 0.01 \cdot \left(1 - W^{\frac{1}{m}}\right)^{\frac{1}{n}} \quad (24)$$

Where K is the conductivity in  $m/day$ .

W is the wetness (volume of soil water per volume of pores).

m is the Van Genuchten m, which is unitless.

n is the shape parameter of the retention curve.

$\psi(W)$  is the matric potential in  $mH_2O$  at wetness W (Philip Kraft 2017).

0.01 is the conversion factor between m and cm.

## Water Flux between Soil Layers

To calculate the horizontal water flux between two soil layers, Richards equation can be used:

$$q_{hor} = \frac{\Delta\psi_{tot}}{d} \cdot K(\theta) \cdot A \quad (25)$$

$$\psi_{tot} = \psi_M(\theta) + h \quad (26)$$

Where  $q_{hor}$  is the flow in  $m^3/day$ .

$\Delta\psi_{tot}$  is the difference of the total water potentials of the two soil layers in m.

d is the distance between the two soil layers in m.

$K(\theta)$  is the geometric mean conductivity in m/day.

A is cross-sectional area of the flux in  $m^2$ .

$\psi_M$  is the matrix potential in m.

h is the height of the soil layer above sea level in m (Philip Kraft 2017).

For computing the lateral flux between two soil layers CMF has implemented several methods: Richards<sub>lateral</sub>, Darcy, Topographic-Gradient-Darcy and Darcy-Kinematic.

Richards<sub>lateral</sub> and the Darcy method is more accurate as they are using the moisture head to calculate the distance gradient between soil layers, which leads to a higher computational time complexity.

Topographic-Gradient-Darcy and Darcy-Kinematic uses the topographic elevation of the cells to compute the distance gradient. This simplifies the solving; however, it is only valid in cases where the water table is approximately parallel to the ground surface.

Only the Darcy-Kinematic is used for this thesis, as it is assumed that the water table is parallel to the ground surface in the case study. The lateral flux can be calculated as:

Where  $q_{lat}$  is the lateral flow in  $m^3/day$ .

$$q_{lat} = \frac{\Delta z_{surf}}{d} \cdot K(\theta) \cdot A_{cross} \quad (29)$$

$\Delta z_{surface}$  is the difference in distance below the surface between the two layers in m.

d is the distance between the two layers in m.

$K(\theta)$  is the harmonic average of the unsaturated conductivity in m/day given the retention curve.

$A_{cross}$  is the cross-sectional area between the two layers in  $m^2$ .

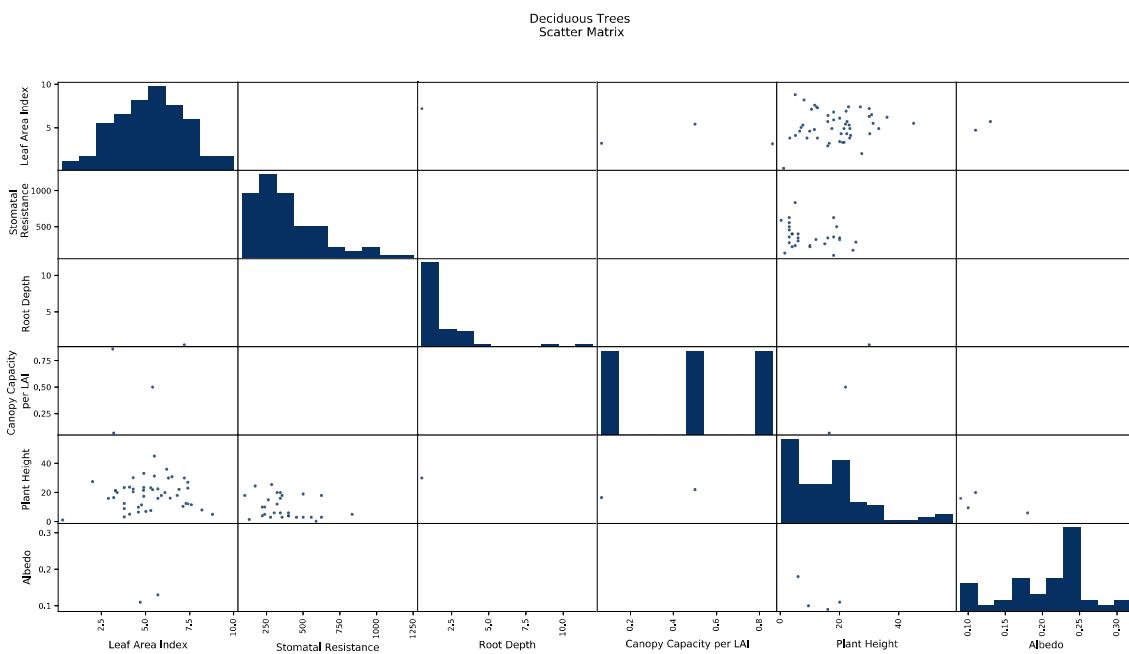


Figure 3.4 - Deciduous Trees - Scatter Matrix

## Soil Model Data

### Soil Data

In order to obtain data on soil properties, the European Soil Database v2.0 will be used (Panagos et al. 2012). It contains data on soil properties throughout whole Europe and have provided a way to translate these data to the coefficients of the Van Genuchten-Mualem curve.

### Vegetation Data

For input to CMF several plant parameters are needed, and for these values the PlaPaDa will be used (Breuer L and Frede 2003). The database contains multiple parameters

	Leaf Area Index	Stomatal Conductance	Root Depth	Interception	Plant Height	Albedo
Count	91	87	64	33	95	43
Mean	5,298	3,747	1,882	0,999	16,275	0,206
StD	1,976	2,431	1,816	0,703	13,069	0,053
Min	0,3	0,8	0,5	0,2	0,25	0,09
25%	3,8	2	1,075	0,51	5,55	0,18
50%	5,4	3,05	1,4	0,8	16	0,22
75%	6,5	4,5	1,85	1,27	22,3	0,24
Max	10	11,71	12	2,7	58	0,32

Table 3.1 - Descriptive statistics for the deciduous tree data.

collected from an extensive literature review. As the data originates from several sources the data is not very consistent. Therefore, a data processing was needed. The data are subdivided into the following landcover categories: understory, crops, coniferous, deciduous, pasture, shrubs and soil. For this thesis only deciduous is used.

## Deciduous Trees

To get an overview of the data available for the deciduous category a scatter matrix and a table that presents the descriptive statistics of the data. These can be seen in Figure 3.4 and Table 3.1 respectively. The scatter matrix clearly shows a lack of consistent data therefore, it was decided to produce a synthetic deciduous tree type. As the plant height is where most of the other parameters correlate to, it has been chosen as the basis. By using least squares to compute the linear relation between the different parameters and

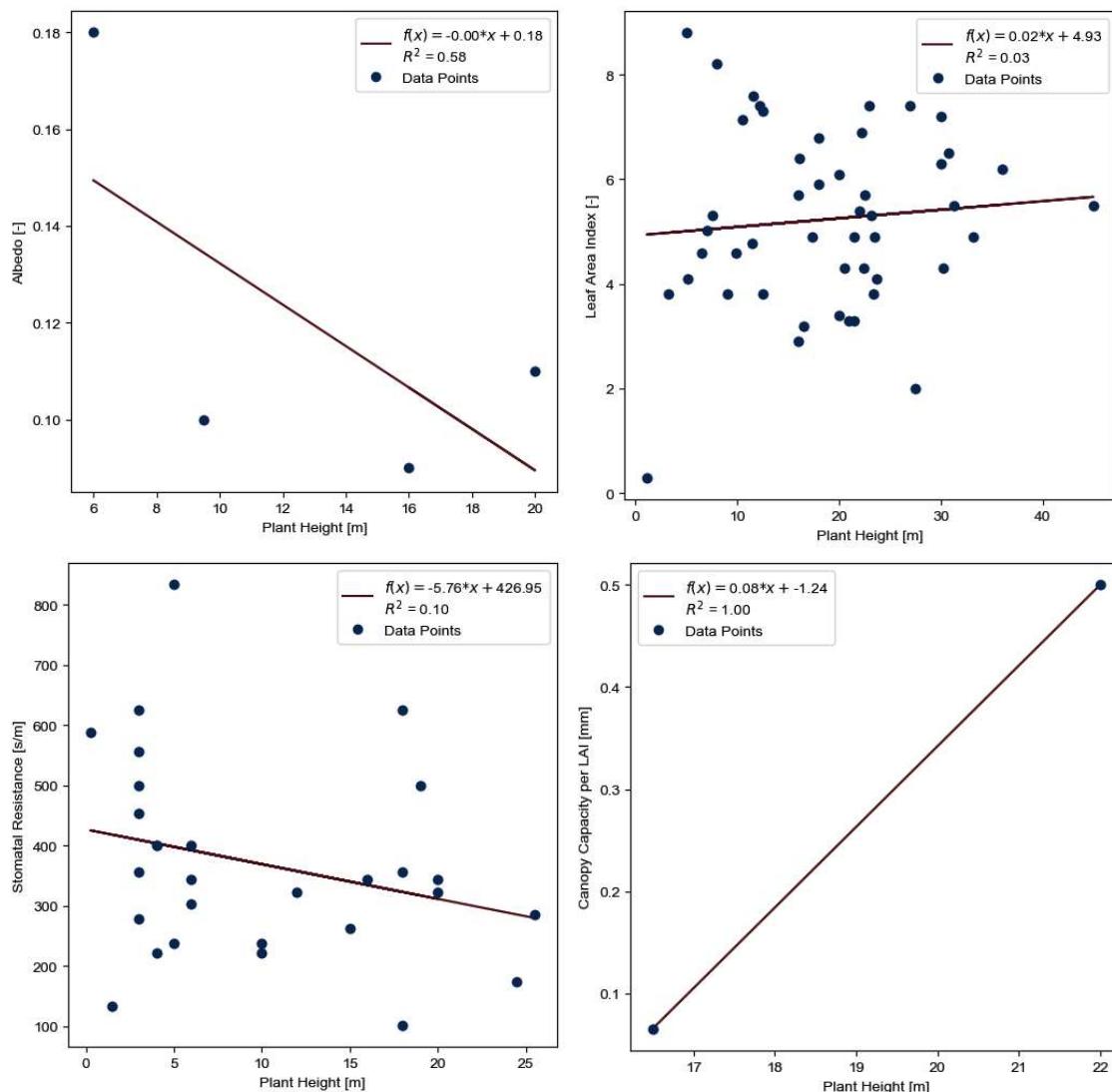


Figure 3.5 - Albedo, Leaf Area Index, Stomatal Resistance and Canopy Capacity plotted against Plant Height.

---

the plant height. The correlations and coefficient of correlation can be seen in Figure 3.5. The only parameter, which the least squares could not be performed on, was the root depth, as there is only one data point correlated with plant height. Furthermore, as most of the root data have a depth of less than two meters, a constant root depth, corresponding to the mean of the root depth data, will be used. A function was developed so that it is possible to compute the parameters for the synthetic tree by inputting the tree height. The function can be found in the documentation for the Component Classes.

As can be seen in Figure 3.5 the coefficient of correlation in all cases are quite bad. As of now, it has not been possible to collect a quality dataset. The purpose is not to investigate and collect vegetation data but use the available data to develop a model. In future development better data is however needed.

## 3.3 - MEAN RADIANT TEMPERATURE

For computing the **Mean Radiant Temperature** (MRT), the Ladybug Tools MRT Calculator will be used. It is a Grasshopper component, which uses the following function to compute the MRT:

$$MRT = \left( \sum_{i=0}^i VF_i \cdot T_i^4 \right)^{\frac{1}{4}} \quad (30)$$

Where VF is the view factor to a surface from the given point.

T is the temperature of the surface in °C.

The view factors will be computed by the Ladybug Tools Surface View Analysis component. It makes use of ray tracing to compute the view factors.

### Soil Temperature

EnergyPlus can calculate the hourly temperature of the soil surface. These values are used for the comfort model as input to the mean radiant temperature. The model EnergyPlus uses to compute the soil temperatures is called the Xing-model (Xing and Spitler 2017).

### Sky Temperature

In order to adjust for the sky temperature, the SolarCal model is used (Arens et al. 2015). It gives the possibility to calculate a  $\Delta MRT$ , which can be added to the MRT, from the ground surfaces and canopies, and thereby adjust for the sky radiation.

$$\Delta MRT = \frac{f_{eff} \cdot h_r \cdot T_a + ERF}{f_{eff} \cdot h_r} \quad (31)$$

Where  $f_{eff}$  is the fraction of the body surface receiving radiation.  $f_{eff}$  is 0.696 for a seated person and 0.725 for a standing person.

$h_r$  is the radiation heat transfer coefficient in W/m<sup>2</sup>K.

$T_a$  is the air temperature in °C

ERF is the effective radiant field, which describes the total radiant energy flux the human body is exposed to. ERF can be expressed as:

$$ERF = \left( \frac{1}{2} \cdot f_{eff} \cdot f_{svf} \cdot (I_{diff} + I_{TH} \cdot R_{floor}) + A_p \cdot f_{bes} \cdot \frac{I_{dir}}{A_D} \right) \cdot \frac{\alpha_{SW}}{\alpha_{LW}} \quad (32)$$

---

Where  $f_{svf}$  is the sky view factor and is unitless.

$I_{diff}$  is the diffuse sky radiation in  $\text{W/m}^2$ .

$I_{TH}$  is the global horizontal radiation in  $\text{W/m}^2$ .

$R_{floor}$  is the reflectivity from the ground. It is assumed to be 0.25.

$A_p$  and  $A_D$  is geometry coefficients of the human body, which are calculated based on sun altitude and azimuth.

$f_{bes}$  is 1 or 0 indicating if the person is exposed to direct sun or not.

$I_{dir}$  is the direct radiation in  $\text{W/m}^2$ .

The  $\alpha$  values are referring to the absorptivity and reflectivity of the clothing.

## 3.4 - COMFORT METRICS

### Introduction

On the contrary to indoor environment there is no European Standard regulating the calculation of outdoor thermal comfort around a building. Where the indoor environment has a well-defined method – PMV/PPD – standardized in (Dansk Standard 2007), there is more than 23 metrics to evaluate the outdoor thermal environment (Coccollo et al. 2016). The authors state that:

*“A perfect model or software to quantify the outdoor human comfort does not exist: the user need to understand the basic equations under each model and choose the best one according to their research’s needs.”*

It is therefore needed to investigate which comfort model would be best suited for this thesis. (Coccollo et al. 2016) states that:

*“(...) PET is the most widely used model, already applied and compared with on-site monitoring in many locations around the world. UTCI, which is based on contemporary science, is the only model applied to all climates, including polar, and to all scales, from micro to macro. (...) OUT\_SET\* and SET are well validated for temperate climates.”*

Physiologically Equivalent Temperature (PET), Outdoor Effective Temperature (OUT\_SET) and Universal Thermal Climate Index (UTCI) were therefore chosen for further investigation as they are well validated and have worldwide application – in case of PET and UTCI – where OUT\_SET is applicable for temperate climates such as Denmark’s.

### Standard Effective Temperature

Standard Effective Temperature (SET\*) is defined as the temperature in a standard isothermal environment with a relative humidity of 50%. A person in the environment, while wearing activity adjusted clothing, has the same skin wetness and heat exchange as the actual environment. To calculate the Standard Environment, the model includes functions for Standard Operative Temperature ( $T_{SO}$ ), Standard Operative Vapour Pressure ( $P_{SO}$ ) and a model to relate activity (MET) to clothing values (clo) and convective heat transfer caused by changes to air speeds around the body as a function of said activity.

### Outdoor Effective Temperature

To adopt SET\* into a model applicable for outdoor conditions one must incorporate the outdoor radiant environment, therefore Outdoor Effective Temperature (OUT\_SET) have been developed. This is done by calculation of a new mean radiant temperature:  $T_{MRT}$ :

$$T_{MRT} = \left( \left( \frac{\epsilon_{SKY}}{2} + \frac{\epsilon_{GND}}{2} \right) \cdot T_{SKY}^4 + \frac{f_p \cdot (1 - \alpha_{cl}) \cdot S \downarrow}{F_{EFF} \cdot \sigma} + \frac{(1 - \alpha_{cl}) \cdot (D \downarrow + \alpha_{GND} \cdot (S \downarrow + D \downarrow))}{\sigma} \right) \quad (33)$$

$\epsilon_{GND}$  is the ground emissivity and is unitless.

$T_{SKY}$  is the sky temperature in K.

$f_p$  is the body's projected area factor and is unitless.

$\alpha_{cl}$  is the albedo of the clothing and is unitless.

$S_\downarrow$  is the direct radiation in W/m<sup>2</sup>.

$F_{EFF}$  is the effective radiation area factor and is unitless.

$\sigma$  is the Stefan Boltzmann constant  $5.67 \cdot 10^{-8} \text{ W}/(\text{m}^2\text{K}^4)$ .

$D_\downarrow$  and the diffuse radiation in W/m<sup>2</sup>.

$\epsilon_{SKY}$  is the sky emissivity and is calculated as a clear sky:

$$\epsilon_{SKY} = 0.7 + 5.95 \cdot 10^{-5} \cdot e_a \cdot e^{\frac{1500}{T_a}} \quad (34)$$

$e_a$  is the partial vapor pressure in hPa.

$T_a$  is the ambient air temperature in K.

OUT\_SET is only applicable in an open field context and not in an urban environment as it does only include radiation exchange with the sky and not any surrounding buildings.  
(Pickup and de Dear 2000)

## Physiologically Equivalent Temperature

The Physiological Equivalent Temperature (PET) is defined as the temperature, independent of location, at which the human heat balance is equal to those of a typical indoor setting. The typical indoor setting is defined as a person having a work metabolism of 80W on top of the basic metabolism, wearing 0.9 clo and with equal ambient air

PET (°C)	Thermal Sensitivity	Physiological Stress
Below +4	Very cold	Extreme cold stress
+4 to +8	Cold	Strong cold stress
+8 to +13	Cool	Moderate cold stress
+13 to +18	Slightly cool	Slight cold stress
+18 to +23	Neutral (comfortable)	No thermal stress
+23 to +29	Slightly warm	Slight heat stress
+29 to 35	Warm	Moderate heat stress
+35 to +41	Hot	Strong heat stress
Above +41	Very hot	Extreme heat stress

Table 3.2 - PET assesment scale

temperature and mean radiant temperature, air velocity of 0.1m/s and a water vapor pressure of 12hPa.

For the human heat balance the Munich Energy-balance Model for Individuals (MEMI) are used. The basic heat balance that must be solved are:

$$M + W + R + C + E_D + E_{Re} + E_{Sw} + S = 0 \quad (35)$$

M is the metabolic rate.

W is the physical work.

R is the total body radiation.

C is the convective heat flow.

$E_D$  is the latent heat flow to evaporate water into water vapor diffusing through the skin.

$E_{Re}$  is the sum of heat flows for heating and humidifying the inspired air.

$E_{Sw}$  is the heat flow due to evaporation of sweat.

S is the storage heat flow for heating or cooling the body mass.

All variables are given in W.

(Matzarakis and Mayer 1996) correlates PET to thermal sensitivity and physiologic stress. See Table 3.2.

## Modified Physiologically Equivalent Temperature

Further development has gone into the Physiological Equivalent Temperature which has led to the Modified Physiological Equivalent Temperature (mPET) (Chen and Matzarakis 2014). It proposes a different way to simulate the human thermoregulation, called the YCC-thermoregulation model, and introduces an improved clothing model. The following equation is proposed for calculation of the heat exchanging fluxes for each segment of the body.

$$k \cdot \left( \frac{\delta^2 T}{\delta r^2} + \frac{\omega}{r} \cdot \frac{\delta T}{\delta r} \right) + q_m + \rho_{bl} \cdot \omega_{bl} \cdot c_{bl} \cdot (T_{bl,a} - T) = \rho \cdot c \cdot \frac{\delta T}{\delta t} \quad (36)$$

k is conductive coefficient of tissue.

T is temperature of tissue.

r is radius of tissue.

$\omega$  is geometric coefficient of tissue. If tissue form is cylinder,  $\omega$  is equal to 1. If tissue form is sphere,  $\omega$  is equal to 2.

$q_m$  is metabolic rate of tissue.

$\rho_{bl}$  is blood density.

$\omega_{bl}$  is perfusion rate of blood.

$c_{bl}$  is heat capacity of blood.

$T_{bl,a}$  is temperature of arterial blood.

$\rho$  is tissue density.

$t$  is time step.

The thermoregulation is applied onto a simple two-cylindrical-body-model. The proposed clothing model are capable of modeling heat transport in multi-layer clothing.

Sensitivity analyses to air temperature, mean radiant temperature, wind speed, relative humidity and clothing done by (Chen and Matzarakis 2014) shows that differences between PET and mPET are mostly significant at sub-zero air temperatures. Most software (to the knowledge of the author) have also only implemented PET – among these is Ladybug Tools (Sadeghipour Roudsari and Pak 2013).

## Universal Thermal Climate Index

“The UTCI is defined as the air temperature ( $T_a$ ) of the reference condition causing the same model response as actual conditions.” (Błazejczyk et al. 2013)

This can be rewritten as:

$$UTCI = T_a + \text{Offset}(T_a, T_{MRT}, U_{wind}, \rho_{vapour}) \quad (37)$$

UTCI (°C)	Physiological Stress
Above +46	Extreme heat stress
+38 to +46	Very strong heat stress
+32 to +38	Strong heat stress
+26 to +32	Moderate heat stress
+18 to +26	Thermal comfort zone
+9 to +18	No thermal stress
0 to +9	Slight cold stress
-13 to 0	Moderate cold stress
-27 to -13	Strong cold stress
-40 to -27	Very strong cold stress
Below -40	Extreme cold stress

Table 3.3 - UTCI assessment scale

$T_a$  is ambient air temperature.

$T_{MRT}$  is mean radiant temperature.

$U_{wind}$  is wind speed.

$p_{vapour}$  is water vapour pressure.

The reference condition is defined as an environment with a wind speed ( $U_{wind}$ ) of 0.5 m/s at 10m height (approximately 0.3 m/s at 1.1m height), a mean radiant temperature ( $T_{MRT}$ ) that equals the ambient air temperature ( $T_a$ ), a vapour pressure ( $p_{vapour}$ ), which corresponds to a relative humidity of 50%, but having a maximum value of 20hPa.

It is furthermore assumed that persons have a metabolic rate of 2.3 MET corresponding to a walking speed of 1.1 m/s and wearing clothing with a thermal resistance modeled by the UTCI-clothing model (Havenith et al. 2012). The paper also suggests a third order regression formula, that can be used to approximate clothing values as a function of air temperatures:

$$I_{cl} = 1.372 - 0.01866 \cdot T_a - 0.0004849 \cdot T_a^2 - 0.000009333 \cdot T_a^3 \quad (38)$$

The expression can be used for clothing values from 0.1 clo to 1.43 clo.

The model uses the UTCI-Fiala model (Fiala et al. 2012) as it's thermoregulation model. It models an average person of 73.4kg and with a surface body area of 1.85m<sup>2</sup>. The body is represented by 12 spherical or cylindrical parts and contains a total of 187 tissue nodes. It uses the same heat transfer equation as the proposed one in mPET Equation 36. The model also includes a model to predict thermoregulatory reactions of the central nervous system, e.g. shivering or sweating.

The model computes a UTCI temperature which can be categorized in terms of thermal stress. The assessment scale can be seen in Table 3.3.

As the complexity of the UTCI is quite large so is the computational power and time needed to perform the analysis. Therefore (Bröde et al. 2012) proposed methods for approximating the UTCI. They came up with two methods: a look-up table of pre-calculated UTCI values and a 6<sup>th</sup> order regression function. They state that the computing speed (number of calculations per second) of the three methods are as following:

Actual UTCI model: 1/s

Look-up table: 100/s

Regression function: >100 000/s

With the higher number of computation per second comes with a cost of lower accuracy for the regression function. On the other hand, the 4<sup>th</sup> dimensional ( $T_a$ ,  $T_{MRT}$ ,  $U_{wind}$ ,  $p_{vapour}$ ) look-up table with 100 steps in each dimension, would require 200Mb of storage space.

Because of the higher number of calculations per second the regression function is most commonly used. The regression function is only valid within the following bounds:

Ambient air temperature: -50 to +50°C

Mean radiant temperature: 50°C below air temperature and 70°C above air temperature

Wind speeds at 10m height: 0.5 to 17m/s

Vapour pressure: 0hPa to 45hPa

This limits UTCI in some extreme cases, but the model covers the most scenarios.

## Comfort Metric Comparison

In order to compare the three metrics and investigate their sensitivity to mean radiant temperature, relative humidity and wind speeds, a parameter investigation was carried out using Ladybug's comfort tools (Sadeghipour Roudsari and Pak 2013). The Grasshopper definition can be found in Appendix A. The parameter investigation has been done on the following background:

Air Temperature: -30 to 70°C in steps of 5°C.

Relative Humidity: 10 to 90% in steps of 5%.

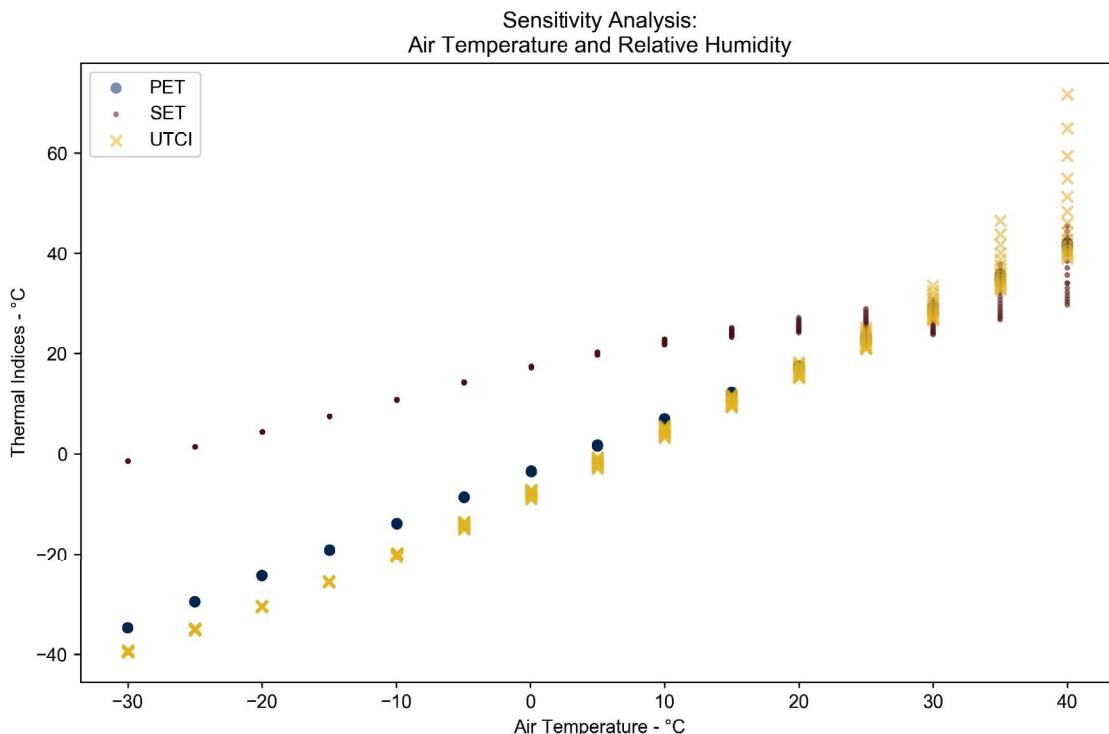


Figure 3.6 - Sensitivity to relative humidity as function of air temperature.

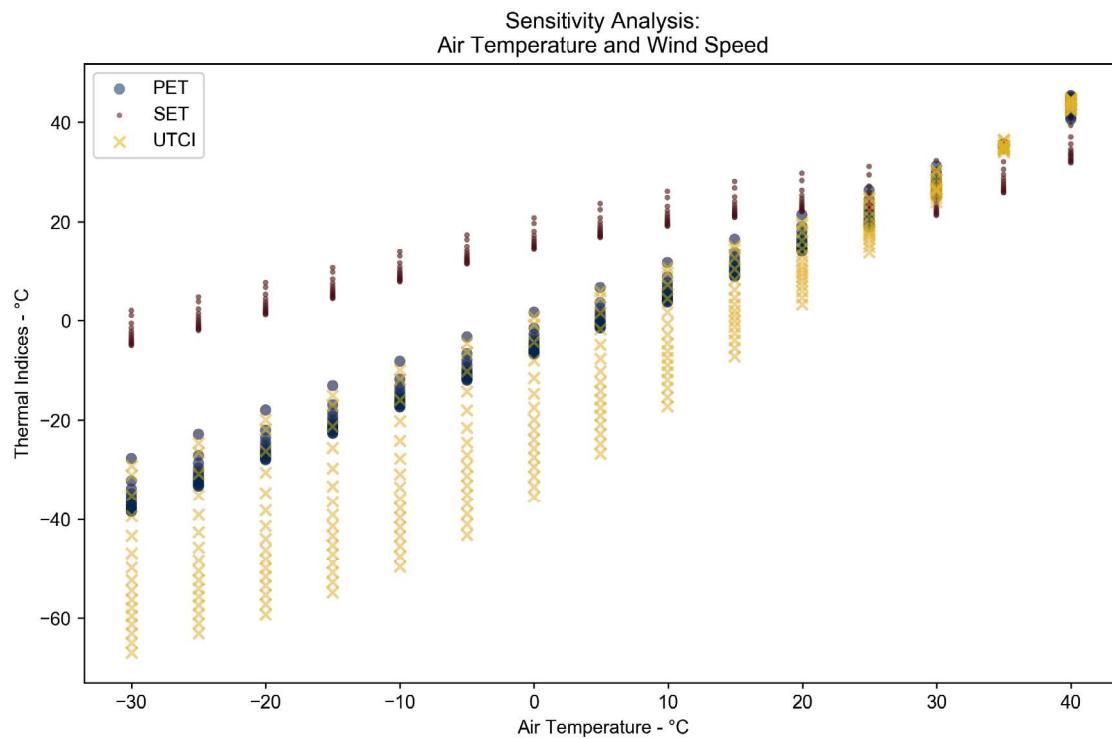


Figure 3.7 - Sensitivity to wind speed as function of air temperature.

Mean Radiant Temperature: -30 to 70°C in steps of 5°C.

Wind Speed: 0 to 15m/s in steps of 1m/s at 10m. Ladybug's Wind Speed Calculator was used to calculate wind speeds at 1.1m for PET and SET.

Clothing: 0.1 to 1.43clo calculated with the regression formula from Equation 38, as a function of air temperature. Only used for PET and SET.

MET: A metabolic rate of 2.3MET was used corresponding to a walking speed of 1.1m/s.

For PET the following body characteristics were used:

Age: 35 years

Gender: Average gender

Height: 175cm

Weight: 73.4kg

Position: Standing

Clothing Albedo: 0.37

Acclimated: True

Activity Duration: 480min

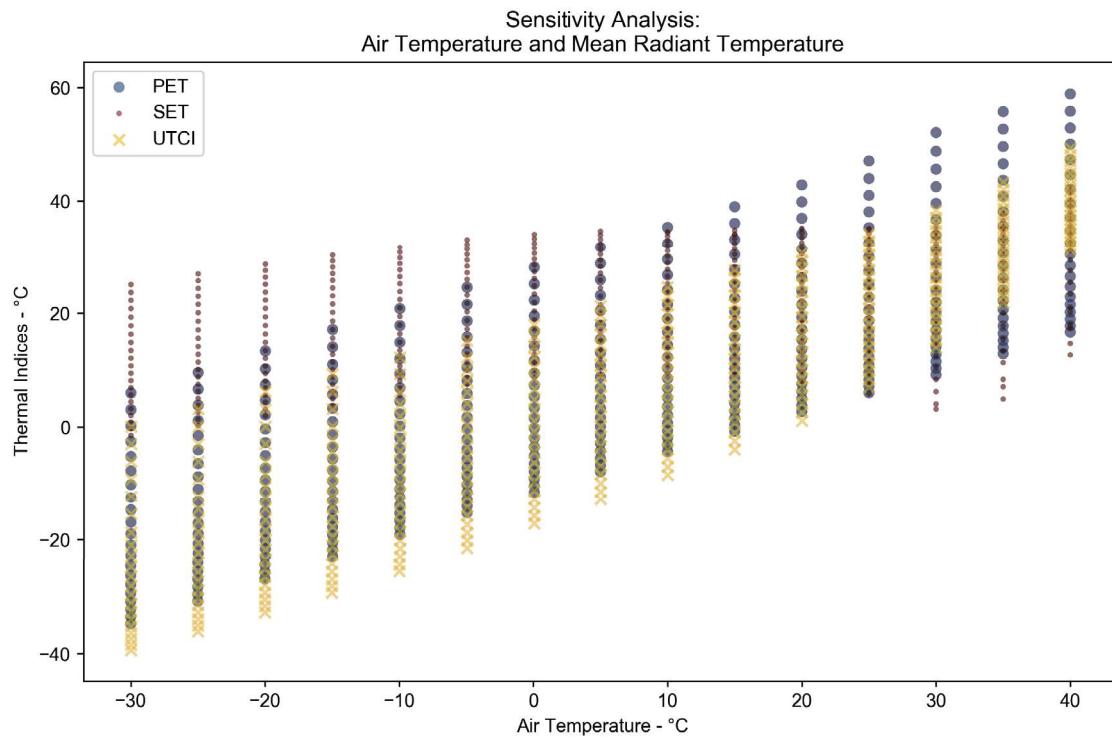


Figure 3.8 - Sensitivity to mean radiant temperature as function of air temperature.

Figure 3.6 shows the sensitivity to relative humidity as function of air temperature. Input variables were air temperature from -30 to 40°C, relative humidity from 10 to 90%, mean radiant temperature equal to air temperature, wind speed at 3m/s and clothing calculated with the regression formula from Equation 38 as a function of air temperature.

It shows that all three comfort metrics are only marginal sensitive to changes in relative humidity. UTCI is the most sensitive. It starts showing variations already at an air temperature of -10°C. At 40°C the variations span in range from 38.9 to 71.7°C for 10 and 90% relative humidity respectively. Both PET and SET are far less sensitive.

Figure 3.7 shows the sensitivity to wind speed as function of air temperature. Input variables were air temperature from -30 to 40°C, wind speed from 0 to 15m/s, relative humidity 50%, mean radiant temperature equal to air temperature and clothing calculated with the regression formula from Equation 38 as a function of air temperature.

Again, UTCI shows the largest sensitivity. UTCI seems to be biased to show a larger sensitivity towards wind speeds at lower temperatures, whereas both PET and SET have a more constant sensitivity independent of air temperature. Both PET and SET vary about 10°C dependent on wind speeds.

Figure 3.8 shows the sensitivity to mean radiant temperature as function of air temperature. Input variables were air temperature from -30 to 40°C, mean radiant temperature from -30 to 70°C, relative humidity at 50%, wind speed at 3m/s and clothing calculated with

---

the regression formula from Equation 38 as a function of air temperature.

All three metrics seems to be highly dependent on mean radiant temperature. Especially PET which shows the larges variations.

## Conclusion

*“A perfect model or software to quantify the outdoor human comfort does not exist: the user need to understand the basic equations under each model, and choose the best one according to their research’s needs.”* (Coccolo et al. 2016)

As cited above there is currently no comfort metric that captures the complex nature of outdoor human comfort. It is therefore necessary to make a choice between the comfort metrics – in this case OUT\_SET\*, PET and UTCI.

All three has been chosen because of their applicability to the temperate northern European climate, all having advantages and disadvantages. Our objective in this thesis is to evaluate the outdoor comfort on a small urban scale around buildings. For that the OUT\_SET\* is not applicable as it does not consider the radiational exchanges from objects other than the sky.

Furthermore, as this thesis objective is to investigate the variations in outdoor human comfort – caused by changes in mean radiant temperature, ambient air temperature, relative humidity and wind speed by introducing variant amounts of vegetation – both PET and UCTI seems feasible. UTCI’s simplified regression formula, that it is applicable to the entire world and the fact that it does not need a detailed wind speed calculation leads to that this author’s opinion that UTCI is the most suited comfort metric for this thesis.

## 3.5 - SOFTWARE

### Applied Software

The backbone of the software tools used are **Python** (Rossum, 1991). Python is used to couple all the different software tools together.

**Rhinoceros 3D** (Robert McNeel & Associates, 1980) – from here on named: Rhino – is a CAD software tool used as a geometry handler and visual displayer.

**Grasshopper** (David Rutten, 2007) are a plugin for Rhino which gives the user the possibility to manipulate geometry with its visual programming. The visual programming makes it possible to use Grasshopper as a parametric design tool. Grasshopper also functions as a platform for interaction between the different software tools used. Essentially all geometry is exchanged through Grasshopper.

**Ladybug Tools** (Sadeghipour Roudsari & Pak, 2013) couples Grasshopper with EnergyPlus, Radiance and other environmental software tools. It also includes tools to compute outdoor comfort.

**EnergyPlus** (Illinois & The Regents of the University of California, 1996) is an open source whole building energy simulation tool. EnergyPlus is a console-based program that reads and writes text files.

**Radiance** (Ward, 1985) is an open source ray tracing software tool.

**Catchment Modelling Framework** (Kraft, Vach, Frede, & Breuer, 2011) is a Python programming library for hydrological models.

### Software Connections

To better understand the connections between the different software Figure 3.9 is provided. As stated before Grasshopper functions as a platform information exchange, both geometry and numerical data, by bringing the components from the different

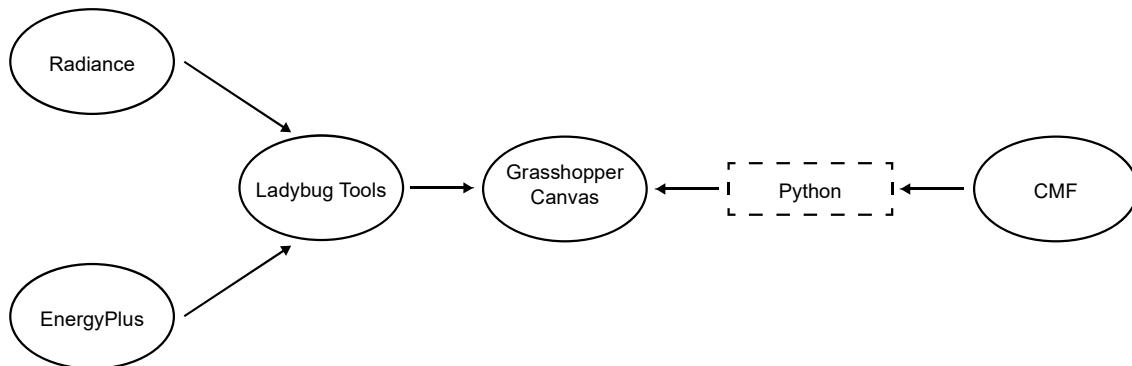


Figure 3.9 - Diagram depicting the connections between the software.

---

tools onto the Grasshopper Canvas. The software which is missing the connection to Grasshopper is CMF. That connection will be written in Python. When the connection is established it is possible for CMF to share information with the other software tools and get the geometry. Furthermore, when a connection is established to Grasshopper it is possible to create a workflow to compute outdoor comfort.

# 4 - DEVELOPED MODELS

## 4.1 - CONCEPTUAL APPROACH

In order to evaluate outdoor thermal comfort at a given location, four environmental values must be known as well as information about the person at the location. The four values are: air speed, air temperature, relative humidity (of the air) and mean radiant temperature. From these four values together with the human parameters the comfort can be computed.

The popular procedure is to collect 3D geometry and measured weather data from a weather station nearby and use the data for wind, energy and radiation models (Mackey et al. 2015). These types of models assume that air temperature and relative humidity are not affected by the surrounding local context. (Mackey et al. 2015) only takes citywide adjustments for the air temperature into account. Envi-Met (Bruse 2014) is a model, which takes evapotranspiration from the surrounding local context into account, but have other disadvantages as stated in (Acero and Herranz-Pascual 2015). They point out that

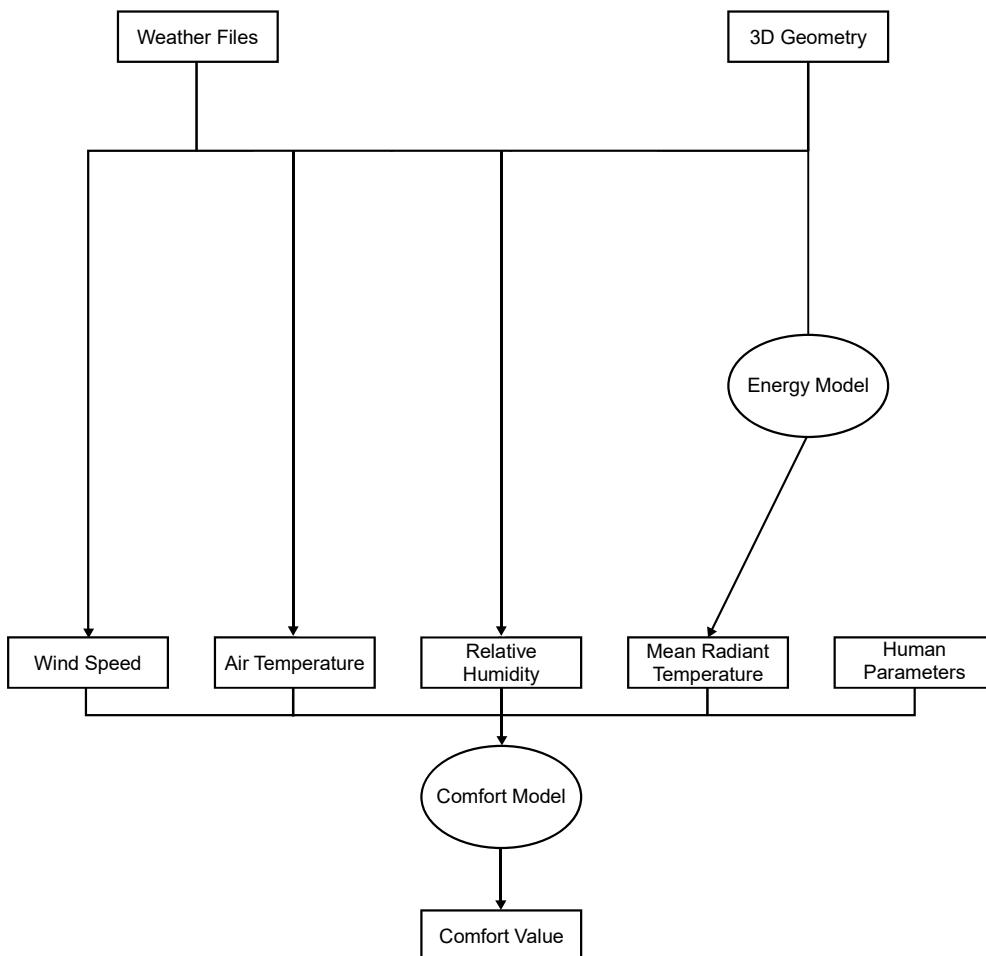


Figure 4.1 - Current model network for outdoor comfort modelling. As described in (Mackey, Galanos, Norford, Roudsari, & Bhd, 2015)

only changes in temperature and relative humidity is possible to simulate. Other variables such as cloudiness, wind speed and direction, remains constant. The model does not include precipitation either and the radiation calculation is inaccurate. The CFD model proposed by (Manickathan et al. 2018) benefits from higher accuracy but is difficult to integrate in a design process. It is computationally expensive in its current steady state form and not well suited for making annual simulations. Therefore, a main objective of this thesis is to develop a simulation tool, which can be used as an alternative.

The tool should be able to compute the changes in air temperature and relative humidity, which occurs around vegetation and water. It is the goal that the tool should be modular and able to adapt to distinctive design conditions. Therefore, the tool will be developed as a plugin for Grasshopper. By implementing the tool into Grasshopper, it becomes possible to integrate the analysis in design workflow with already existing plugins especially Ladybug Tools plugin (Sadeghipour Roudsari and Pak 2013). Finally, should the tool be open-source and free to use.

	Ladybug Tools	Envi-Met	CFD	Livestock <sup>1</sup>
Accuracy	Low	Low	High	Medium
Computation Expense	Medium	High	High	Medium
Includes Evapotranspiration	No	Yes	Yes	Yes
Annual Simulations	Yes	Yes	No <sup>2</sup>	Yes
Available in Grasshopper	Yes	Yes <sup>3</sup>	No	Yes

Table 4.1 - Comparison of included features in the different methods. 1) Livestock is the method proposed by this thesis. 2) An annual simulation would be possible, but be practical impossible due to computation costs. 3) Envi-Met is available in Grasshopper through Ladybug Tool as a beta-version.

The aim of the tool is to bridge the gaps between some of the current software tools when computing outdoor comfort. This will be done by developing two models. The Soil Model will be developed to compute the evapotranspiration from the ground and vegetation. The Atmosphere Model will convert the evapotranspiration into a changed temperature and relative humidity of the air.

For the rest of the inputs of environmental parameters the following models will be used:

- The air speed will be taken from the weather file, which consists of measured values at 10m height. The mentioned values can be translated into values for the wished height above the ground.
- Mean radiant temperature will be computed through two models. A radiation model to compute the view factors from the different surfaces and an energy model to compute the temperatures on ground surface.
- The human parameters are a collection of values describing a “test” person.

In Figure 4.2 an overview of the, here, shortly described models can be seen.

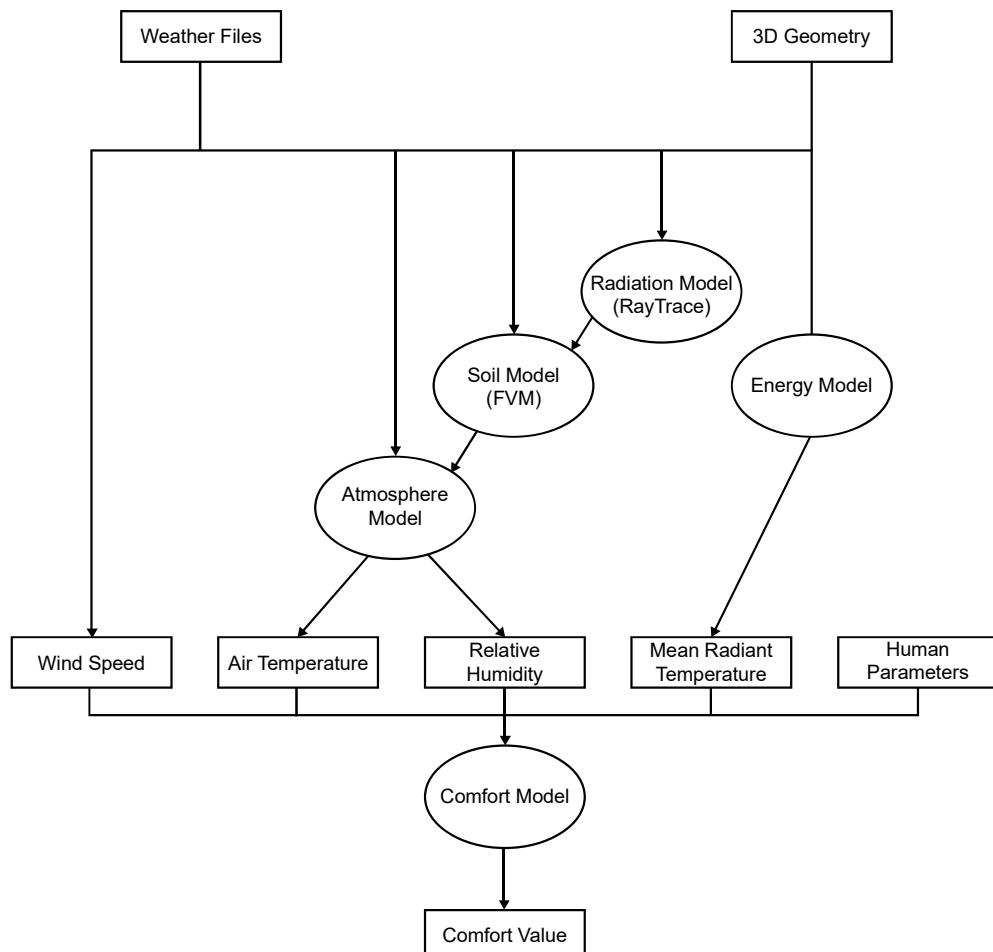


Figure 4.2 - Overview of the collection of models used in this thesis. The ellipses represents models and the black rectangles are values or parameters.

---

## Developed Models

A plugin for Grasshopper has been developed as a framework for the two proposed models. The plugin is named Livestock. The models will be built into Grasshopper components, which will be bundled with the plugin.

Together with developing models that can investigate the impact of vegetation on the microclimate, it is also an aim to demonstrate a workflow of how the models can be used to compute outdoor comfort. This workflow will be done in Grasshopper, which makes it possible to use already integrated plugins to calculate outdoor comfort.

This chapter provides an insight in the developed models and the methodology behind them:

- Section 4.2 will describe Livestock and the methodology behind it.
- Section 4.3 will describe the Atmosphere Model and how it works.
- Section 4.4 will describe how the Soil Model is coupled with CMF and the Livestock components through Python.

---

## 4.2 - LIVESTOCK

Livestock is the name of the library of components that has been developed. Livestock consists of a series of Grasshopper Python Script components and an underlying collection of Python scripts and a PyPI – **Python Package Index** - (Foundation 2017) package.

The components are divided into subcategories:

**0 | Miscellaneous.** The group Miscellaneous is for components that are meant for setting up a Grasshopper script. The group contains the following components:

- Livestock Python Executor
- Livestock SSH Connection
- Livestock Update
- Livestock Hour to Date

**1 | Geometry.** The group Geometry is for components doing generic geometry functions. The group contains the following components:

- Livestock Load Mesh
- Livestock Save Mesh

**2 | CMF.** This group contains all components related to working with CMF. The group contains the following components:

- Livestock CMF Ground
- Livestock CMF Outputs
- Livestock CMF Results
- Livestock CMF Retention Curve
- Livestock CMF Solve
- Livestock CMF Solver Settings
- Livestock CMF Surface Flux Result
- Livestock CMF Vegetation Properties
- Livestock CMF Weather
- Livestock Outlet
- Livestock Synthetic Tree

---

**3 | Air.** This group contains components that relates to the atmosphere model. The group has the following components:

- Livestock New Air Conditions
- Livestock Load Air Results

## Background

Each component is modelled as a placeholder with the Grasshopper Python Script for a Python class implementation of the component. This makes it possible to create and edit the components without having Grasshopper open, thus making it possible to use better Python editors than the build-in Grasshopper one – in this case PyCharm (JetBrains 2017) was used.

Besides easing the workflow, it also solves the problem of updating the components in the future. Some Grasshopper plug-ins such as Ladybug Tools (Sadeghipour Roudsari and Pak 2013), have written most of the source code directly in the Grasshopper Python Script components and every time an update is made or a bug is fixed the component has to be replaced with the new version, which is troublesome. By having the Grasshopper component as a placeholder, this can be avoided. The only time a Livestock component should be replaced is when new inputs or outputs are added to the component.

Livestock has a package on PyPI. The PyPI package is not a duplicate of the Python scripts imported into Grasshopper. The Grasshopper Python Script component runs IronPython (Hugunin and Viehland 2008), which is an implementation of Python for the .NET framework. IronPython comes as Python 2 and has been without development (Hugunin and Viehland 2008) for some time. Standard Python, from here on named CPython, is well maintained and is at the time of this writing on version 3.6, although a beta version of 3.7 has been released. CPython has undergone a lot development that IronPython still has to catch up to, including a whole bunch of new features. Furthermore, are the packages on PyPI mainly targeting CPython and many cannot be used in IronPython. This decreases the usability of the IronPython as a general-purpose implementation of Python. Specific CMF cannot be used with IronPython, therefore a CPython package and scripts were necessary. The CPython Livestock package – from now on mentioned as the Livestock package – is used whenever larger computations or CMF are needed.

## Documentation

As part of the thesis documentation for the code has been written. The documentation has been separated into two parts, corresponding the structure of the code. One part is for the IronPython/Grasshopper code and components and the second part is for the CPython package.

Documentation for Livestock, both the IronPython/Grasshopper and the PyPI package source code can be found here:

<http://ocni-dtu.github.io/>

Or in Appendix B and C.

## Grasshopper Components

The following sections is an in-depth explanation of the created Grasshopper components and their internal functions. All components are build up in a similar way, to act as the placeholder.

```
# Livestock Load Mesh Component
# Import statement:
from livestock.components.geometry import LoadMesh

# Body:
1) comp = LoadMesh(ghenv)
2) comp.config()
3) comp.run_checks(Filename, Load)
4) comp.run()

# Configure Outputs:
Mesh = comp.mesh
MeshData = comp.data
```

Above, the code inside the Livestock Load Mesh component is presented. The code consists of three parts: The import statement, the body and the outputs.

The import statement imports the component class object from the python scripts.

The body sets up the class object. All Grasshopper components have the same four lines:

- 1) The class object is created. A single input is given: ghenv, which is the Grasshopper Environment object.
- 2) The components configuration function is executed. This function controls that the component has the right name, inputs, outputs, description, version number and category.
- 3) The function run\_checks is executed. It is given the Grasshopper component's inputs as arguments, which vary from component to component. The inputs are then loaded into the class and checked.

- 4) The functionality of the component is executed with the function “run”.

The outputs section connects the results of the class object with the outputs of the Grasshopper component.

## GHComponent

Many functions in the component classes are the same, therefore a Super Class – GH-Component – was created to collect these. This way all component classes inherit the same functions. Below, the class of GHComponent is presented. For readability the code itself has been hidden, but the doc-strings are kept.

```
class GHComponent:
    def __init__(self, ghenv):
        pass

    def config_component(self, component_number):
        """
        Sets up the component, with the following steps:
        - Load component data
        - Generate component data
        - Generate outputs
        - Generate inputs
        :param component_number: integer with the component number
        """

    def add_warning(self, warning):
        """
        Adds a Grasshopper warning to the component.
        :param warning: Warning text.
        """

    def add_output_parameter(self, output_):
        """
        Adds an output to the Grasshopper component.
        :param output_: Output index.
        """

    def add_input_parameter(self, input_):
        """
        Adds an input to the Grasshopper component.
        :param input_: Input index.
        """

    def add_default_value(self, parameter, param_number):
        """
        Adds a default value to a parameter.
        :param parameter: Parameter to add default value to
        :param param_number: Parameter number
        :return: Parameter
        """

    def component_data(n):
        """
        Function that reads the grasshopper component list and returns the
        component data
        """
```

## Data Formats

To exchange data between software a data file format is needed. Two file formats for exchanging data will be used. The .txt format will be used when simple structured data needs to be exchanged. The file format XML (Extensible Markup Language) will be used when more structured data are needed. Both file formats are easy to read and write for most software, also Python. In .xml it is possible to give the stored data a chosen hieratic structure, which makes it easy to read again, therefore it is used for structured data. .txt files are simpler but lacks the hieratic structure. So, both file formats have their own use case: .xml for hieratic data and .txt for flat data. Geometry is saved as obj files. Obj files are in a simple file format and can be read by almost every software.

## Data Flow

The data flow for the CMF connection can be depicted as in Figure 4.3. In the Grasshopper environment data are collected and organized and saved as a .xml file for information or .obj file for geometry. Together with the data files a Python file is written, which contains instructions on how to process the data. Two things will happen:

- 1) Python will execute the instruction in the Python file and save the results as .xml, .txt and, or .obj files, thereafter Grasshopper will read them.
- 2) The instructions in the Python file is to send the files to another computer for processing. In that case two Python files are send from Grasshopper: one that tells Python to send on the files and one with instructions on how to process the files as in case 1. Python will establish a SSH connection (Secure Shell) to the remote computer, wherethrough the files will be send. The remote computer will execute the Python instruction file and the result files will be send back through the SSH connection, afterwards it will be closed. The local Python will then save the result files to the right place, where Grasshopper is able to read the files.

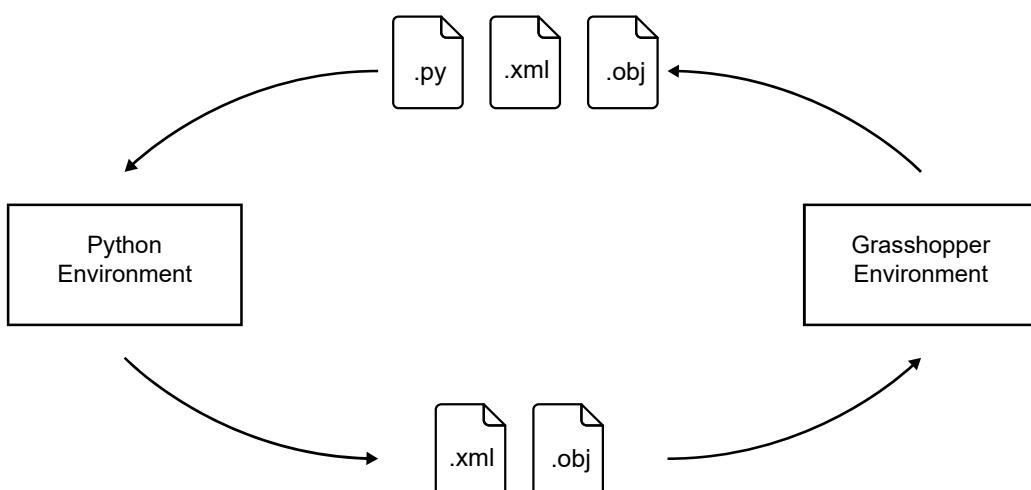


Figure 4.3 - Information exchange between the Grasshopper environment and the Python environment.

The reason for sending the files forward to remote processing is to reduce processing time, as most cases are quite computational heavy. Furthermore, it is freeing your work station, as it is not pinned down from the workload, while the computation is ongoing. This is desirable especially in a professional context.

## Grasshopper to CPython Link

In order to leave Grasshopper and utilize the full functionality of CPython, a method had to be developed. The method works by writing templates from the Grasshopper Python Script followed by spawning a CPython subprocess, which executes the template. To examine this method the component class of Livestock New Air Conditions are given as an example:

```
class NewAirConditions(GHComponent):
    1) def __init__(self, ghenv):
    1)     def check_inputs(self):
    1)         """
    1)             Checks inputs and raises a warning if an input is not the correct
    1)             type.
    1)         """
    1)     def config(self):
    1)         """
    1)             Generates the Grasshopper component.
    1)         """
    1)     def run_checks(self, mesh, evapotranspiration, temperature, relhum,
    1)                 wind_speed, boundary_height, investigation_height, cpus, folder,
    1)                 run):
    1)         """
    1)             Gathers the inputs and checks them.
    1)         """
    2)     def get_mesh_data(self):
    2)         """
    2)             Extracts the data needed from the mesh.
    2)         """
    3)     def write_files(self):
    3)         """
    3)             Write the files.
    3)         """
    4)     def do_case(self):
    4)         """
    4)             Runs the case. Spawns a subprocess to run either the local or ssh
    4)             template.
    4)         """
    4a)     template_to_run = self.folder + '/new_air_conditions_template.py'
    4b)     # Run template
    4c)     thread = subprocess.Popen([self.py_exe, template_to_run])
    4c)     thread.wait()
    4d)     thread.kill()
```

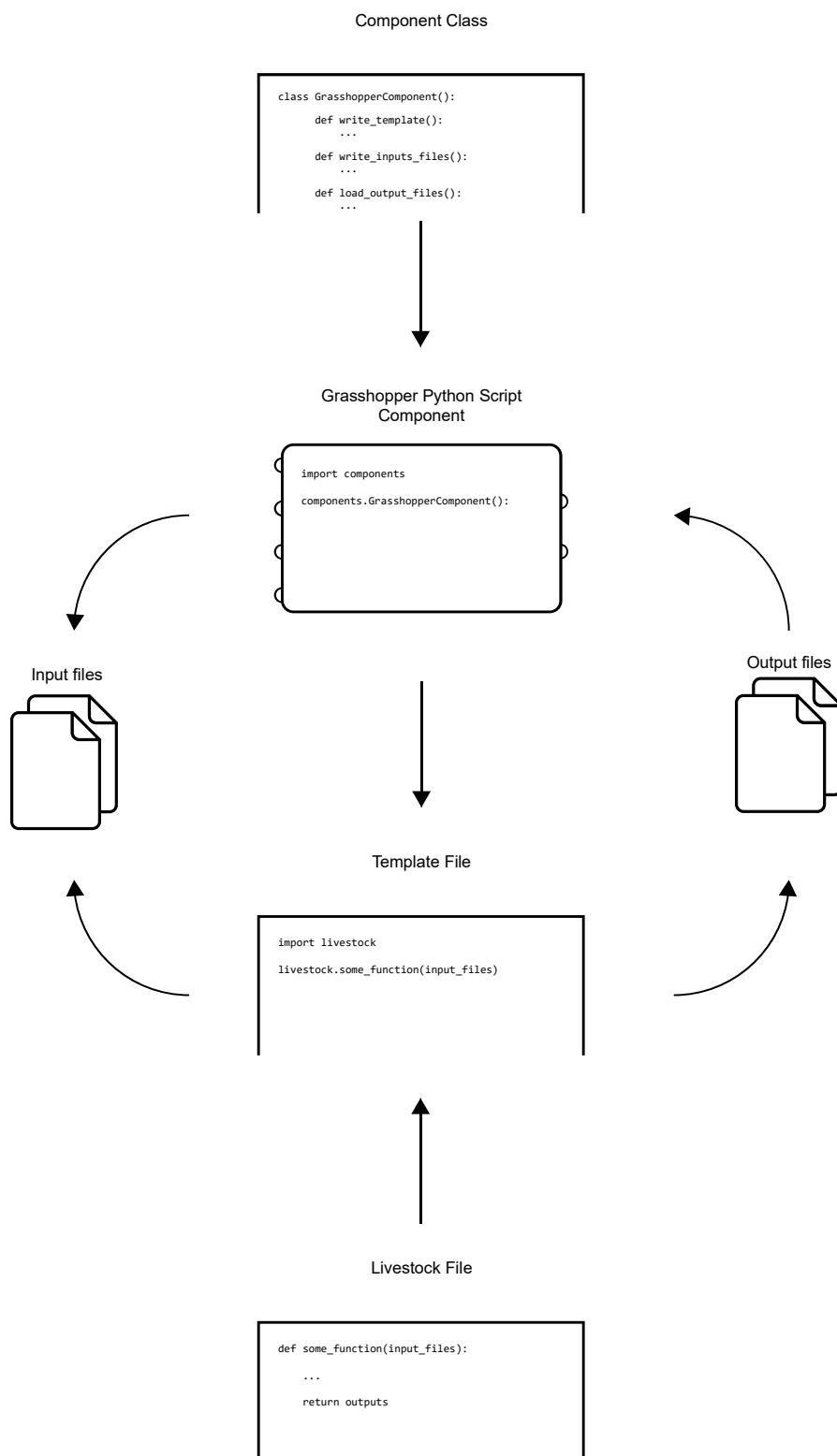


Figure 4.4 - Visualization of the Livestock Template Method

```

        return True

5)    def load_results(self):
        """
        Loads the results from the results files and adds them to
        self.results.
        """

6)    def run(self):
        """
        In case all the checks have passed and run is True the component
        runs.
        The following functions are run - in this order.
        get_mesh_data()
        write_files()
        do_case()
        load_results()
        """

```

Again, the code itself have been hidden, leaving only the doc-strings behind, besides from the function: `do_case()`.

- 1) Initialization of the component. `__init__()`, `check_inputs()`, `config()` and `run_checks()` sets up and configures the component.
- 2) `get_mesh_data()` gathers the wanted data from the mesh – in this case the mesh face areas.
- 3) The needed data for the template is written, including the template itself.
- 4) The CPython subprocess is spawned.
  - a) The location of the template file is defined.
  - b) The CPython subprocess is spawned. `self.py_exe` is the CPython path that Livestock Python Executor collects.
  - c) The component waits until the subprocces finishes.
  - d) `thread.wait()` should terminate the subprocess thread, but to be sure it is terminated, `thread.kill()` is called.
- 5) `load_results()` reads the results from the template.
- 6) `run()` calls the functions in the right order.

The template is just a Python file, which has a single function call:

```

1)    # Imports
        import livestock.air as la

2)    # Run function
        if __name__ == '__main__':
            la.new_temperature_and_relative_humidity(r''' + path + ''')

        # Announce that template finished and create out file
        print('Finished with template')

```

- 1) The `livestock.air` module is imported
- 2) `new_temperature_and_relative_humidity()` is executed.

As the template is executed as a CPython file it is possible to import the Livestock Package and all its functionalities. The `new_temperautre_and_relative_humidity()` will run and compute the results of the atmosphere model then write the results to files, thereafter it will terminate and `load_results()` of the Grasshopper component will collect the results.

## SSH Tunnel

The SSH functionality works like the template method. Livestock CMF Solver is the only component that is currently utilizing the SSH method. The component takes different inputs and has some different internal processing but the core functionality – point 3 and 4 from above – is still the same. With a minor change the SSH functionality can be implemented: When the files are written, not only the data files and the template file are written but also a SSH template and a SSH input file.

The SSH template file is similar to any other template file of this thesis; it has a single function call – `ssh_connection()`. `ssh_connection` reads the data in the SSH input file and opens the SSH tunnel, transfers the files, makes the remote machine execute the template file, and transfers the results back and cleans up the remote folder. The SSH input file contains all the information needed for this and only a few lines of information are needed:

- IP address of the remote machine.
- Port where through the connection should be made.
- Username and password for the remote machine.
- Filenames of the files that should be transferred i.e. the data files and the template.
- Filename of the file which should be run of the remote machine i.e. the template.
- Filenames of what files should be returned i.e. the result files.

When the results are back the SSH connection will be closed and the SSH template will be terminated and the CPython part of the Livestock CMF Solve component will be finished.

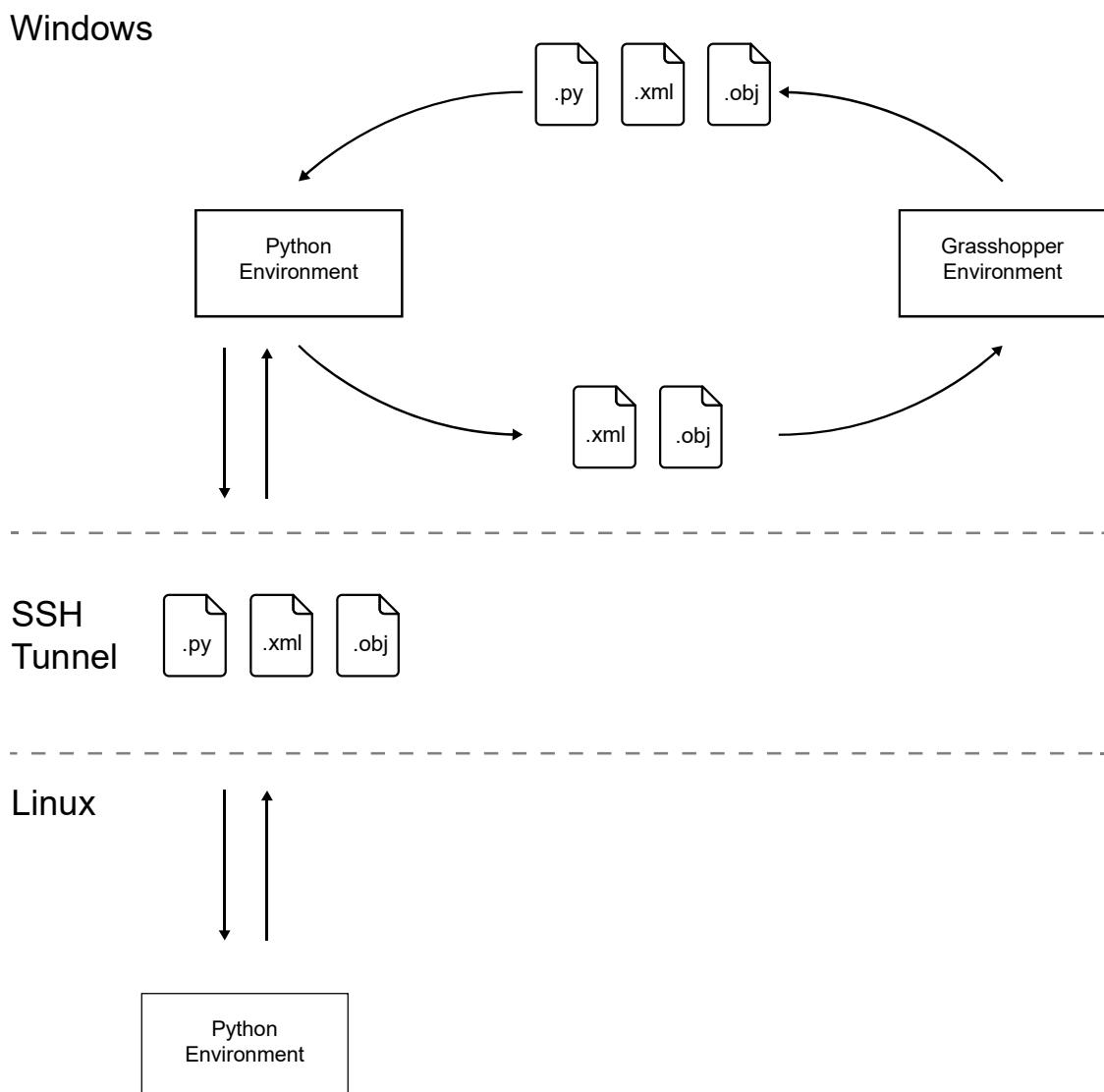


Figure 4.5 - Information exchange between Grasshopper Environment, through Python onwards to remote computer for processing and back again.

# 4.3 - ATMOSPHERE MODEL

## Description

The atmosphere model is made on the basis of the equations described in chapter 3.1 – Atmosphere Model. The equations are converted to Python code and is included in the Livestock Package.

The main function of the atmosphere model is `new_temperature_and_relative_humidity`:

```
def new_temperature_and_relative_humidity(folder: str) -> bool:
    """
    Calculates a new temperatures and relative humidity for air volumes.
    """

    # Helper functions
    def get_files(folder_: str) -> tuple:
        """
        Reads the case files and returns them as a tuple of lists
        """
    def reconstruct_results(folder_, processed_rows_):
        """
        Collects and saves the results when they have been simulated.
        """

    # Run function
    1) temperature, relative_humidity, wind_speed, area, height_top,
       height_stratification, vapour_flux, cpu = get_files(folder)

    rows_ = [i
             for i in range(0, len(vapour_flux)-1)]

    2) input_packages = [(index,
                          temperature[index],
                          convert_relative_humidity_to_unitless(
                              relative_humidity[index]),
                          vapour_flux[index],
                          wind_speed[index],
                          area,
                          height_stratification,
                          height_top)
                         for index in rows_]

    3) pool = multiprocessing.Pool(processes=cpu)
    4) processed_rows = pool.map(run_row, input_packages)
    5) reconstruct_results(folder, processed_rows)
```

- 1) The needed files are loaded into memory from the specified folder.
- 2) The information is packed into a variable: `input_packages`.
- 3) A CPU worker pool is created with `multiprocessing.Pools`.
- 4) The workload – the function `run_row()` applied to a input package – is distributed the workload among the given CPUs.

- 5) When all computation is over the new\_temperature\_and\_relative\_humidity function gathers the information into four files; one with the new temperature, one with the new relative humidity, one with the latent heat flux and one with the excess vapour flux. For more details about the functions see the code documentation.

It is possible to make the computation in parallel because the process can be split into subparts. The vapour flux, which is the actual computational input to the function, can be seen as a two-dimensional matrix. It has a distribution in the x direction, which can be thought of as the cells, and they have a distribution in the y direction, which can be conceptualized as the time. The functions also take the area of the cells, which is a one-dimensional vector in the x-direction. Finally, the function need temperature, relative humidity and wind speed from the weather file. All three parameters are one-dimensional vectors in the y-direction i.e. time dimension. As there is no transfer of information in the x- or y-direction, the computation can be done independently on a whole row. Finally, the results can be composed together to a complete file.

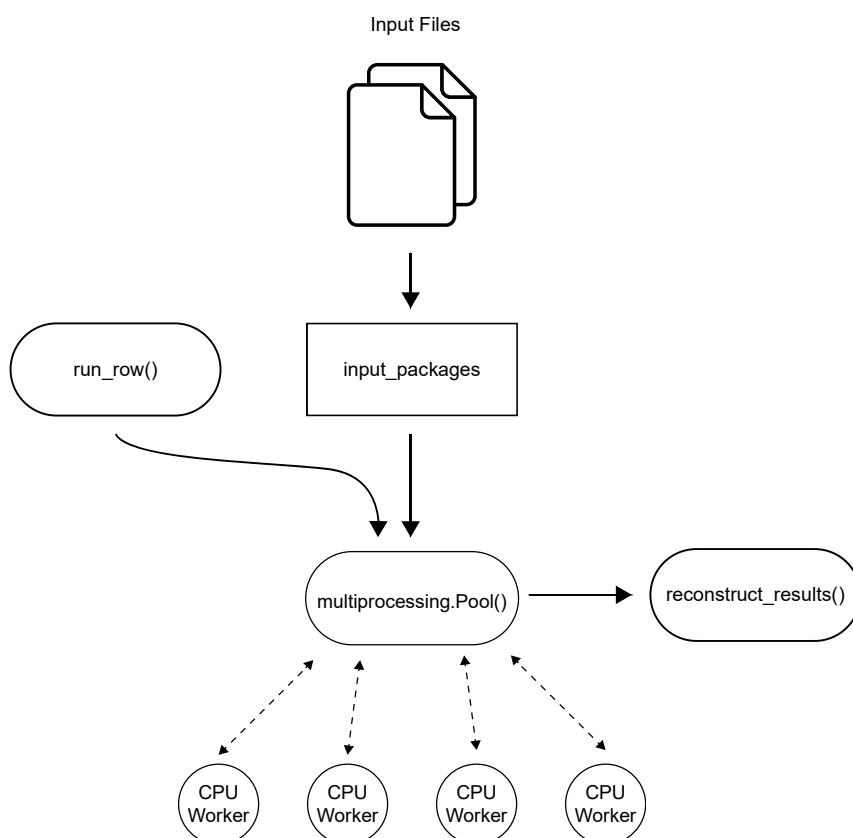


Figure 4.6 - Process diagram of the new\_temperature\_and\_relative\_humidity function.

The function `run_row()` computes a single row of the Vapour Flux Matrix. It does it with the following steps:

- 1) Air temperature is converted from Celsius to Kelvin.
  - 2) The wind speed corrected air volume is computed.
  - 3) `compute_temperature_relative_humidity()` is called.
    - a) The modified temperature for each cell is computed, given the latent heat flux (which is again given by the vapour flux) and the current vapour pressure in the air.
    - b) The cells are looped over; checking if the new vapour pressure, given the modified temperature and the vapour flux, exceeds the saturated vapour pressure of the modified temperature.
    - c) If the check is passed; the vapour flux of the cell are stored.
    - d) If the new vapour pressure exceeds the saturated vapour pressure another function will be called – `max_possible_vapour_flux()`. The function is placed within a root finding algorithm – `scipy.optimize.brentq()` – thereby

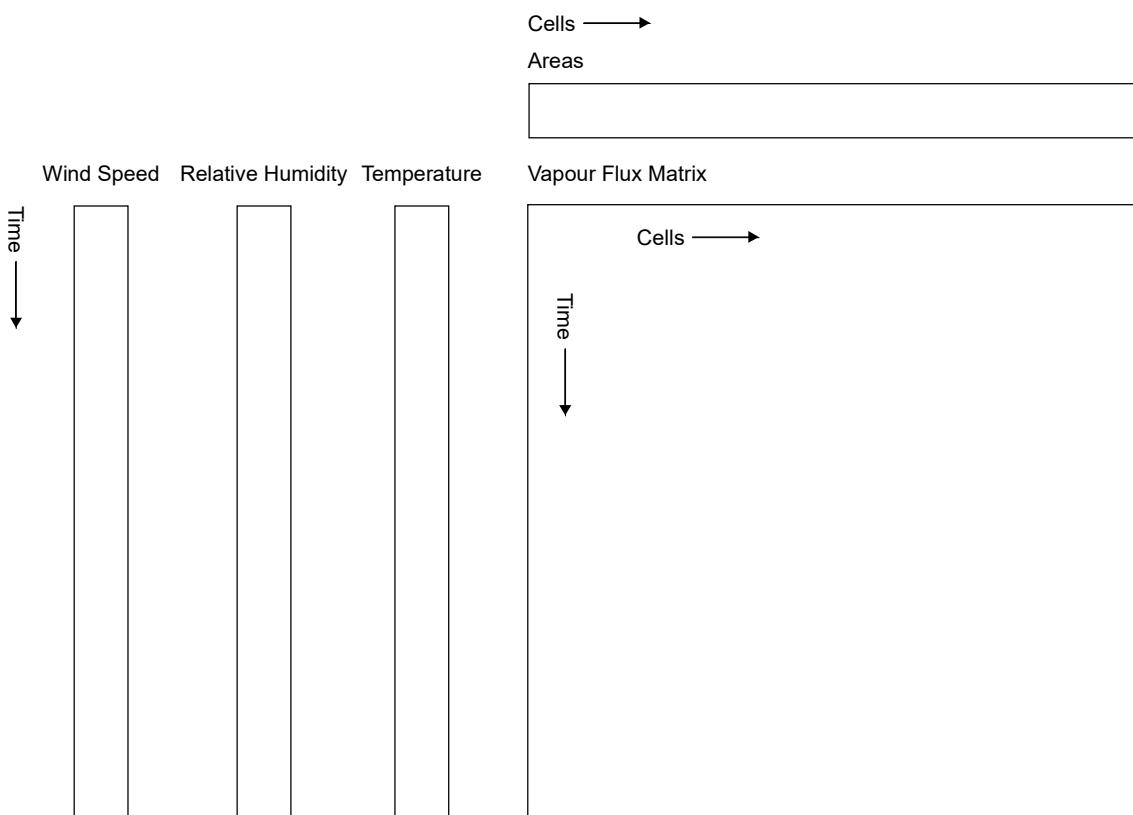


Figure 4.7 - Illustration of the matrices and vectors with heat and vapour fluxes, weather data and areas.

---

is it possible to find the vapour flux of the corresponding vapour pressure that will not exceed the saturated vapour pressure of a temperature given that vapour flux. When that vapour flux is found it is stored.

- e) The corresponding latent heat flux is computed, given the vapour flux for each.
  - f) The corresponding modified temperature is computed based on the latent heat flux.
  - g) The new relative humidity is computed based on that temperature and the vapour.
  - h) The new temperature, relative humidity, heat flux and used vapour flux is returned to the original function.
- 4) The stratified values for the relative humidity is computed.
- 5) The stratified values for the temperature is computed.

The function `compute_temperature_relative_humidity()` computes the new temperature and relative humidity and checks that the relative humidity does not exceed 100%. If it does exceed 100%, the function will compute the vapour flux that corresponds to 100% relative humidity. Since the relative humidity check comes first, the root finding algorithm, which is the computational most expensive, will only be called in the cases, where it is needed.

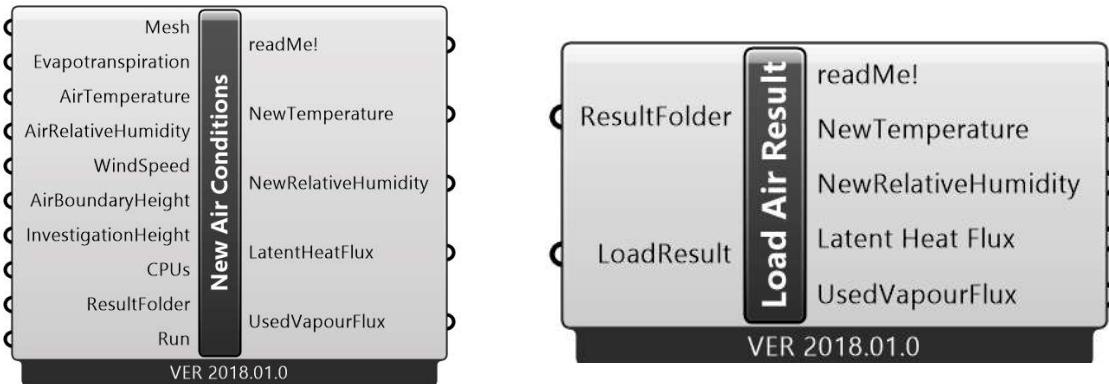


Figure 4.8 - Components of the atmosphere model: New Air Conditions and Load Air Results.

## Grasshopper Components

Two Grasshopper components have been developed for the atmosphere model. A detailed description of their inputs and outputs can be found in the documentation.

### Script setup

Figure 4.9 presents a sample script of how the atmosphere model can be applied. The script consists of six groups of components.

- 1) *Livestock Setup* prepares the script with information about the location of the CPython interpreter and SSH connection (in this case SSH is not used).
- 2) The group *Geometry* generate the mesh geometry.
- 3) *Preview* lets the user see it in the Rhino viewport.
- 4) *Weather* loads and EPW file with Ladybug Tool components.
- 5) *Load Results* loads the results from the soil model with the component CMF Results.
- 6) *Run Atmosphere Model* is the last group and it is this group that prepares and writes the information to files and spawns the subprocess with the CPython interpreter.

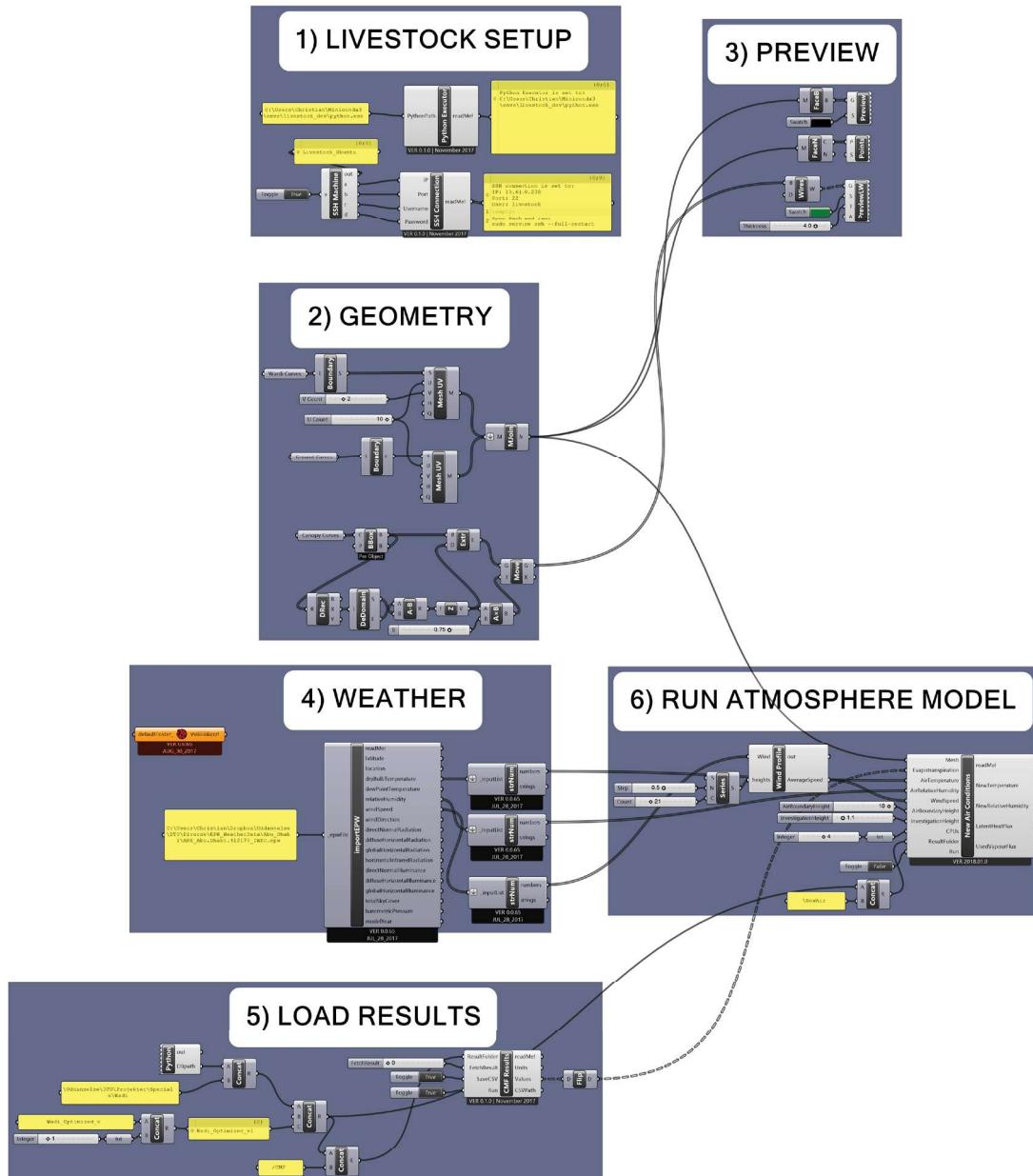


Figure 4.9 - Sample script of how the New Air Conditions component can be used.

## 4.4 - SOIL MODEL

### Description

The soil model is a Grasshopper wrapper around the Catchment Modelling Framework. The Grasshopper components collects inputs and passes them on to Livestock CMF Solve, which writes out files and uses the developed Livestock SSH Template Method to let the computation be carried out remotely. There has also been developed Grasshopper components to load and visualize the results of the Soil Model.

In the background it is the `livestock.hydrology.CMFModel` class that turns the written files into a CMF project and computes the results.

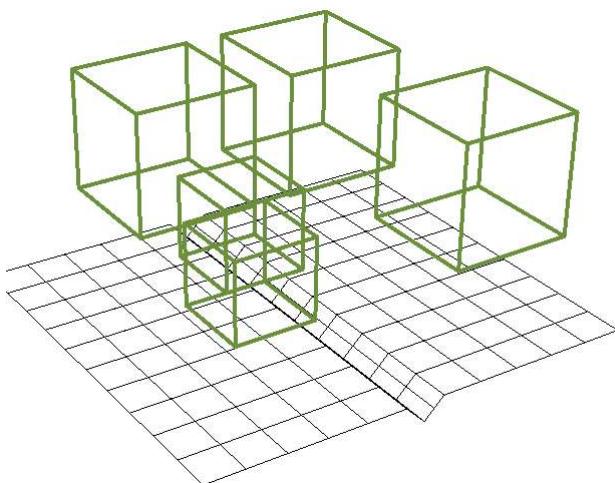
### Grasshopper Components

The soil model components are the largest group of components developed for this thesis. They are located under the “2 | CMF” category. A complete list and detailed description of their functionality can be found in the documentation.

### Script Setup

Figure 4.11 presents a sample script of how the soil model can be applied. This script consists of 7 groups.

- 1) Livestock Setup is identical to the Livestock Setup of the Atmosphere Model. The group configures the SSH connection and CPython interpreter.
- 2) Geometry generates the geometry for the script.
- 3) Preview displays it for the user in the Rhino viewport.



44	49	54	59	64	69	74	79	84	89
43	48	53	58	63	68	73	78	83	88
42	47	52	57	62	67	72	77	82	87
41	46	51	56	61	66	71	76	81	86
40	45	50	55	60	65	70	75	80	85
21	22	23	24	29	31	33	35	37	39
20	19	18	16	30	32	34	36	38	36
19	18	14	12	10	8	9	4	3	1
94	99	104	109	114	119	124	129	134	139
93	98	103	108	113	118	123	128	133	138
92	97	102	107	112	117	122	127	132	137
91	96	101	106	111	116	121	126	131	136
90	95	100	105	110	115	120	125	130	135

Figure 4.10 - Axometry of the model (to the left) and top view of the model (to the right). Each black square is a mesh face - each with its own index. The green boxes represent tree canopies.

---

4) Ground sets up the ground properties for the model.

- Three soil layers are set, each one meter deep.
- The retention curve of the ground is set to the standard CMF retention curve.
- The ground surface properties are set as dry soil with the component CMF Vegetation Properties.
- CMF Ground collects the information and sets the saturated depth to 5m.
- Penman-Monteith is set as evapotranspiration method.
- The properties are applied to the cells from 0 to 139, i.e. all cells.

5) This group sets the properties for the trees with the component Synthetic Tree.

- There are different heights, 3 m and 4.5 m, but only one type of tree – Synthetic Deciduous.
- The 3 m tree properties are applied to the cells: 113, 114, 118, 119, 122, 123, 127 and 128.
- The 4.5 m tree properties are applied to the cells: 45 - 47, 50 - 52, 55 - 57, 71 - 73, 76 - 78, 81 - 83, 97 - 99, 102 - 104 and 107 - 109.

6) Weather is on Figure 4.11 visually the largest group, but it only consists of one Livestock component – CMF Weather. The weather component requires some preprocessing that is why the group is so large.

- Initialization of Ladybug Tools.
- Loading of the annual radiation results.
- Loading of the EPW weather file data.
- Generation of rain
- Collection of information in the CMF Weather component.

7) Solver is the last and final component group.

- CMF Solve is the component that writes the gathered information in the previous groups and spawns the subprocess with the CPython interpreter.
- CMF Solve takes an input from CMF Outputs, which configures the wanted outputs, and CMF Solve Settings, which configures the solver from the CMF library.

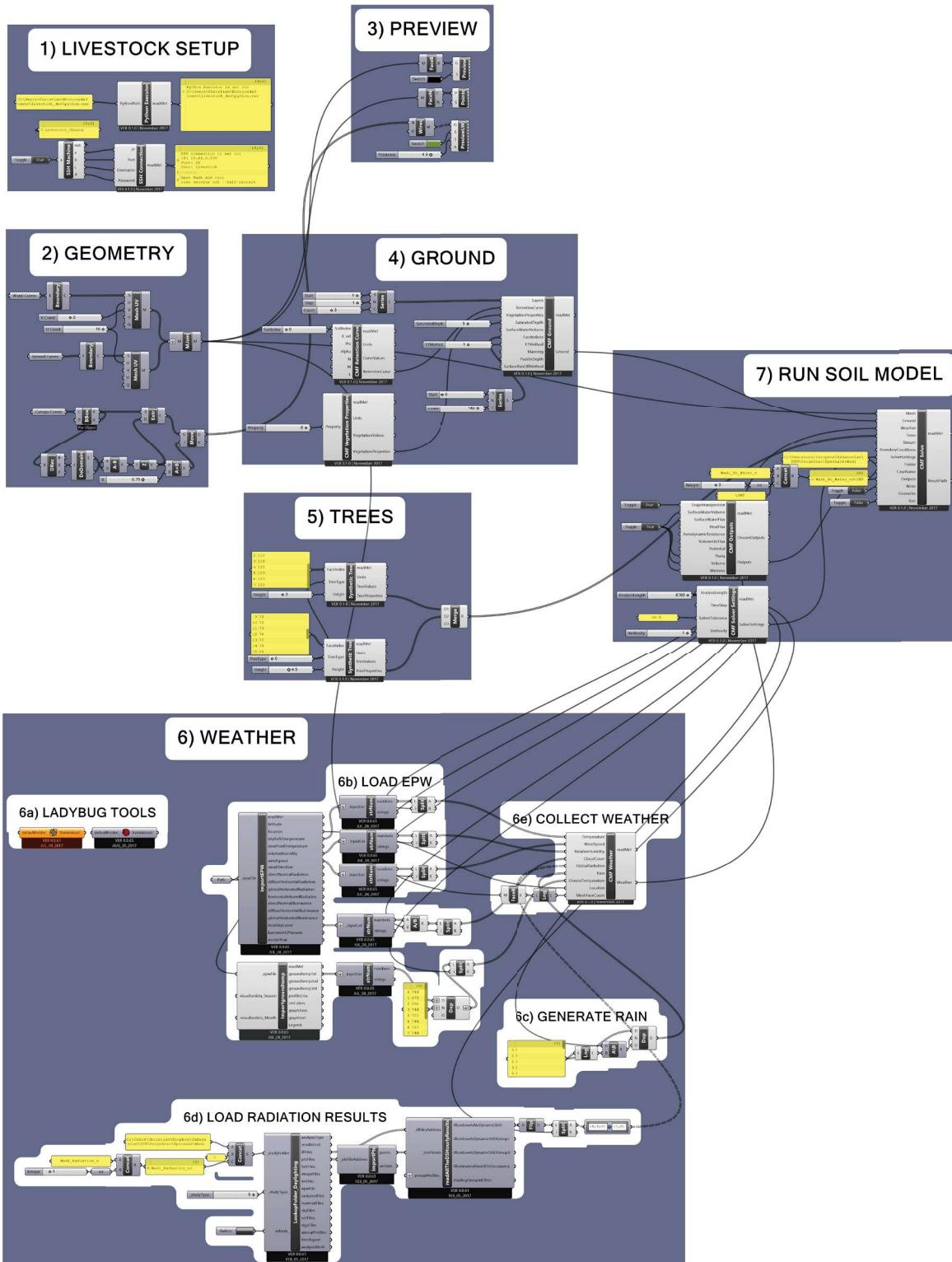


Figure 4.11 - Setup of the soil model in a Grasshopper script.

# 5 - CASE STUDY

# 5.1 - ABU DHABI CASE

## Case Description

This case study is based on a project from BIG – Bjarne Ingels Group. The case study is located within the master plan of an area in Abu Dhabi, United Arab Emirates. Through the area runs a wadi. There is an interest in investigating the possibility of improving the outdoor comfort by having trees around the wadi and/or water running through it. These wishes fit well with the capabilities of the developed models and purpose of this thesis.

For this case a small part of the master plan will be analyzed. The area is located around the wadi, which runs through a densely build up area. It is the area enclosed by a red ellipse on Figure 5.1. The area has been simplified for easing the computations. The simplified model can be seen in Figure 5.3. The analyzed area has the dimensions: 20m times 22m and the wadi goes right through it. The wadi is 2m wide, and 1m deep and its cross section is modelled as a triangle. In the model there is placed five trees. Three trees on the south side of the wadi and two on the north.

The case study has been approached in the following way:

- 1) An analysis of the weather in Abu Dhabi is carried out together with an analysis of the weather in Copenhagen as a comparison



Figure 5.1 - Master plan of the Abu Dhabi case area. The red ellipse encloses the area of interest for this thesis.

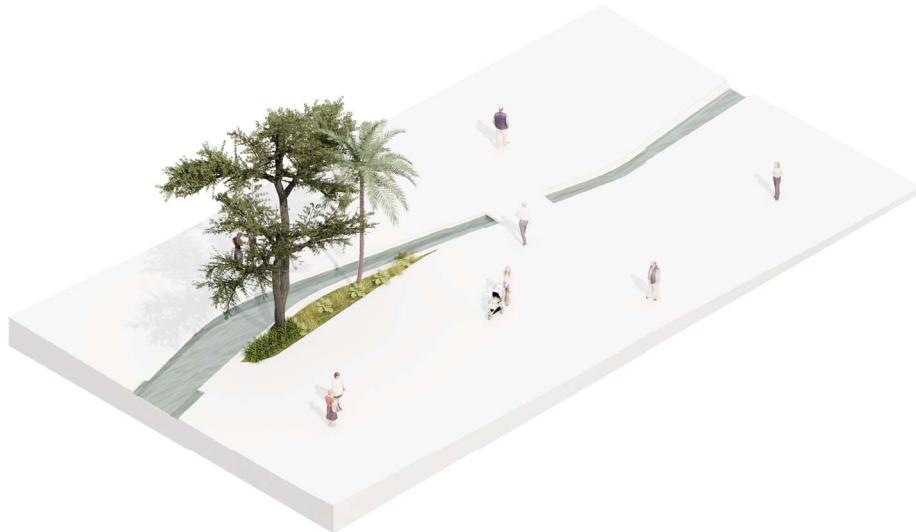


Figure 5.2 - The architect's vision of the area around the wadi.

- 2) The theoretical limit of the effects of evapotranspiration on a single mesh cell is investigated.
- 3) The water balance of the model has been investigated to optimize the evapotranspirational flux.
- 4) An optimized model is transferred to the second part of the study, where the optimized evapotranspiration is used to modify the air with the atmosphere model.
- 5) The modified air conditions are used to compute the thermal comfort of the case and to compare them with popular approach of thermal comfort modelling, presented in section 4.1.

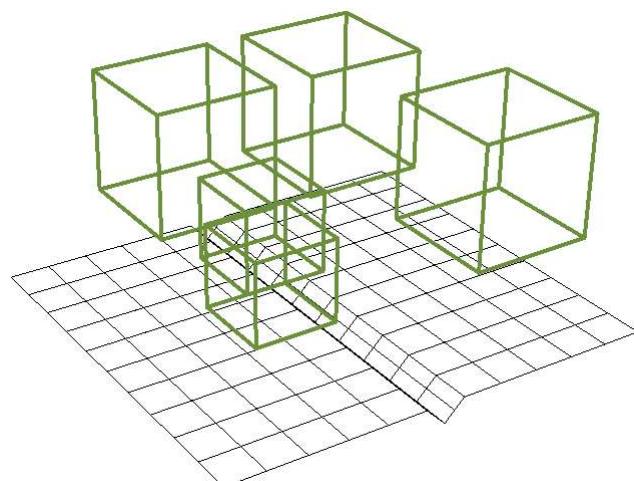


Figure 5.3 - Simplified computational model of the wadi.

## 5.2 - WEATHER ANALYSIS

As the thermal comfort highly depend on the climate of location in question an initial analysis of the weather in Abu Dhabi will be conducted as wells as a comparison with Copenhagen.

### Abu Dhabi

Abu Dhabi is in the Köppen climate class BWh – Hot Dessert Climate. June through September is the hottest months with temperatures above 32°C, as a daily average, and with a maximum up to 47°C. November to March is regarded as the cooler season (Wikipedia 2018a).

For hourly weather data an EPW file was collected. It was possible to acquire such a file from EnergyPlus' weather catalog (Illinois and The Regents of the University of California 1996). The source of the file is from the International Weather for Energy Calculations - IWEA.

### Temperature

The monthly average temperature varies from 18.0°C in January to 34.7°C in August. A typical day can be plotted with maximum, average and minimum values for every hour of the day. The typical day is constructed by taking all the values for each hour of the day for all days and producing the minimum, average and maximum. Such a plot can be seen in Figure 5.5. It shows that the average temperature at midnight is around 25°C. It falls until the sun sets around 07:00 from where it increases until it peaks at midday with

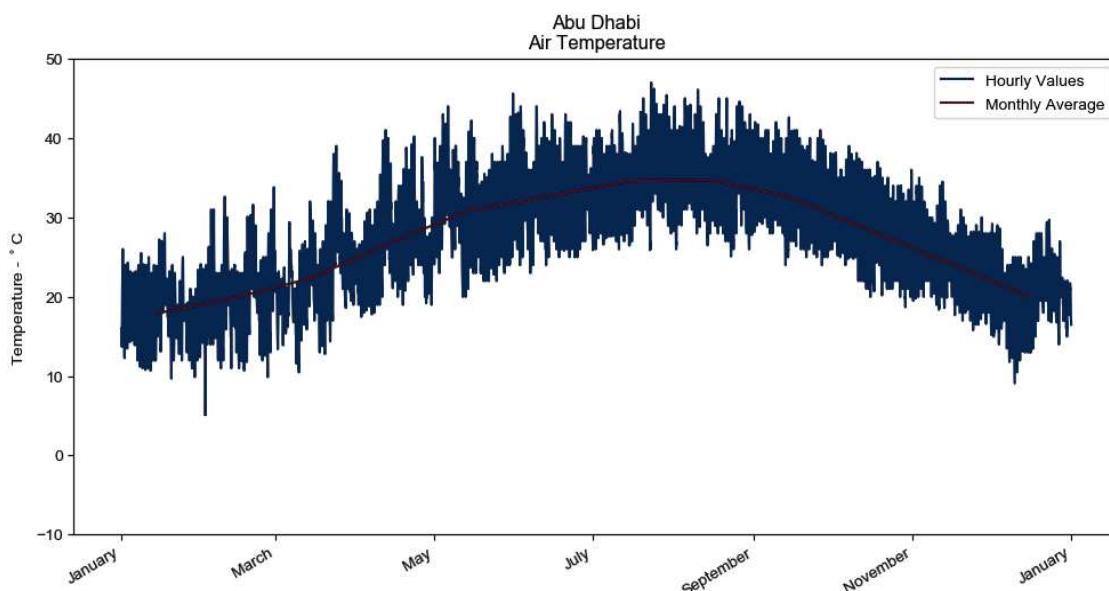


Figure 5.4 - Temperature in hourly values, monthly maximum, monthly average and monthly minimum for Abu Dhabi.

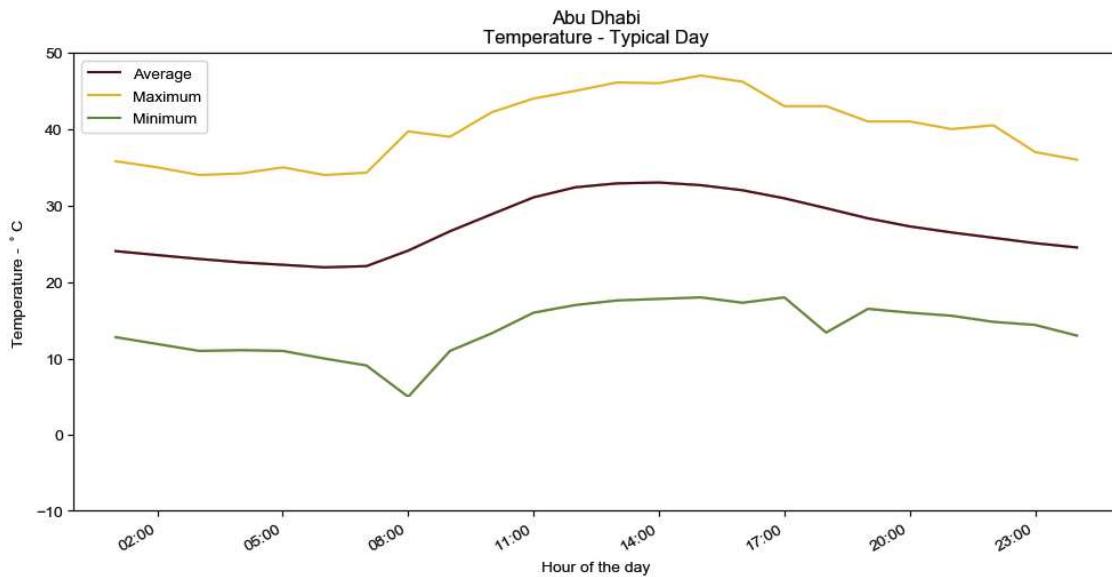


Figure 5.5 - Maximum, average and minimum temperatures for Abu Dhabi over the course of a day.

around 32°C, hereafter in declines again. Both the maximum and minimum curves follow the profile with a shift in temperature of +/- 10°C.

### Relative Humidity

It is harder to generalize the relative humidity over the year as shown in Figure 5.6. In general the average relative humidity from April to October is lower than the rest of the year. The average for this period is 55.2 %. The average for the rest of the year is 68.2%. The minimum values for April to October is also lower than the minimum values for the rest of the year. However, every month – besides April and December – has a maximum

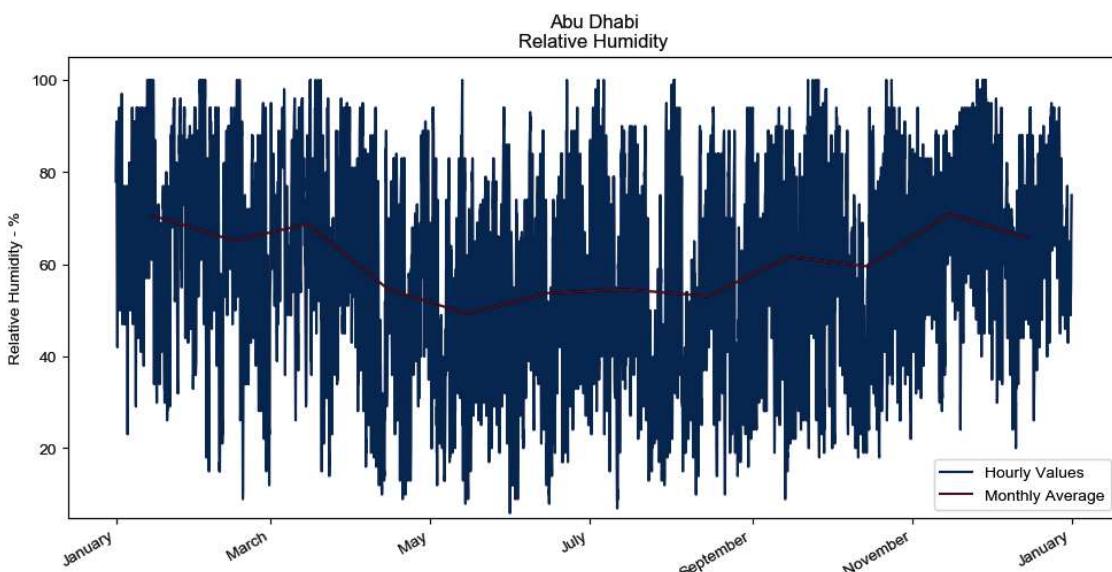


Figure 5.6 - Relative humidity in hourly values, monthly maximum, monthly average and monthly minimum for Abu Dhabi.

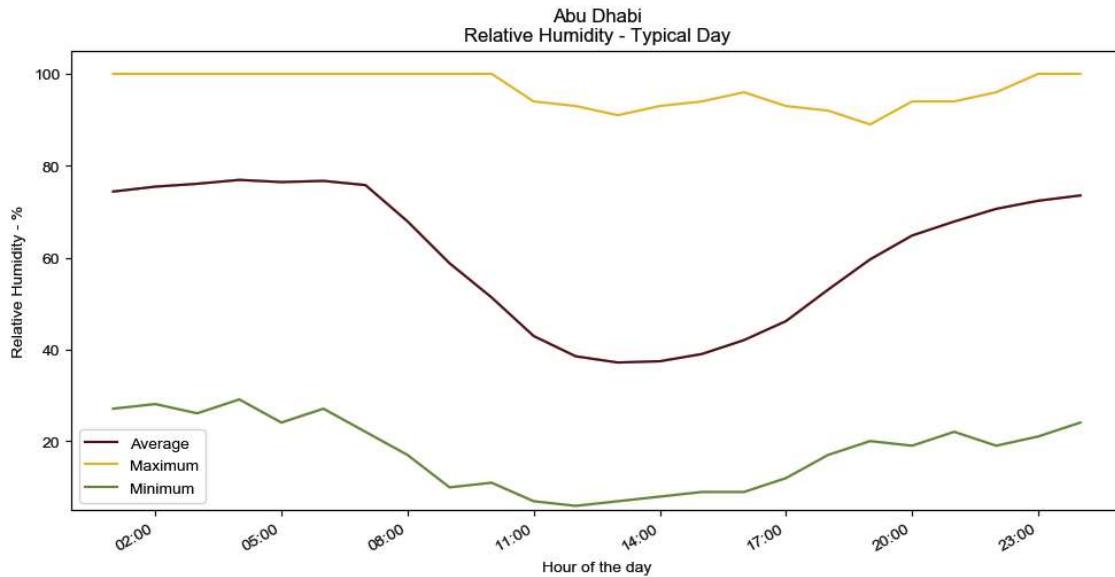


Figure 5.7 - Maximum, average and minimum relative humidity for Abu Dhabi over the course of a day.

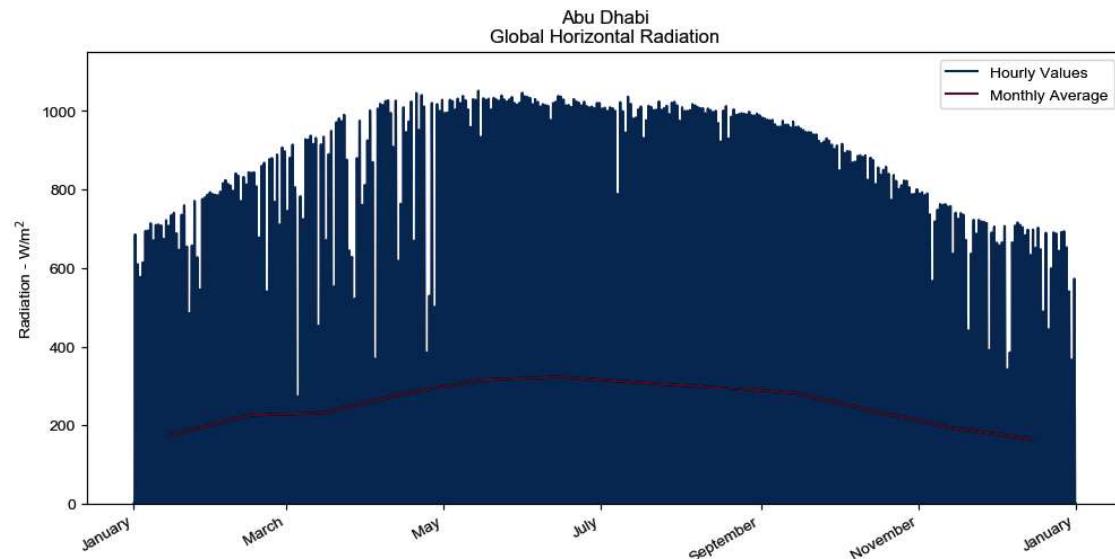


Figure 5.8 - Horizontal Global Radiation over a year.

relative humidity of 100%. There is a much bigger fluctuation over the typical day. From midnight until 07:00 the relative humidity stays almost stable at 75%. It drops to 37.2% at its minimum at 13:00, where after it rises again until it reaches 75% again.

## Radiation

The yearly and the typical day plots of the radiation can be seen in Figure 5.8 and 5.9. April to August all have maximum horizontal global radiation values above 1000 W/m<sup>2</sup>, with the peak in May at 1051 W/m<sup>2</sup>, while the rest of the months have lower radiation values. The monthly average is for all months significant lower but also follows the same

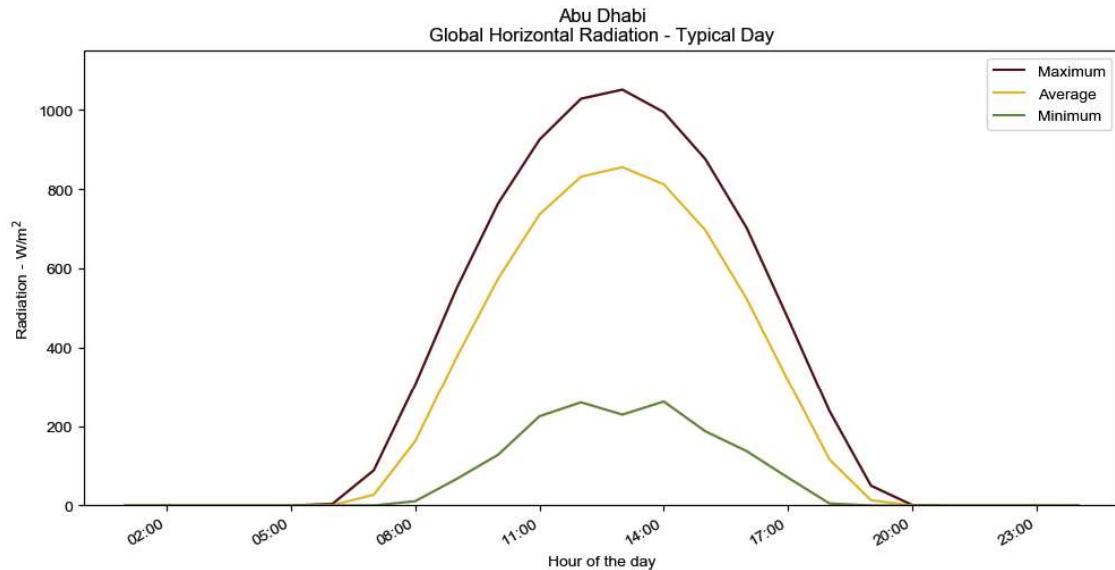


Figure 5.9 - Horizontal Global Radiation plotted for a typical day.

tendency curve as the temperature with higher values from April to August. The daily radiation plot shows that from 07:00 to 19:00 is exposed to the highest radiation shaped almost as a normal distribution around 13:00.

### Wind

To visualize the wind patterns a wind rose have been created, it is presented in Figure 5.10. The most frequent wind direction is North-West (12.6% of the year), followed by East (9.8% of the year), West-North-West (9.5% of the year) and North-North-West (8.9% of the year). The average wind speed is 3.6 m/s and 74% of the year (6464 hours) has a wind speed below 5 m/s. 1% (88 hours) of the year has wind speeds above 10 m/s with a maximum of 24.2 m/s. There is clam wind conditions for 140 hours of the year.

### Precipitation

EPW weather files does not include precipitation as a default. It has therefore not been possible to find reliable hourly data on precipitation for Abu Dhabi. (Wikipedia 2018b) states that it only rains 57mm during the year in Abu Dhabi. Most of that precipitation happens in February and March. It is therefore assumed in the modelling process that it does not rain in Abu Dhabi.

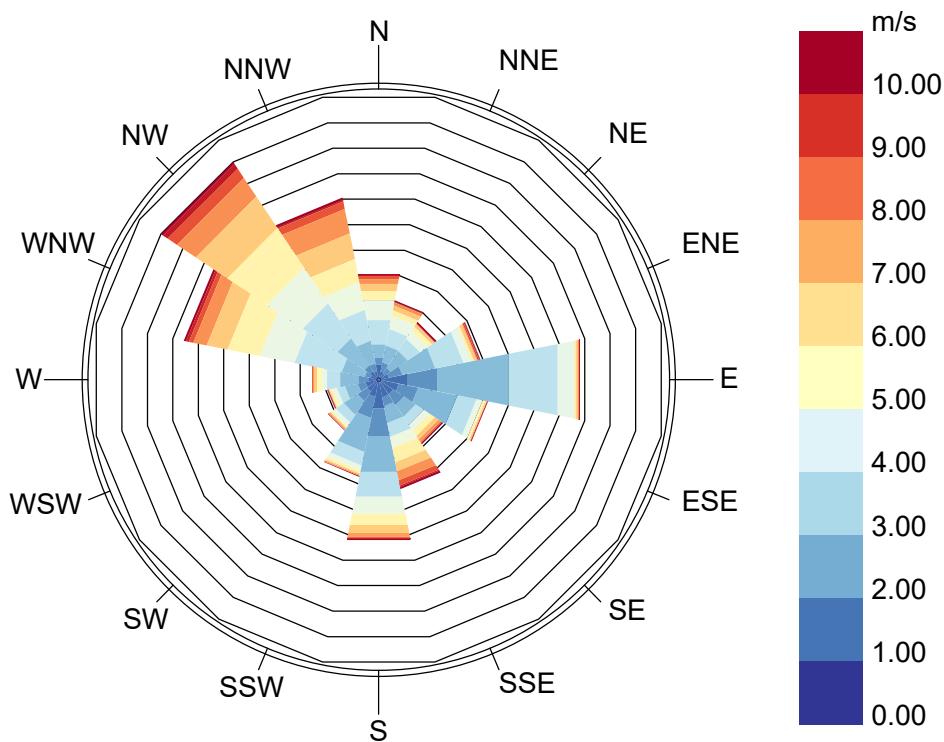


Figure 5.10 - Wind Rose for Abu Dhabi. Each circle represents a frequency of 1.3% equal to 110 hours.

## Copenhagen

Copenhagen is located at the Köppen climate class Cfb – Temperate Oceanic Climate (Weatheronline.co.uk 2018).

For hourly weather data the 2013 Design Reference Year (DRY) for Copenhagen is used. As with weather file for Abu Dhabi precipitation is not included in the DRY file. For the precipitation data for Copenhagen, data from Climate for Culture (“Climate for Culture” 2018) is used. Climate for Culture has collected real weather data from 1961-1990 and has projected that data to 2021-2050 regulating it for climate change. Precipitation from year 2021 has been used here.

### Temperature

The temperature in Copenhagen is significantly lower than in Abu Dhabi as seen in Figure 5.11. The average temperature for December to March is below 1°C and peaks in August with 17.9°C. The average temperature for a typical day in Copenhagen is around 8°C from 22:00 to 07:00, thereafter it rises to approximately 10°C during the day. So a much more constant average temperature compared to Abu Dhabi.

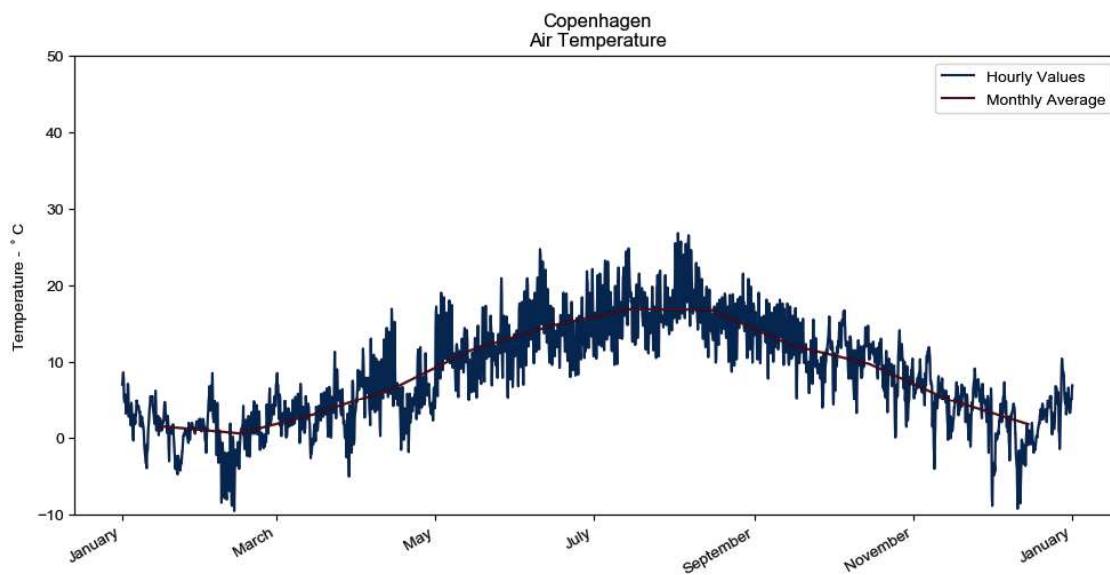


Figure 5.11 - Temperature plotted for the year.

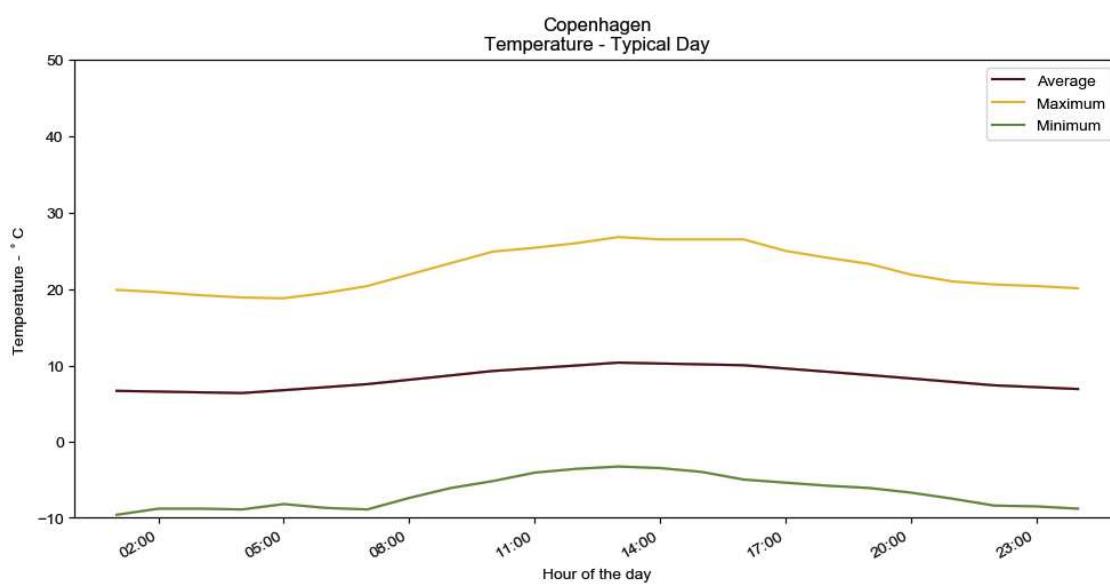


Figure 5.12 - Temperature plotted for a typical day.

## Relative Humidity

The typical day has on average a higher relative humidity in Copenhagen compared to Abu Dhabi as seen in Figure 5.14. It is at around 90% from midnight to 07:00 and has its minimum at 15:00 with 72.5%. During the summer Copenhagen has a lower relative humidity than during the winter as seen in Figure 5.13.

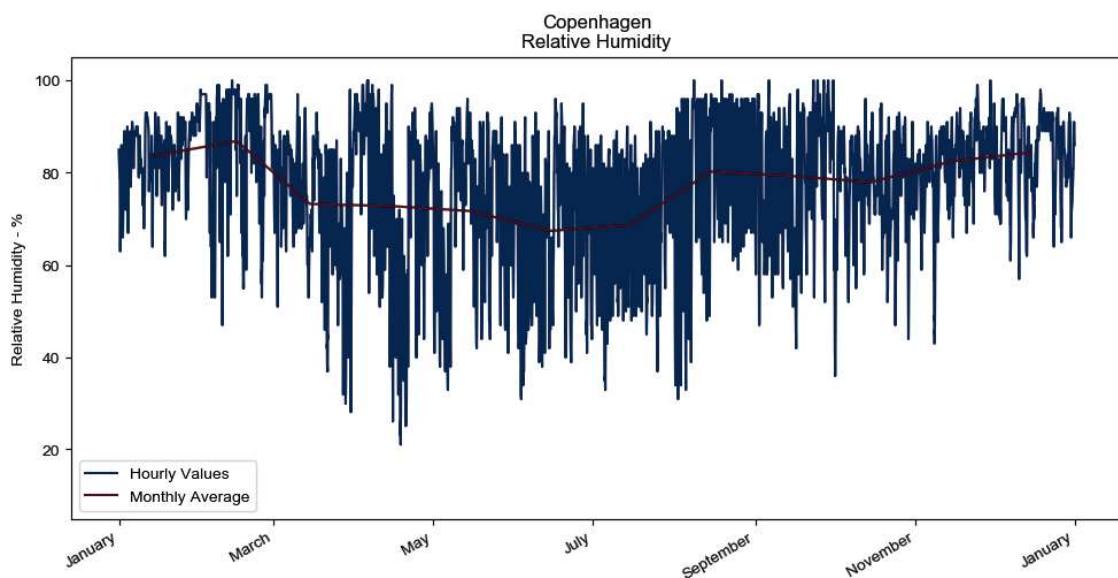


Figure 5.13 - Relative humidity for the whole year.

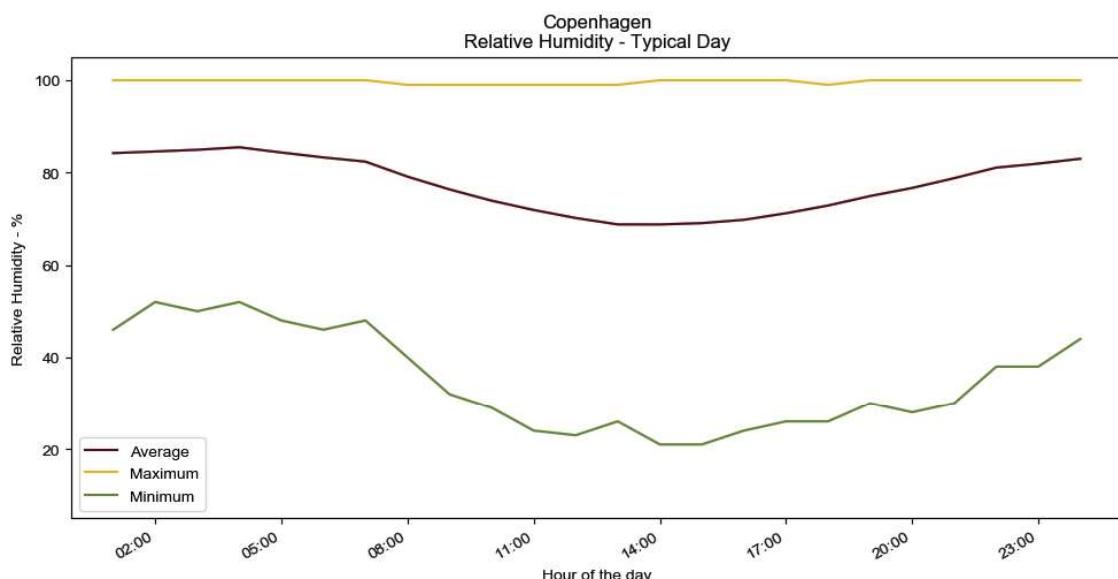


Figure 5.14 - Relative humidity for a typical day.

## Radiation

The global horizontal radiation for Copenhagen has been plotted and shown in Figure 5.15. It has a maximum value in July at  $874 \text{ W/m}^2$ . Other than July only May and June has peak radiation values above  $800 \text{ W/m}^2$ . These months are also the only ones having an average radiation above  $200 \text{ W/m}^2$ . The plot of the typical day reveals the long days in summer and short days in winter, with the plots maximum and minimum curves. The maximum curve already starts to rise at 4:00 and returns to zero at 22:00. The minimum curve hardly rises at all.

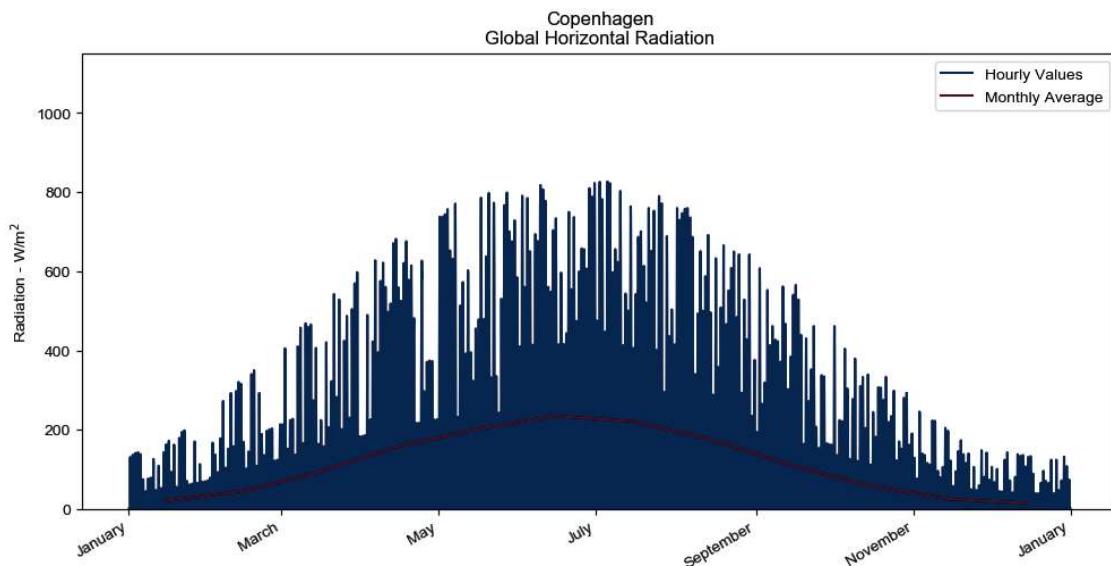


Figure 5.15 - Horizontal Global Radiation for a whole year

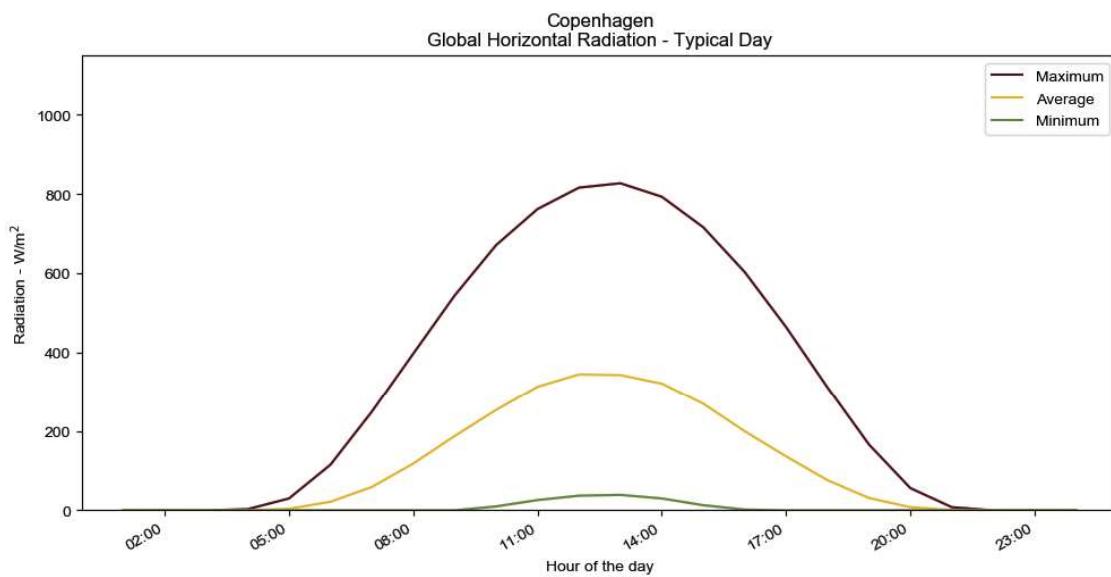


Figure 5.16 - Horizontal Global Radiation over a typical day.

## Wind

For Copenhagen, a wind rose has been created to investigate the wind pattern, it can be found in Figure 5.17. The most frequent wind direction in Copenhagen is West-South-West (11.5% of the year), followed by West (11% of the year) and South-South-West (10.1%). The average wind speed is 4.4 m/s in Copenhagen and thereby higher than in Abu Dhabi. 61% for the year (5340 hours) has wind speeds below 5 m/s. 1.5% (135 hours) of the time there is wind speeds above 10 m/s with a maximum of 13.8 m/s. There is no wind for 316 hours of the year. It can be concluded hereof that Copenhagen has a higher average wind speed, but with more consistently wind speeds and with lower extremes.

## Precipitation

The monthly precipitation for Copenhagen is shown in Figure 5.18. On average it rains 65.6mm per month in Copenhagen, so more than it does in Abu Dhabi on an entire year. July is the wettest month with 122.7mm followed by December with 114.2mm. October and March the driest with 18.9mm and 20.5mm respectively.

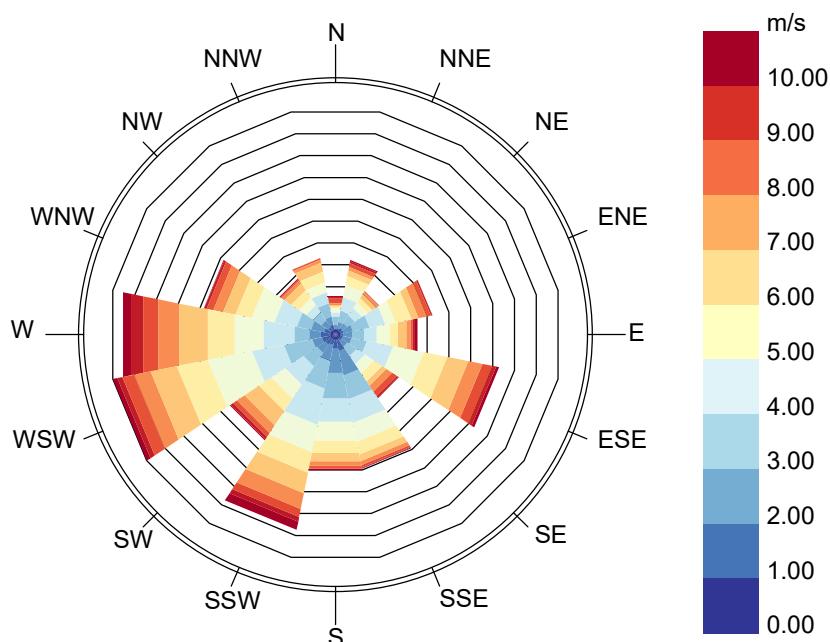


Figure 5.17 - Wind Rose for Copenhagen. Each circle represents a frequency of 1.1% equal to 100 hours.

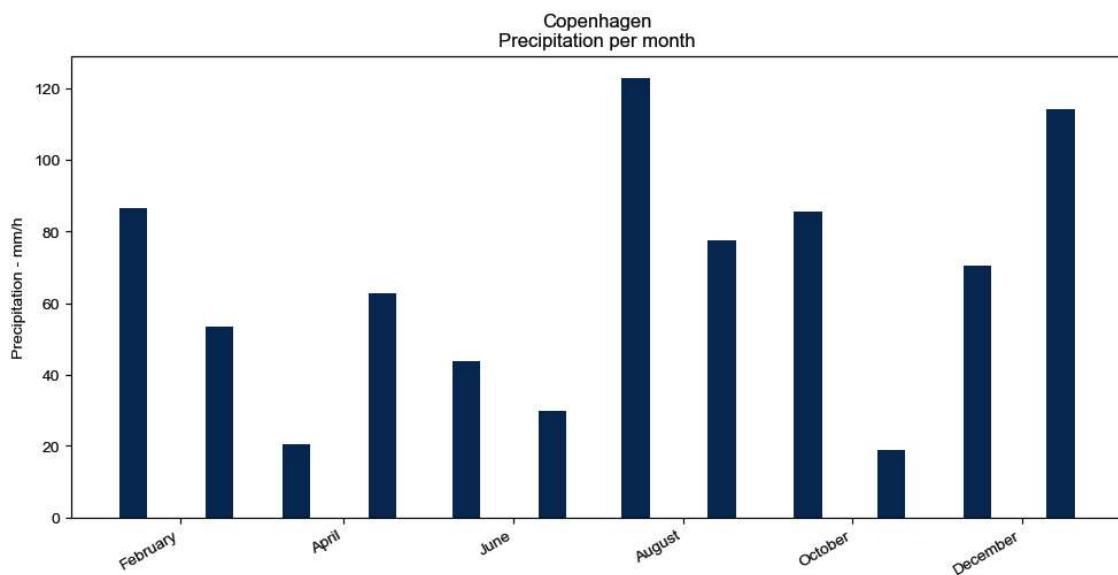


Figure 5.18 - Precipitation per month over an entrie year.

## 5.3 - THEORETICAL LIMIT

To see the potential of a hygrotermic controlled microclimate an investigation has been carried out to see what amounts of water vapour is needed to achieve maximum benefit i.e. 100% relative humidity and what the corresponding air temperature would be. This investigation is done based on the Abu Dhabi EPW weather file and the methods developed for this thesis. The analysis assumes an air volume with a footprint of 3.26 m<sup>2</sup>, which is the average mesh face area for the case and a height of 10m and the temperature is taken at 1.1m.

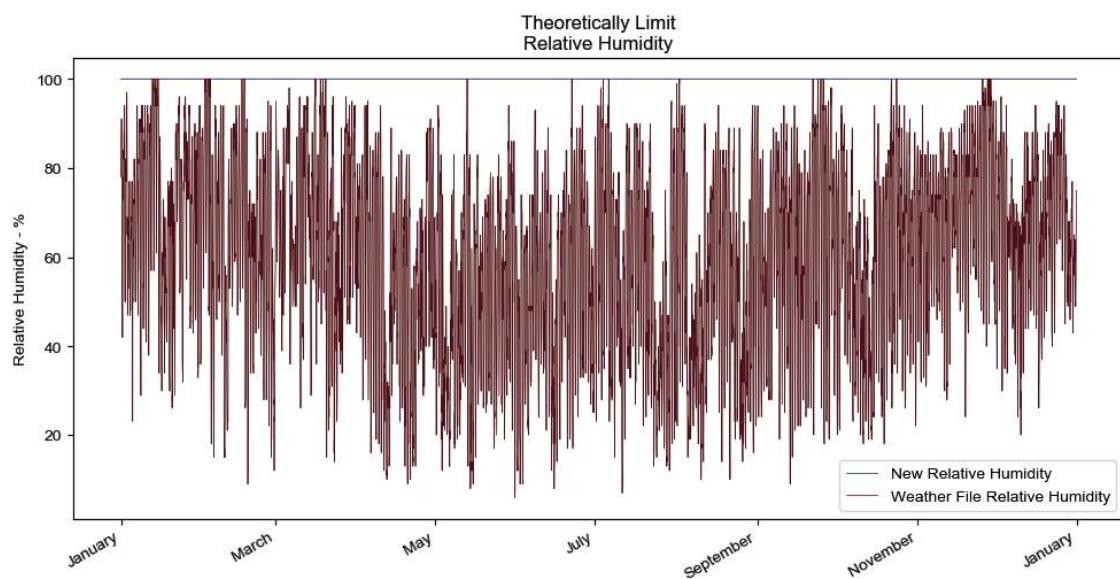


Figure 5.19 - Relative humidity is at 100% when the theoretical limit is reached.

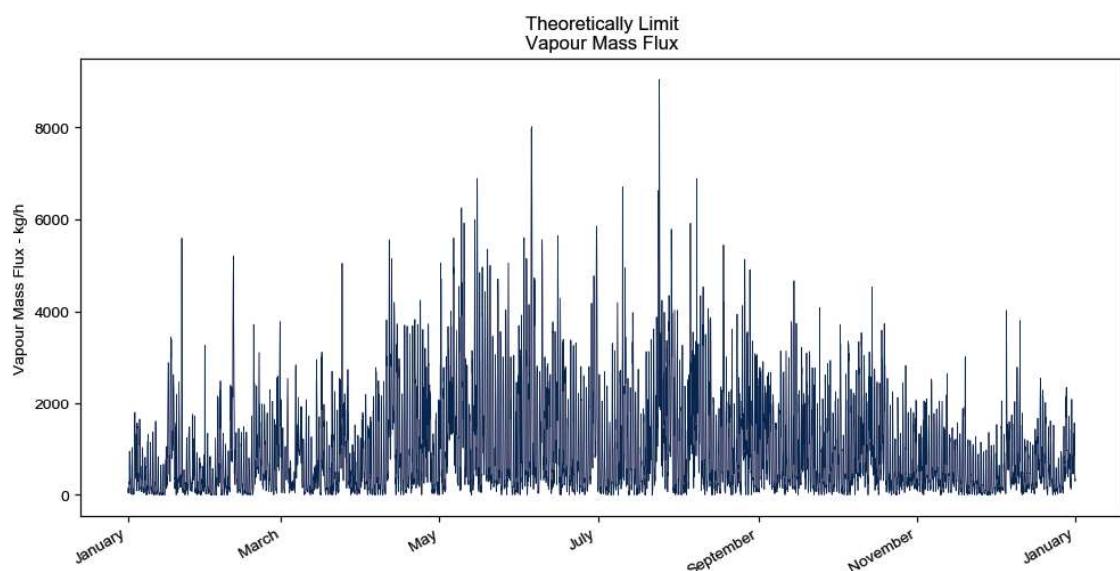


Figure 5.20 - Water vapour mass flux needed to reach the theoretical limit.

In Figure 5.19 it is visible that the attended outcome is achieved as the blue curve is 100% relative humidity at all times. The vapour mass flux needed to achieve this can be seen in Figure 5.20. In that figure the spread of the vapour flux seems quite large, from 0 to peaks in June and August at 8000 and 9000 kg/h. From Figure 5.20 it is hard to say how often which size of volume is needed, only when. Therefore, a histogram has been made and it can be seen in Figure 5.21. The frequency and span of the first 4 bars can be seen in Table 5.1. It can be observed that 29% of the year less than 244 kg/h and 17% of the time less than 122 kg/h is needed. The reason for these seemingly quite high numbers in 32.6m<sup>3</sup> of air is that the air volume is corrected for wind speeds. As seen in Figure 5.22, some of the resulting air volumes are large – up to 175 000 000 m<sup>3</sup>. This of course leads to some of the massive amounts of water vapour calculated. With the substantial amounts of vapour needed likewise sizes of heat will be needed and we therefore have latent heat flux up to more than 20GJ/h.

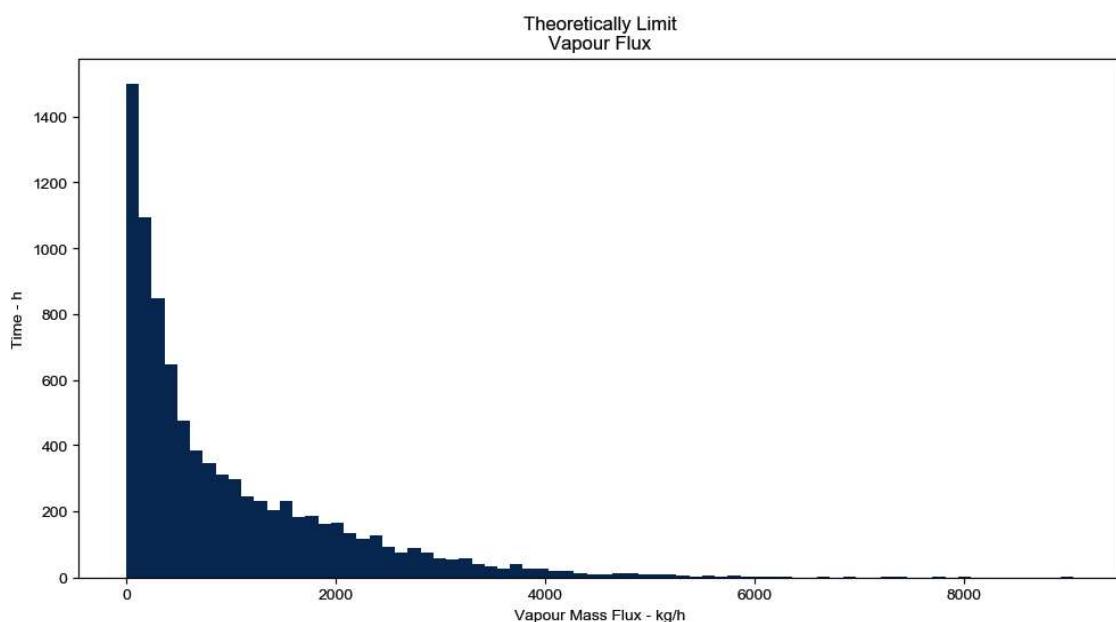


Figure 5.21 - Histogram of the needed water vapour mass flux to reach the theoretical limit.

Number of Hours	Value Span - kg/h	Percentage of the year
1501	0 - 122	17%
1094	122 - 244	12%
848	244 - 366	10%
648	366 - 488	7%

Table 5.1 - First 4 bars of the histogram of Figure 5.21

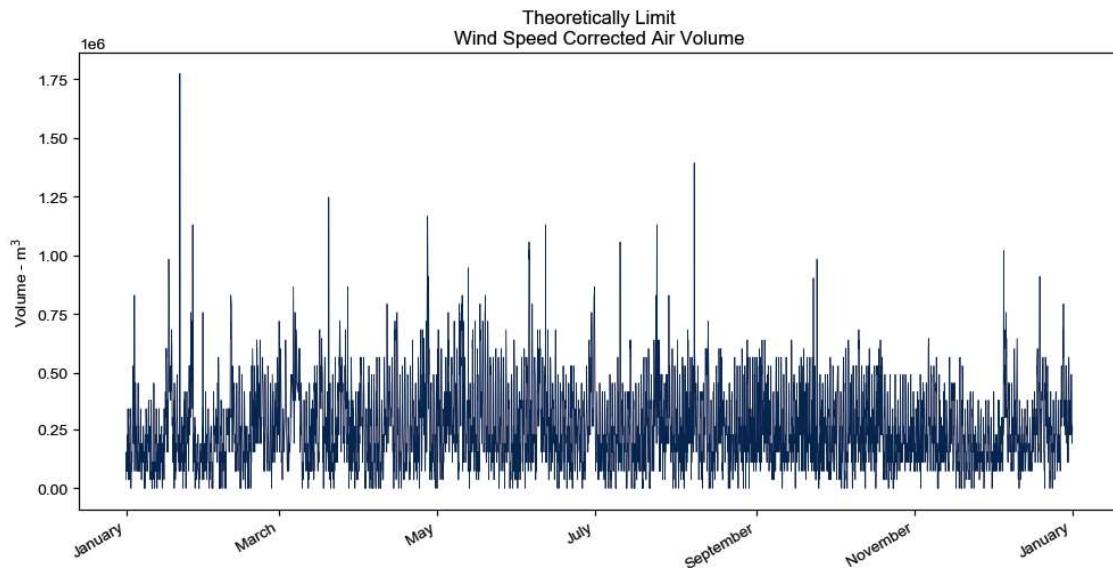


Figure 5.22 - Wind speed corrected air volume used in the calculation of the theoretical limit.

We also benefit for these efforts as seen in Figure 5.24 and 5.25. On an average hour we can lower the air temperature with 6.5°C. Luckily for us the largest reductions in air temperature is seen during the hot season where an reduction in temperature is mostly needed.

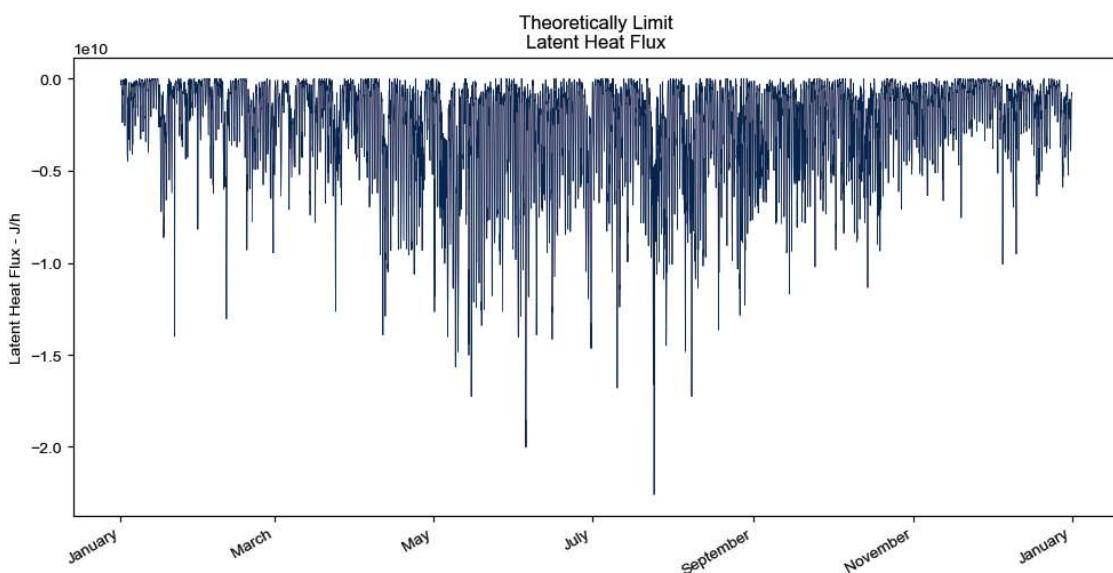


Figure 5.23 - Corresponding latent heat flux to the needed water vapour flux.

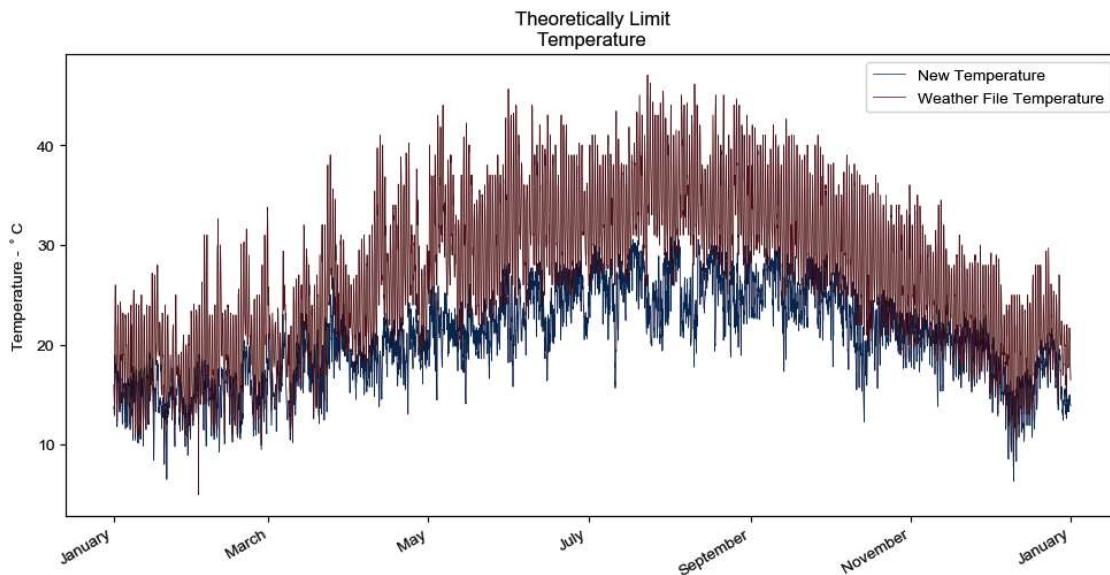


Figure 5.24 - Air temperature when the theoretical limit is reached.

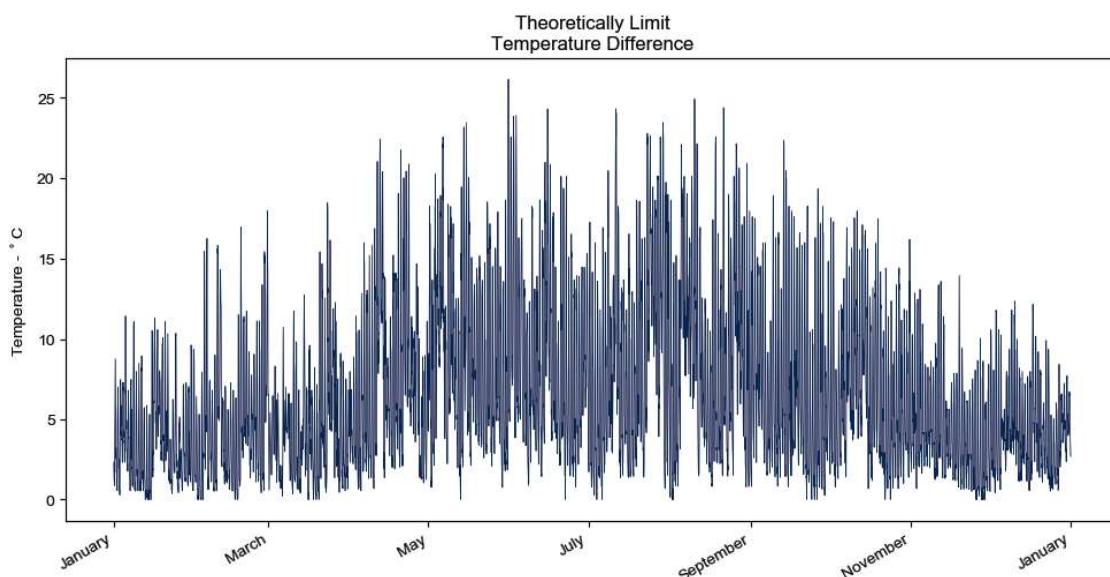


Figure 5.25 - Difference in air temperature between the weather file and the computed air temperature, when the theoretical limit is reached.

---

## 5.4 - WADI ANALYSIS

### Initial Model

As a first attempt to analyze the case an initial model has been setup. It uses the geometry as shown in Figure 5.3, the default CMF soil parameters and the Abu Dhabi EPW weather file as input for the soil model.

A Grasshopper script has been created to make the soil model and extract results about the water volume and evapotranspiration in the model.

### Ground Conditions

The ground conditions script sets up the soil model. The script can be seen in Figure 5.26. It is subdivided into 7 parts:

1. Livestock Setup. Identical to the setup described in the section about the soil model.
2. The geometry of the case is created.
3. Preview of the geometry to the Rhino viewports. Identical to the setup described in the section about the soil model.
4. The ground parameters are set. Here the components Livestock CMF Retention Curve, Livestock CMF Vegetation Properties and Livestock CMF Ground are used. Their parameters can be seen in Table 5.2 – 5.4.
5. The tree parameters are set. Livestock Synthetic Tree are used. Input values can be seen in Table 5.2.
6. The weather parameters are collected.
  - a. Initialization of Ladybug Tools.
  - b. Temperature, wind speed, relative humidity, cloud cover and ground temperature are taken from the EPW file.
  - c. Rain is set to 0mm/h for all hours.
  - d. Global radiation is calculated with a script with a workflow for using Ladybug Tools with Radiance. The script can be found in Appendix A.
  - e. Everything is collected in Livestock CMF Weather.
7. Everything is collected and solved in the Livestock CMF Solve component. It is also feed with inputs from Livestock CMF Solver Settings and Livestock CMF Outputs.

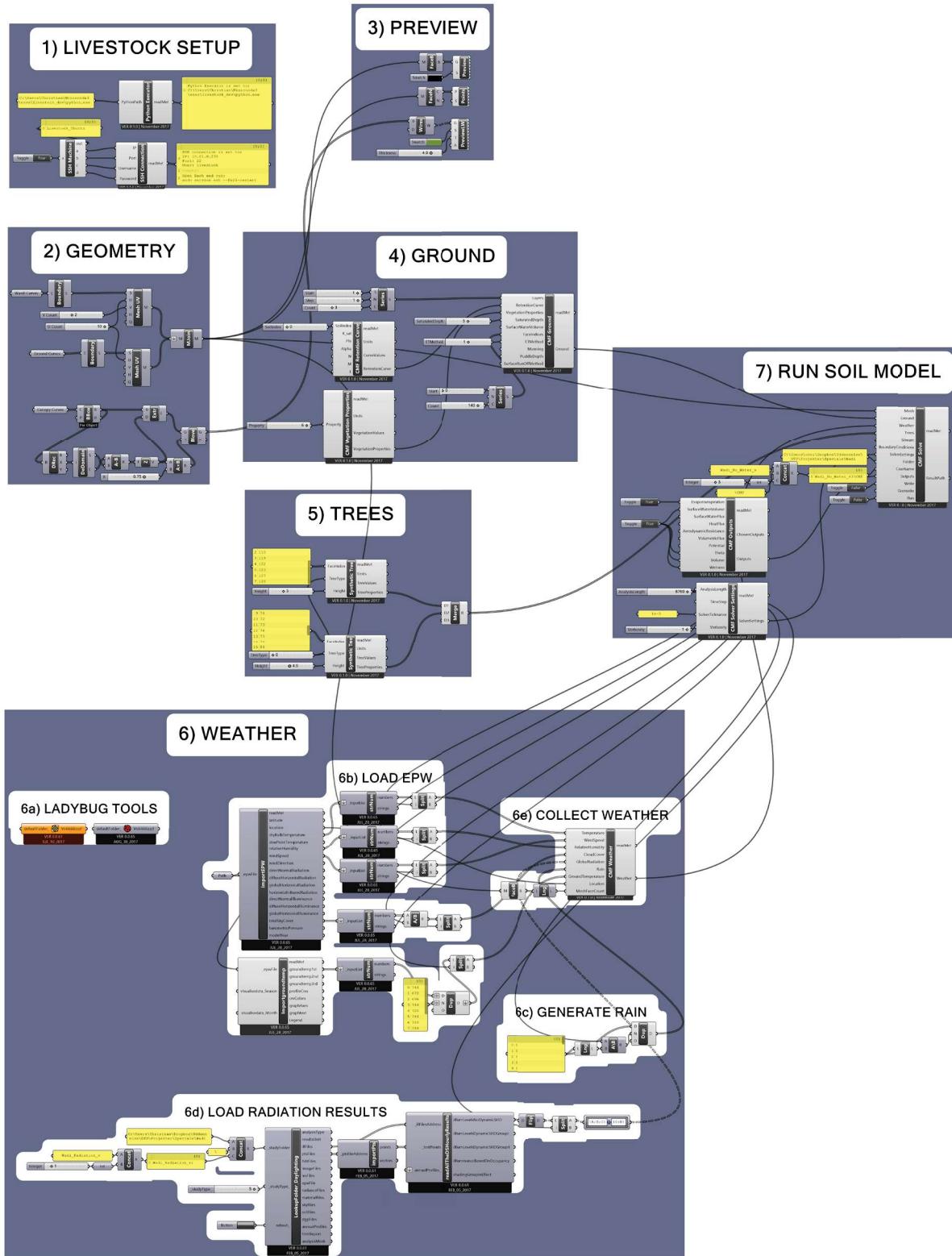


Figure 5.26 - Grasshopper script used to setup the soil model.

**Tree Properties**

Name	Synthetic Deciduous	Synthetic Deciduous
Height [m]	3	4,5
Leaf Area Index [-]	5,0	5,0
Albedo [-]	0,2	0,2
Canopy Closure [-]	1	1
Canopy PAR Extinction [-]	0,6	0,6
Canopy Capacity per LAI [mm]	-1,0	-0,9
Stomatal Resistance [s/m]	409,7	401,0
Root Depth [m]	1,9	1,9
Fraction at Root Depth [-]	1	1
Leaf Width [m]	0,1	0,1

Table 5.2 - Input for the tree properties

**Retention Curve**

Type	standard_cmf
K_sat [m/day]	15
phi [m <sup>3</sup> /m <sup>3</sup> ]	0,5
alpha [1/cm]	0,2
n [-]	1,2
m [-]	-1
l [-]	0,5

Table 5.3 - Used values for the retention curve.

**Vegetation Properties**

Name	Dry Soil
Height [m]	0
Leaf Area Index [-]	0
Albedo [-]	0,2
Canopy Closure [-]	0
Canopy PAR Extinction [-]	0
Canopy Capacity per LAI [mm]	0
Stomatal Resistance [s/m]	0
Root Depth [m]	0
Fraction at Root Depth [-]	0

Table 5.4 - Used values for the vegetation properties

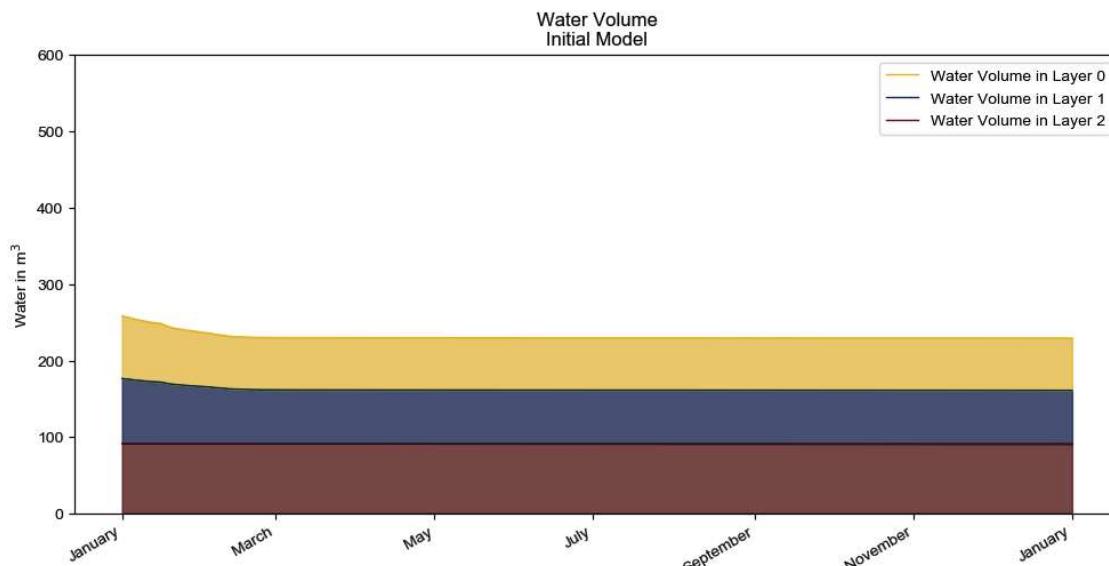


Figure 5.27 - Water volume in the different layers.

## Water Balance

With the components it is possible to output the water in the soil layers and on the surface of the mesh. Together with the data from the evapotranspiration it is possible to study the water balance within the model. In Figure 5.28 the cumulative evapotranspiration is plotted together with the total cumulative water loss from the soil layers. The red and yellow curve follow each other nicely – as expected – they are just mirrored around the x-axis. In the end of February, the evapotranspiration stops. This is because the trees use all the available water in the soil. Figure 5.27 displays that the water volume declines in the top two soil layers until the end of February, where after they stay constant. The third layer does not have any change in the water volume, this is due that the root from the trees does not penetrate that deep and can therefore not suck up the water from the deepest layer. From this it can be concluded that the evapotranspiration of the trees simply dries out the soil. Two measures can be done to prevent the drying out of the soil: a different type of trees can be chosen with different properties that does not dry out the soil so fast, but that would mean that the evapotranspiration to cool the air would be equally decreased. The second more sensible thing to do in this case is to inject water into the model. It does not seem far fetched that the trees would be watered in such an environment.

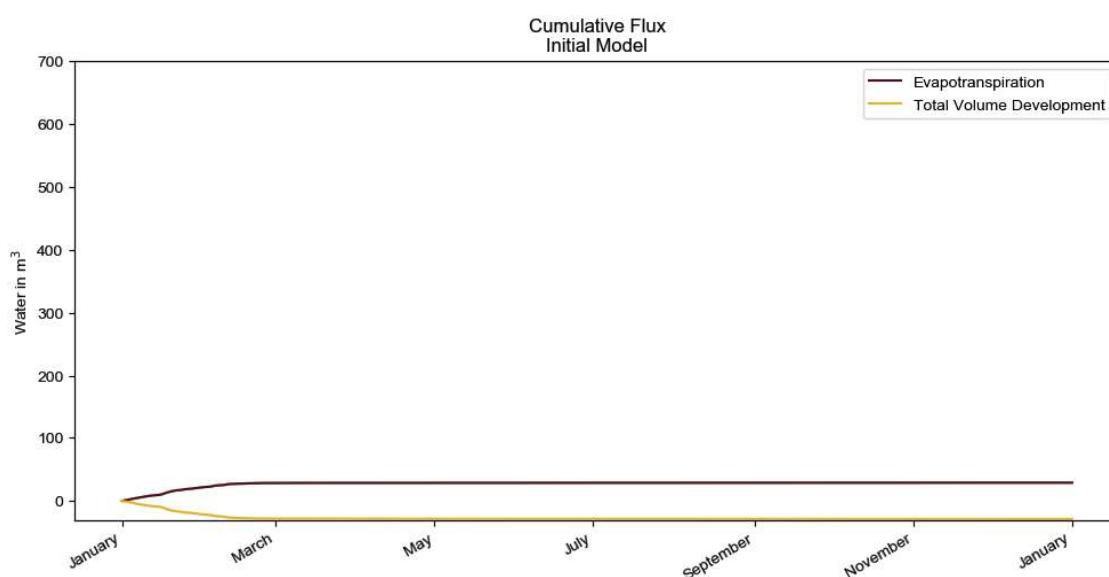


Figure 5.28 - Cumulative water flux.

## Model with Sprinklers

A model with sprinklers are investigated. The sprinklers are introduced to prevent the drying out of the soil and thereby stopping the evapotranspiration. The setup is identical to the model above besides on one parameter: the sprinklers are modelled as rain, which falls in the morning. Therefore, in the weather component is inputted a rain set with 1mm of rain on all mesh faces at 04:00 in morning every day.

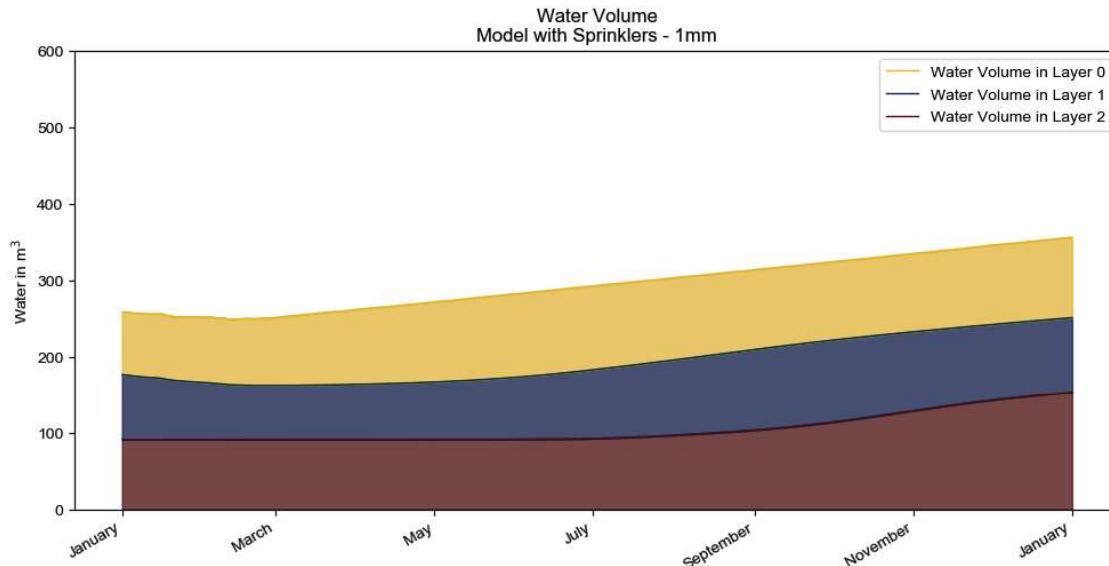


Figure 5.29 - Water volume in the different soil layers for the model sprinkled with 1mm per day

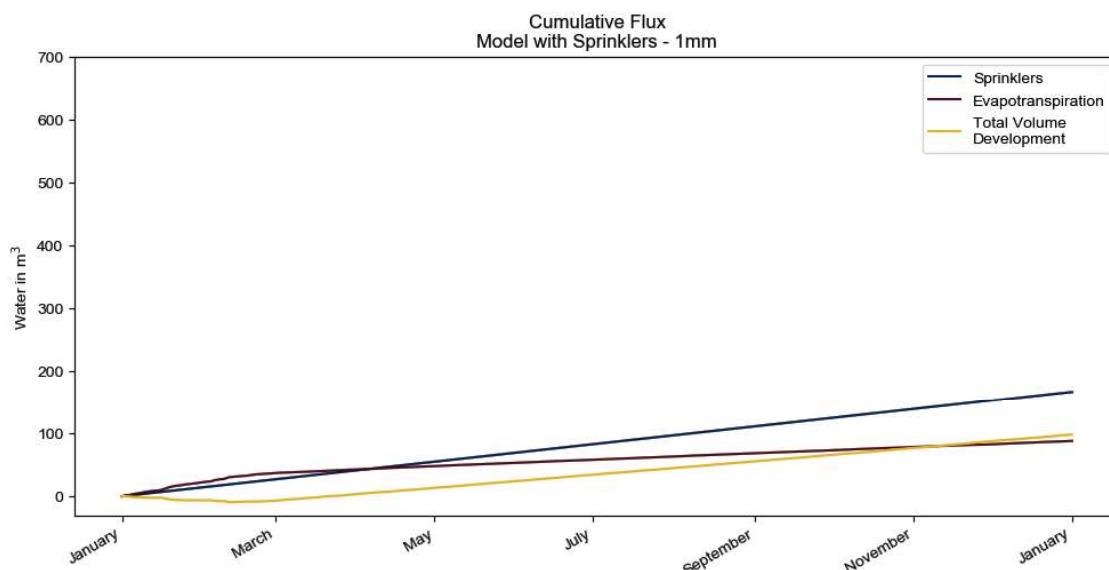


Figure 5.30 - Cumulative water flux for the model sprinkled with 1mm per day

## Water Balance

In contrast to the previous model this model experiences an increase in the water volume of the soil layers. It has a slight drop of water in the beginning of the year primarily due to a loss of water in the middle soil layer. After March the water volume in all soil layers start to increase. This development can also be seen in Figure 5.30, where a plot of the cumulative flux from the case are displayed. Every morning the sprinklers provide  $0.46\text{m}^3$  of water to the model unaffected of everything else. The amount of water evapotranspirated has a steep slope until the beginning of March where after it flattens out and the tangent becomes lower.

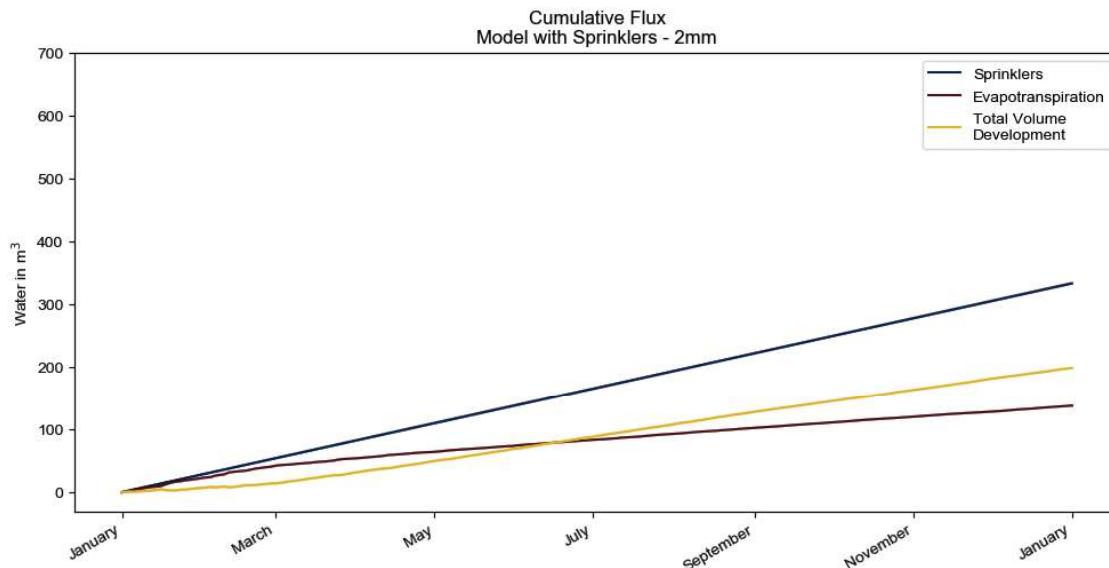


Figure 5.31 - Cumulative water flux for the model sprinkled with 2mm per day

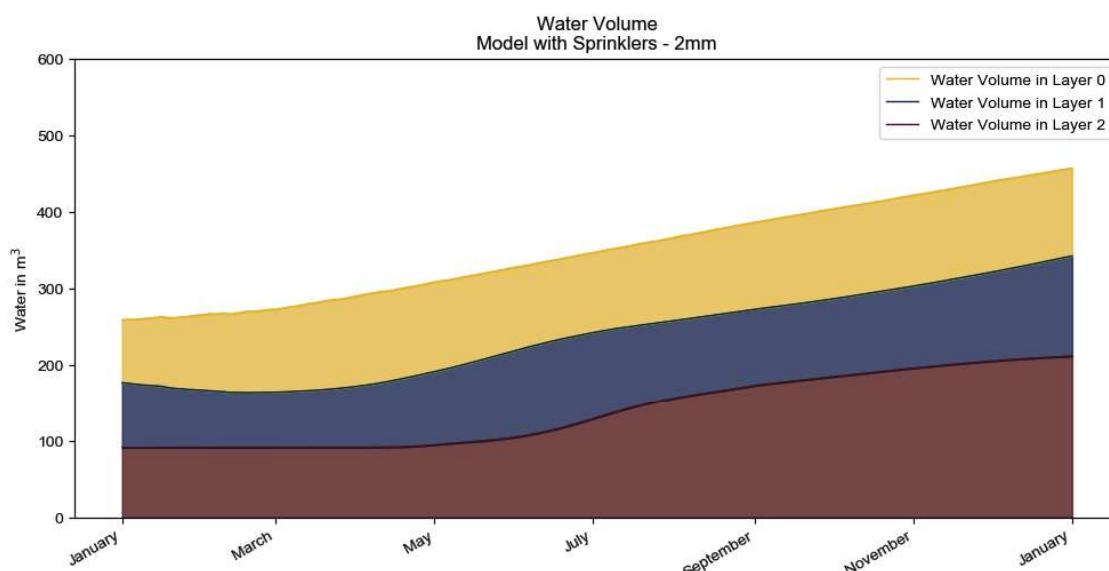


Figure 5.32 - Water volume in the different soil layers for the model sprinkled with 2 mm per day

---

## Double Sprinkler

To investigate if it is possible to maintain the steep evapotranspiration slope, seen in the beginning of the year, a model like the previous one, but with 2 hours of sprinkling, meaning a total 2mm of water on all mesh faces every morning, has been implemented. As seen in Figure 5.31 this leads to a slightly increased evapotranspiration, but not so much as it prevents the total volume of water in the model not to rise. The model ends up storing almost 100m<sup>3</sup> more water in the end of the year compared to the model with only 1mm of sprinkling. In Figure 5.35 the wetness of the top layer is showed. This figure explains why there is only a slight increase in evapotranspiration. It can be seen that in the middle of the year the area under the trees are dried out while the rest of the soil are saturated.

## Focused Sprinkler

The previous model did not seem capable of achieving the goal of the maintaining the steep evapotranspiration slope, therefore a different approach is needed. In this model only, the mesh faces with trees will receive sprinkling, but a higher amount – 3mm for 3 hours each morning. Looking at Figure 5.34, we finally achieve the goal of maintaining the steep slope. What more is that we can see that almost all water put into the model is evapotranspirating out again. Figure 5.33 also shows this in the way that the total volume of water in the model always stays below 300m<sup>3</sup>. The methodology of focused sprinkling will be carried on to the optimized model, as it is shown, that it achieves the best results.

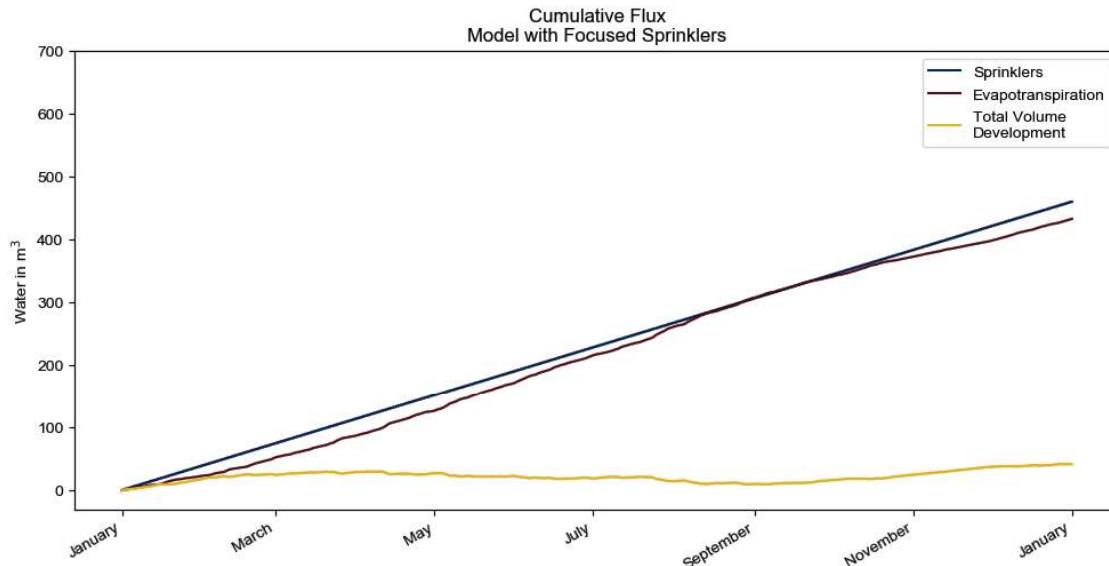


Figure 5.33 - Cumulative water flux for the model with focused sprinkling

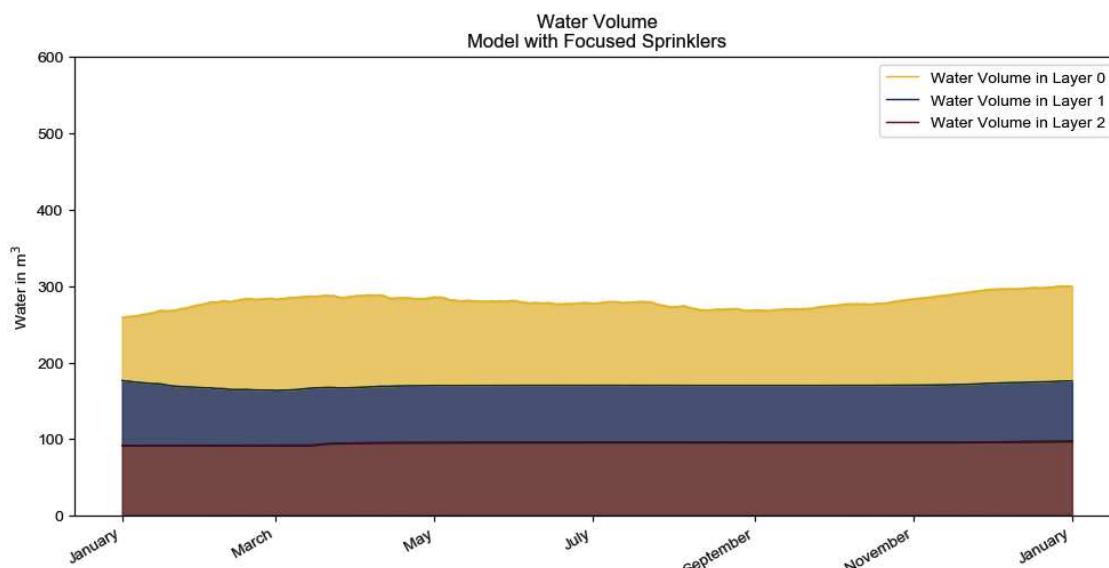


Figure 5.34 - Water volume in the different soil layers for the model with focused sprinkling

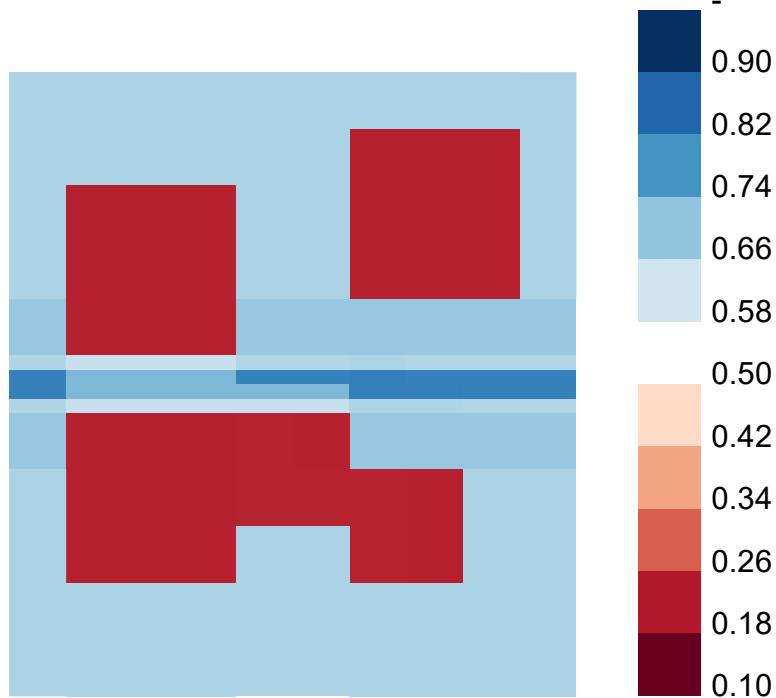


Figure 5.35 - Wetness in the top layer at 12:00 on the 1<sup>st</sup> of June for the model which has 2mm of sprinkling every morning.

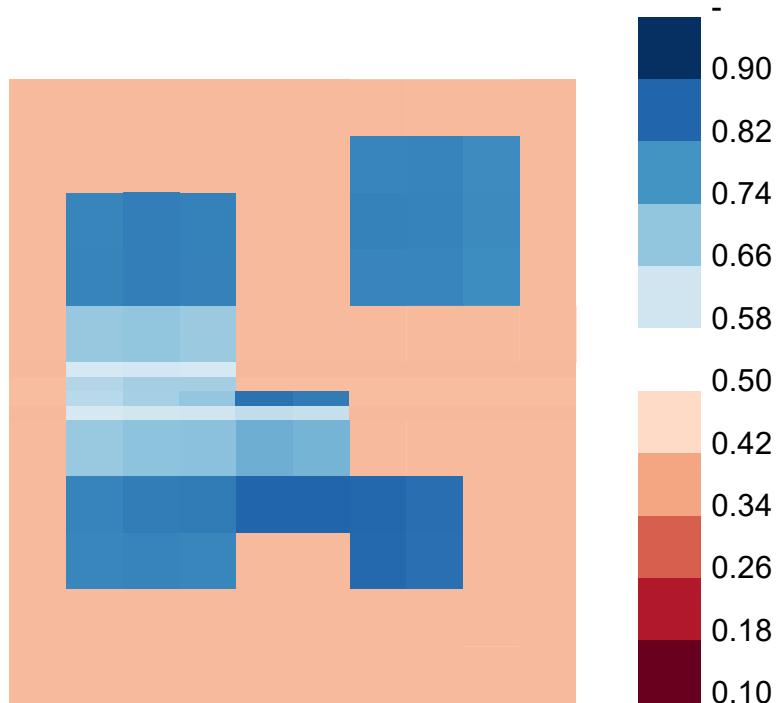


Figure 5.36 - Wetness in the top layer at 12:00 on the 1<sup>st</sup> of June for the model, which utilizes focused sprinkling.

## Parameter Variations of Ground Properties

As it has not been possible to obtain any information on the soil parameters for the case a simple parameter investigation has been conducted in order to see how the retention curve influence the evapotranspiration of the trees. The soil parameters used in this parameter variation study is the default CMF values and the 5 soil types from the European Soil Database – HYPRESS (Panagos et al. 2012). The values can be seen in Table 5.5.

### Water Balance

In Figure 5.37 the resulting water volume given the different soil parameters is displayed over the course of a year. The values vary greatly, but they all follow the same trend; there is a drop in volume in the beginning of the year, where after the volume again start to increase. The lowest curve is the Coarse which has approximately half the water volume at all times compared to the highest curve; Very Fine. The CMF Default soil parameters is the version with the second lowest water volume at all times.

Looking at the cumulative flux for evapotranspiration in Figure 5.38, a relatively wide spread in the resulting cumulative flux can be observed. As with the water volumes in Figure 5.37, the evapotranspiration all follows the same tendency with a larger flux in the beginning of the year – corresponding to the drop in water volume observed in Figure 5.37 – followed by a more gentle increase afterwards. The Medium-Fine soil parameters is the one where the large evapotranspiration continues the longest and is also the ones which ends up on the highest value at the end of the year ( $143.2\text{m}^3$  water evapotranspirated). The default CMF parameters are the ones that evapotranspire the least and ends on  $87.9\text{m}^3$  at the end of the year. This means that with the default CMF soil parameters the trees only evapotranspire 62.9% of what they could if they were in the Medium-Fine soil.

As the soil conditions in Abu Dhabi is unknown and therefore any of these soil types are equally likely. Furthermore, the whole area is going to modified and build upon and it therefore does not seem unlikely to assume that the soil could be changed to maximize the evapotranspiration and therefore the medium-fine soil will be used in the optimized model as it perform the best.

**Soil Parameters**

Type	standard_cmf	coarse	medium	medium_fine	fine	very_fine
K_sat [m/day]	15.0	60.0	12,061	2,272	24.8	15.0
phi [m <sup>3</sup> /m <sup>3</sup> ]	0.5	0.5	0.5	0.5	0.5	0.5
alpha [1/cm]	0.2178	0.0383	0.0314	0.008	0.0367	0.0265
n [-]	1,211	13,774	11,804	12,539	11,012	11,033
m [-]	-1.0	0.274	0.1528	0.2025	0.0919	0.0936
l [-]	0.5	1.25	-23,421	-0.58843	-19,772	2.5

Table 5.5 - Overview of the soil parameters used in the parameter variation study.

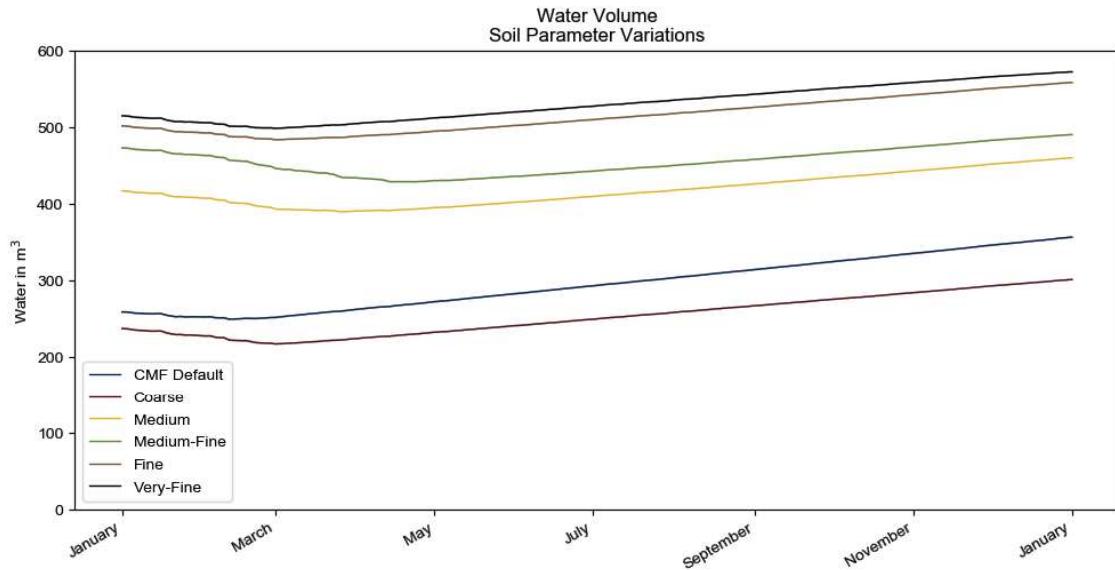


Figure 5.37 - Water volume over the course of a year with different soil parameters.

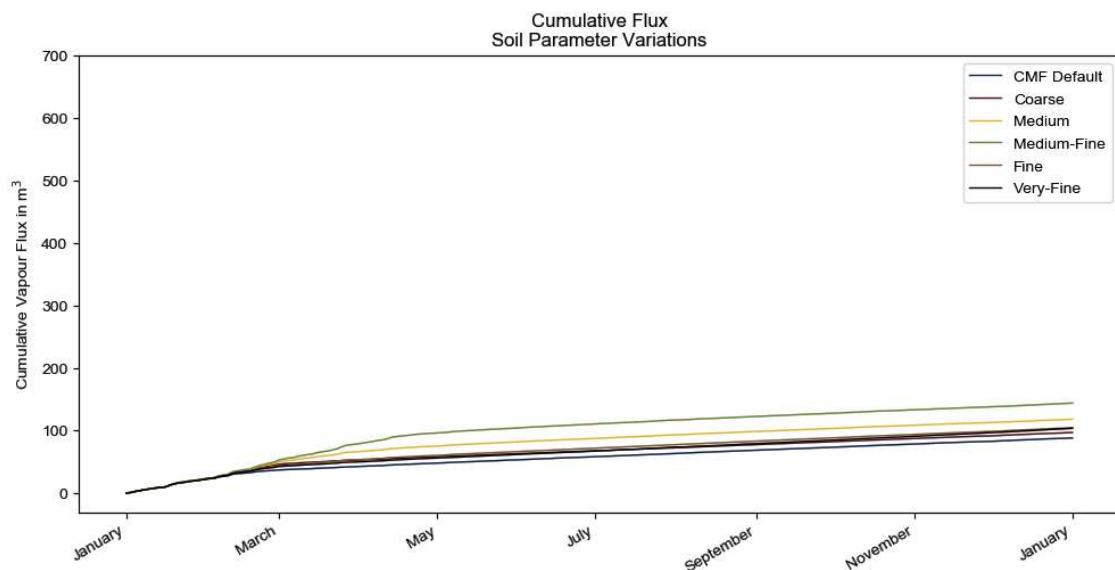


Figure 5.38 - Cumulative evapotranspiration over the course of a year given different soil parameters.

## Wadi in Copenhagen

To study the influence of the weather on the evapotranspiration rate of the case, an identical case study has been done in Copenhagen. The Copenhagen case setup is similar to the initial model from Abu Dhabi, the only difference is that the weather file has been changed. A comparison of the weather data between the two locations can be found earlier in this chapter.

### Water Balance

The development of the water volume stored in the soil layer over the year is quite different to what is seen in Abu Dhabi in the initial model. Even with the same starting point of total water volume the Copenhagen case ends up with approximately 500m<sup>3</sup> compared to Abu Dhabi's 350m<sup>3</sup>. This is caused by the rain fall in Copenhagen. After a year it has rained in total 350m<sup>3</sup>, where there has only been sprinkled 170m<sup>3</sup> in Abu Dhabi. This is the case even though the total evapotranspiration flux in Copenhagen is higher; 95.05m<sup>3</sup> compared to Abu Dhabi's 70m<sup>3</sup>.

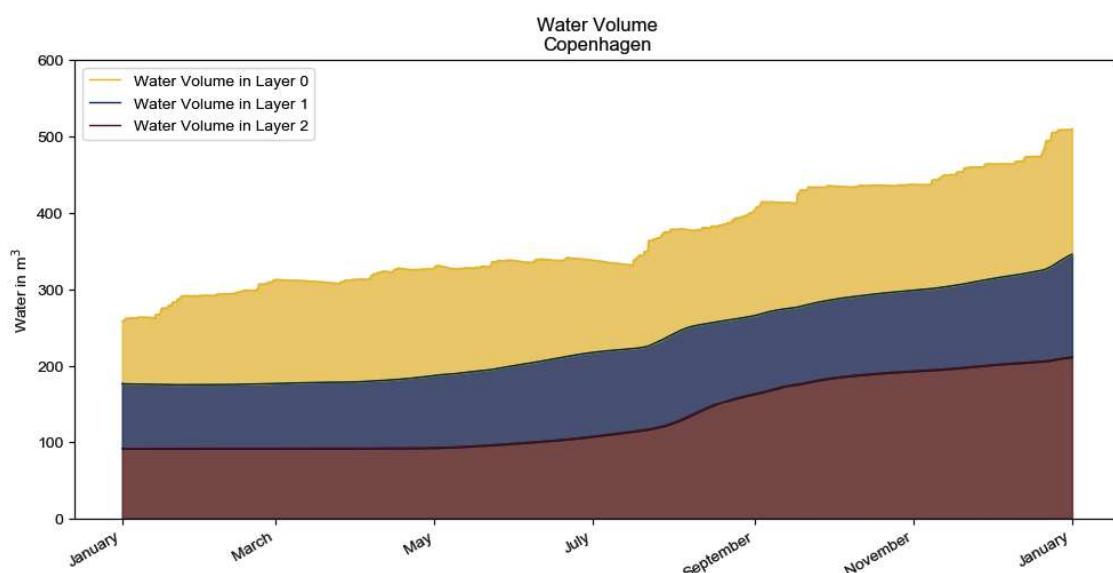


Figure 5.39 - Water volumes for the different soil layers.

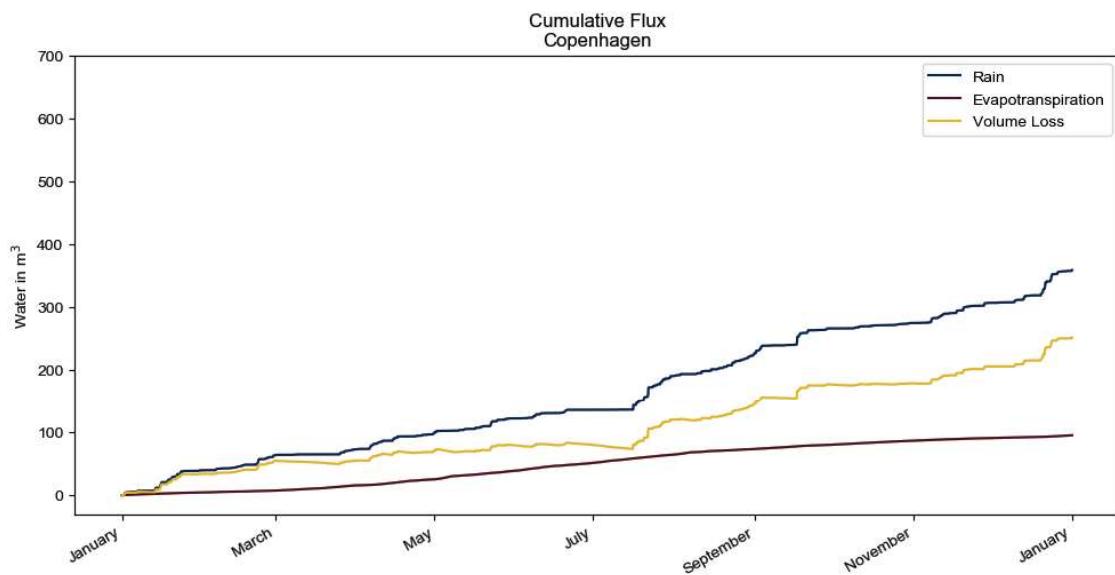


Figure 5.40 - Cumulative flux for the Copenhagen case.

## Wadi with Water

The effects of filling the wadi with water has also been modelled. The model setup is similar to the previous ones besides that soil layers beneath the wadi has been changed to only be one with a thickness of 10cm and the  $K_{sat}$  value of the retention curve is set to 0.1 m/day both measures in an effort to prevent water from infiltrating the wadi soil too much and mostly stay as surface water. On the 40 mesh faces in the wadi 1m<sup>3</sup> of water have been placed initially. To prevent the wadi from drying out, 0.2mm of rain falls each hour on the wadi mesh faces to keep the water level. The rain is used as the supplier for water in the wadi, as it is not possible to supply water to the wadi otherwise with the current version of Livestock.

In Figure 5.41, the water volume of the case can be seen. After a drop in the beginning of the year, the total volume of water stays constant. This can again be examined with the flux. The drop in total volume is caused by the evapotranspiration from the trees in the beginning of the year, just as we have seen before. The evaporation from the water in the wadi is almost identical to the added water from the rain and therefore does not contribute to any significant change in the water balance. This also means that the water level in the wadi stays constant throughout the year.

In the end it is possible to evaporate 65m<sup>3</sup> from the wadi over a year. A part of the architects' proposal for master plan is that wadi is filled with water – at least a portion of the time. Therefore, the water filled wadi will be used in the optimized model as well.

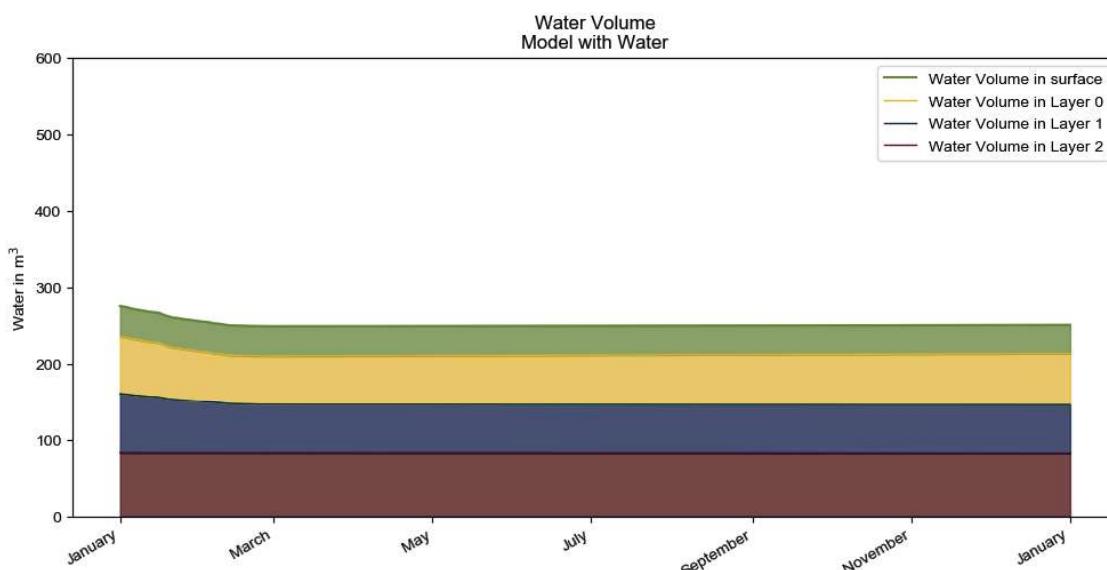


Figure 5.41 - Water volumes for the different soil layers.

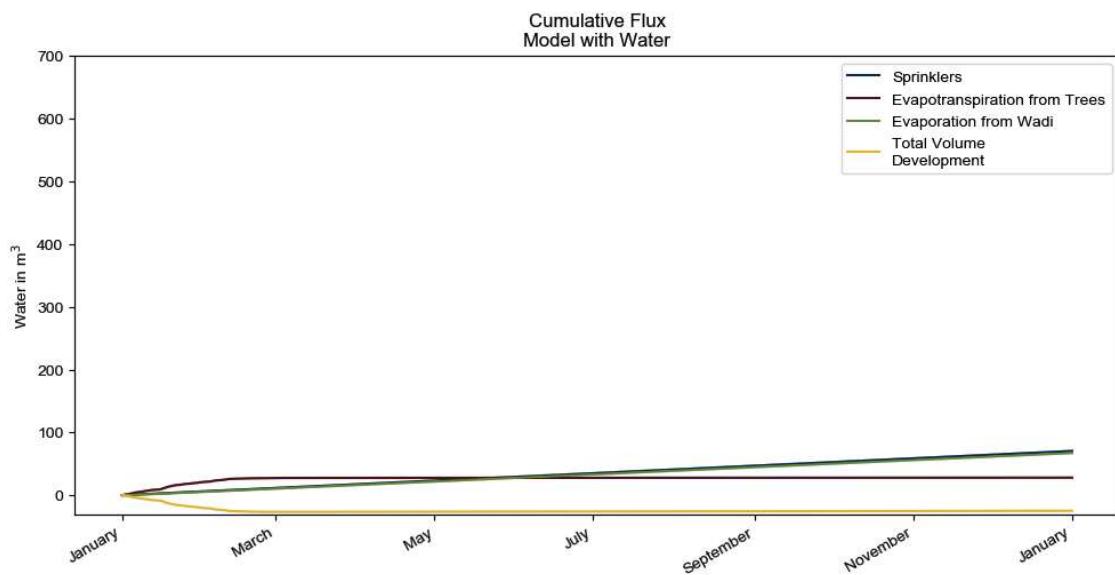


Figure 5.42 - Cumulative flux for the Copenhagen case.

## Optimized Model

In an effort to get the highest evapotranspirational flux from the case the above listed investigations have been combined into an optimized model. The model has the medium-fine retention curve, utilizes focused sprinkling and the wadi is filled with water. It was possible to increase the amount of evapotranspiration by increasing the amount of water sprinkled. The optimized model has for the first 121 days, an usage of 9mm of water on each mesh face with trees, after that the usage increases further to 18mm and then it decreases to 12mm for the last 91 days of the year.

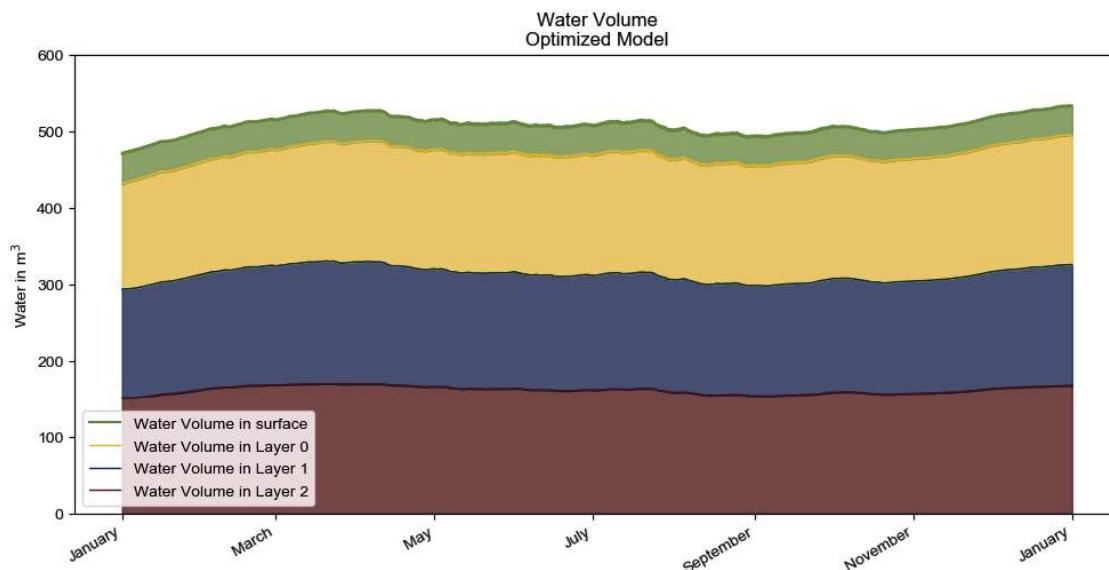


Figure 5.43 - Water volume in the different layers for the optimized model.

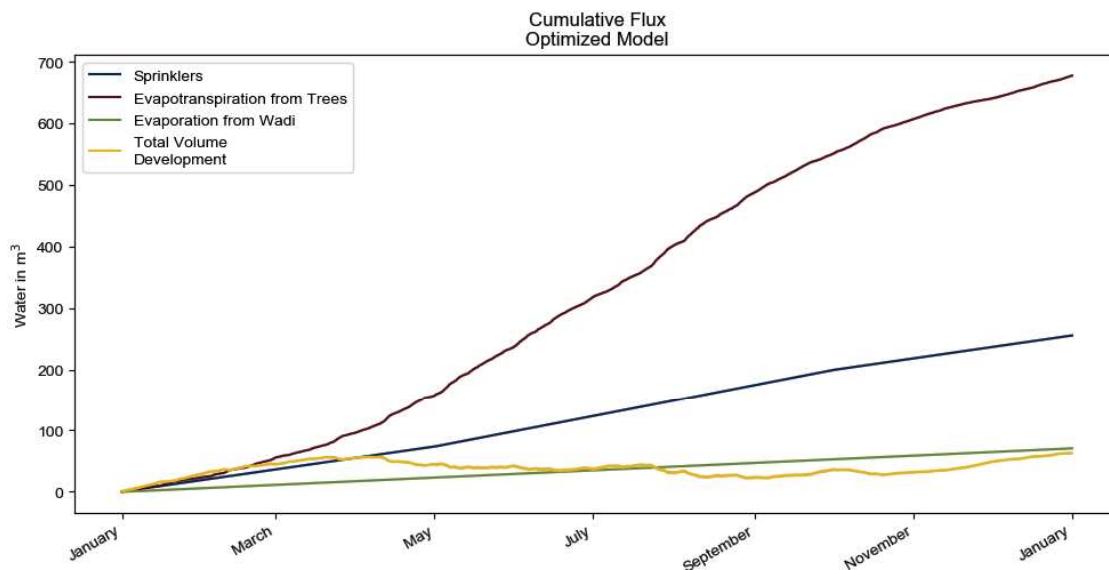


Figure 5.44 - Cumulative flux for the whole optimized model.

---

## Water Balance

The total volume of water in this case is fluctuating around  $500\text{m}^3$ . As seen in Figure 5.43, it has been possible to significant increase the evaporation of the trees by combining the previous findings. The total evapotranspiration from the trees ends a bit below  $700\text{m}^3$  after a year. The evaporation from the wadi on the other hand ends at  $60\text{m}^3$  and can almost be considered neglectable in this comparison.

# 5.5 - AIR CONDITION ANALYSIS

## Optimized Model

The developed atmosphere model is used to compute new air temperatures and relative humidity for the optimized model. It is identical described in the section about the atmosphere model.

In Figure 5.45 and 5.46 the corresponding new air temperature and relative humidity is shown. As it can be seen it is almost impossible to see the change as compared to the weather file. If you investigate the temperature differences between the weather file and temperature at the big tree cells – as done in Figure 5.47 – this marginal difference becomes clear. 98.5% of the time it is only possible to decrease the air temperature with less than 0.3°C. This is a very small change and for most people unrecognizable.

The explanation for this shall be found in how the atmosphere model handles the air volume. Looking at Equation 10 - 12 it becomes clear that the wind speeds effect on the size of the volume is the governing parameter here. To make an example: Let us say that we have a mesh face of 2m x 2m and an air volume boundary height of 10m. This gives us a total volume of 40m<sup>3</sup>. Following Equation 10 this would be air volume input to the atmosphere model at a wind speed at 0m/s. Let us now say we have the same geometry but a wind speed of 1m/s. This will lead to an air volume of 81,283.3m<sup>3</sup>! This is a size of air volume that the trees can not influence. We also saw it in the section about the theoretical limit that some enormous amounts of water are needed. Amounts of vapour that the trees can not produce.

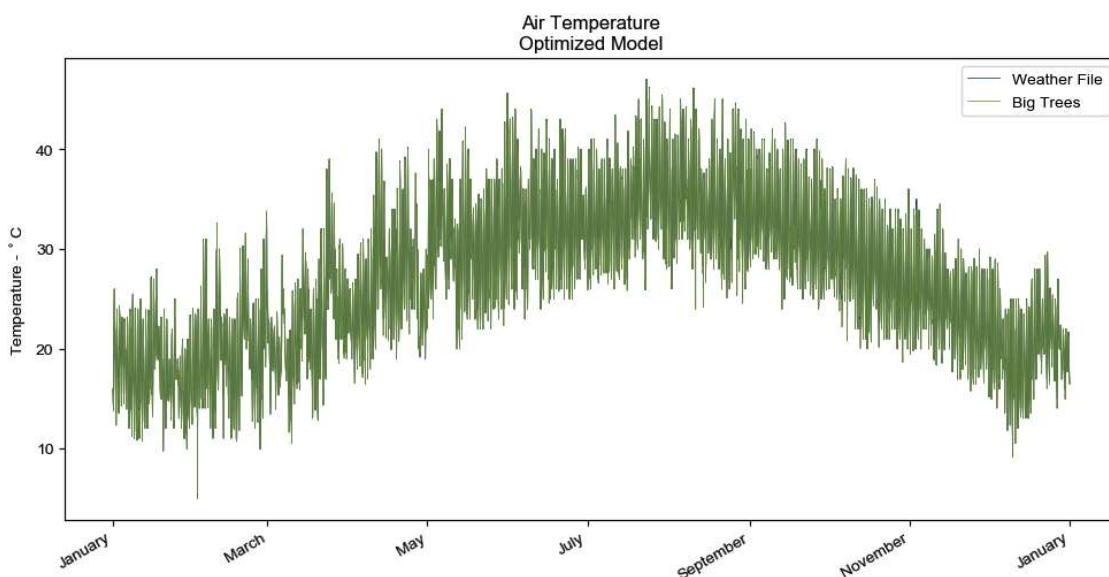


Figure 5.45 - Resulting air temperature for the optimized model.

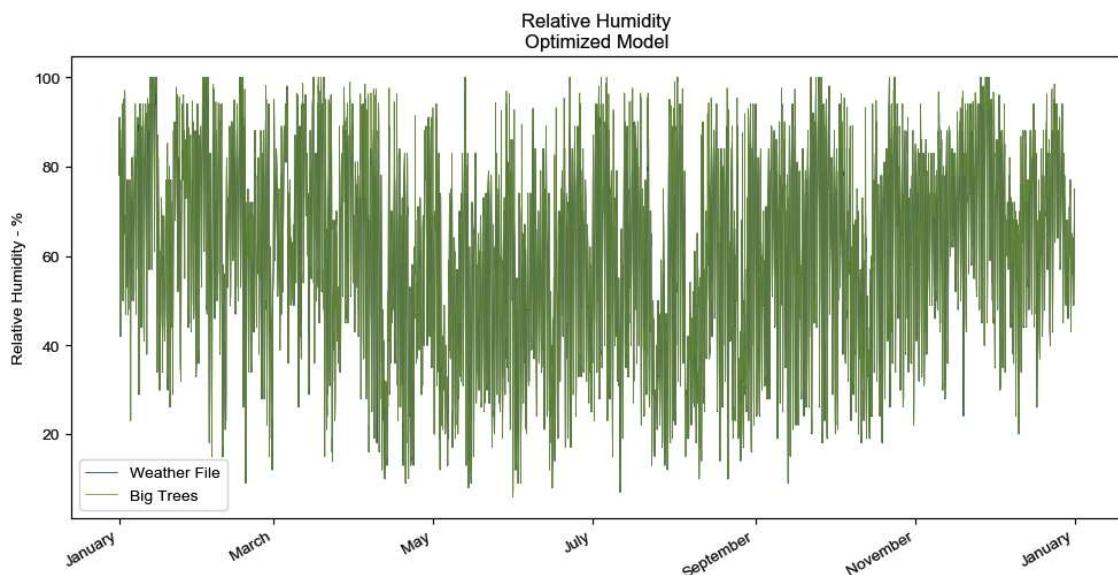


Figure 5.46 - Resulting relative humidity for the optimized model.

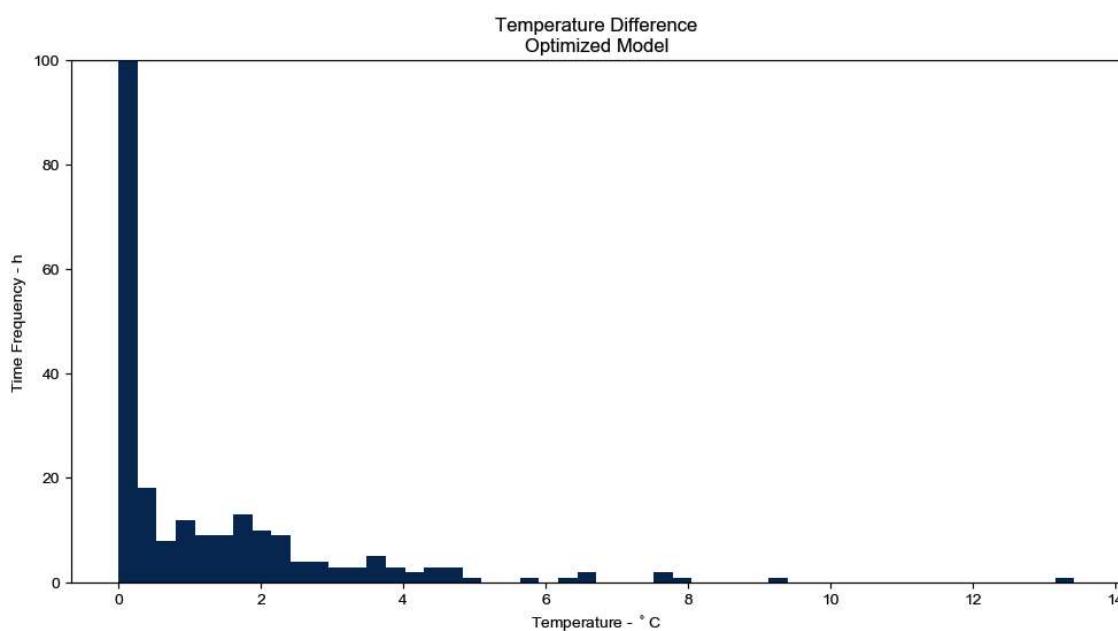


Figure 5.47 - Histogram for the temperature difference between the resulting air temperature under the big trees and the weather file's temperature.

## Optimized Model Without Wind

In Figure 5.48 a similar histogram to the one described in the previous section are presented. This histogram however shows the impact that different wind speeds have on the temperature reduction. Four scenarios are showed. The scenario from before with the actual wind speed data from the weather file, a scenario where the wind speed is kept constant at 1m/s throughout the year, a second one where it is kept at 0.5m/s and lastly one where there is no wind at all during the year. For the two scenarios where the wind speed is kept constant at 0.5m/s and 1m/s it is not possible to decrease the temperature with more than 1°C and the majority lines much lower. The scenario with actual wind speed data behaves as described before with the majority below 0.3°C and some hours with high impact (when the wind is calm). But the scenarios with no wind shows satisfactory results with a median of 4°C and a 2.2°C 25% quartile and a 7.3°C 75% quartile.

Examining the case where there is no wind; a substantial effect from the evapotranspiration can be observed. When looking at the yearly values – Figure 5.49 and 4.50– it can be observed that most of the time the relative humidity is increased, and the air temperature is lowered. A closer look at the 1<sup>st</sup> of January and June – Figure 5.51 and 5.52 – the same trend is visible. The relative humidity is in general increased over both days but especially during midday the difference becomes large, where the weather file's relative humidity drop drastically while the modified relative humidity only drops a bit. The effect from the increased relative humidity is also visible on the air temperature, as it stays low and only increase a little bit during daytime, while the air temperature from the weather file increases a lot – particularly in June.

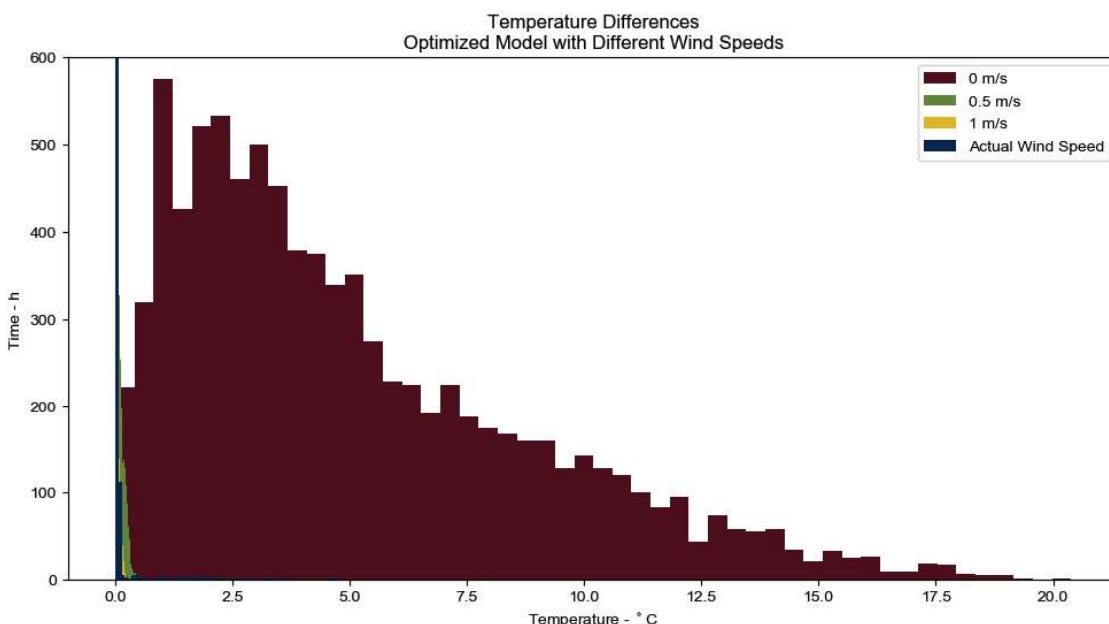


Figure 5.48 - Histogram for the temperature difference between the resulting air temperature under the big trees with different wind speeds and the weather file's temperature.

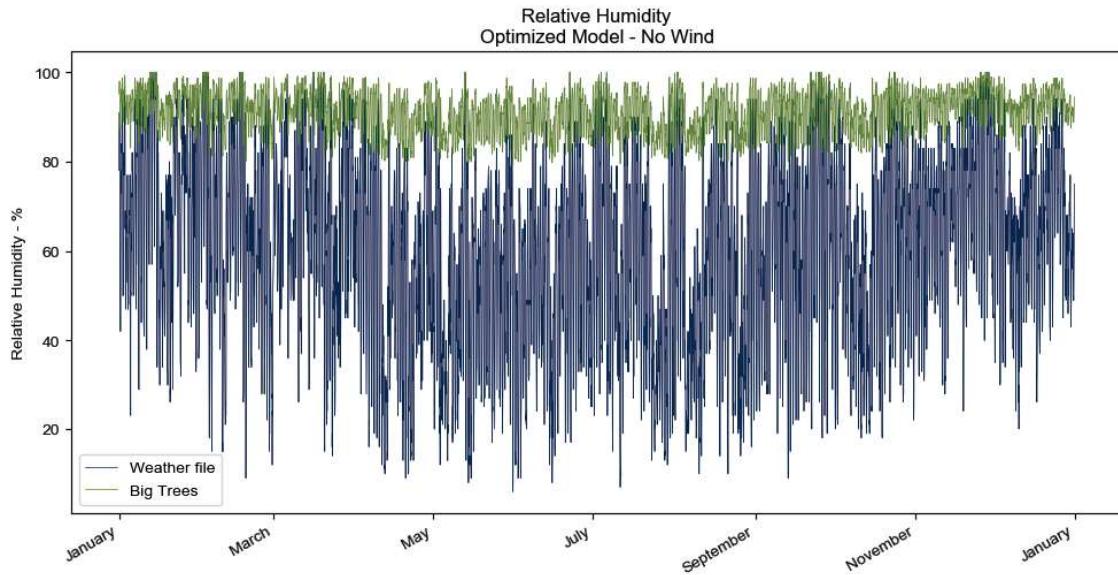


Figure 5.49 - Resulting relative humidity for the optimized model with no wind.

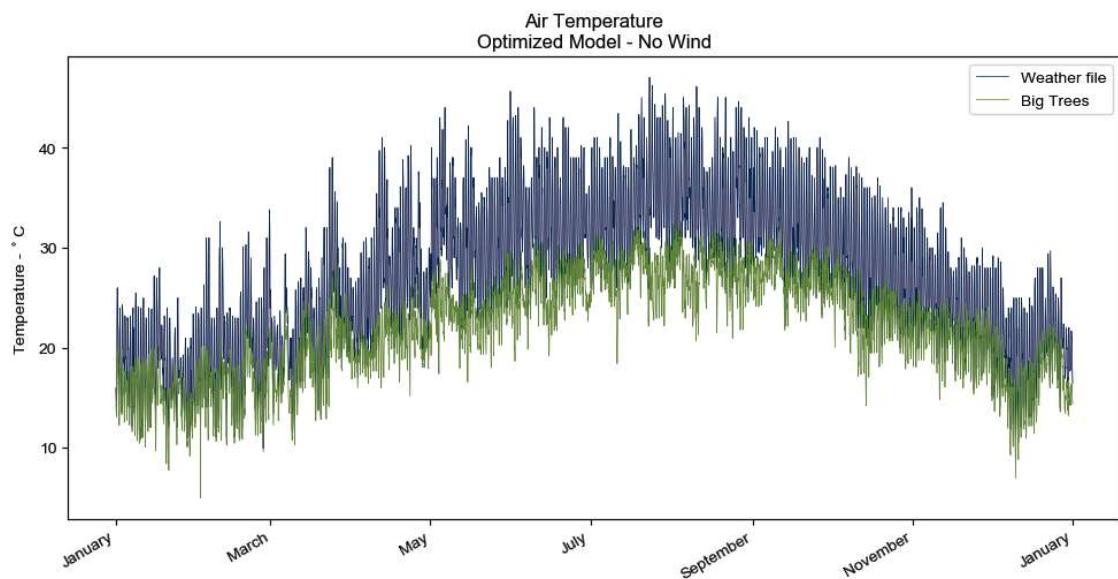


Figure 5.50 - Resulting air temperature for the optimized model with no wind.

In Figure 5.51 and 5.52 only the values from under the big trees are shown. Examining the difference in temperature and relative humidity between the wadi, small trees and the big trees will reveal no difference. This is due to the way the model is constructed. The model does not allow the full amount of water vapour to be used if that would lead to excess 100% relative humidity.

As there is so big differences in between the case with wind and without, and they rely on different assumptions, both with their pros and cons, they will both be used in the comfort calculations as edge scenarios of how much it is possible to alter the thermal comfort in the model.

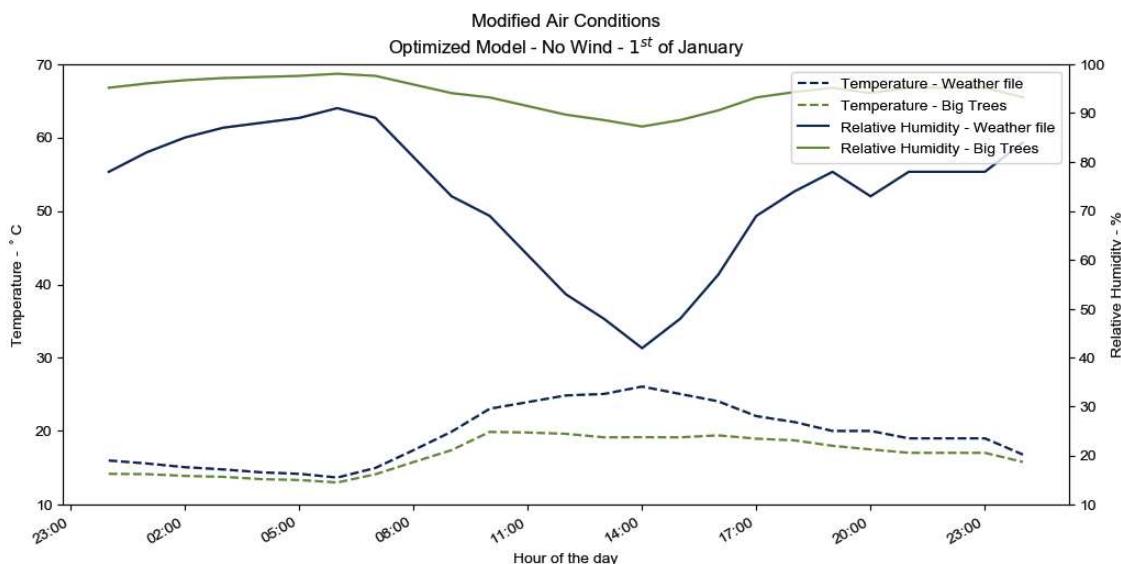


Figure 5.51 - Resulting relative humidity for the optimized model with no wind on the 1<sup>st</sup> of January.

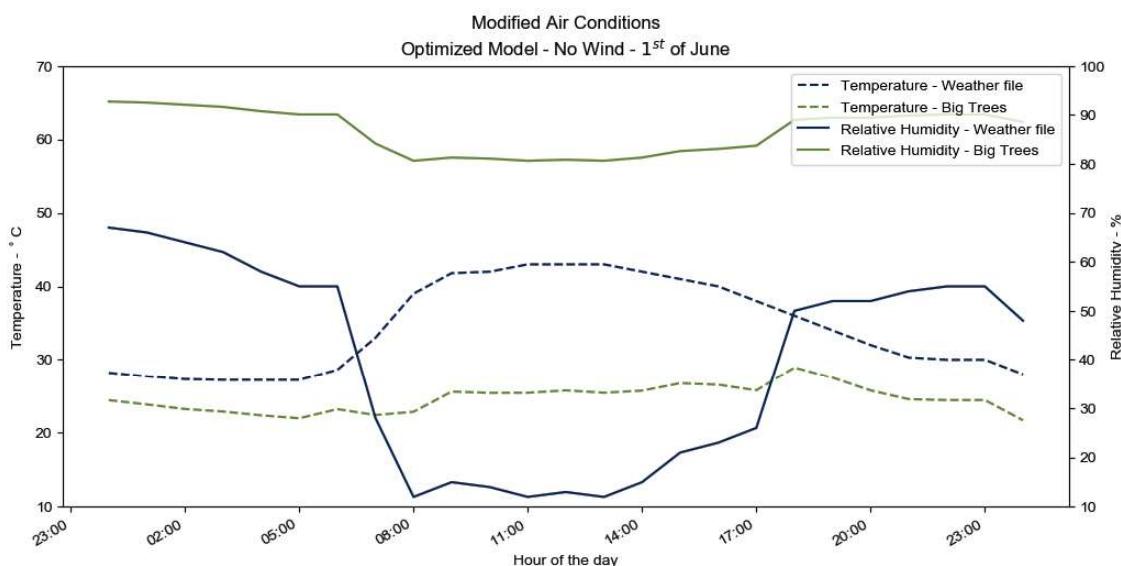


Figure 5.52 - Resulting relative humidity for the optimized model with no wind on the 1<sup>st</sup> of June.

## 5.6 - MRT ANALYSIS

The mean radiant temperature is one of the substantial contributors for thermal comfort as seen in Figure 3.8, where changing the MRT will lead to substantial changes in UCTI. As stated in Section 3.3, MRT consist of two components the surface temperature and the view factor to that surface. This section will examine the MRT of the Wadi case study.

### Script

In Figure 5.53 a Grasshopper script for setting up the MRT computation is shown.

- 1) Initialization of Ladybug Tools
- 2) Geometry generation of the Wadi
- 3) Preview displays it for the user in the Rhino viewport
- 4) Load the EPW weather file
- 5) The ground temperature is computed.
  - a) A Honeybee zone is generated to represent the ground
  - b) EnergyPlus is ran to compute the surface temperature of the ground throughout the year.
  - c) The temperature results are loaded into Grasshopper.
  - d) A postprocessing of the surface temperature is needed to match the correct surface temperature with the mesh faces of the geometry.
- 6) The mean radiant temperature is computed.
  - a) The results from the atmosphere model is loaded
  - b) The view factors are calculated
  - c) The MRT are calculated and adjusted for the influence of the sky
  - d) The results are saved to a text file.
- 7) Thermal Comfort is calculated. This will be described in its own section later.

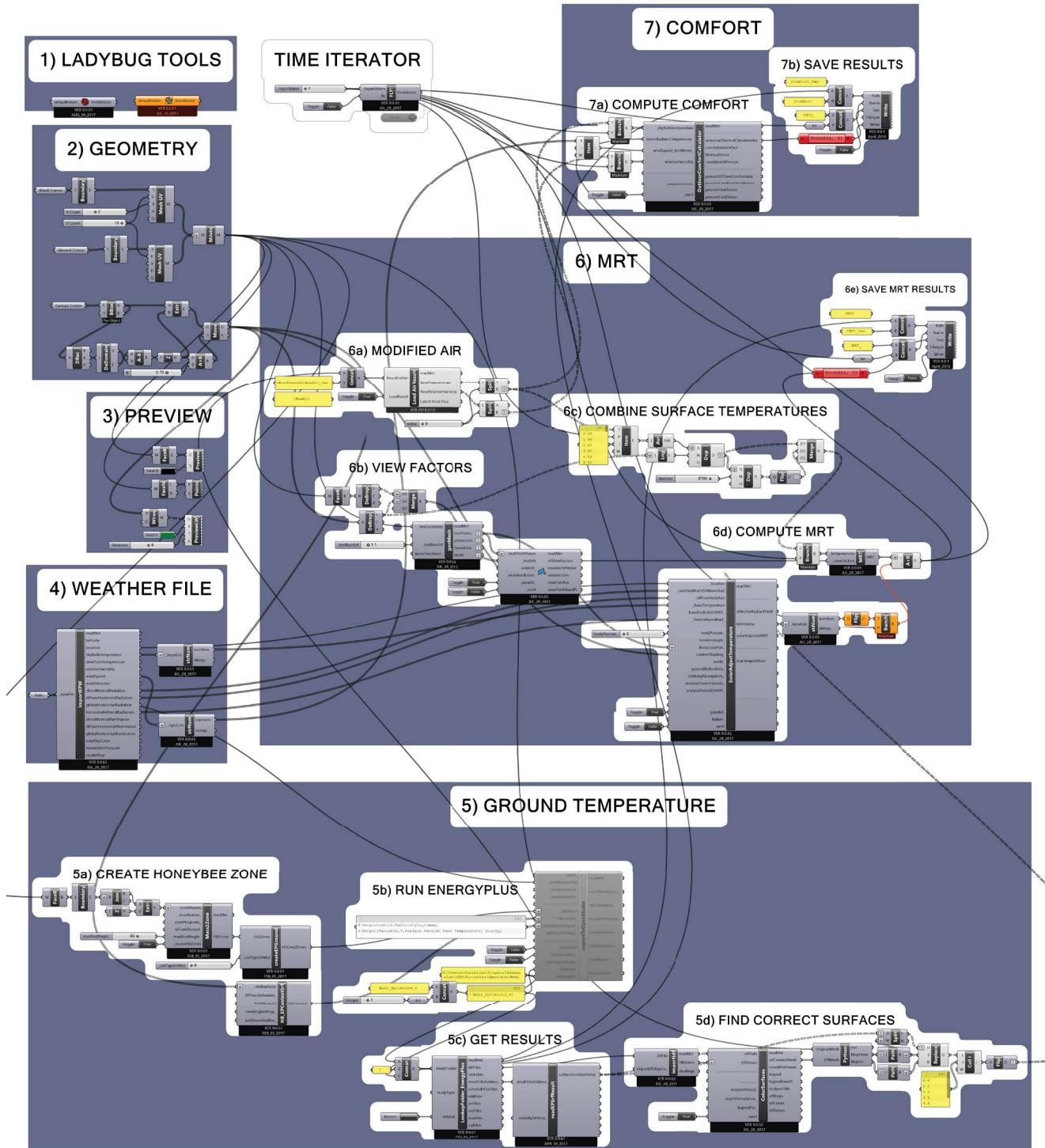


Figure 5.53 - Grahopper script for computing MRT

## Ground Temperature

The surface temperature of the ground is a substantial contributor to the MRT as it is “close” to people and therefore has a large view factor. In this case study the ground is modelled as dry sand. In Figure 5.54 the surface temperatures of the ground have been visualized for the 1<sup>st</sup> of January and the 1<sup>st</sup> of June, both at 12:00 and 17:00.

In January at 12:00 the mesh faces, which lies within the shadows of the trees have a surface temperature between 20°C and 40°C. The southern bank of the Wadi is also in shadow and therefore also has a surface temperature of around 20°C. The mesh faces outside of the tree shadows have a higher temperature around 50°C. This is a quite high temperature and would probably be uncomfortable to touch with the naked hand. At 17:00 the sun is set, and the temperature of the ground surface is almost uniform below 30°C.

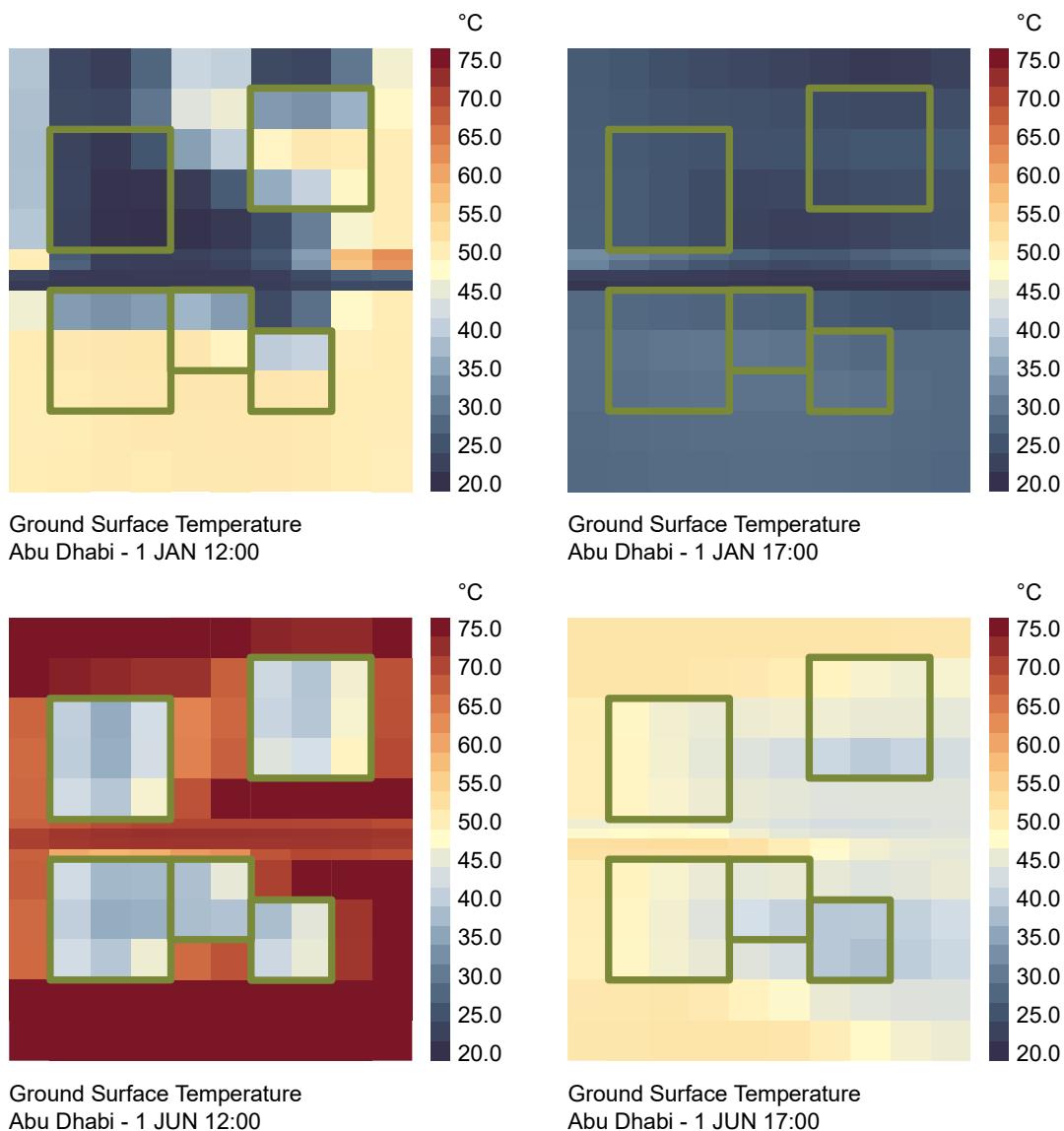


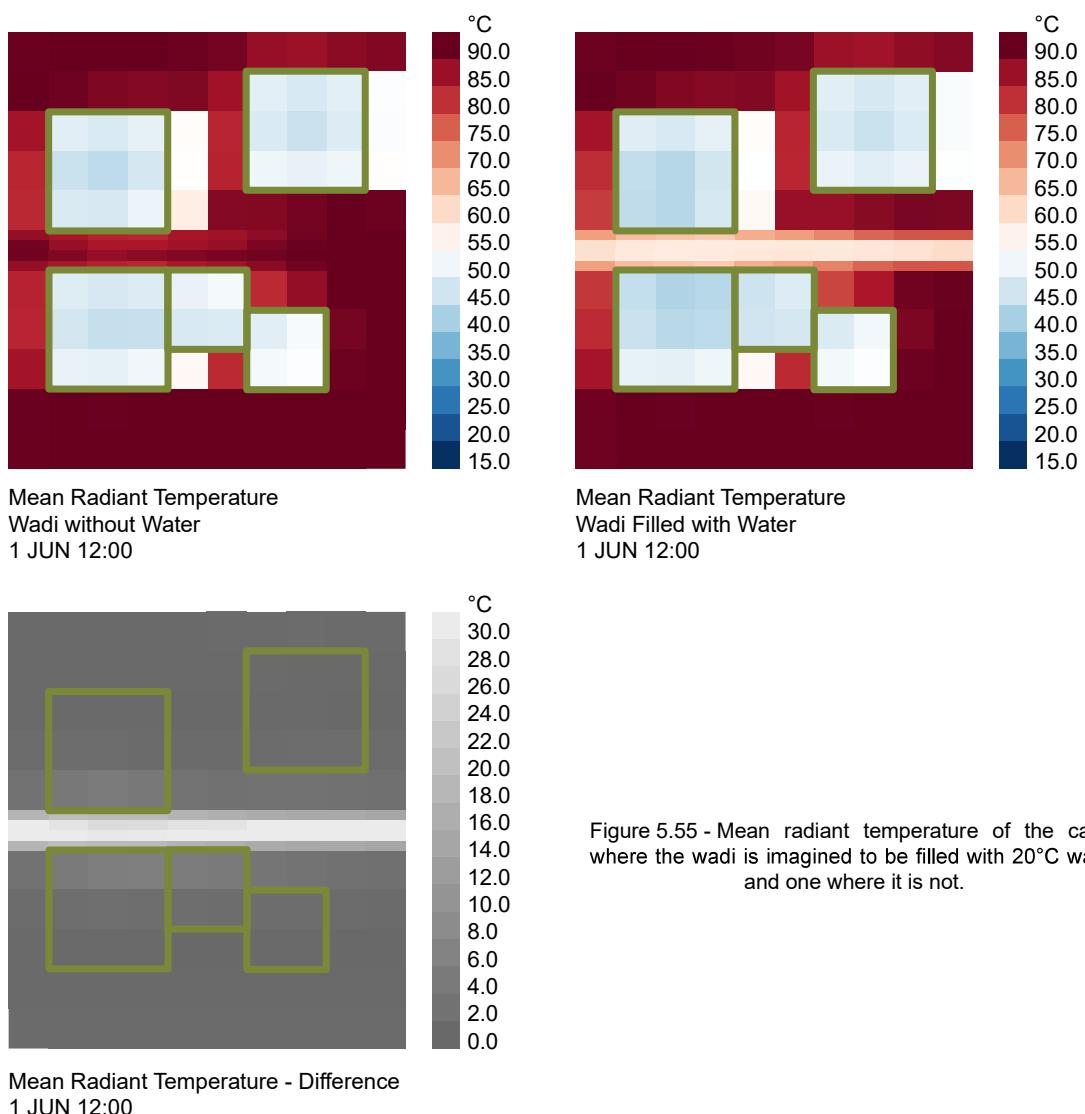
Figure 5.54 - Ground surface temperatures two hours in January and two hours in June.

---

In June the solar radiation has increased and so has the ground surface temperatures. The sun is also much higher in the sky and we see that the shadows of the trees are not pointing north anymore but are under the trees now. Under the trees and in their shadows the surface temperatures are around 40°C in their periphery the temperature increases with around 20°C. In the areas away from the trees the surface temperature reaches a staggering 75°C! At 17:00 the temperature has dropped to between 45°C - 50°C for the most part of the model besides in the shadows of the trees where down to 35°C can be seen.

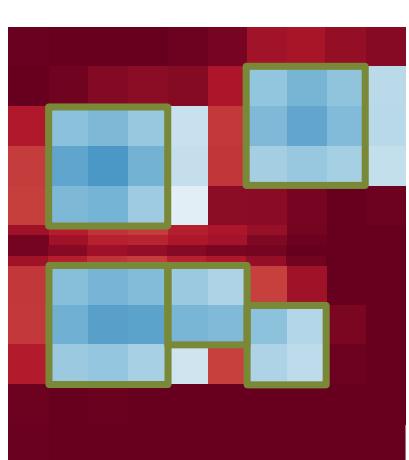
## Water Filled Wadi

To investigate the effect filling the wadi might have on the mean radiant temperature a comparison has been made and can be seen in Figure 5.55. The ground surface temperatures and sky temperature are simulated the same way as before and the surface temperature of the canopies are assumed equal to the air temperature from the weather file. Figure 5.55 (top left) depicts that setup, where Figure 5.55 (top right) is the same but all the surface temperatures in the wadi are changed to 20°C to mimic the surface temperature of water flowing through. Comparing the two figures – as done in Figure 5.55 (bottom left) – it is visible that has a significant impact on the mean radiant temperature above the wadi, up to 30°C, while it drastically decreases the further you move away – when you are 2m away from the wadi the effect has decreased to 1°C.

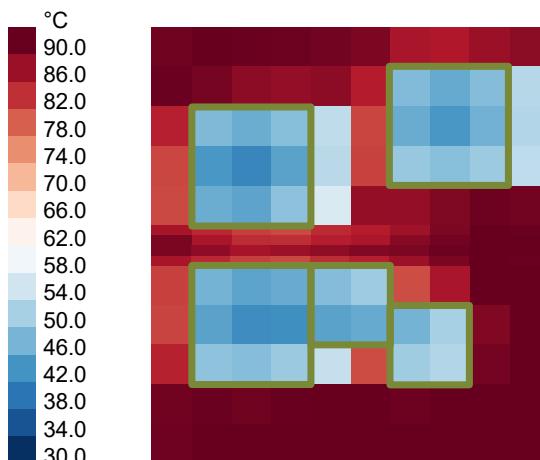


## Canopy Temperatures

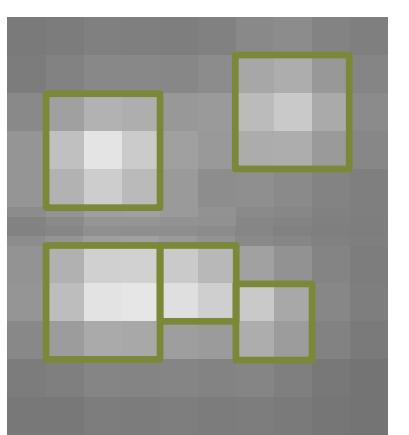
Besides lowering the ambient air temperature, it is interesting to look at what effect the evapotranspiration of the trees have on the mean radiant temperature, if you assume that the canopies have the same temperature as the ambient air. Figure 5.56 visualizes this difference. Figure 5.56 (top left) shows the mean radiant temperature for the optimized model. As seen earlier in this chapter the difference in air temperature between the output of the optimized model and the weather file are almost neglectable and can in this case therefore be seen as representable of the weather file. Figure xx is displaying the mean radiant temperature for the optimized model without any wind. In figure xx it can be observed that the mean radiant temperature below the canopies of optimized model is around 2°C higher than the mean radiant temperature below the canopies in the case of the optimized model without any wind. 2°C is of course a smaller difference than the 10°C achieve by the wadi, but as you cannot really utilize the effect the wadi gives – as you cannot stand on water – the effect of lowering the temperature of the canopies are more useful.



Mean Radiant Temperature  
Optimized Model  
1 JUN 12:00



Mean Radiant Temperature  
Optimized Model without Wind  
1 JUN 12:00



Mean Radiant Temperature - Difference  
1 JUN 12:00

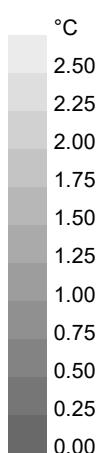


Figure 5.56 - Mean radiant temperature of the case, where the canopies have the surface temperature of the modified air temperature from the optimized model and one wehre they takes the temperature from the optimized model without any wind.

# 5.7 - COMFORT ANALYSIS

## Comfort with Current Methods

To compute the outdoor comfort for the Wadi Case without taking any evapotranspiration into account a series of Ladybug Tools have been used. The Grasshopper script used to do this is almost identical with the one shown in Figure 5.53. However, it differs on two points:

6a) is not present in this setup and the values from the EPW file are used instead for the canopy temperature.

7) Is used. Here the mean radiant temperature, relative humidity, air temperature and wind speed – the last three parameters are all from the EPW weather file – are combined in the UTCI comfort component from Ladybug Tools.

For each mesh face the Universal Thermal Climate Index is calculated. 1<sup>st</sup> of January and 1<sup>st</sup> of June will again be used to study the scenario.

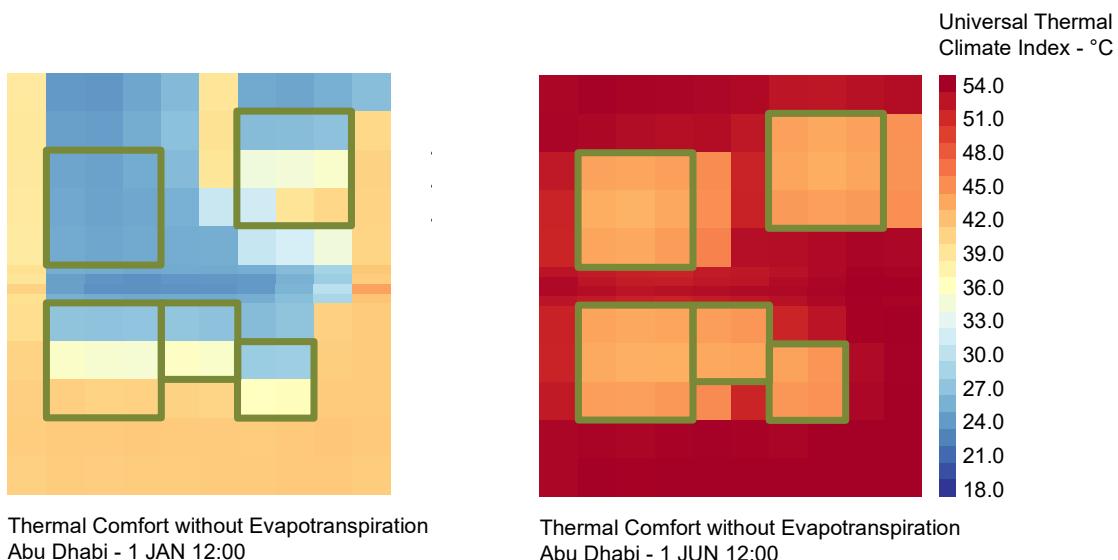
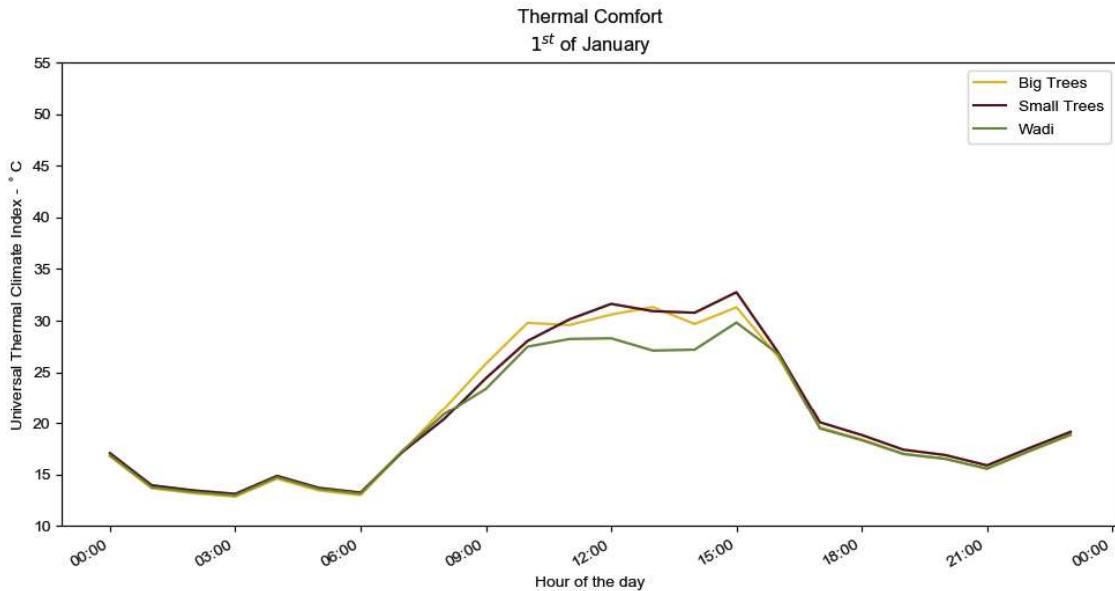
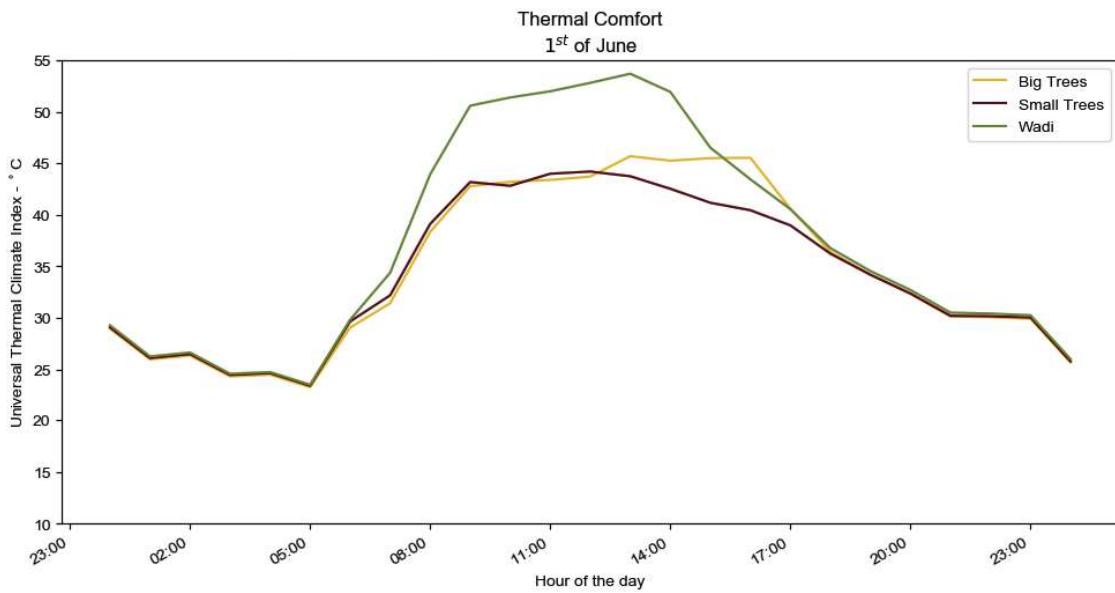


Figure 5.57 - Universal Thermal Climate Index at 1<sup>st</sup> of January (to the left) and 1<sup>st</sup> of June (to the right).

Figure 5.58 - UTCI at the 1<sup>st</sup> of JanuaryFigure 5.59 - UTCI at the 1<sup>st</sup> of June

### 1<sup>st</sup> of January

Figure 5.58 shows the universal Thermal Climate Index for the 1<sup>st</sup> of January. On the figure are plotted three curves; Big Trees, Small and Wadi. They represent the average value found on the mesh faces, where the big trees, small trees and the wadi are located. The wadi curve also represents the non-shade areas of the model. It can be seen that the temperature is stable around 15°C until sunrise around 6:00. Thereafter all three locations on the model experience an increase in UTCI until around 10:00, where after the curves again are stable around 30°C until 15:00, where a drop in temperature can be observed until midnight.

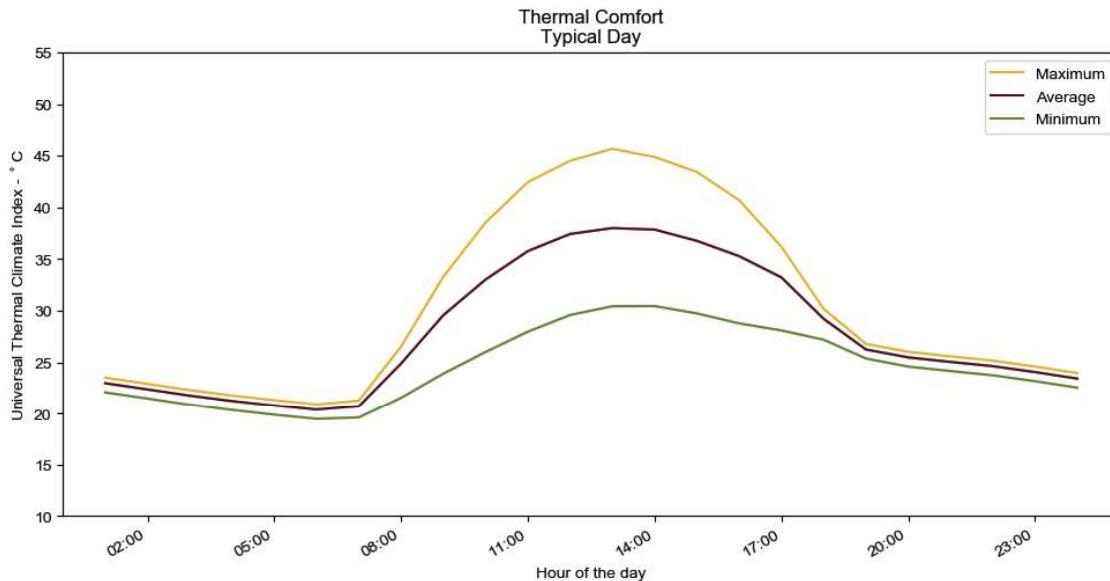


Figure 5.60 - Universal Thermal Index for a typical day.

The situation at 12:00 is investigated further and displayed in Figure 5.57. The sun has risen to zenith and everything which is not in the shadows of the trees has a UTCI of 40°C. This means that it is only if you are under the trees that you are thermal comfortable otherwise you are at Strong Heat Stress (>32°C). In the shadows of the trees you would only experience a Moderate Heat Stress at 30°C.

### 1<sup>st</sup> of June

Figure 5.59 shows the universal Thermal Climate Index for the 1<sup>st</sup> of June. Here the UTCI is stable at 25°C before sunrise for all three curves. After 6:00 the curves begin to rise, quite early on the wadi curve's rate of change is higher than the two others. When all three curves stabilize again the wadi is more than 10°C hotter. The two others stay just below 45°C. The wadi curve and small tree curve starts to decrease some hours before the big tree but at 18:00 they again all have the same UTCI.

Again 12:00 is investigate further and can be seen in Figure 5.57. At this time of the year not even the shadows of the trees will keep you from a Strong Heat Stress. Under the trees there is ~43°C and outside of their shadows the UTCI is even higher: 54°C. It can thereby been seen that you experience the same thermal comfort under the shade of the trees in June as you do in the sun in January.

---

## Typical Day

In Figure 5.60 a plot of UTCI for the typical day are presented. It is constructed by taking the average of all mesh faces for each hour. It shows us that you can expect an UTCI between 20°C and 25°C from 18:00 to 06:00. During the day it is higher and the UTCI peaks at 13:00. The maximum is at 45°C and the minimum at 30°C at 13:00 resulting in a spread of 15°C between the average UTCI of the model between the hottest and coolest day of the year.

## Conclusion

During the cooler season the shadows of the trees are enough to keep you thermally comfortable, but when you transition into the hot season the shadows are no longer enough to keep you comfortable. It is therefore an obvious place to try to improve the thermal comfort. Since the relative humidity is so low at the problematic hours it is obvious to try to improve the thermal comfort with evapotranspiration.

## Comfort Optimized Model

To compute the outdoor comfort for the Wadi Case with evapotranspiration a similar setup to the previous section has been used. The Grasshopper script used to do this is almost identical with the one shown in Figure 5.53. However, it differs on one point:

7) Is used. Here the mean radiant temperature, wind speed from the EPW weather file and the modified relative humidity and air temperature from the atmosphere model– are combined in the UTCI comfort component from Ladybug Tools.

For each mesh face the Universal Thermal Climate Index is calculated. 1<sup>st</sup> of January and 1<sup>st</sup> of June will again be used to study the scenario.

### 1<sup>st</sup> of January

The development of the UTCI over the 1<sup>st</sup> of January is very similar to the development for the model, where evapotranspiration is not considered, we will see that later. It starts out around 15°C for all three curves. The UTCI starts to increase at 06:00 until about 10:00, where they find a stable temperature around 30°C with the wadi a bit lower around 27°C. From 16:00 onwards they again begin the decrease in temperature.

### 1<sup>st</sup> of June

The 1<sup>st</sup> of June has likewise the 1<sup>st</sup> of January a similar development to the model without evapotranspiration.

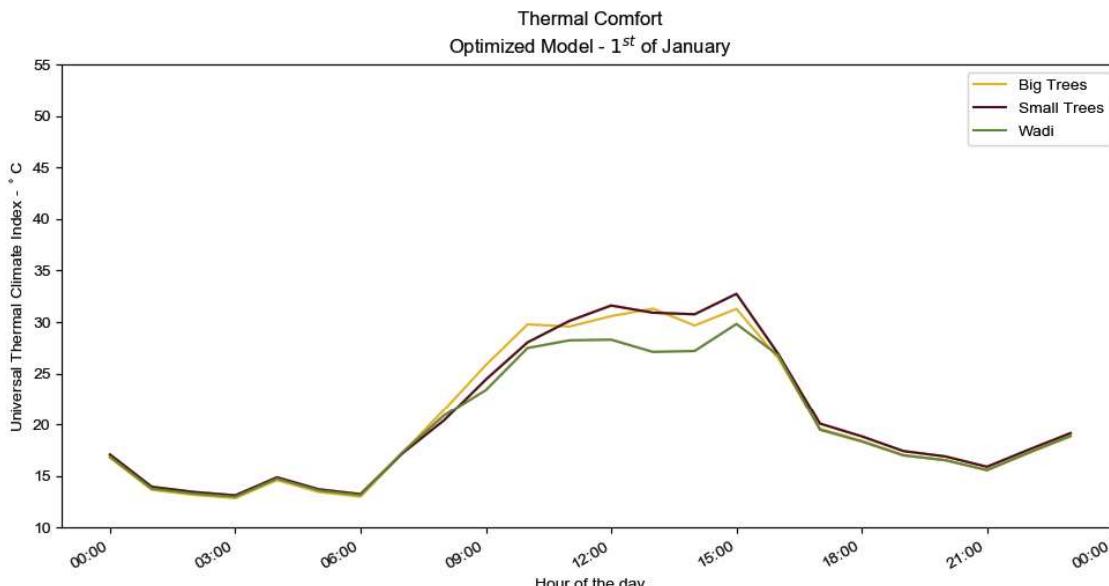
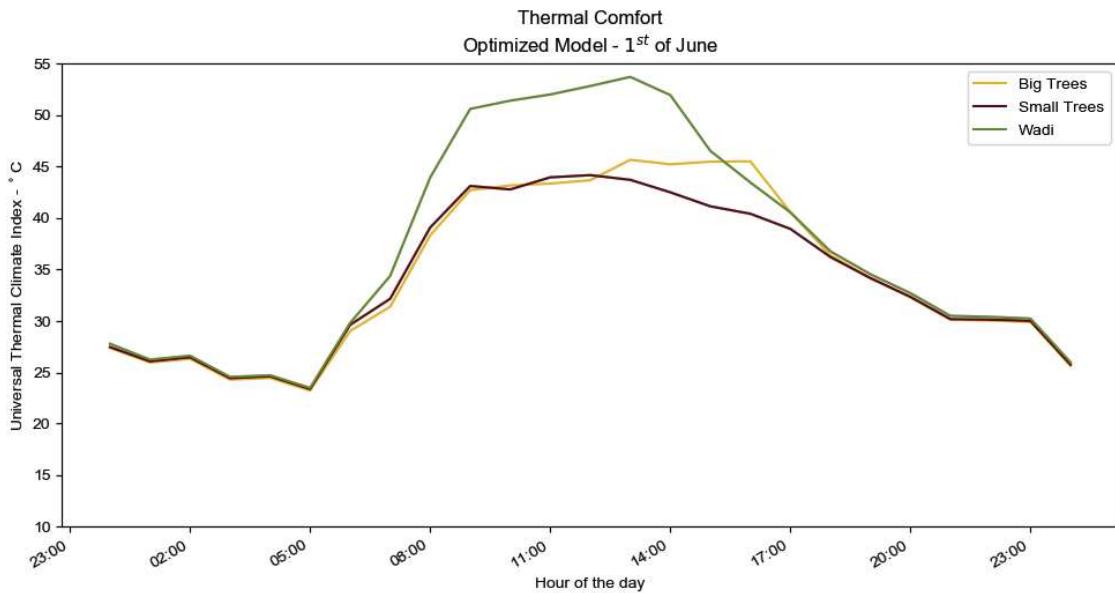
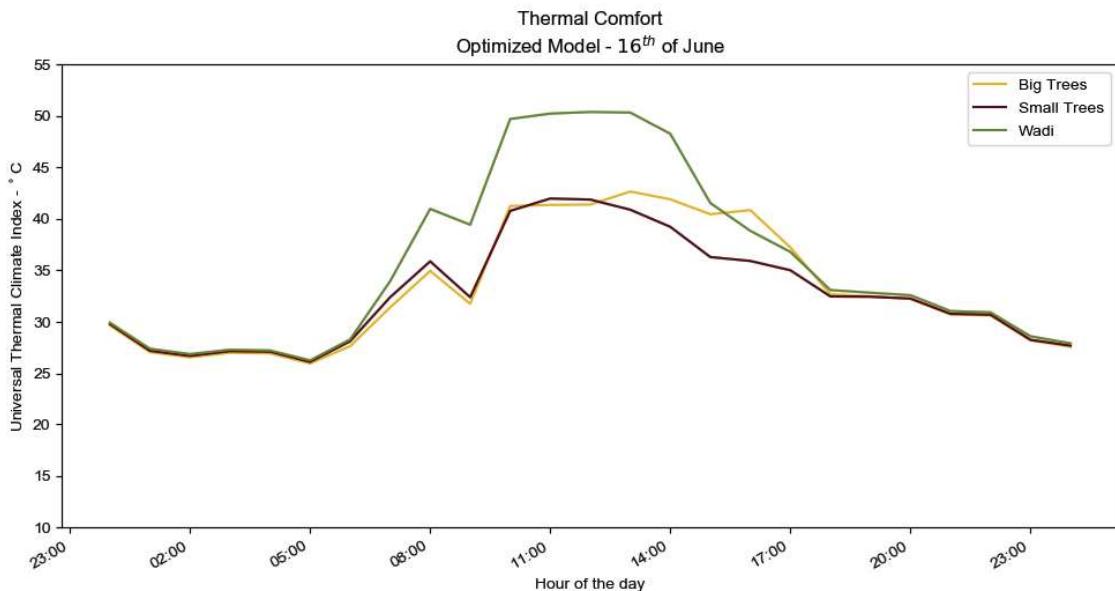


Figure 5.61 - UTCI the 1<sup>st</sup> of January for the optimized model

**16<sup>th</sup> of June**

The 16<sup>th</sup> of June is much more interesting to look at. At first it might look like it is similar to the 1<sup>st</sup> of June, which for the most part it is. However, an interesting phenomenon happens. At 10:00 there is a drop in UTCI for all three curves. Besides the decrease in UTCI, which is quite substantial for the big and small tree curve ~5°C, the values before and after seem unaffected by this drop.

Figure 5.62 - UTCI the 1<sup>st</sup> of June for the optimized modelFigure 5.63 - UTCI the 16<sup>th</sup> of June for the optimized model

To further examine this and find the course of this drop in UTCI Figure 5.64 and 5.65 has been made. The two figures show the environmental components of the UTCI together with the resulting UTCI for the two dates. Where UTCI, MRT, air temperature and relative humidity seem to follow the same pattern both days – low values during the night and high during the day, and the opposite in case of relative humidity – the wind speed does not. The 1<sup>st</sup> of June it is fluctuating around 3m/s the whole and at the 16<sup>th</sup> it is decreasing from midnight until 10:00 where it hits a local minimum, where after it increases to 7m/s and again from 17:00 starts to decrease again.

The interesting phenomenon that happens at 10:00 is caused by the wind speed. It is obvious, when it is displayed as it is in Figure 5.65. The wind speed reaches 0m/s and suddenly the relative humidity peaks and the UTCI and air temperature drops. The 16<sup>th</sup> has been chosen exactly because of this. It is not so often that there are calm wind conditions in Abu Dhabi during day, that mostly happens at night. As we can see in both the 1<sup>st</sup> and the 16<sup>th</sup> have calm wind conditions at night, which seem to affect the relative humidity and air temperature a bit, that effect is not large enough to cause any substantial change to the UTCI.

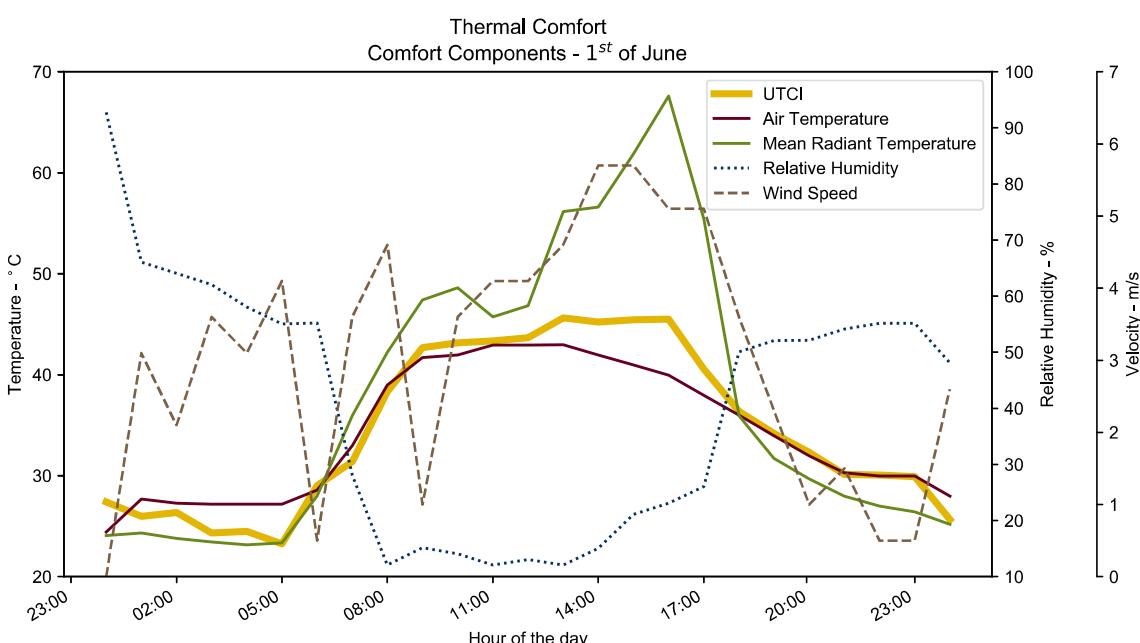
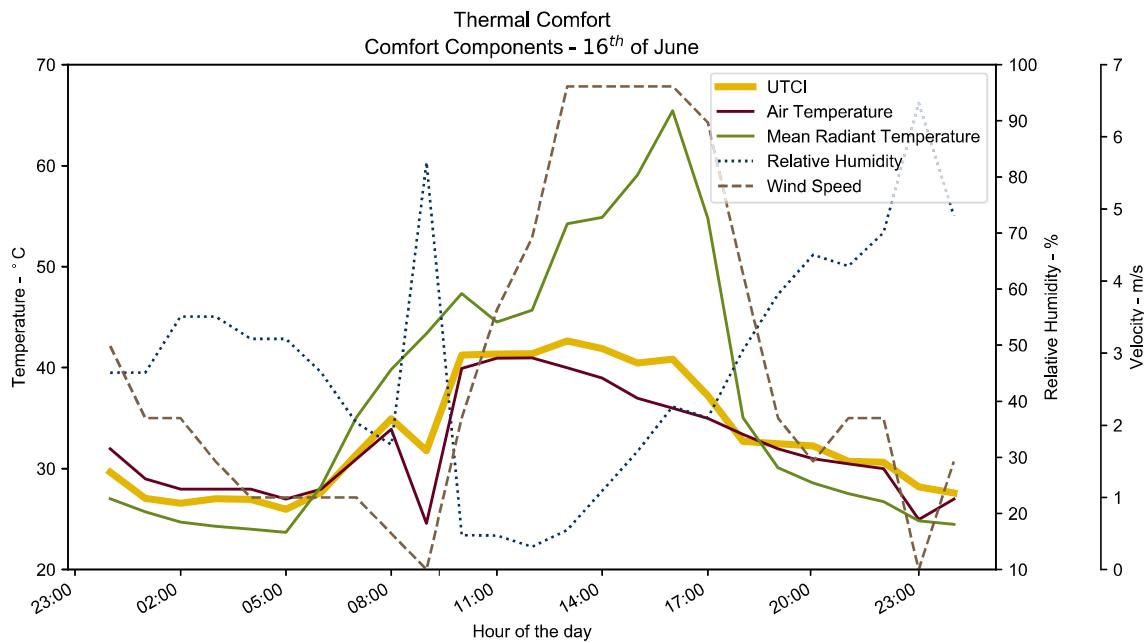


Figure 5.64 - Environmental inputs that make up the UTCI for the 1<sup>st</sup> of June

Figure 5.65 - Environmental inputs that make up the UTCI for the 16<sup>th</sup> of June

## Comfort Comparison Between Models

As hinted in the previous section the difference between the model with and without evapotranspiration is most of the time neglectable. Figure 5.66 and 5.67 plots the average UTCI of the big tree cells for both the model with and without evapotranspiration. The 1<sup>st</sup> of January it is not possible to see any difference. The 1<sup>st</sup> of June has a small difference in UTCI at midnight caused by the calm wind conditions that we observed in

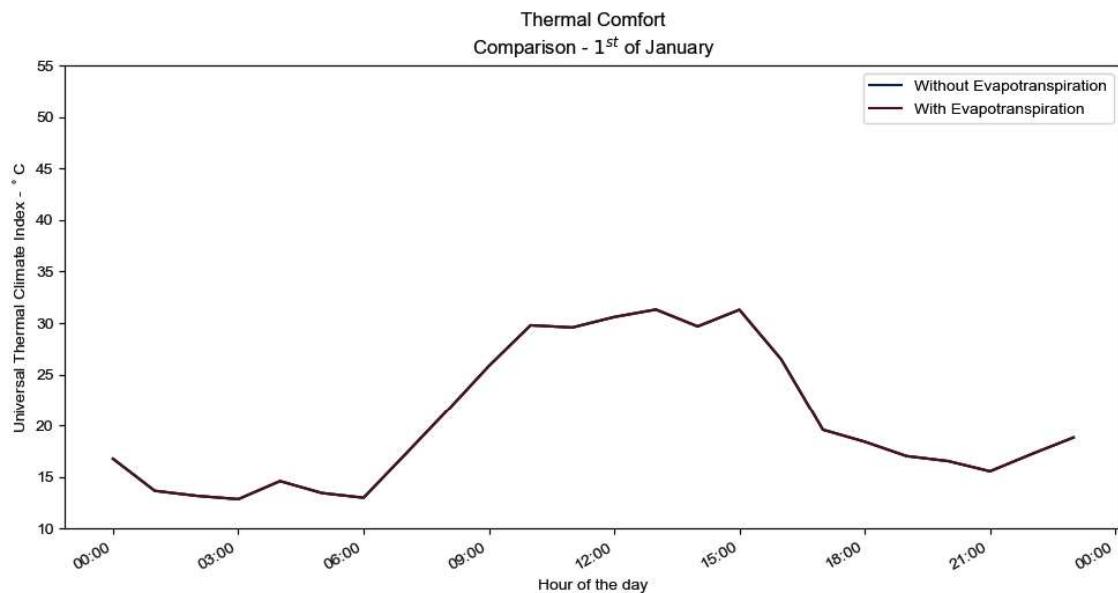


Figure 5.66 - Comparison of UTCI between the model with and without evapotranspiration in the 1<sup>st</sup> of January. The values presented are the average at the cells of the big trees

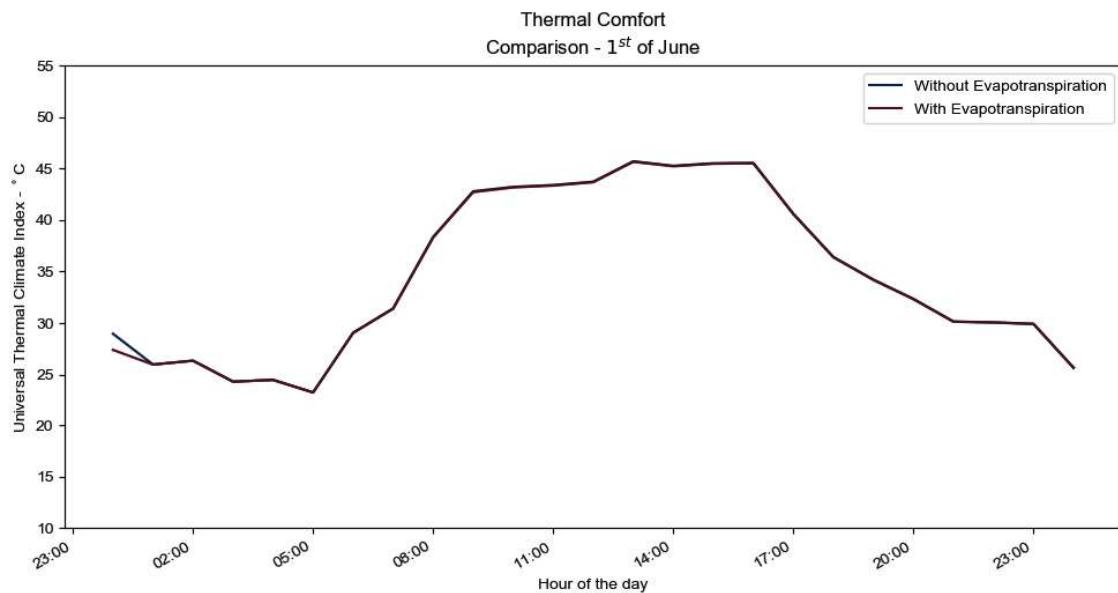


Figure 5.67 - Comparison of UTCI between the model with and without evapotranspiration in the 1<sup>st</sup> of June. The values presented are the average at the cells of the big trees

Figure 5.64. In Figure 5.68, we can see how significant the calm wind conditions are at 10:00. Here a large gap between the curves can be observed it displays a difference of ~7°C. Again at 23:00 we can see the effect of the calm conditions and the UTCI of the model with evapotranspiration is a bit lower.

To be able to generalize these observations further a histogram has been created, it can be seen in Figure 5.69. The histogram together with Table 5.6 shows a clear picture. In

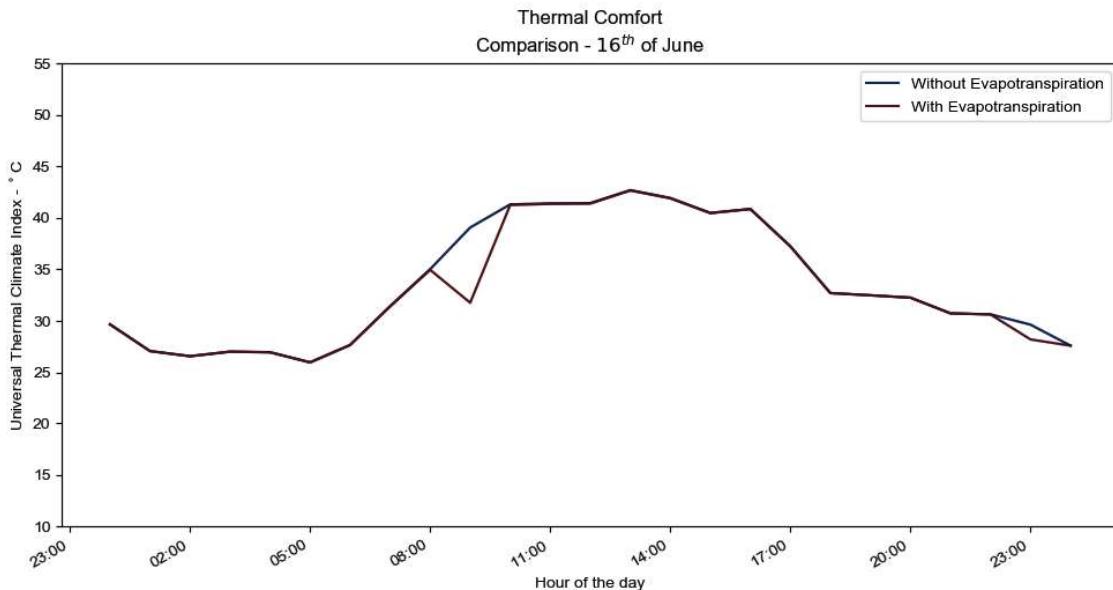


Figure 5.68 - Comparison of UTCI between the model with and without evapotranspiration in the 16<sup>th</sup> of June. The values presented are the average at the cells of the big trees

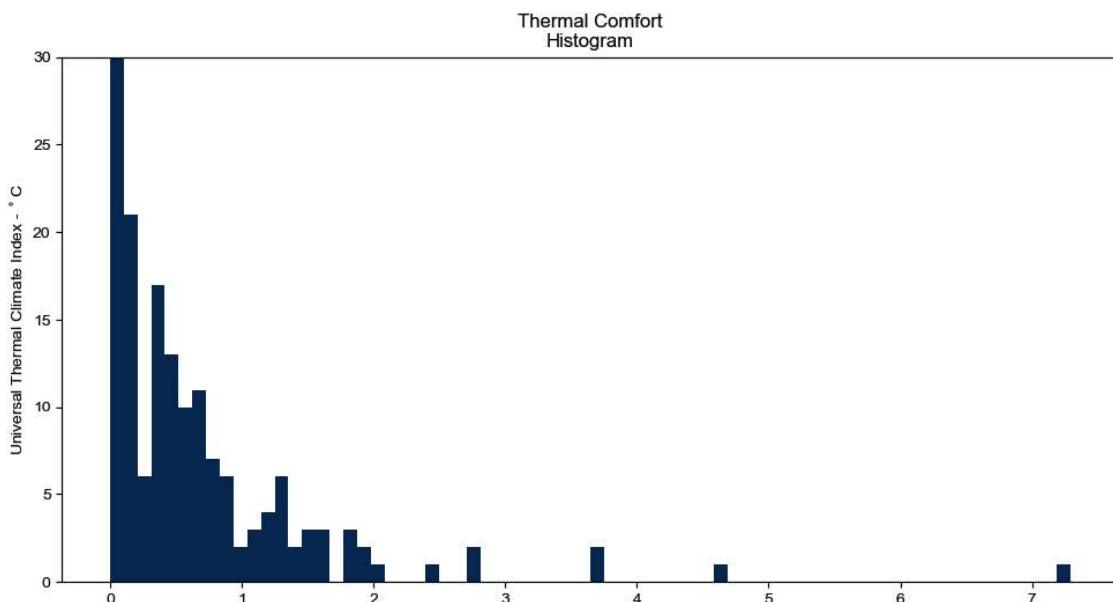


Figure 5.69 - Histogram of the difference between the model with and without evapotranspiration. The full length of the first bar is not in the figure. The values presented are the average at the cells of the big trees

Frequency - hours	Frequency - %	Start - °C	End - °C
8633	98,6%	-0,0	0,1
21	0,2%	0,1	0,2
6	0,1%	0,2	0,3
17	0,2%	0,3	0,4
13	0,1%	0,4	0,5

Table 5.6 - The first 5 bars of the histogram. The figure shows the frequency both in hours and percentage together with their spans. The values presented are the average at the cells of the big trees

98.6% of the time it is only possible to alter the UTCI with less than 0.1°C. In Section 5.2 – *Weather Analysis* it is stated the Abu Dhabi experience calm wind conditions 140 hours of the year. That statement corresponds nicely with the hypothesis laid out over the previous section that it is only in the hours with calm wind conditions that it is possible to affect the UTCI with evapotranspiration from the trees and wadi.

## Comfort Model Without Wind

As observed in the previous sections the effect of the evapotranspiration boils down to if there is wind or not. Like in Section 5.5 – *Air Condition Analysis*, we will look at the consequences of setting the wind speed to 0m/s for all hours of the year.

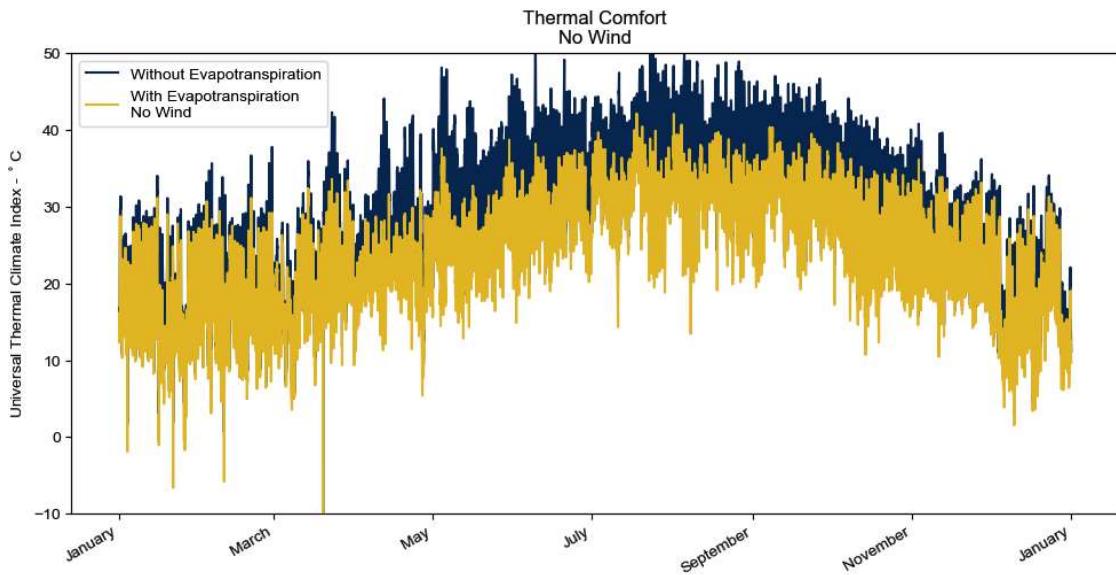


Figure 5.70 - Difference in UTCI of the year between the two models. The values are the average of the cells under the big trees

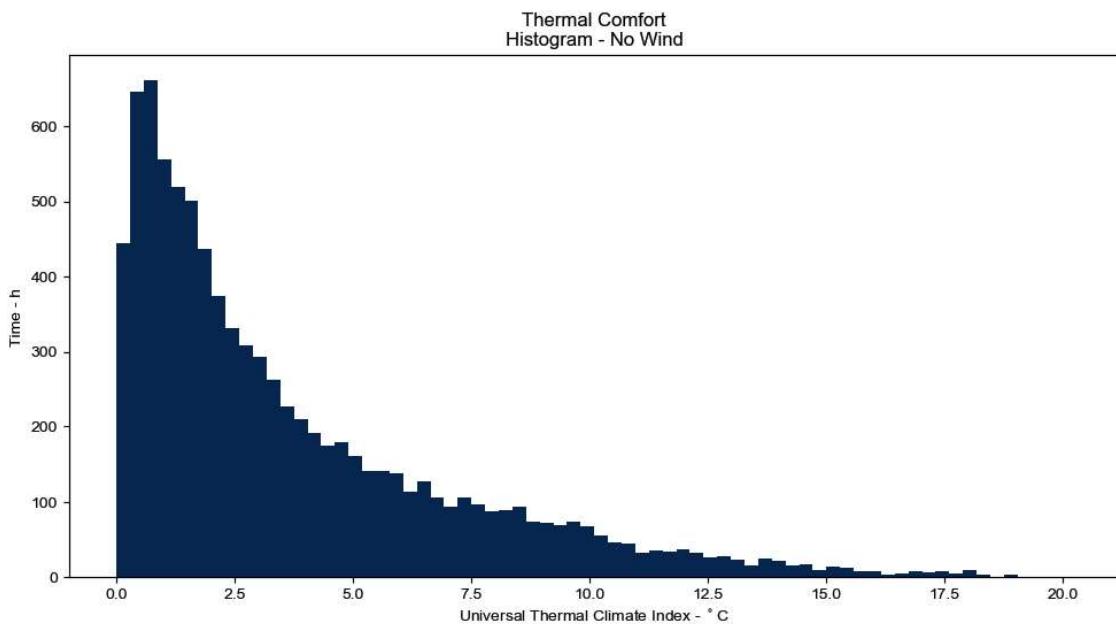


Figure 5.71 - Histogram of the difference in UTCI between the two models. The values are the average of the cells under the big trees

<b>Thermal Comfort Summary Statistics - No Wind</b>	
Minimum	0,0 °C
25 <sup>th</sup> Percentile	1,1 °C
Mean	3,7 °C
Median	2,5 °C
75 <sup>th</sup> Percentile	5,4 °C
Maximum	20,2 °C
Standard Deviation	3,5 °C

Figure 5.72 - Summary statistics of the difference in UTCI between the two models. The values are the average of the cells under the big trees

In Figure 5.70; UTCI for the model without evapotranspiration and the model with evapotranspiration (but without wind) are plotted for an entire year. It clearly shows that it is possible to gain an effect with the evapotranspiration, especially during the hot season, where it sometimes is possible to get an effect above 10°C. This can be supported by Figure 5.71, where a histogram of the difference is shown. The median is 2.5°C and the highest frequencies can be found below that difference in UTCI, between the two models. However, the average difference for the entire year is 3.7°C. Again, these two plots are based on the average of the cells with the big trees.

### 1<sup>st</sup> of January

Zooming in on the two example days, 1<sup>st</sup> of January and 1<sup>st</sup> of June, we see in Figure 5.73 that for the 1<sup>st</sup> of January only a small difference in UTCI can be achieved. It is first after 9:00 that the two curves start to separate, and they stay that way, with a few degrees of difference, for the rest of the day. Figure 5.73 only displays the difference in UTCI between the big trees. How some of the other areas in the model behave can be seen in Figure 5.74. There is not that big of a difference between the three locations but it can be seen that the wadi has the lowest UTCI.

### 1<sup>st</sup> of June

Figure 5.75 displays the difference in UTCI between the two models in June. Compared to January there is a much larger difference. Until sunrise the two curves only differ a few degrees but afterwards the model without evapotranspiration starts to increase in UTCI, while the other stays relative constant. During midday there is a difference in UTCI at about 15°C, which is quite a substantial difference. Looking at the different location of the model with evapotranspiration we can see that the two curves for the trees stay mostly together, besides in the late afternoon, while the unshaded wadi increase in UTCI.

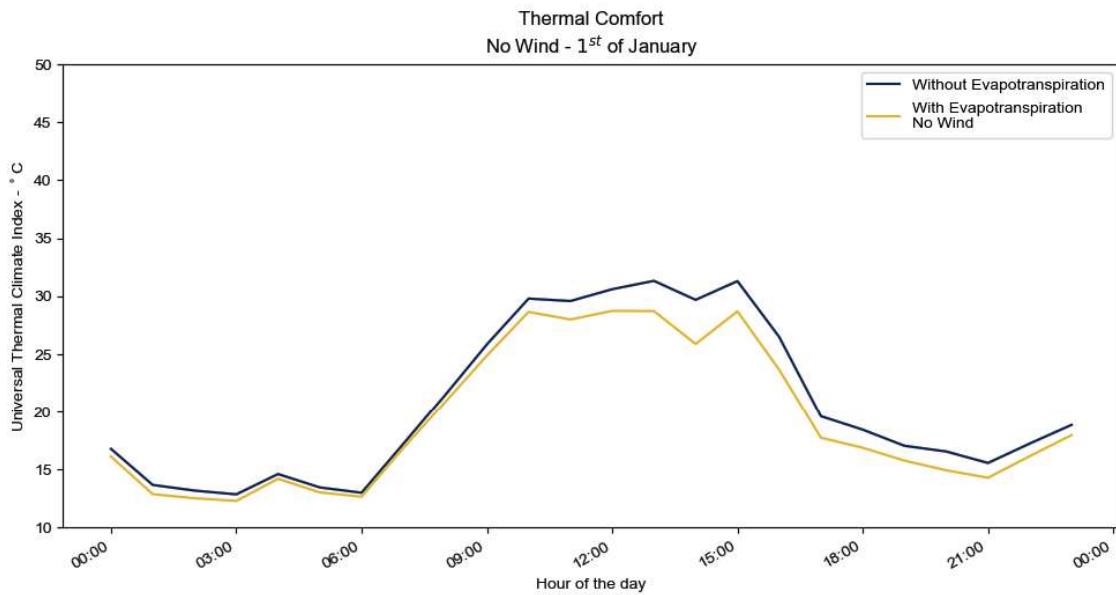


Figure 5.73 - Difference in UTCI between the two models at the 1<sup>st</sup> of January. The values are the average of the cells under the big trees

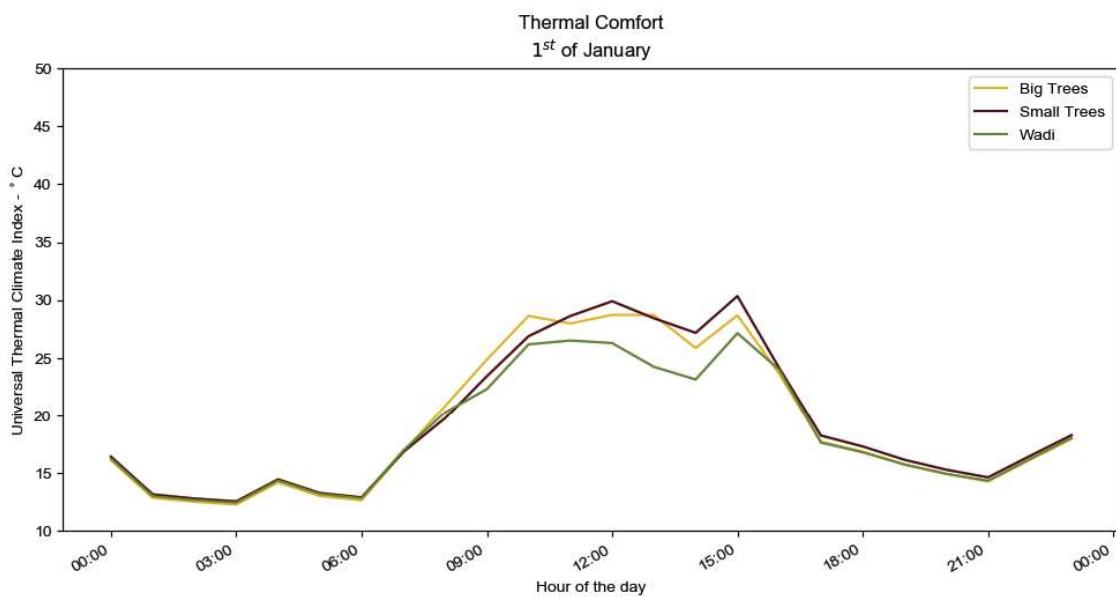


Figure 5.74 - Difference in UTCI between three locations in the model with evapotranspiration and without wind at the 1<sup>st</sup> of January. The values are the average of the cells under the big trees

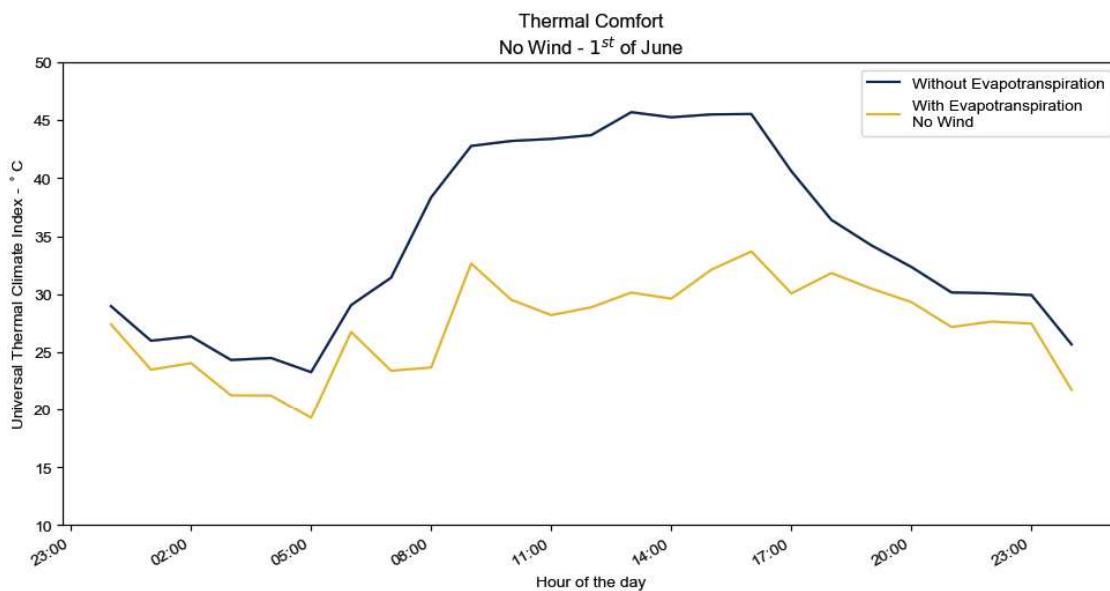


Figure 5.75 - Difference in UTCI between the two models at the 1<sup>st</sup> of June. The values are the average of the cells under the big trees

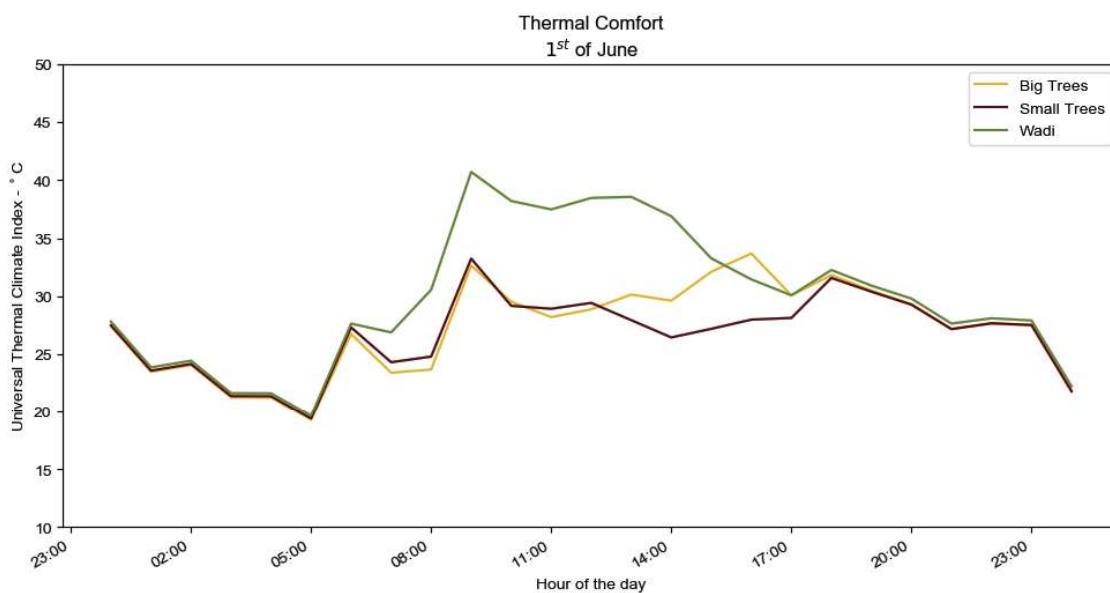


Figure 5.76 - Difference in UTCI between three locations in the model with evapotranspiration and without wind at the 1<sup>st</sup> of June. The values are the average of the cells under the big trees

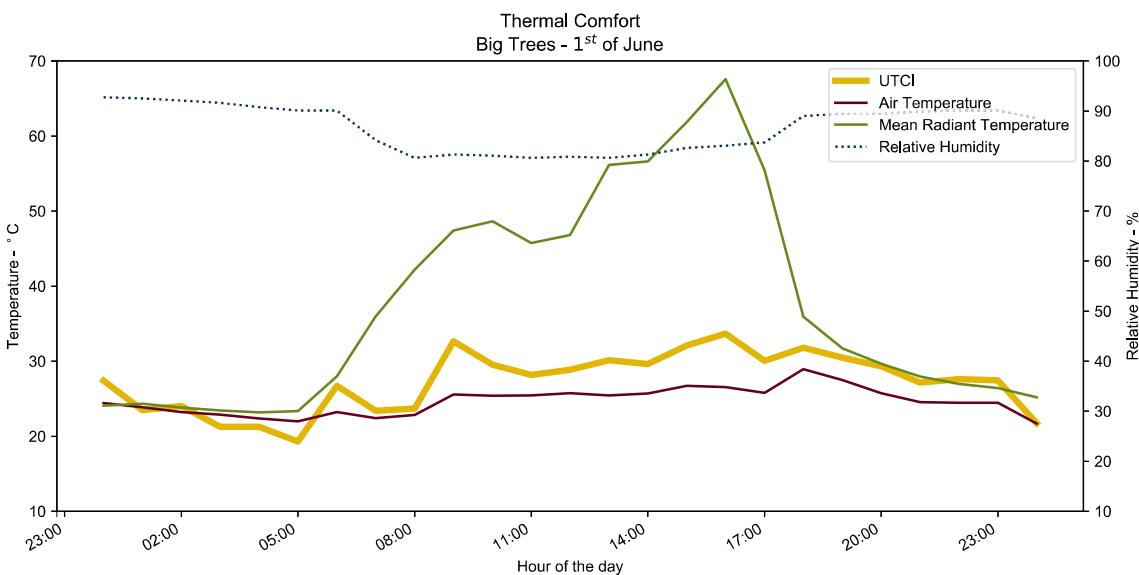
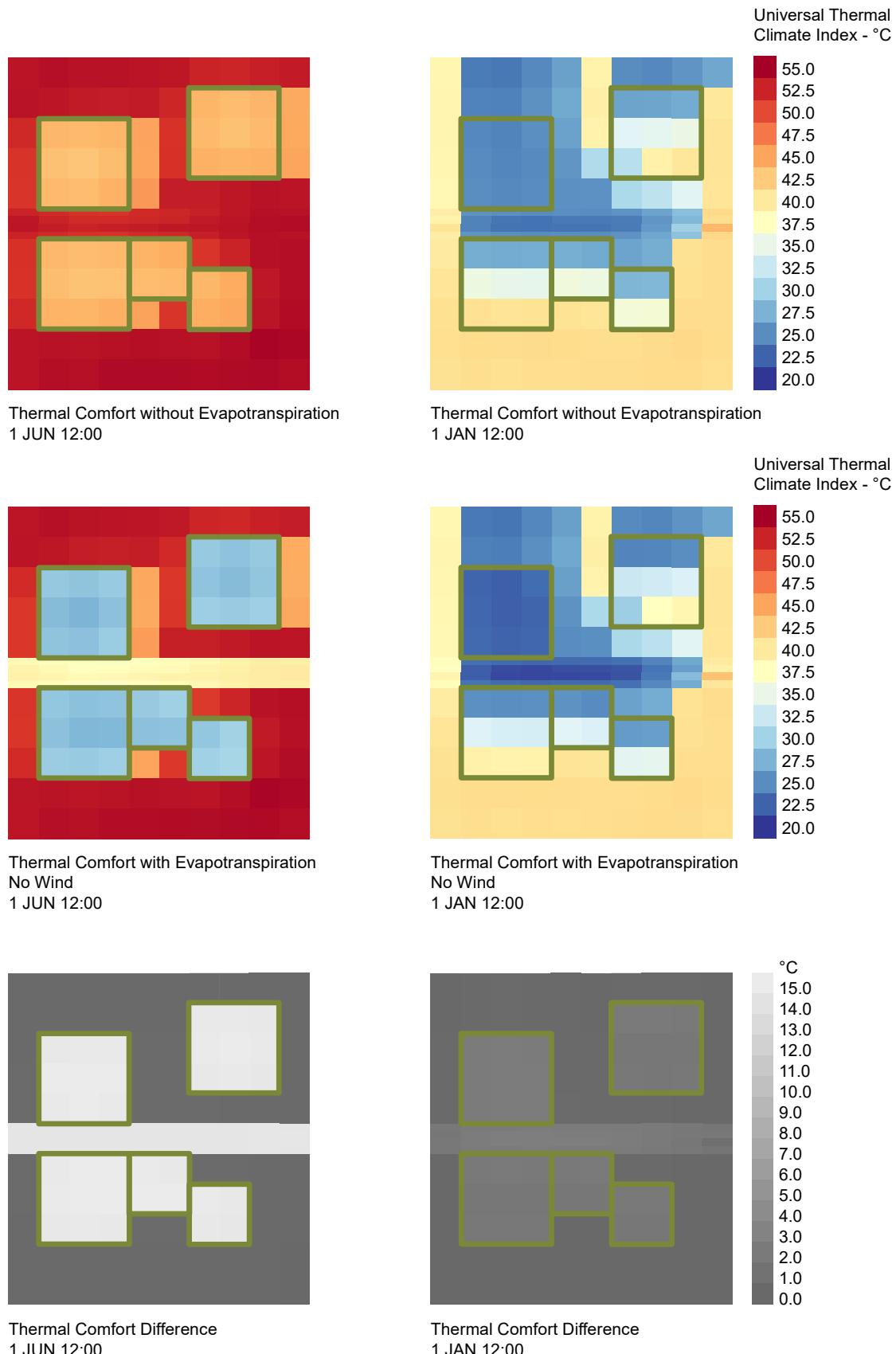


Figure 5.77 - Environmental inputs that make up the UTCI for the 1<sup>st</sup> of June. The values are the average of the cells under the big trees

It is also possible to break the UTCI at the 1<sup>st</sup> of June down into its components as done in Figure 5.77. In this figure the wind speed is not shown as it is constant at 0 m/s. We can see that it is possible for the trees to keep an almost stable relative humidity and air temperature, while the MRT just rises during the day.

### Spatial Differences

In Figure 5.78 the spatial differences in the models are visible for the 1<sup>st</sup> of January and 1<sup>st</sup> of June at 12:00. Here it can be seen how it is under the trees and over the wadi differences arise and it also displays the model assumption that the air volumes does not mix or exchange any kind of heat or vapour flux as it strictly the wadi and the tree cells that experience a difference.

Figure 5.78 - Spatial difference between the two models at the 1<sup>st</sup> of June 12:00

# 6 - DISCUSSION AND CONCLUSION

---

# 6.1 - DISCUSSION

## Atmosphere Model

The atmosphere model is created as a simple analytical model. It has been the objective not to overcomplicate it or create a full blooded CFD (Computational Fluid Dynamic) model. The model is conceptually a standing column of still air above the ground cell. To correct this the stratification and the wind speed correct volume have been introduced. The stratification was introduced to deal with the assumption that the relative humidity and temperature is uniform for the whole column in the basic model. In reality this is not the case, but there will be a temperature and relative humidity gradient from the source of the evapotranspiration to the ambient air. The gradient is complex to define as it depends on numerus factors such as; convection, radiation, advection and diffusivity. It is possible to construct such a model as it has been done in (Manickathan et al. 2018). There a CFD model is used to handle these phenomena. The problem is that it is computational heavy and would not work well in a design process. Furthermore, their approach is focused on a steady state scenario and transforming that into a transient CFD simulation would only increase the already heavy computation.

Transforming the Atmosphere Model into a CFD model might be the way to go, as it would make it possible to address some of the other simplifications of the current model. Just as there is a temperature and relative humidity gradient vertically, is there also one horizontally between the cells. This has not been addressed in the current implementation, but it is governed by the same phenomena as the vertical gradient.

## Wind Speed Correct Volumes

In an effort to introduce the wind into the model the wind speed corrected volumes emerged. The way they are constructed have a huge effect on the result of the model. The volumes are constructed in a way, one would normally calculate a volume flow through a pipe or similar i.e. cross-sectional area times velocity. The velocity is the wind speed, but the cross-section of the air column needs to be constructed. In principle the shape of the surface cell is assumed unknown, which is not entirely correct as the cell shape can be extracted from the ground mesh. Relying on the actual shape of the mesh face, might make the size of the wind speed correct volume more precise, but it would be an insignificant change. Therefore, the assumption of using a circle with the same area as the mesh face arose.

The circle has the same cross-section no matter, which direction the wind blows. That simplifies the calculation, by omitting a potential complicated geometric computation with the cell. The details of the cross-section are not interesting, the main assumption is, if it is correct that the wind, which flows through the air column, should receive an equal amount of water vapour? You could argue that it is only the boundary layer around the vapour source – be it a leaf, grass or a whole canopy – will receive the water vapour

---

and that it would mix with the surrounding air. A derived assumption is that the air, which flows through the cross-section of the air column, is not affected of what happens “up-stream”. To exemplify: if there are two neighboring cells both emitting a vapour flux. The air flowing would first be affected by the evapotranspiration of the first cell and then move to the next and further be affected by the second cell’s evapotranspiration. In this model it is assumed that two air flows go through the two cells without ever mixing. This seems like the root to the substantial difference between still air conditions and where wind is present. At this time, it has not been possible to address this problem more in-depth, but a better model is needed in the future.

## Soil Model

### Catchment Modeling Framework

The Soil Model is implemented as a wrapper around the CMF and it turned out satisfactory. The documentation for CMF has been clear and therefore made it relatively easy to create a wrapper around. Being able to only implement a wrapper and not a whole library for handling water fluxes within the soil, have been of immense importance. It has made it possible to focus on the Atmosphere Model and thermal comfort and not dig too deep into hydrology. Relying on already developed tools makes it possible to explore their capabilities in the future and not just keeping a singled sided focus on thermal comfort. In the case of CMF, with the current implementation, or with relatively small adjustments, it should be possible to investigate surface water flows and infiltration in a storm water management context.

### Wadi Model

The model of the wadi an its local context with a few trees is as most computational models a simplification of reality. The model is taken out of any bigger surrounding context, such as the nearby buildings. These buildings would have an impact on the thermal comfort as they also create shade and would have an influence on the mean radiant temperature. As it has not been the topic of this thesis to optimize the buildings, to improve the thermal comfort around the wadi, they have been neglected from the model, as they would add unnecessary complexity to the model. By being disconnected from the surrounding context, it is indirectly assumed that the model is a catchment, which is not correct. There would be some exchange of water between the edges of the model and the imaginary surroundings. This seems like a good assumption as the model itself is completely flat and there is no transport caused by topographically differences. Making the model entirely flat is an assumption that was made. Hardly anything is ever flat, but it seemed like a fair assumption, which would simplify the model and leave out complications with topographical height differences. Furthermore, within architecture it is widespread practice create the surrounding areas flat, unless a deliberate topographical feature is present.

---

## Soil Data and Tree Data

It has not been possible to get hold of any soil data from Abu Dhabi, it has therefore not been possible to verify if the data in the HYPRESS (Panagos et al. 2012) database are representative of the soil in Abu Dhabi. As argued in section 5.4 - Wadi Analysis, it seems conceivable to choose a soil type that would be beneficial for the optimization of evapotranspiration in the area, as the whole neighborhood is going to be constructed.

Most of the parameters influencing the soil model have been investigated for their effect on the model and in the end the evapotranspirational flux. An obvious choice would also have been to look at the tree properties. This have not been done, because of the poor data available in the PlaPaDa (Breuer L and Frede 2003). It makes little sense to compare synthetical composed data, which has low correlation. As stated in the paragraph about Vegetation Data (Page 16) a more consistent dataset is needed. There is a wide variety in tree types and thereby properties and some might be better suited for a climate than others. In Miami-Dade Country, USA, palm trees have been removed from the city's Urban Forestry Master Plan as they do not provide enough shade and cooling effect compared to their water consumption (Davidovich 2011). More detailed data might also enable the user of the models make cost-benefit analyses such as this example, besides giving a more precise insight in the evapotranspirative output.

### Tree model

For further research a dynamic tree model capable of predicting the trees parameters given different inputs would be beneficial. In this thesis the trees are assumed static, which in reality is not correct. A tree may have different properties depending on the time of year or time of the day. The leaf fall in temperate climates is an obvious phenomenon, which comes to mind, that would alter the trees' properties drastically. Other properties such as stomatal resistance depend on climatic conditions such as the short-wave radiative flux and vapour pressure deficit (Manickathan et al. 2018). The type of model could also be imagined predicting leaf temperatures giving a more precise input to the mean radiant temperature, rather than assuming it is the same as the air temperature.

### Mesh Generation

The generation of the ground surface mesh has not been touched upon. Since the model for the case study is simple, it has not been necessary to deal with. The meshing capabilities in Grasshopper are good enough for simple to medium complex meshes. For larger and more complex models, more attention must be paid on creating a correct mesh. Other computational techniques, such as CFD or FEM (Finite Element Method), depend on high quality mesh generations, therefore there is a variety of Python libraries out there ("Mesh Generation for Computational Science in Python" 2018), which can be integrated into Livestock.

---

## Comfort Computation

The ease of computing the comfort metrics with Ladybug Tools was one reason why a workflow in Grasshopper was chosen. With already developed components it was easier to adjust to the new inputs from the Atmosphere model. As stated above; it was assumed that the tree canopies had the same temperature as the ambient air. This is of course not correct. At the current point in time, it has been the best assumption to be made without having a tree model.

The choice between which comfort metric to use was not easy. Both UTCI and PET seem like capable models. Even though UTCI was picked in the end, it could as well have been PET. The fast computation of the UTCI regression model was an advantage and the ultimo deciding factor. The regression formula is both a blessing and a curse. On one hand it delivers fast computations at a high accuracy, with few inputs. On the other hand, it loses some on the flexibility, which PET possesses. Restraining this flexibility is the trade-off that was made to create the regression formula. The openness or flexibility of PET gives the user the possibility to investigate more scenarios, where different body types or metabolic rates comes into play.

The case study is operating at the edge, sometimes even surpassing, of the boundaries of the UTCI regression formula. In section 3.4 Comfort Metrics it is stated that the regression formula only is accurate within an ambient air temperature of -50°C and 50°C and a wind speed at 10m height between 0.5m/s to 17m/s. The ambient air temperature does not surpass the 50°C at any point of the year but it is certainly above 45°C at more accounts. As pointed out previously, is there 140 hours with 0m/s of wind speed.

## Case Study Results

The case study demonstrates an application of the developed models. First by applying the Soil Model to investigate the water volume in the model and optimize the effect of the evapotranspiration. Thereafter, the Atmosphere Model is used to investigate the effect, of the resulting evapotranspiration, on the microclimate. Finally, these changes are transformed into thermal comfort and a comparison with the current methodology.

### Highlighted Dates

Together with analyses of annual progress, two dates were chosen for further investigation: 1<sup>st</sup> of January and 1<sup>st</sup> of June. Those two dates should be representative of the cooler season and the hot season. It can be argued if those two really are representative. Often in these annual analyses the solar solstice, 21<sup>st</sup> of December and 21<sup>st</sup> of June, are selected. 21<sup>st</sup> of December and 21<sup>st</sup> of June are commonly not selected because they are representative or for being the coldest and hottest days, but because they have the shortest and longest day. There is no evidence, that shows that the 1<sup>st</sup> of January and

---

1<sup>st</sup> of June are the most representative dates for the two seasons. However, as seen on Figure 5.11, they both lie comfortable within the two seasons.

## Air Results

The analysis of the altered air conditions shows that an effect ranges from neglectable ( $0.3^{\circ}\text{C}$ ) to an average of  $4^{\circ}\text{C}$  is possible. The effect depends on whether the model do not use the wind speed correct volumes, or it does. The two situations should be thought of as extremes, where the correct answer lies in between. However other studies report that a cooling effect of  $3^{\circ}\text{C}$  to  $4^{\circ}\text{C}$  of the air temperature can be expected from trees (Shashua-Bar et al. 2009). That suggests that the correct answer is closer to the situation without the wind correct volumes, than with.

## Comfort Results

The change in air temperature lies at the heart of the difference between the developed methodology and the current methodology to compute thermal comfort. Therefore, the results are transferred onwards and affect both the mean radiant temperature, however only to a minor degree, and the UTCI. There is also a span in UTCI ranging from neglectable to an average of  $3.7^{\circ}\text{C}$  to a maximum of  $20.2^{\circ}\text{C}$ . As this span is rather large at some points during the year ( $15^{\circ}\text{C}$  during the day of the 1<sup>st</sup> of June), one should be cautious with using it as the absolute truth, without further investigation. What can be said, is that the current methodology of neglecting the effect of vegetation on the microclimate and thermal comfort, is incorrect. The knowledge have been around of a while, documented by research like (Shashua-Bar et al. 2009). The developed models facilitates a way to use this in a design process with a reasonable accuracy, without having to rely on heavy computational methods those presented in (Manickathan et al. 2018).

## Theoretical Limit

An upper limit for the effects of the evapotranspiration have also been investigated. Besides showing that enormous amounts of vapour is needed to maximize the situation with wind speed corrected volumes, it also gives an insight into what cooling effects are possible with vapour mists. Several mist products exist on the market today. They provide cooling by spraying very small droplets of water into the air, which evaporates. Figure 6.1 shows what cooling effect can be achieved at different relative humidity and air temperature. The announced cooling effects on that figure fits well with the achieved cooling effects observed in Figure 5.73 and 5.75. This suggest that the real solution lies closer to the situation without the wind speed corrected volumes.

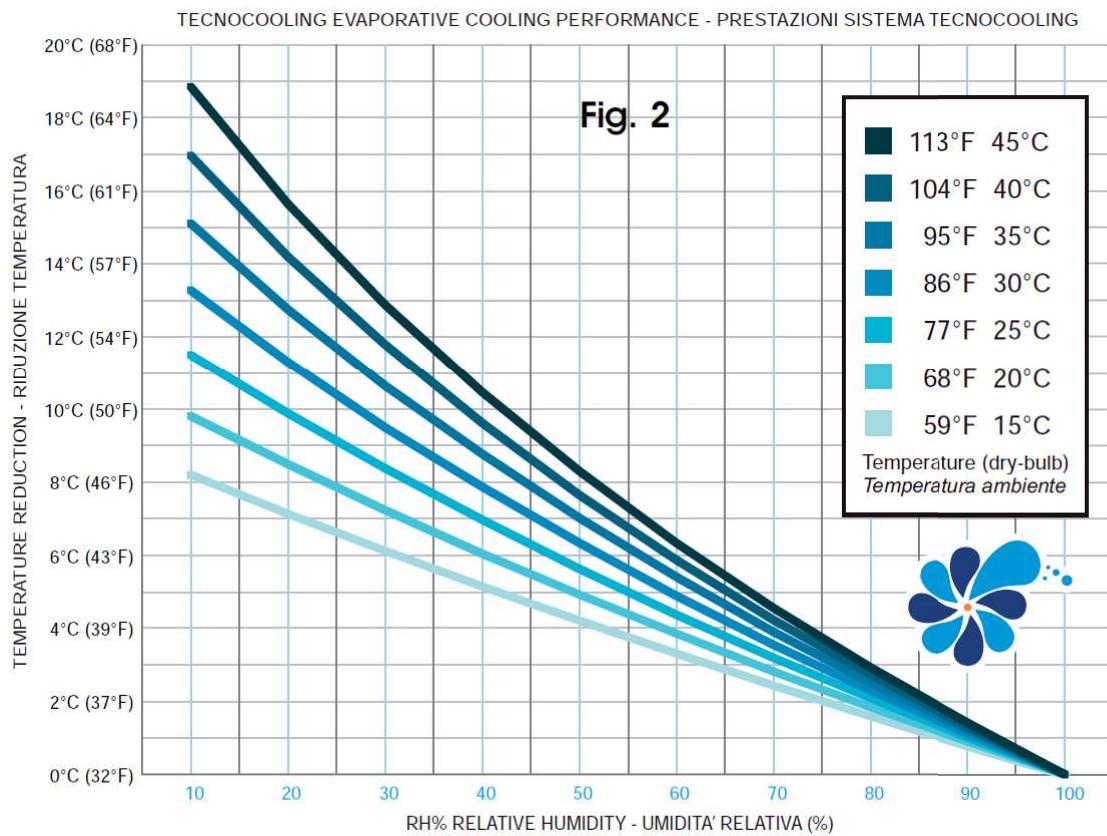


Figure 6.1 - Cooling effect achievable with TecnoCooling's products at various air temperatures and relative humidity.

### Mean Radiant Temperature

An investigation of mean radiant temperature was conducted to explore the consequences of modifying the temperature in the canopy and wadi. It analyses showed that the wadi could significantly reduce the mean radiant temperature close to the wadi banks. The effect would quickly decrease with increased distance to the banks. A smaller effect was found, when the canopy temperature was modified. Here only a reduction of 2.5°C could be achieved. These two small studies show that not only will the evapotranspiration impact the air temperature and relative humidity it will also influence the mean radiant temperature.

An important parameter in the mean radiant temperature is the ground surface temperature. Therefore, a visualization of the ground surface temperature was presented in Figure 5.54. It showcases the span in temperature there is between the 1<sup>st</sup> of January and 1<sup>st</sup> June, as well as between midday and afternoon. The figure should function as a reminder that trees and water might be important for the thermal comfort, but paying attention to the ground and choosing the correct surface coverage might be equally important.

---

## Model Validation

The model has not been validated against measured data. A validation would be needed to quantify the accuracy of the model's predictions. However, it has been difficult to find such measured data to validate against. It has not been possible to make those measurements either, because of time restrictions. For further research it would be advantageous to assign more time to either conduct those measurements or find data from other studies to validate against.

In the quick comparison given here, the model predictions seem to agree to some degree with the measured values and what can be expected. The model is built upon exciting knowledge and tools, which gives it some creditability. It is in the opinion of this author that the model is valid enough to be used in a design process.

---

## 6.2 - CONCLUSION

The main aim was to develop a method to evaluate the influence of vegetation on thermal comfort in the microclimate of outdoor spaces.

Current and past theories on hydrology and hygrothermal comfort has been presented. Significant theories explaining how soil water content, evapotranspiration, radiation and vapour content in the air effect the human thermal comfort through trees and plants have been examined.

It was concluded that UTCI – Universal Thermal Climate Index – was the most suited comfort metric. It benefits from being worldwide applicable, short computation time and does not need a detailed wind analysis. In other studies, other metrics, such as the PET – Physiologically Equivalent Temperature, might be a better fit.

In order to evaluate the influence of evapotranspiration on the microclimate and thermal comfort, two models have been developed: The Atmosphere Model and The Soil Model. The Soil Model computes the movements of water, within its defined cells and makes it possible to compute the evapotranspiration from the cells. The Atmosphere model makes use of the Soil Model's output and converts them into a change in air temperature and relative humidity. These two models are created to function within the parametric tool Grasshopper and be used in an analysis workflow with geometry and comfort computations. To make this work a methodology and framework for exchanging data between Grasshopper Canvas and CPython were developed. The methodology makes it possible to implement the Soil Model as a wrapper around the already existing hydrological library for CPython – Catchment Modeling Framework (CMF) – and use native CPython high speed computation libraries, such as NumPy and SciPy, for mathematical operations. The developed methodology allows the user to benefit from remote calculation machines to speed up the computations. The models and its Grasshopper implementations are bundled and freely available under the MIT license and the name Livestock.

The conducted research shows that with the developed models; that evapotranspiration can have a beneficial effect on the thermal comfort of a case study in Abu Dhabi. The implementation of the developed models makes it possible to document the change in air temperature and relative humidity, which affect the thermal comfort. The largest benefits are during the summer, where a decrease of more than 10°C in UTCI is not unusually, during daytime. During the night and in the cooler months decreases of less than 5°C in UTCI are present. On average it is possible to decrease the UTCI with 3.7°C over the year. The values presented here are valid for the ground cells under the big trees. The conducted research shows a difference in the reduction of UTCI, between the different mediums of evapotranspiration i.e. big trees, small trees and wadi. The research also shows it is not only one of the mediums, which shows the best performance, but it alters over the course of the day and year.

# 7 - POSTFACE

## 7.1 - BIBLIOGRAPHY

- Allen, R. G. (Rick G.), Luis S. Pereira, Dirk Raes, and Martin Smith. 1998. *Crop Evapotranspiration : Guidelines for Computing Crop Water Requirements*. Food and Agriculture Organization of the United Nations. <http://www.fao.org/docrep/x0490e/x0490e00.htm#Contents>.
- Arens, Edward, Tyler Hoyt, Xin Zhou, Li Huang, Hui Zhang, and Stefano Schiavon. 2015. "Modeling the Comfort Effects of Short-Wave Solar Radiation Indoors." *Building and Environment* 88. Elsevier Ltd: 3–9. doi:10.1016/j.buildenv.2014.09.004.
- Acero, Juan A., and Karmele Herranz-Pascual. 2015. "A Comparison of Thermal Comfort Conditions in Four Urban Spaces by Means of Measurements and Modelling Techniques." *Building and Environment* 93 (P2). Elsevier Ltd: 245–57. doi:10.1016/j.buildenv.2015.06.028.
- Błazejczyk, Krzysztof, Gerd Jendritzky, Peter Bröde, Dusan Fiala, George Havenith, Yoram Epstein, Agnieszka Psikuta, and Bernhard Kampmann. 2013. "An Introduction to the Universal Thermal Climate Index (UTCI)." *Geographia Polonica* 86 (1): 5–10. doi:10.7163/GPol.2013.1.
- Breuer L, and H Frede. 2003. "PlaPaDa - an Online Plant Parameter Data Drill for Eco-Hydrological Modelling Approaches." <http://www.staff.uni-giessen.de/~gh1461/plapada/plapada.html>.
- Bröde, Peter, Dusan Fiala, Krzysztof Blazejczyk, Ingvar Holmér, Gerd Jendritzky, Bernhard Kampmann, Birger Tinz, and George Havenith. 2012. "Deriving the Operational Procedure for the Universal Thermal Climate Index (UTCI)." *International Journal of Biometeorology* 56 (3): 481–94. doi:10.1007/s00484-011-0454-1.
- Bruse, Michael. 2014. "ENVI-MET 4." <http://www.envi-met.info>.
- Chen, Yung-chang, and Andreas Matzarakis. 2014. "Modification of Physiologically Equivalent Temperature." *Journal of Heat Island Institute International* 9: 26–32.
- Coccolo, Silvia, Jérôme Kämpf, Jean Louis Scartezzini, and David Perlmutter. 2016. "Outdoor Human Comfort and Thermal Stress: A Comprehensive Review on Models and Standards." *Urban Climate* 18. Elsevier Inc.: 33–57. doi:10.1016/j.ulclim.2016.08.004.
- Dansk Standard. 2007. "DS/EN 15251:2007: Input-Parametre Til Indeklimaet Ved Design Og Bestemmelse Af Bygningers Energimæssige Ydeevne Vedrørende Indendørs Luftkvalitet, Termisk Miljø, Belysning Og Akustik.," 54. <http://sd.ds.dk/globalproxy.cvt.dk/Viewer?ProjectNr=M204572&Status=60.61&VariantID=&Page=0>.
- Davidovich, Joshua. 2011. "The Problem With Palm Trees." *CityLab*. <https://www.citylab.com/solutions/2011/12/problem-palm-trees/748/>.
- Eslamian, Saeid. 2014. *Handbook of Engineering Hydrology: Fundamentals and Applications*. Francis and Taylor, CRC Group. doi:10.1016/0022-1694(75)90105-5.
- Federer, C. Anthony. 2017. "The BROOK90 Hydrologic Model." Accessed November 29. <http://www.ecoshift.net/brook/brook90.htm>.

- 
- Fiala, Dusan, George Havenith, Peter Bröde, Bernhard Kampmann, and Gerd Jendritzky. 2012. "UTCI-Fiala Multi-Node Model of Human Heat Transfer and Temperature Regulation." *International Journal of Biometeorology* 56 (3): 429–41. doi:10.1007/s00484-011-0424-7.
- Foundation, Python Software. 2017. "PyPI - the Python Package Index : Python Package Index." Accessed December 22. <https://pypi.org>.
- Gehl, Jan. 2001. *Cities for People*. 1sted. Copenhagen: Bogværket.
- Havenith, George, Dusan Fiala, Krzysztof Blazejczyk, Mark Richards, Peter Bröde, Ingvar Holmér, Hannu Rintamaki, Yael Benshabat, and Gerd Jendritzky. 2012. "The UTCI-Clothing Model." *International Journal of Biometeorology* 56 (3): 461–70. doi:10.1007/s00484-011-0451-4.
- Hornberger, George M. 1998. *Elements of Physical Hydrology*. Johns Hopkins University Press. <http://findit.dtu.dk/en/catalog/2304900439>.
- Hugunin, Jim, and Dino Viehland. 2008. "IronPython.net." <http://ironpython.net/>.
- Illinois, T. B. of T. of the U. of, & The Regents of the University of California. (1996). EnergyPlus | EnergyPlus. Retrieved from <https://energyplus.net/>
- JetBrains. 2017. "PyCharm: Python IDE for Professional Developers by JetBrains." Accessed December 22. <https://www.jetbrains.com/pycharm/>.
- Kraft, Philip. 2017. "CMF: Documentation." Accessed November 29. <http://fb09-pasig.umwelt.uni-giessen.de/cmf/chrome/site/doxygen/index.html>.
- Kraft, Philipp, Kellie B. Vach, Hans Georg Frede, and Lutz Breuer. 2011. "CMF: A Hydrological Programming Language Extension For Integrated Catchment Models." *Environmental Modelling and Software* 26 (6): 828–30. doi:10.1016/j.envsoft.2010.12.009.
- Københavns Kommune, Teknik of Miljøforvaltningen. 2015. "Fremtidens Gårdhaver Med LAR."
- Mackey, Christopher, Theodore Galanos, Leslie Norford, Mostapha Sadeghipour Roudsari, and Neapoli Sdn Bhd. 2015. "Wind , Sun , Surface Temperature , and Heat Island : Critical Variables for High-Resolution Outdoor Thermal Comfort Payette Architects , United States of America Massachusetts Institute of Technology , United States of America University of Pennsylvania ,."
- Manickathan, Lento, Thijs Defraeye, Jonas Allegrini, Dominique Derome, and Jan Carmeliet. 2018. "Parametric Study of the Influence of Environmental Factors and Tree Properties on the Transpirative Cooling Effect of Trees." *Agricultural and Forest Meteorology* 248 (May 2017). Elsevier: 259–74. doi:10.1016/j.agrformet.2017.10.014.
- Matzarakis, Andreas, and H Mayer. 1996. "Another Kind of Environmental Stress: Thermal Stress." *WHO Newsletter* 18 (JANUARY 1996): 7–10.
- "Mesh Generation for Computational Science in Python." 2018. *Stack Overflow*. Accessed March 14. <https://stackoverflow.com/questions/7975522/mesh-generation-for-computational-science-in-python>.

- 
- Meuser, Andreas. 1990. "Effects of Afforestation on Run-off Characteristics." *Agricultural and Forest Meteorology* 50 (1–2): 125–38. doi:10.1016/0168-1923(90)90143-T.
- Nieuwenhuis, Marlon, Craig Knight, Tom Postmes, and S. Alexander Haslam. 2014. "The Relative Benefits of Green versus Lean Office Space: Three Field Experiments." *Journal of Experimental Psychology: Applied* 20 (3): 199–214. doi:10.1037/xap0000024.
- Oke, T R. 1981. "Canyon Geometry and the Nocturnal Urban Heat-Island - Comparison of Scale Model and Field Observations." *Journal of Climatology* 1 (3): 237-.
- Panagos, Panos, Marc Van Liedekerke, Arwyn Jones, and Luca Montanarella. 2012. "European Soil Data Centre: Response to European Policy Support and Public Data Requirements." *Land Use Policy* 29 (2): 329–38. doi:10.1016/j.landusepol.2011.07.003.
- Peuhkuri, Ruut, and Carsten Rode. 2016. "Heat and Mass Transfer in Buildings."
- Pickup, J., and R. de Dear. 2000. "An Outdoor Thermal Comfort Index (OUT\_SET\*) - Part I - The Model and Its Assumptions." *Selected Papers from ICB-ICUC*, no. January 1999: 279–83.
- Robert McNeel & Associates. (1980). Rhinoceros 3D. Retrieved from <https://www.rhino3d.com/>
- Roth, M. 2013. "Urban Heat Islands." *Handbook of Environmental Fluid Dynamics, Volume Two*, 143–59. doi:doi:10.1201/b13691-13.
- Rossum, G. van. (1991). Python. Retrieved from <https://www.python.org/>
- Rutten, David. (2007). Grasshopper - algorithmic modeling for Rhino. Retrieved from <http://www.grasshopper3d.com/>
- Sadeghipour Roudsari, Mostapha, and Michelle Pak. 2013. "Ladybug Tools." <https://github.com/ladybug-tools/>.
- Shashua-Bar, Limor, Oded Potchter, Arieh Bitan, Dalia Boltansky, and Yaron Yaakov. 2009. "Microclimate Modelling of Street Tree Species Effects within the Varied Urban Morphology in the Mediterranean City of Tel Aviv, Israel." *International Journal of Climatology* 4 (December 2007): n/a-n/a. doi:10.1002/joc.1869.
- UNDESA. 2014. *World Urbanization Prospects*. Undesa. doi:10.4054/DemRes.2005.12.9.
- Ward, G. (1985). Radiance. Berkeley: Lawrence Berkeley National Laboratory. Retrieved from <https://www.radiance-online.org/>
- Wikipedia, the free encyclopedia. 2017. "Wind Profile Power Law." [https://en.wikipedia.org/wiki/Wind\\_profile\\_power\\_law](https://en.wikipedia.org/wiki/Wind_profile_power_law).
- Xing, Lu, and Jeffrey D. Spitler. 2017. "Prediction of Undisturbed Ground Temperature Using Analytical and Numerical Modeling. Part I: Model Development and Experimental Validation." *Science and Technology for the Built Environment*

# 8 - APPENDIX

## Appendix A

<https://tinyurl.com/hygrotermic>

Contains data files, Grasshopper scripts and Python Scripts

## Appendix B

Documentation for the Livestock CPython Package

## Appendix C

Documentation for Livestock Grasshopper

Both Appendix B and C can be accessed through: <https://ocni-dtu.github.io>