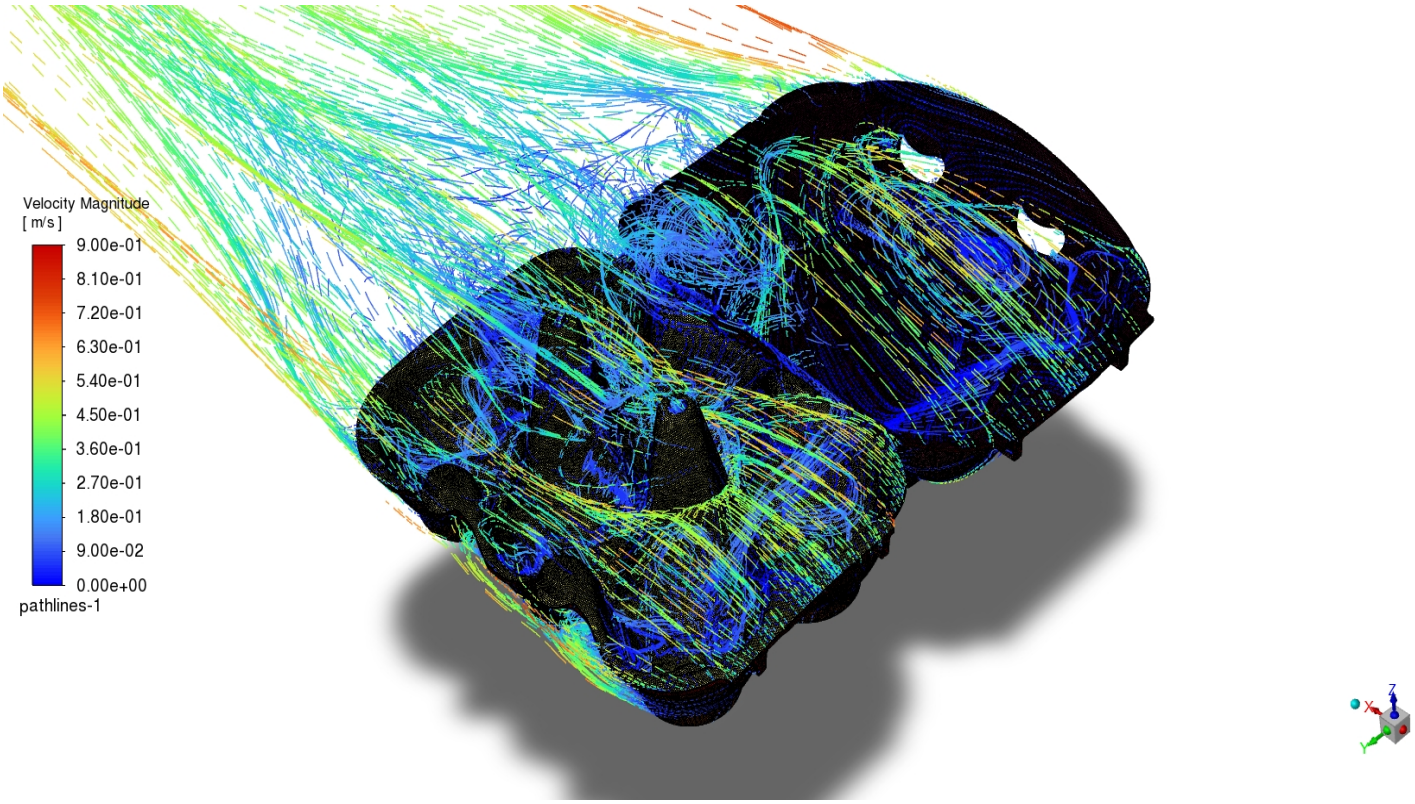


Computational Fluid Dynamics Analysis of Recycled Paper Packaging Drying Processes

Master thesis

10th Semester Project

Jakob Rosenkilde Kristensen

STUDENT REPORT

AALBORG UNIVERSITY

Department of Energy Technology
Aalborg University
http://www.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title:

Computational Fluid Dynamics Analysis of Recycled Paper Packaging Drying Processes

Theme:
Master thesis

Project Period:
Spring semester 2024

Project Group:
PECT-4-S24

Participant(s):
Jakob Rosenkilde Kristensen

Supervisor(s):
Matthias Mandø

Copies: 1

Page Numbers: 89

Date of Completion:
June 3rd 2024

Preface

This report was written by PECT-4-S24, studying at Aalborg University Esbjerg. The project was prepared in the Spring of 2024 from February to June and processes the overall subject: Master thesis.

This project builds upon previous findings in the numerical modeling of paper drying from last semester. The project was conducted in collaboration with a company with no interest in being named. The company has provided material and access to test facilities for this project. The drying models developed during this project will be validated against the experimental results obtained from the previous research. The previous project is governed by a Non-disclosure agreement.

Special thanks are extended to Michael Bjerre for his exceptional mentorship, invaluable experience, and knowledge-sharing throughout the project.

Aalborg University, June 3rd 2024

Instructions for reading

This report contains animations, which require a PDF reader capable of supporting such features. Adobe Acrobat Reader is recommended for viewing these animations. The PDF reader will appear as static images if they do not support animations.

List of abbreviations

CFD Computational Fluid Dynamics

DTRM Discrete Transfer Radiation Model

EES Engineering Equation Solver

S2S Surface to Surface

DO Discrete Ordinate

UDF User-Defined Function

List of Figures

List of Tables

Abstract

The primary aim is to determine how CFD can effectively model the drying process, incorporating various physical phenomena such as heat and mass transfer. The drying process is divided into steady-state and transient models. The steady-state model aims to capture the consistent evaporation rate during the middle phase of drying, while the transient model addresses the dynamic changes in temperature and moisture content. Both models use Ansys Fluent for CFD simulations, and the transient drying model utilizes Python for evaluating transport phenomena within the paper. The experimental setup involves a drying oven where conditions such as temperature, humidity, and airflow velocity are controlled and measured. Results from the CFD simulations indicate that the k-ω SST turbulence model provides the most accurate predictions for evaporation rates, aligning closely with experimental data. However, achieving an energy balance that matches experimental results remains a challenge. The research suggests that a two-way coupling between Ansys Fluent and Python could enhance future calculations and improve the overall accuracy of the simulations. In conclusion, the study demonstrates that CFD, combined with numerical drying models, can effectively predict the drying behavior of complex paper geometries.

Contents

1 Introduction

Paper recycling contributes to the conservation of natural resources [1] [2]. The European Declaration on Paper Recycling has committed to achieving a 76% paper recycling rate by 2030 [3]. In 2022, 70.5% of all paper and cardboard were recycled, totaling 77.9 million tons in Europe [3]. In contrast, Brazil recycles about 3.017 million tons annually, accounting for only 40% of its total consumption [1]

The composition of the paper is based on fibers and void spaces. These fibers are a composite matrix of lignin, cellulose, and hemicellulose, interconnected to create a cohesive structural framework [4] [5]. When fibers are used for the first time for a product, they are classified as virgin fibers. [3]. With each recycling process, the paper fibers shorten. In Europe, paper fibers are reused an average of 3.5 times, compared to the global average of 2.5 times [3].

Larger fibers require less energy to dry the product [3] [6]. Therefore, the war in Ukraine has adversely affected the use of recycled paper due to the significant increase in natural gas prices, which became a more dominant cost factor than the price of recycled paper. Even though virgin fibers are more expensive to purchase, it became overall cheaper to produce a product using them because the energy required for drying is lower compared to recycled paper [3] [6].

Producing new white paper, cardboard, or using recycled paper for packaging involves a general process of water dehydration [7]. Most of the moisture content in the production process is removed prior to the oven stage, either by applying pressure or using a vacuum pump [8]. However, even though the oven removes the least amount of water, it represents the most costly and energy-intensive step in the production process [7].

The oven drying process consists of three stages [9] which is illustrated in figure 1.1:

1. An increase in temperature and the drying product's evaporation rate as the prod-

uct enters the oven.

2. A period of constant temperature and evaporation rate for the drying product.

3. An increase in temperature is accompanied by a decrease in the evaporation rate of the product as the majority of moisture has been removed.

Figure 1.1a illustrates the development of moisture content and temperature throughout the drying process, under the premise of constant oven temperatur. The solid line represents the moisture content, while the dotted blue line indicates the temperature. Figure 1.1b demonstrates the drying rate as a function of moisture content. The section from points A to B highlights the stage where the drying rate increases, the segment from B to C depicts a constant evaporation rate, and the part from C to D shows a decrease in the drying rate [9].

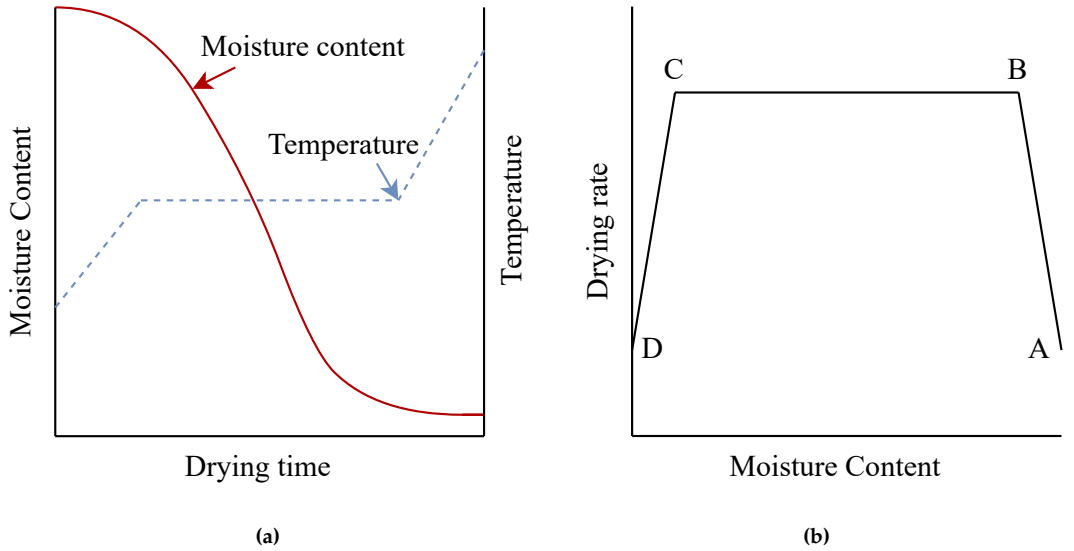
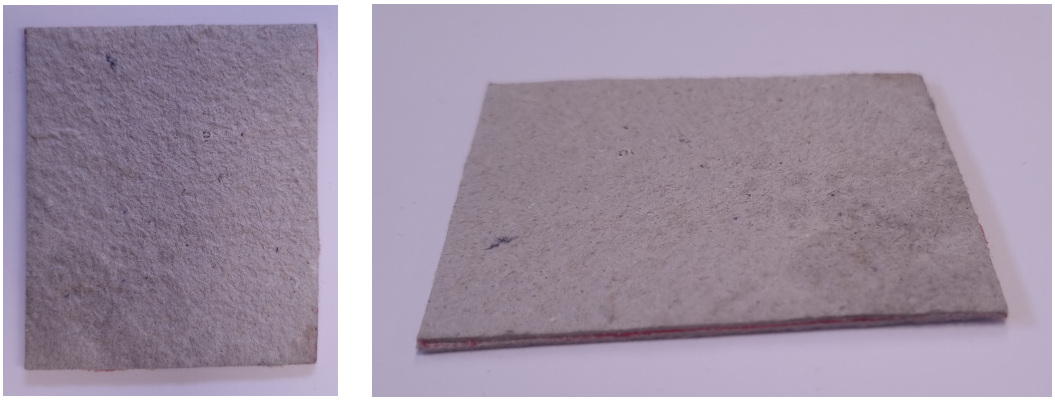


Figure 1.1: Figure 1.1a shows the moisture and temperature evolvement during drying, and figure 1.1b shows the drying rate over the moisture content.

1.1 Project goal

This report builds upon the previous work by the author [5], applying the drying model developed by Baggerud E. [4], which was originally intended for use with blank white paper for copying and printing, to assess the moisture ratio and temperature in recycled-paper packaging. Figure 1.2 shows one of the test samples utilized in the dry-

ing oven.



(a) Top view **(b)** Side view - illustrate the two combined pieces

Figure 1.2: Drying samples used in the previous report [5]

The numerical drying model was utilized to determine the temperature and moisture content in one dimension, incorporating a mass and an energy balance. The mass balance included mass diffusion and convection, while the energy balance comprised conduction, convection, and radiation. The boundary conditions were characterized by dimensionless numbers, including Reynolds, Schmidt, Nusselt, Sherwood, and Prandtl numbers [5]. The model underwent validation through comparison with experimental results. A small oven was constructed to maintain constant temperature and velocity, while the temperature and mass of the drying piece of recycled paper were measured [5].

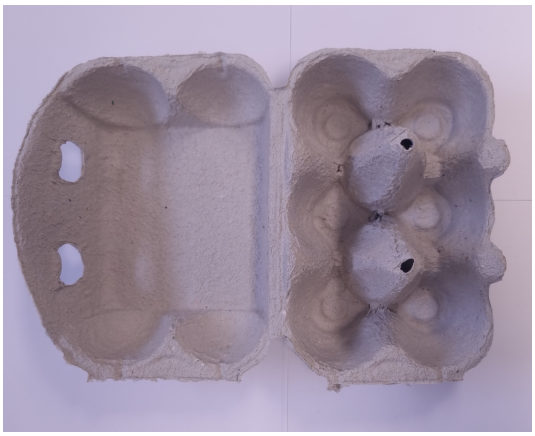
The drying model developed by Baggerud E. [4] neglected the effect of radiation on the boundary conditions, an assumption that proved inadequate for recycled paper. The inclusion of radiation was found to be crucial for achieving accurate results.

While the numerical drying model was capable of accurately predicting the drying behavior during the first and second phases, its accuracy diminished as drying progressed. This happened as the sorption energy started to affect the dynamics, and the model's predictions for temperature and moisture content began to diverge from actual observations. A critical assumption in the model is that saturated water is always present in the boundary conditions. This proved to be a poor assumption.

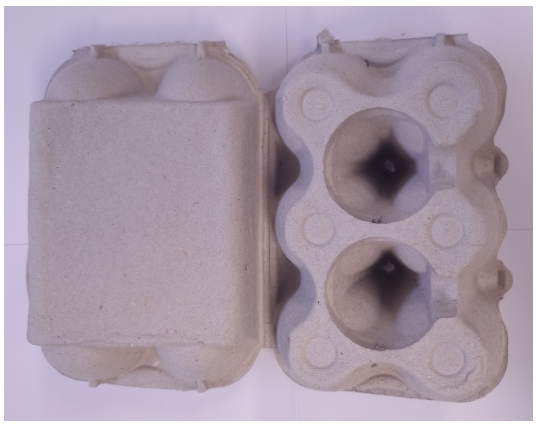
The importance of boundary conditions in the drying process is a key finding from the

model. These conditions depend on various factors such as mass diffusion, heat convection, radiation, conduction, and latent heat. Additionally, the model and experimental results were based on a simple square geometry, examining temperature and moisture content in a one-dimensional context. The model revealed that ambient temperature and airflow velocity significantly impact drying time. These insights underscore the need for a comprehensive analysis of the airflow patterns around the drying material.

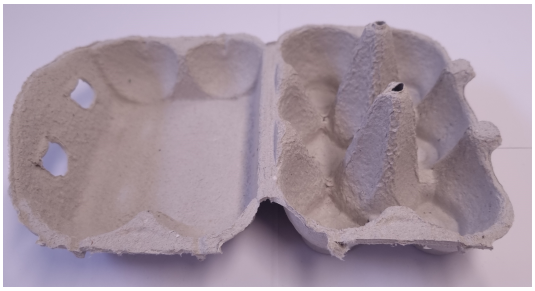
The original idea behind the drying model was to predict where an egg packaging was drying the slowest. Describing the geometry of egg packaging numerically is a complex task, including many assumptions. Therefore, this project evaluates the boundary condition by employing CFD. Figure 1.3 shows the complex geometry of an egg packaging.



(a) Top view



(b) Bottom view



(c) Front-top view

(d) Front-bottom view

Figure 1.3: Figures of an actual egg packaging, illustrated from the four different angles

1.1.1 Problem formulation

Building upon the previous results and the challenges in assessing the actual airflow across the drying recycled paper, this project will focus on how Computational Fluid Dynamics CFD, can enhance the understanding of boundary conditions' impact on the drying process. This leads to the following problem statement:

How can CFD be effectively applied to predict the drying process of paper with complex geometries?

The experimental results from the previous report serve as a benchmark for validating the CFD analysis. This analysis investigates the effects of boundary layers, turbulence models, species transport, and material properties specific to the paper. The investigation unfolds in two stages. Initially, a simple geometry is examined, where the findings are cross-verified with experimental results from the previous project [5]. Subsequently, the study expands to assess CFD potential in predicting drying processes in more complex structures beyond simple square geometries.

1.1.2 Curriculum and personal goals

The goals for the master's thesis are based on the curriculum of MSc in Sustainable Energy Engineering and personal goals. Below is a list of some of the curriculum goals and the personal goals for the project.

Curriculum goals

- Be able to use advanced laboratory set-ups or use real measured data series combined with data analysis methods and analysis and modeling methods within fluids and process systems.
- Have obtained skills related to the industrial area within process engineering and combustion technology.
- Be able to control complex or unexpected working and development situations within fluids and process systems and develop new solutions.
- Independently be able to continue own development in competence and specialization.

Personal Goals

- Learn to utilize Ansys Fluent and Ansys Meshing at a professional level with an increased focus on species transport and evaporation modeling.
- Develop skills in implementing user-defined functions in Ansys Fluent.
- Understand and effectively communicate the value of scientific research to industry stakeholders.

1.2 State-of-the-art of paper drying

Modeling the paper drying process involves complex heat and mass transfer mechanisms [6]. Diffusion and convection mass transfer primarily govern moisture movement within paper structures[4]. There are generally two approaches to modeling this phenomenon[10]:

1. Treating the paper as a homogeneous medium. This approach considers the moisture flux directly proportional to the moisture content gradient [10]
2. Considering the paper as a composite material of fibers and void spaces [10] [4]

Both methods offer unique perspectives on the drying process, contributing to a comprehensive understanding of moisture dynamics in paper materials.

The moisture transport within the paper is a complex process influenced by the presence of free and bound water [4]. The paper fibers contain bound water, while free water is located between the fibers. Figure 1.4 depicts the general fiber matrix structure.

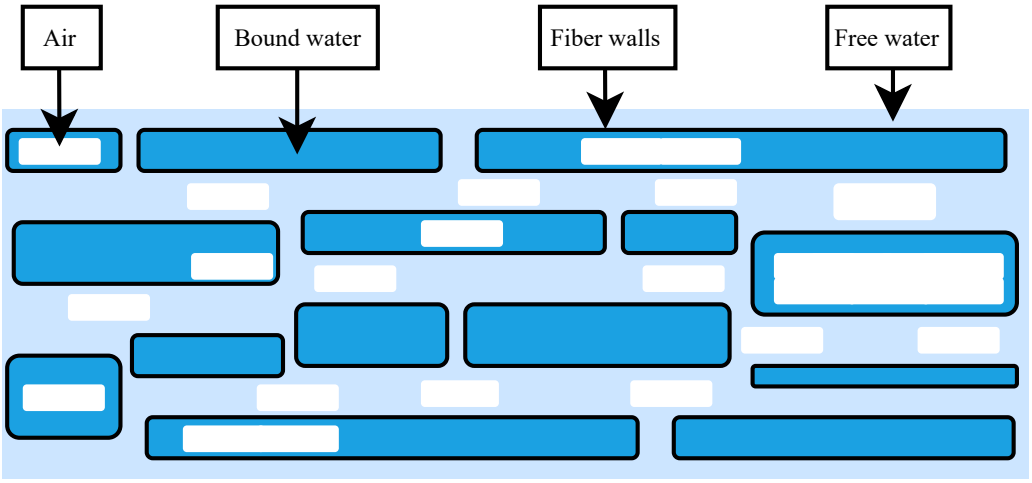


Figure 1.4: Illustration of the paper matrix consisting of air, bound and free water, and fibers.

During the drying process, the removal of water occurs in two stages: first, the evaporation of free water, and subsequently, the removal of bound water, which requires additional energy known as sorption energy [4]. Free water typically remains until the dry matter content reaches 75-80% [11].

Moisture transport in the paper drying process is commonly described using Fick's law. This concept is often encapsulated through the effective diffusion. Various researchers have proposed different expressions for this effective diffusion. Radhakrishnan et al. [10], Chen P. and David C. [12], and Baggerud E. and Stenström S. [13] have provided formulas for effective diffusion in the context of moisture transport.

Ambient conditions during the drying process significantly affect the paper's final properties, such as porosity, shrinkage, apparent density, and deformation. Higher drying temperatures produce more brittle and fragile paper, leading to more significant deformation and generally lower quality [2] [14]. The shrinkage in the thickness of white paper typically ranges between 40-60% and 2-6% in width [15].

An alternative to the one- and two-dimensional modeling discussed in previous research is the utilization of CFD. This approach allows for the detailed evaluation of flow, temperature, and concentration within complex structures [16]. The drying process involves simultaneous heat and mass transfer occurring at the surface.

CFD analysis is performed by solving the fundamental conservation equations for mass,

momentum, and energy. These three conservation equations are combined with a turbulence model to account for the turbulence. Although the k-ε and k-ω models are commonly used, studies have shown that the k-ω SST turbulence model often yields superior results for evaporation modeling [16] [17] [18].

To model diffusion between water-vapor and humid air, it is required to include the species transport in the CFD analysis [16].

Mass transfer at the interface where evaporation occurs, there are, in general, five models that describe mass transfer dynamics: Theoretical Analysis, Film Theory, Sharp Interface Model, Schrage Model, and Lee Model [16]. Theoretical Analysis, the Schrage Model, and the Lee Model have been specifically applied to water/air mixtures.

W. Ambrosini utilized Theoretical Analysis to investigate the dynamics of heat and mass transfer in falling film evaporation [19]. Similarly, X. Wang applied Theoretical Analysis to assess falling film evaporation within passive containment cooling systems for nuclear power plants [20].

Lee et al. adopted the Schrage Model to study falling film evaporation over an elliptical tube in a counter-current airflow scenario [16] [21] [22].

Furthermore, Wang et al. employed the Lee Model to evaluate the film cooling effect on photovoltaic cells, demonstrating the model's applicability in enhancing the efficiency of such systems [16] [23] [24].

The choice of a radiation model is pivotal in accurately simulating the drying process. Radiation models such as the Discrete Transfer Radiation Model (DTRM), Surface to Surface (S2S), and Discrete Ordinate (DO) have been evaluated, especially in heat transfer analyses within baking ovens [25]. A comparative study indicated a negligible temperature prediction discrepancy (less than 0.2%) between the S2S and DO models [25]. Despite the similarity in outcomes among the three models, the DO model was noted for its higher computational demand compared to DTRM and S2S [25].

The search for literature on applying CFD in predicting the paper drying process has yielded scant results. Drying simulations based on CFD are still at an early stage [17] [26]. Most previous studies have concentrated on spray drying, surface evaporation, and fluidized bed drying [16] [26]. One study used CFD to calculate local heat transfer coefficients to export the heat transfer coefficient into an external program to finalize

the drying process [27]. Due to the shortage of research on paper drying with CFD, the literature research has shifted to the drying process of foods.

The drying processes of paper and food are subject to impacts from both external and internal factors. Externally, the characteristics of the bulk fluid enveloping the drying object are predominant. Internally, the drying object's specific properties significantly influence the drying outcome. Table 1.1 below shows the primary external and internal properties that play a pivotal role in the drying process [4] [26] [17] [25]:

Table 1.1: The external and internal properties which influence the drying process.

External influence	Internal influence
Temperature	Density
Velocity	Porosity
Humidity	Permability
	Specific heat
	Mass diffusion
	Thermal conductivity
	Sorption isotherm

The spectrum of drying methodologies includes, but is not limited to, vacuum drying, spray drying, freeze drying, microwave-assisted drying, infrared drying, and convection drying. Currently, forced convection drying methods are employed in over 85% of industrial drying operations within the food sector [26].

A study applying CFD for predicting mushroom drying in a forced convection oven demonstrated a strong correlation with experimental data [18]. This research utilized experimental results to derive source terms for mass during moisture evaporation, momentum due to the viscous and inertial resistance of the mushrooms, and energy due to the continuous evaporation process [18].

Based on the research, this project employs the CFD simulation to evaluate the boundary conditions for the drying process. These include the heat at mass transfer at the surface. The results from the boundary condition are coupled to a numerical drying model in Python, which returns the temperature at the surface while computing the internal moisture ratio and temperature for the drying paper.

2 Steady-state drying modeling

This chapter outlines the geometry and numerical setup for the steady-state drying process (second stage). This includes comparing a two- and three-dimensional CFD simulation. The primary aim of this section is to critically assess the performance and accuracy of the numerical drying model; this includes a mesh independence study and evaluation of different turbulence models. To ensure the reliability and validity of the model, the results derived from the CFD analysis will be validated against existing experimental data and new experimental results.

2.1 Geometry

Experimental results from the previous project [5] are based on the simple square geometry, with the dimensions of the drying piece being 7 cm in length, 6 cm in width, and 0.4 cm in thickness. The drying piece of paper's length was parallel with the flow direction. Figure 2.1 illustrates the experimental setup for the drying process.

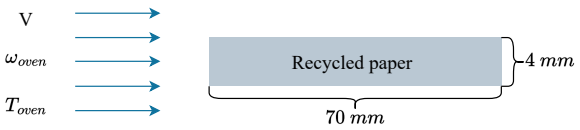


Figure 2.1: Schematic of the experimental drying process of recycled paper.

Within the experimental framework, the recycled paper was positioned inside an oven and subjected to constant velocity (V), humidity ω_{oven}, and temperature T_{oven} throughout the drying period. Initially, the recycled paper exhibited a thickness of 4 mm when fully saturated and shrunk to approximately 2 mm upon complete drying. This initial thickness was selected for the CFD simulation to represent the steady-state condition (second stage - constant evaporation), characterized by the presence of saturated vapor at the surface.

2.1.1 Two-dimensional geometry

A two-dimensional geometry of drying oven and drying paper is constructed to evaluate the performance of the CFD simulation. The two-dimensional simulation balances achieving a quick solution and maintaining accurate results. The CFD simulation for the two-dimensional geometry is designed with a planar symmetry to enhance the mesh quality. Figure 5.1 outlines the geometry of the two-dimensional simulation.

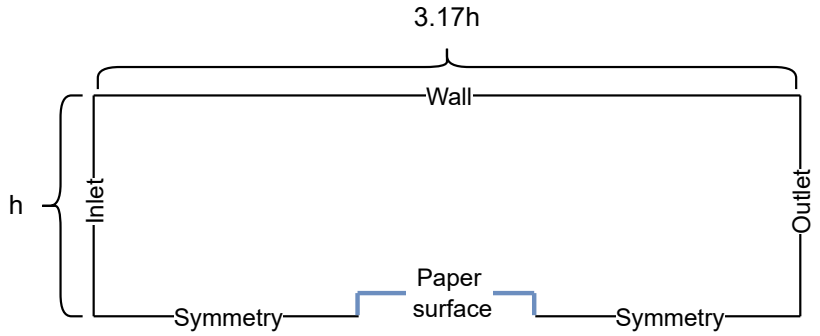


Figure 2.2: Schematic of the two-dimensional drying process.

The height h is half the height of the actual drying oven, as introduced in Chapter 4. This geometry neglects the side effects of the oven and the drying piece of paper.

2.1.2 Three-dimensional geometry

The three-dimensional geometry represents a quarter of the drying oven, including two walls that represent the top and sides of the oven, along with half the width and thickness of the drying piece of paper. The simulation is configured with two symmetry planes to represent the oven's geometry visually. Figure 2.3 illustrates the fluid domain of the CFD simulation. The gray planes represent walls, the green planes indicate the symmetry planes and the blue lines mark the placement of the drying piece of paper.

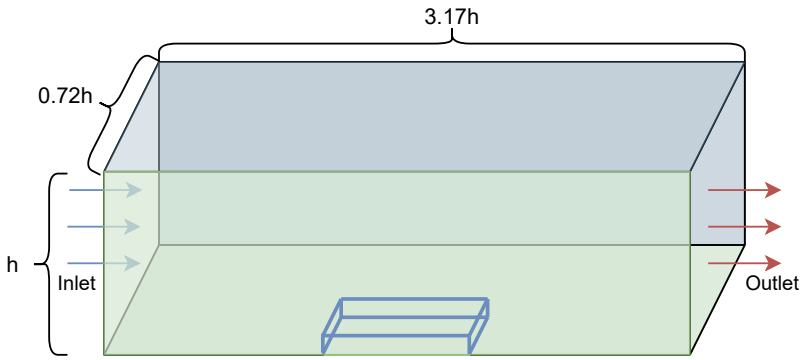


Figure 2.3: Schematic of the three-dimensional drying process.

The three-dimensional geometry includes the flow conditions between the oven walls and the piece of drying paper and the mass and heat transfer on the edges.

2.2 Numerical steady-state drying model

The numerical steady-state drying model aims to capture the evaporation rate for the drying piece of paper. The steady-state simulation enables fast and stable simulations, which is the foundation of which type of mesh and turbulence model will be used for the real egg packaging geometry.

The CFD simulation targets the consistent evaporation rate observed in the second stage of the drying process as seen in figure 1.1b. This phase is identified as a steady state due to the constant evaporation rate and temperature over time. The inputs are based on experimental data, including recorded temperature, humidity, and airflow velocity measurements. The underlying assumptions of the model are as follows:

- The air-water vapor mixture near the paper surface and in the ambient environment is treated as an ideal gas.
- The relative humidity at the paper's surface is assumed to be at 100%.
- The temperature of the paper's surface is considered constant throughout the process.
- The fluid mixture only consists of water vapor and dry air.

This model will incorporate the governing equations for mass, momentum, energy balance, and species transport, which are all written for the three-dimensional simulation.

2.2.1 Mass balance

The overall mass conservation equation is calculated with equation 2.1.

$$\nabla(\rho\vec{v}) = 0 \tag{2.1}$$

ρ is the density, $\vec{v}$ is the velocity vector,

2.2.2 Momentum balance

The momentum conservation for the simulation is shown in equation 2.2 [28].

$$\nabla \cdot (\rho\vec{v}\vec{v}) = -\nabla p + \nabla\bar{\bar{\tau}} \tag{2.2}$$

p is the static pressure and $\bar{\bar{\tau}}$ is the stress tensor [28].

The stress tensor is determined by equation 2.3 [28].

$$\bar{\bar{\tau}} = \mu\left((\nabla\vec{v} + \nabla\vec{v}^T) - \frac{2}{3}\nabla\cdot\vec{v}I\right) \tag{2.3}$$

I is the unit tensor [28].

2.2.3 Energy balance

The energy balance in the CFD simulation accounts for the enthalpy transport resulting from species diffusion [29]. This balance is represented by equation 2.4.

$$\nabla \cdot (\vec{v}(\rho E + p)) = \nabla \cdot \left(-\sum_i h_i\vec{J_i} + (\bar{\bar{\tau}}_{eff}\cdot\vec{v})\right) \tag{2.4}$$

E denotes the total energy, further detailed in Equation 2.5. The term h_j represents the sensible enthalpy of species i [29]. Additionally, $\bar{\bar{\tau}}_{eff}$ refers to the effective stress tensor [28]. The first term on the right-hand side of the equation addresses species diffusion, and the second term represents the viscous dissipation.

$$E = h - \frac{p}{\rho} + \frac{v^2}{2} \tag{2.5}$$

h is the sensible enthalpy [29].

For ideal gas is the sensible enthalpy calculated with equation 2.6.

$$h = \sum_j Y_j h_j \tag{2.6}$$

Y_j is the mass fraction of species j [29].

The enthalpy of species j is determined with equation 2.7.

$$h_j = \int_{T_{ref}}^{T} c_{p,j} dT \tag{2.7}$$

$c_{p,i}$ is the specific heat of species j, T is the temperature, and T_{ref} is 298.15 K [29].

2.2.4 Species transport

The species transport calculates the local mass fraction for each species throughout the solution of convection-diffusion [30]. The species transport for the simulations is shown in equation 2.8.

$$\nabla \cdot (\rho \vec{v} Y_i) = -\nabla \cdot \vec{J}_i \tag{2.8}$$

Y_i is the mass fraction of species i, and $\vec{J}$ is the mass diffusion of species i [30].

The mass diffusion is determined by equation 2.9 [30].

$$\vec{J}_i = -\left(\rho D_{i,m} + \frac{\mu_t}{Sc_t}\right) \nabla Y_i - D_{T,i} \frac{\nabla T}{T} \tag{2.9}$$

$D_{i,m}$ is the mass diffusion coefficient for species i, μ_t is the molecular viscosity, Sc_t is the turbulent schmidt number, and $D_{T,I}$ is the thermal diffusion coefficient [30].

2.2.5 Turbulence models

Three different turbulence models were used to evaluate the results for steady-state drying. This was done to identify the most suitable turbulence model for accurately predicting drying behavior. The three turbulence models are:

1. k - ε: This model is widely used for general-purpose turbulence simulations and is effective for fully developed turbulent flows [31].

2. k - ω: This model performs well in near-wall regions and for low Reynolds number flows, making it suitable for detailed analysis of boundary layers [31].

3. k - ω SST: This hybrid model combines the advantages of both the k-epsilon and k-omega models, providing improved accuracy for complex flows with adverse pressure gradients and separation [31].

The empirical constants in all three turbulence models were set to their default values in Ansys Fluent.

2.2.6 Physical parameters

The CFD simulation requires inputs of the mass fraction of H_2O at both the inlet and the surface of the drying paper, as well as the velocity at the inlet. This velocity is determined based on the velocity measured in the experiments. However, the mass fraction is calculated using temperature, relative humidity, and ambient pressure.

The calculation of the mass fraction is given by equation 2.10.

$$Y_A = \frac{\rho_A}{\rho_A + \rho_B} \tag{2.10}$$

Y_A epresents the mass fraction of a specific species, while ρ_A and ρ_B denote the densities of the species in question.

The density of each species is determined using Equation 2.11.

$$\rho_i = \frac{P_i M_i}{RT} \tag{2.11}$$

ρ_i is the density, P_i is the partial pressure, M_i is the molar mass, R is the universal gas constant, the subscript i indicates the specific species being considered.

The partial pressures are calculated in Engineering Equation Solver (EES), which utilizes the International Association for the Properties of Water and Steam 1997 (IAPWS-97) formulation [32].

The mass fraction of H_2O on the paper's surface is calculated to achieve 100% relative humidity. This calculation uses a User-Defined Function (UDF) where the surface temperature dictates the mass fraction. For simplicity is the saturation pressure of water in the air, the UDF calculated with the Magnus-Tetens formula [33].

$$P_{sat} = 610.94 e^{\frac{17.625T}{T+243.04}} \tag{2.12}$$

P_{sat} is the saturation pressure.

3 Transient drying model

This chapter outlines the numerical setup for the transient drying model. The primary aim of the model is to include all three stages of the drying process. The significant difference in the transient model, compared to the steady-state model, is to include the moisture content of the drying paper. Meanwhile, the steady-state model only provided an evaporation rate in the second stage. The transient drying model is set up in both Ansys Fluent and Python. Ansys fluent is used to evaluate the surface heat transfer coefficient at the boundary conditions, and Python evaluates the internal properties of the drying paper, such as the temperature and moisture content. The thickness of the paper is discretized with five nodes in Python. Figure 3.1 outlines the principles of the transient drying model.

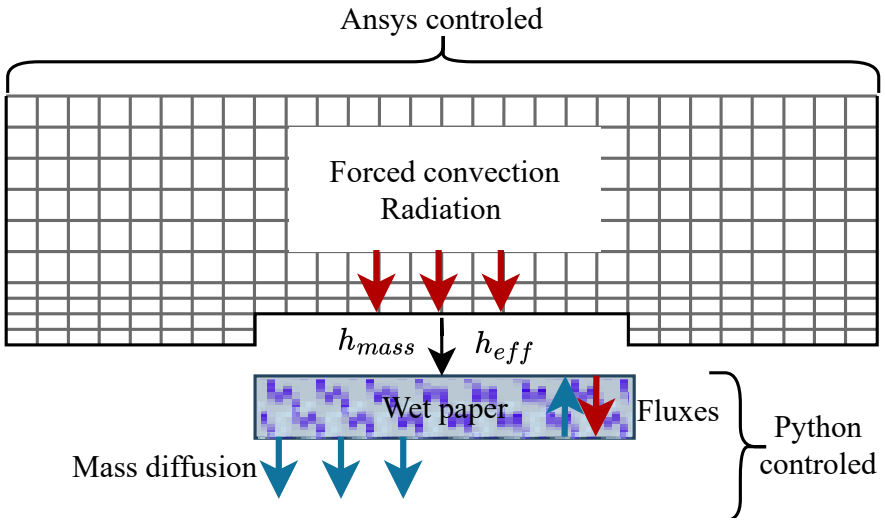


Figure 3.1: Transient drying model

The transient drying model incorporates governing equations for mass, energy, and momentum. The Ansys Fluent simulation is used to evaluate the surface heat transfer

coefficient. This includes forced convection and radiation heat transfer. The Python simulation accounts for transport phenomena within the paper, encompassing convective and diffusive air fluxes, water vapor, and liquid water. This facilitates the assessment of the moisture and temperature gradients within the drying paper.

Assumptions for the transient drying model
The transient drying model operates under the following assumptions, as noted in [4] [5]:

1. The ambient fluid mixture consists solely of water vapor and dry air.
2. The ambient fluid mixture behaves as an ideal gas.
3. The structural integrity of the pulp is preserved, with no loss of solid material during the drying process.
4. Shrinkage of the paper is confined to the direction perpendicular to the paper surface (in the thickness direction).
5. The specific heat capacity of the dry paper remains constant throughout the drying period.
6. The water/moisture content is assumed to be uniform at the start of the drying simulation.
7. The mechanisms of evaporation, radiation, and convective heat transfer are limited to the paper's surface.
8. Moisture and temperature within the paper are modeled as spatially uniform.
9. The removal of liquid water proceeds by first evaporation of free liquid, followed by bound water.

3.1 Transient drying model

The transient drying model is derived into the two segments of the Ansys Fluent part and Python part. The drying mechanisms for the paper are predominantly derived by Baggerud E [4], which provides the dynamics of the drying behavior for paper.

3.1.1 Conservation equations

The numerical drying model's conservation equation includes energy and mass conservation equations. In contrast, the CFD simulation also incorporates momentum conservation. The energy conservation aspect of the numerical drying model is depicted in equation 3.1 [4] [5].

$$\begin{aligned} &\underbrace{(c_{p,s} + \sum_{i=a,v,l} (c_{p,i}\ X_i))\frac{dT}{dt}}_{Accumulation} + \underbrace{\frac{d}{dB}(J_v^s(\triangle h_{vap} + \triangle h_{sorp}) + q_{cond})}_{Divergence} = \\ &\underbrace{-\sum_{i=a,v,l}\left(c_{p,i}\ J_i^s \frac{dT}{dB}\right)}_{Convective} + \underbrace{J_l^s \frac{d}{dB}(\triangle h_{sorp}) + J_v^s \frac{d}{dB}(\triangle h_{vap} + \triangle h_{sorp})}_{Latent\ Heat} \end{aligned} \tag{3.1}$$

In this equation, $c_{p,s}$ represents the heat capacity of the solid paper, and $c_{p,i}$ denotes the heat capacity of the individual components. The variable X_i refers to the concentration of the solid base, J_v^s is the flux of water vapor through the paper, $\triangle h_{vap}$ indicates the enthalpy of evaporation, $\triangle h_{sorp}$ is the enthalpy of sorption, and q_{cond} refers to the conduction heat transfer. Furthermore, J_l^s represents the liquid flux within the paper. The subscript i denotes different phases: air (a), water vapor (v), and liquid water (l).

The energy conservation equation in the CFD simulation is shown in equation 3.2 [29].

$$\nabla \cdot (\vec{v}(\rho E + p)) = \nabla \cdot (\lambda_{eff} \nabla T + \bar{\bar{\tau}}_{eff} \cdot \vec{v}) \tag{3.2}$$

λ_{eff} is the effective thermal conductivity.

The mass conservation equation for the Python model is divided into one for the water, liquid, and vapor and one for the air. The mass conservation equation is shown in 3.3 [4] [5].

$$\frac{dU}{dt} + \frac{d}{dB}(J_l^s + J_v^s) = 0 \tag{3.3}$$

In this equation, U is the moisture ratio.

The conservation equation for air is shown in equation 3.4.

$$\frac{dX_a}{dt} + \frac{dJ_a^s}{dB} = 0 \tag{3.4}$$

The mass conservation for the CFD simulation is shown in equation 3.5.

$$\nabla \cdot (\rho \vec{v}) = 0 \tag{3.5}$$

The conservation of momentum for the steady-state model is also used for the transient model; therefore, equations 2.2 and 2.3 are applied.

3.1.2 Boundary conditions

The drying paper is assumed to be fully exposed to the ambient conditions from all sides. Therefore, the boundary conditions are evaluated based on exposure to forced convection, radiation, conduction, and evaporation. The energy balance at the boundary condition is shown in equation 3.6 [4] [5].

$$\begin{aligned}(J_v^s(\triangle h_{vap} + \triangle h_{sorp}) - q_{cond}) = h_{heat}(T_s - T_{air}) \\ +h_{mass}(\triangle h_{vap} + \triangle h_{sorp}) + q_{rad}\end{aligned} \tag{3.6}$$

In this equation, h_{heat} is the convective heat transfer coefficient, T_s is the surface/boundary temperature of the paper, h_{mass} is the mass transfer coefficient, q_{rad} is the radiation flux. The boundary equation 3.6 is applied in the divergence term in equation 3.1.

The diffusion mass transfer at the boundary condition is calculated based on the difference in water vapor concentration. The mass transfer at the boundary condition is determined by equation 3.7 [4] [5].

$$F_m = h_{mass}(\rho_{vs} - \rho_{v,air})F(u) \tag{3.7}$$

F_m is the mass transfer, h_{mass} is the mass transfer coefficient, ρ_{vs} is the density of water vapor at the surface, and $\rho_{v,air}$ is the density of water vapor in the air. $F(u)$ is a function that limits the moisture content at the surface, and is further evaluated in subsection 3.1.9.

The mass transfer at the boundary is used for evaluating the flux; this is done with the equation 3.8 [4] [5].

$$F_m = (J_l^s + J_v^s) \tag{3.8}$$

3.1.3 Fluxes inside the drying paper

The fluxes within the drying paper consist of conduction, water vapor, liquid, and air. The conduction flux is calculated using equation 3.9 [4] [5]:

$$q_{cond} = -\frac{\lambda_{eff}}{\phi}\frac{dT}{dB} \tag{3.9}$$

ϕ is the bulk density of the solid.

The water vapor flux is determined using equation 3.10 [4] [5]:

$$J_v^s = -\rho_{v,s}\frac{K_{g,eff}}{\mu_g \phi}\frac{dP_{g,g}}{dB} - \rho_{g,g}\frac{D_{va,eff}}{\phi}\frac{dY_v}{dB} \tag{3.10}$$

$K_{g,eff}$ denotes the effective gas permeability, μ_g is the viscosity of water vapor, P_{tot} is the total pressure, ρ_{tot} is the total gas density, $D_{va,eff}$ refers to the effective diffusion of water in air, and Y_v is the mass fraction of water vapor.

The air flux is calculated using equation 3.11 [4] [5]:

$$J_a^s = -\rho_{air}\frac{K_{g,eff}}{\mu_g \phi}\frac{dP_{g,g}}{dB} - \rho_{g,g}\frac{D_{va,eff}}{\phi}\frac{dY_a}{dB} \tag{3.11}$$

ρ_{air} is the air density, and Y_a is the air mass fraction.
The water liquid flux is calculated using equation 3.12 [4] [5]:

$$J_l^s = \frac{D_L(U)}{\phi^2}\frac{dU}{dB} \tag{3.12}$$

$D_L(U)$ represents the diffusion coefficient for liquid water.

3.1.4 Surface to Surface radiation

The S2S radiation model is utilized in the CFD simulation for obtaining the radiation flux. The energy transfer between two objects depends on distance, orientation, and size. These parameters are accounted for by the view factor [34]. The radiation flux is calculated with equation 3.13 [34].

$$q_{rad,k} = \varepsilon_k \sigma T_k^4 + \rho_k \sum_{j=1}^{N} F_{kj} q_{rad,j} \tag{3.13}$$

$q_{rad,k}$ represents the energy flux at surface k, ε_k is the emissivity of surface k, σ is the Stefan-Boltzmann constant, and ρ_k is the reflectivity at surface k. F_{kj} is the view factor from surface k to surface j, and $q_{rad,j}$ is the energy flux emitted from surface j. The indices k and j represent the surfaces between which the energy flux is emitted and absorbed.

The view factor is calculated with equation 3.14.

$$A_j F_{jk} = A_k F_{kj} \; for \; j = 1, 2, 3...N \tag{3.14}$$

3.1.5 Calorimetric properties

The calorimetric properties include the enthalpy of evaporation and the sorption enthalpy. The enthalpy of evaporation in the Python model is obtained from tables provided by IAPWS-97 [32]. The sorption enthalpy is calculated using equation 3.15 [4] [5].

$$\triangle h_{sorp}(U,T) = \frac{RT^2}{M_v} \frac{1-\psi(U,T)}{\psi(U,T)} 0.10085 U^{1.0585} \tag{3.15}$$

R is the universal gas constant, M_v is the molar mass of water vapor, and ψ represents the water activity, a function of the moisture content and temperature.

Water activity is determined by equation 3.16.

$$\psi(U,T) = 1 - e^{-(47.58U^{1.877} + 0.10085TU^{1.0585})} \tag{3.16}$$

3.1.6 Physical Parameters

The physical parameters include pressure and density. Density is calculated using equation 2.11, while the total pressure is calculated using equation 3.17:

$$\rho_{tot} = \sum_{i=v,a} \rho_i \tag{3.17}$$

Here, ρ_{tot} represents the sum of air and water vapor densities.

The total pressure is calculated using equation 3.18.

$$P_{tot} = \sum_{i=v,a} P_i \tag{3.18}$$

P_{tot} is the sum of the partial pressures of air and water vapor.

3.1.7 Solid bulk density

The solid bulk density of paper depends on its composition, specifically the length of its fibers. This density describes the paper's specific volume based on its moisture content

and is derived from experimental results. For this project, the solid bulk density was calculated using equation 3.19.

$$\phi(U) = \frac{1}{\rho_{ss}} + U\frac{1}{\rho_{sl}} \tag{3.19}$$

ϕ is the solid bulk density, ρ_{ss} is the specific density of the paper, and ρ_{sl} is the specific density of liquid water.

3.1.8 Volume Fractions

The volume fractions of drying paper include solid paper, liquid water, and a gas mixture of water vapor and air. The volume fraction of solid paper is calculated with equation 3.20.

$$\varepsilon_s = \frac{1}{\phi\rho_{ss}} \tag{3.20}$$

ε_s represents the solid volume fraction.
The volume fraction of liquid water is determined using equation 3.21.

$$\varepsilon_l = \frac{U}{\phi\rho_{sl}} \tag{3.21}$$

ε_l is the liquid volume fraction.

The gas volume fraction is computed to ensure the total volume fraction equals unity, as shown in equation 3.22.

$$\varepsilon_g = 1 - \varepsilon_s - \varepsilon_l \tag{3.22}$$

ε_g is the gas volume fraction.

3.1.9 Tranpsport parameters

The transport mechanisms in the drying model include conduction, diffusion, and convection. Each mechanism is governed by an equation that depends on a transport coefficient. The heat transfer coefficient for conduction is calculated using equation 3.23.

$$\lambda_{eff} = \frac{\lambda_s + U\lambda_l}{1 + U} \tag{3.23}$$

λ_s represents the heat transfer coefficient for solid paper, and λ_l for liquid water.

The diffusion coefficient between air and water vapor is determined using equation 3.24 [35].

$$D_{va,eff} = 1.87 \times 10^{-10} \frac{T^{2.072}}{P_{atm}}, \quad 280K < T < 450K \tag{3.24}$$

Liquid diffusion varies with the moisture content. As moisture is removed, initially free water and then bound water are extracted. The diffusion coefficients for these stages are calculated with equation 3.25.

$$D_L(U) = \begin{cases} D_0(1 - e^{-174U^{3.7}}) & if U \leq U_{FSP} \\ D_0(1 - e^{-174U^{3.7}})\, e^{\beta(U - U_{FSP})} & if U > U_{FSP} \end{cases} \tag{3.25}$$

D_0, β, and U_{FSP} are empirical parameters that depend on the paper's composition. U_{FSP} denotes the fiber saturation point where all free water has evaporated.
Gas permeability is calculated using the equation 3.26, based on the experiments by Nilsson and Stenström [36].

$$D_{g,eff} = 5.15 \times 10^{-10} e^{-\frac{0.0136}{\phi}} \tag{3.26}$$

The surface heat transfer coefficient is a combined heat transfer coefficient to account for radiation and convection heat transfer and is calculated with equation 3.27.

$$h_{eff} = \frac{q_{conv}}{T_s - T_{air}} \tag{3.27}$$

h_{eff} is the surface heat transfer coefficient.

The mass transfer coefficient is calculated using the Lewis relation heat and mass transfer analogy derived from the Chilton-Colburn analogy. The mass transfer coefficient is calculated with equation 3.28.

$$h_{mass} \cong \frac{h_{eff}}{\rho c_p} \tag{3.28}$$

As the moisture content of the paper is removed, the surface is not fully saturated at some point because the evaporation rate exceeds the moisture diffusion internally. To account for this phenomenon is the $F(U)$ function added to boundary evaporation. The $F(u)$ is an exponential function of the moisture ratio at the boundary condition. Equation 3.29 shows the exponential function.

$$f_f(U) = (1 - e^{-\chi \cdot U^{\gamma}}) \tag{3.29}$$

χ and γ is two fitting parameters.

The two fitting parameters are used to ensure the evaporation rate is not limited in the first and second stages of the drying process. Figure 3.2 represents a few examples of the function.

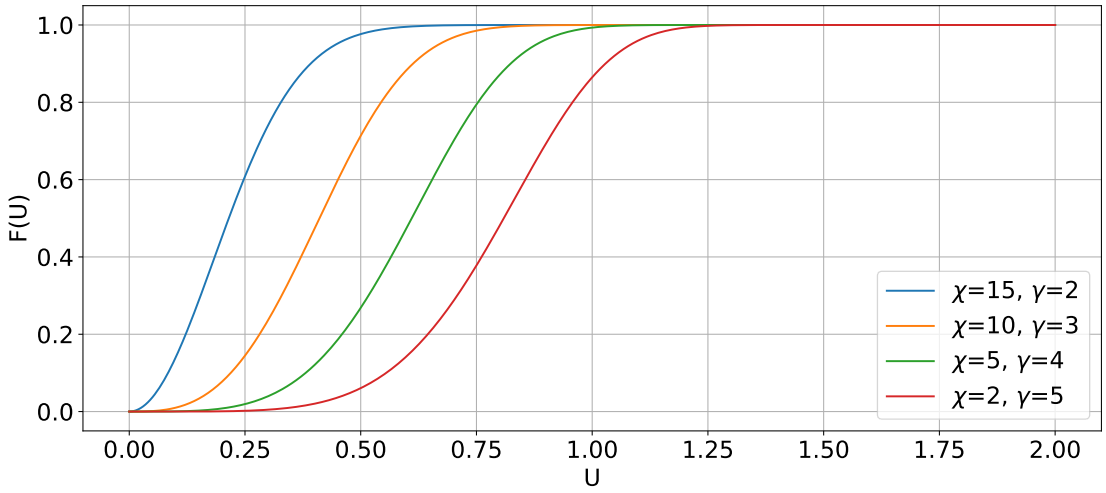


Figure 3.2: Illustration of the limiting evaporation function $F(U)$

The figure shows how the two fitting values can limit the evaporation rate between 0 and 100% based on the moisture content. This allows the function to be fitted to limit evaporation as the third stage of the drying process starts.

3.1.10 Transient Drying Model Implementation

The transient drying model is set up by first evaluating the heat and mass transfer coefficients using Ansys Fluent. These coefficients are then transferred into Python. Within Python, the drying model is established by defining the initial temperature, moisture content conditions, and fitting parameters.

The Python model is designed to solve 60 equations per node per time step. The model runs with a time step size of 0.1 seconds; using larger time steps results in stability and convergence issues. The simulation algorithm is depicted in figure 3.3, and the Python code can be found in Appendix A.

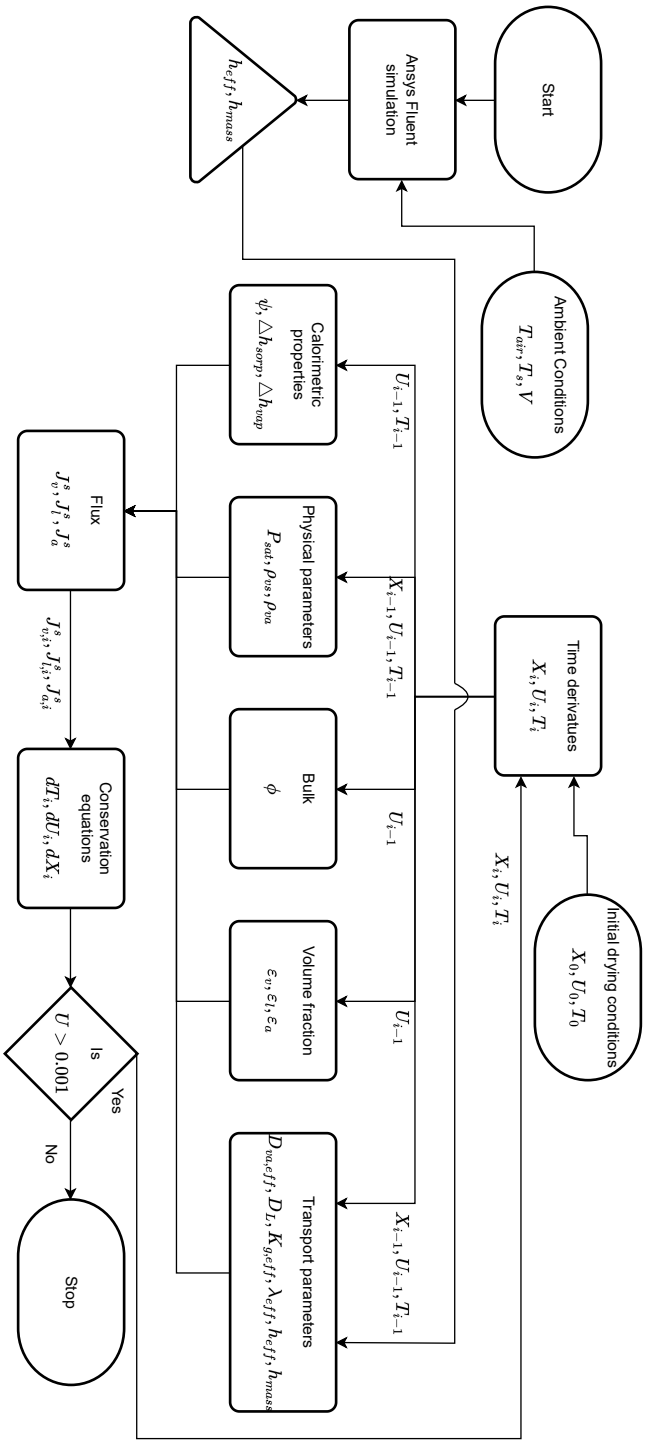


Figure 3.3: Simulation flow chart of the drying process

4 Experimental setup

This chapter outlines how the experimental results for the drying process are obtained. It includes a description of the geometry as well as the selection of measuring instruments and their associated uncertainties.

4.1 Drying oven

In the previous project, as described by [5], a small drying oven was constructed to dry paper squares. This oven was made from polystyrene, and heat was provided by an electric heater. It could monitor temperature, humidity, and weight throughout the drying process. The same drying oven is being used for the current project, which involves drying egg packaging. However, the oven has been slightly modified, and a FLIR camera has been added to enhance its functionality. The updated design of the oven is illustrated in figure 4.1.

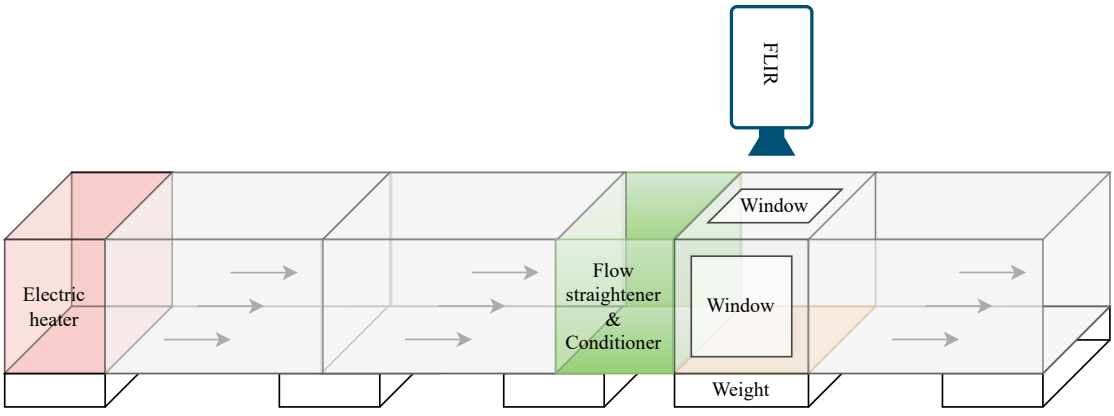


Figure 4.1: Drying oven, where the flow straightener and condition has been moved close to the drying paper

The modification in the new drying oven design involves relocating the flow straightener and conditioner. Previously positioned between the electric heater and the drying

paper, it is now placed directly before the paper to be dried. The drying area is 3.2 meters in length, with an internal width and height of 0.26 meters and 0.36 meters, respectively. For visualizing the real size, is, the real oven shown in figure 4.2.

Figure 4.2: Photo of the real drying oven located in the office.

4.1.1 Instruments Used in the Drying Oven

The temperature inside the drying oven is regulated by an electric heater, which also provides airflow through the oven. The electric heater features three distinct operational settings:

1. The first setting turns on only the fan, providing airflow without heat.
2. The second setting activates a 1.5 kW heating element in addition to the fan.
3. The third setting increases the heating to its maximum capacity of 3 kW, while the fan continues to operate.

These settings enable the maintenance of three different temperature ranges, all of which are affected by the ambient temperature surrounding the oven.

Positioned between the electric heater and the flow straightener and conditioner is a Digital ALMEMO FHAD36R [37], used to measure temperature, relative humidity, and pressure. The collected data is logged using an ALMEMO 2590 [38]. These measurements—temperature, relative humidity, and pressure are recorded as averages over the entirety of the drying process. The measurement range and uncertainties are listed in table 4.1

The weight is monitored continuously throughout the drying period by a KERN PCV 3000-2-2023 [39] scale, logging data four times per second.

The FLIR camera, a Testo 865 model [40], captures an image every minute throughout the drying period. Between the FLIR camera and the drying egg packaging, a plastic sheet is placed. To compensate for the presence of this plastic barrier, the emissivity setting of the FLIR camera is adjusted. This adjustment process involves measuring the temperature of the egg packaging using an R2E4 temperature probe. Subsequently, the FLIR camera's emissivity setting is modified to match the temperature reading obtained from the egg packaging, ensuring accurate temperature measurements are captured [41].

To capture the velocity profile at the oven's exit, measurements are conducted with a miniature digital vane anemometer (model FVAD 15 S120/S140) [42]. Velocity is measured at nine distinct points across the exit area. At each of these points, data is collected for a duration of one minute. The average value from these measurements is then used to construct the overall velocity profile. The measured values is linear extrapolated to create the velocity profile. Figure 4.3 show the velocity profile for the drying oven.

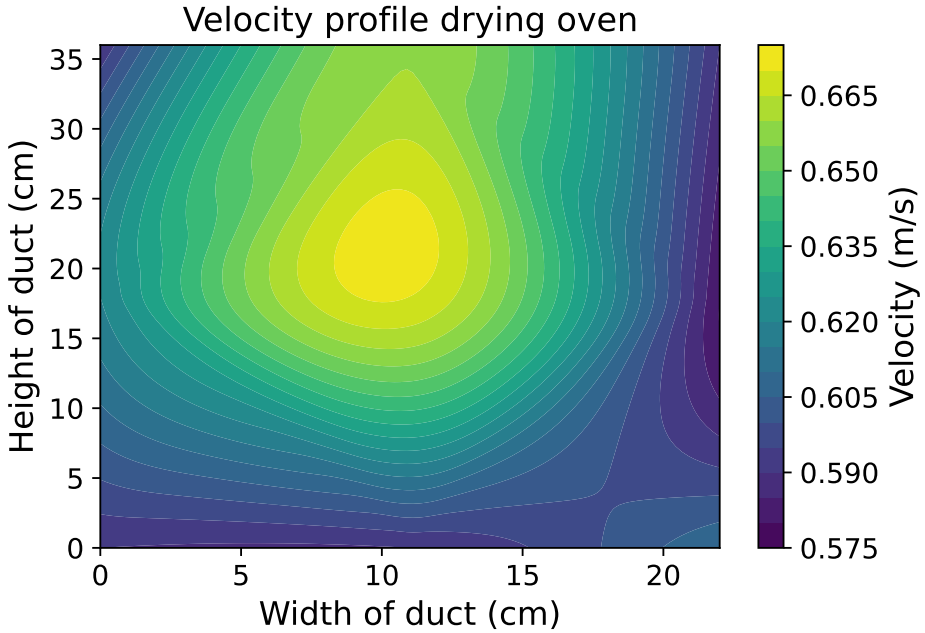


Figure 4.3: Velocity profile of the drying oven

The velocity profile indicates a fully developed flow, with the highest velocity in the center. There is, however, a slightly higher velocity in the top center, which might be caused by the density difference close to the ambient conditions.

Table 4.1 details the instruments utilized in the drying oven along with the uncertainties associated with their measurements.

Table 4.1: Instrument used for the drying process, MV is the measured value

Instrument	Manufacturer	Model	Uncertainty	Measuring interval
Weight	KERN [39]	PCB 3000-2-2023	$\pm$ 0.05g	0-3.6 kg
FLIR Camera	Testo [40]	Testo 865	$\pm$2.0 C	-20°C to +280°C
Anemometer	Kayteck Instruments [42]	FVAD 15 S120/S140	$\pm$ 1.5 % MV	0.4-20 m/s
Humidity Probe	Rotronic [43]	HC2A-IC102	$\pm$ 0.1 C $\pm$ 0.8 % RH	-100°C to 200°C 0 to 100 % RH

5 Validation of the drying process

This chapter compares the experimental results obtained from the previous project [5] with the steady-state and transient drying model. This is done before expanding the CFD simulation into the more complex geometry.
The preceding experimental findings from [5] are the benchmark for the CFD simulations. The experiment was conducted with the ambient conditions shown in table 5.1.

Table 5.1: The operating condition for the drying oven and paper temperature at stage two.

Input	Oven temperature	Oven humidity	Oven velocity	Paper temperatur
Value	55.0 °C	0.00575 kg_{H_2O}/kg_{air}	0.700 m/s	25.2 °C

The moisture content and temperature of the drying paper are shown in figure 5.1, where the three stages of the drying phase are depicted in distinct colors.

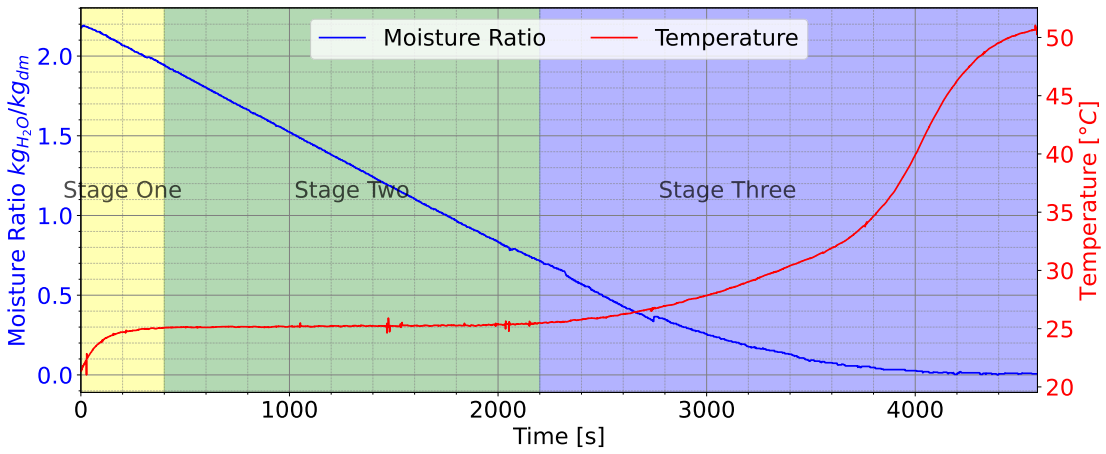


Figure 5.1: Experimental data for the moisture ratio and temperature [5].

The evaporation rate during stage two is calculated to be $2.04 \cdot 10^{-6}$ kg_{H_2O}/s.

5.1 Mesh of two-dimensional model

The mesh for the two-dimensional model is meshed in Ansys mesh. The geometry for the simulations encompasses only the fluid volume surrounding the drying piece of paper. Figure 5.2 illustrates the general schematic and the mesh used two-dimensional simulation.

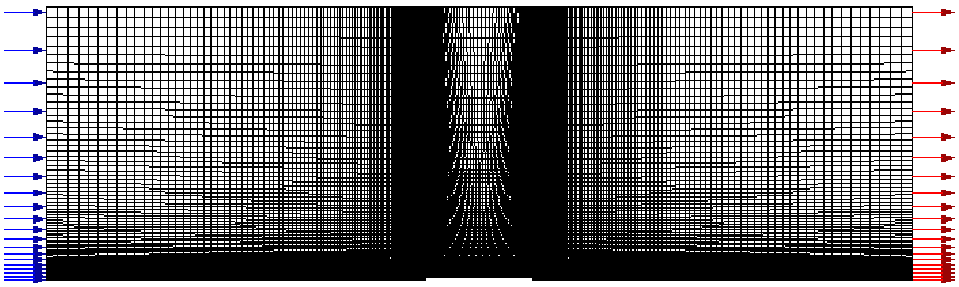


Figure 5.2: Schematic of the geometry and mesh of the two-dimensional CFD simulation

The mesh is constructed as a structured mesh, supplemented with a specified number of divisions around the drying piece of paper. These divisions have been biased to ensure higher resolution near the paper and larger cells further away. Figure 5.3 displays the mesh in close proximity to the surface of the drying paper.

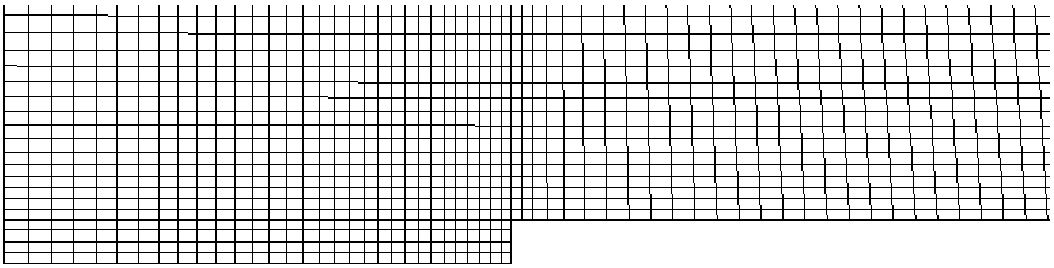


Figure 5.3: Schematic of the geometry and mesh of the first part of the CFD simulation

5.2 Mesh of three-dimensional model

The mesh for the three-dimensional model has been constructed using Ansys Fluent. This simulation encompasses both the oven's fluid volume and the drying paper's solid volume. Figure 5.4 presents the surface mesh from a top view, wherein the top of the geometry has been rendered transparent for clarity; figure 5.5 illustrates the volume mesh in close proximity to the drying paper.

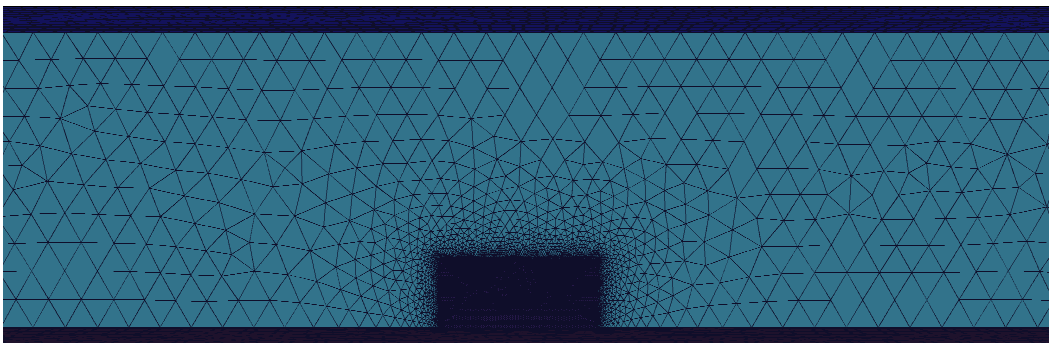


Figure 5.4: Top view of the surface mesh for the symmetry plane, with a refined mesh close to the drying paper

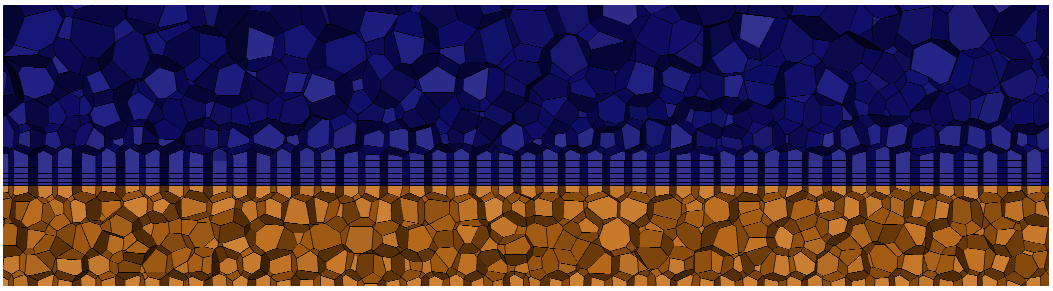


Figure 5.5: Volume mesh close to the drying paper where inflation layers have been applied towards the fluid domain.

The mesh has been designed with local sizing around the drying paper to enhance simulation accuracy. The solid volume of the paper and the fluid volume share a surface. Additionally, inflation layers have been applied to all walls adjacent to the fluid region. The volume mesh comprises polyhedral cells.

5.2.1 Simulation results

The simulations were run with three mesh sizes, and four turbulence models were evaluated for each mesh. The evaporation results for the two-dimensional simulation are

shown in table 5.2, and the results for the three-dimensional simulations are shown in table 5.3.

Table 5.2: Evporation rates for the two-dimensional simulation, where different mesh sizes and turbulence models have been used

	Cells		
Turbulence model	20.7k	45.0k	81.2k
$k-\varepsilon$	$1.88 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$1.84 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$1.80 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$
$k-\omega$	$1.23 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$1.23 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$1.24 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$
$k-\omega\ SST$	$1.2 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$1.21 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$1.22 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$
Transition 4 eq	$1.15 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$1.15 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$1.15 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$

The mesh independence study indicates that k-epsilon is more sensitive to the number of cells than other models. The variations in results from the k-ω, k-ω SST, and the transition 4 eq models are minimal, barely deviating between 19.2k and 82.9k cells.

The y^+ values for the two-dimensional simulation are shown in figure 5.6. The plot is based on the simulation with 82.9k cells.

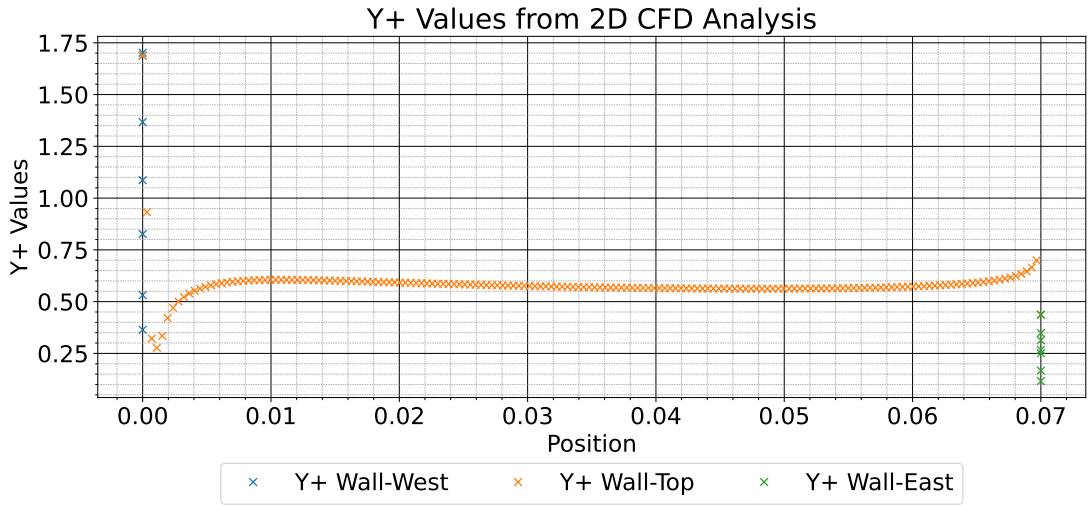


Figure 5.6: Distribution of y^+ values across different walls of the two-dimensional drying.

The results from the y^+ values indicate that all the turbulence models can predict the turbulence generated at the walls. The highest y^+ value is at the front wall perpendic-

ular to the flow direction at 1.7, with an average of 0.578.

Figure 5.7 displays the mass fraction distribution for the two-dimensional simulation, including the symmetry plane.

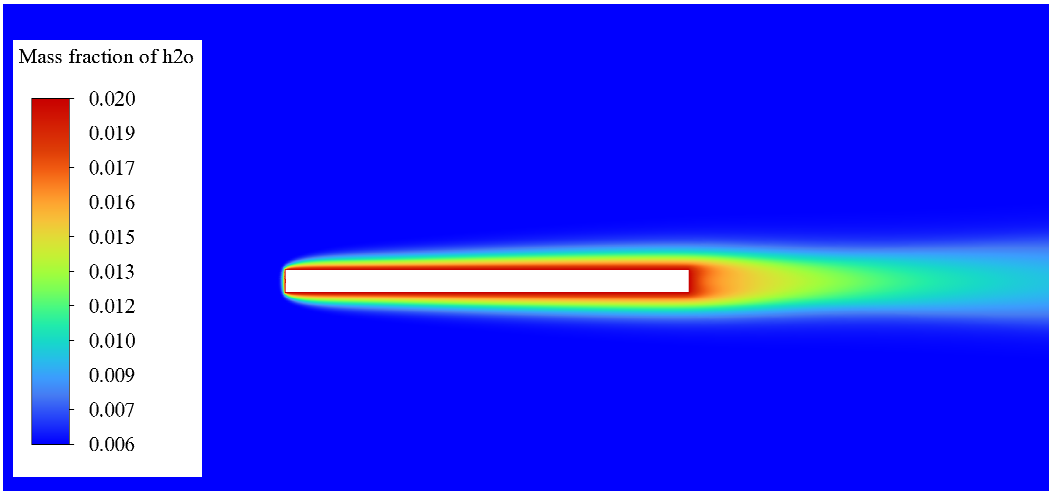


Figure 5.7: Contour plot of the mass fraction distribution from the two-dimensional steady-state simulation.

The contour plots of the mass fraction are nearly identical among the different turbulence models and mesh sizes.

The evaporation rate for the three-dimensional simulation results is shown in table 5.3.

Table 5.3: Evporation rates for the three-dimensional simulation, where different mesh sizes and turbulence models have been used

	Cells		
Turbulence model	19.2k	41.0k	82.9k
$k-\varepsilon$	$2.38 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$2.42 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$2.47 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$
$k-\omega$	$2.16 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$2.19 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$2.21 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$
$k-\omega\ SST$	$2.15 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$2.18 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$2.20 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$
Transition 4 eq	$2.15 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$2.18 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$	$2.20 \cdot 10^{-6}\ \mathrm{kg_{H_2O}/s}$

The findings from the three-dimensional simulations indicate that the $k-\varepsilon$ turbulence model yields the most inaccurate predictions, starkly contrasting the two-dimensional

simulation results. The $k-\omega\ SST$ and transition 4 eq models align more closely with the experimental results, emerging as the most accurate. Notably, these two turbulence models produce similar outcomes unaffected by variations in the number of cells.

Figure 5.8 presents a top-view contour plot of the mass fraction of H_2O. The side view resembles the plot from the three-dimensional simulation, as depicted in figure 5.7.

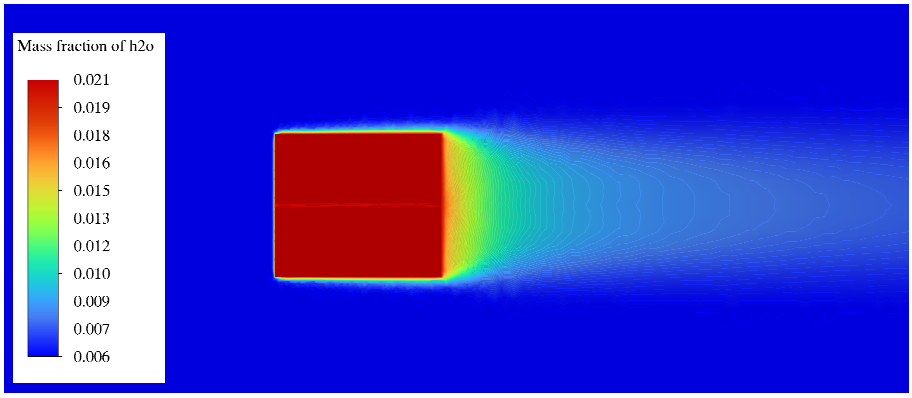


Figure 5.8: Top view - Contour plot of the mass fraction distribution from the three-dimensional steady-state simulation.

Figure 5.8 clearly shows the effect from the sides of the drying paper, which is neglected in the two-dimensional simulation.

The see flow crossing the drying paper is the pathlines of the mass fraction of H_2O shown in animation 5.9. The animation requires a PDF reader, which supports animations; Adobe Acrobat Reader supports this feature.

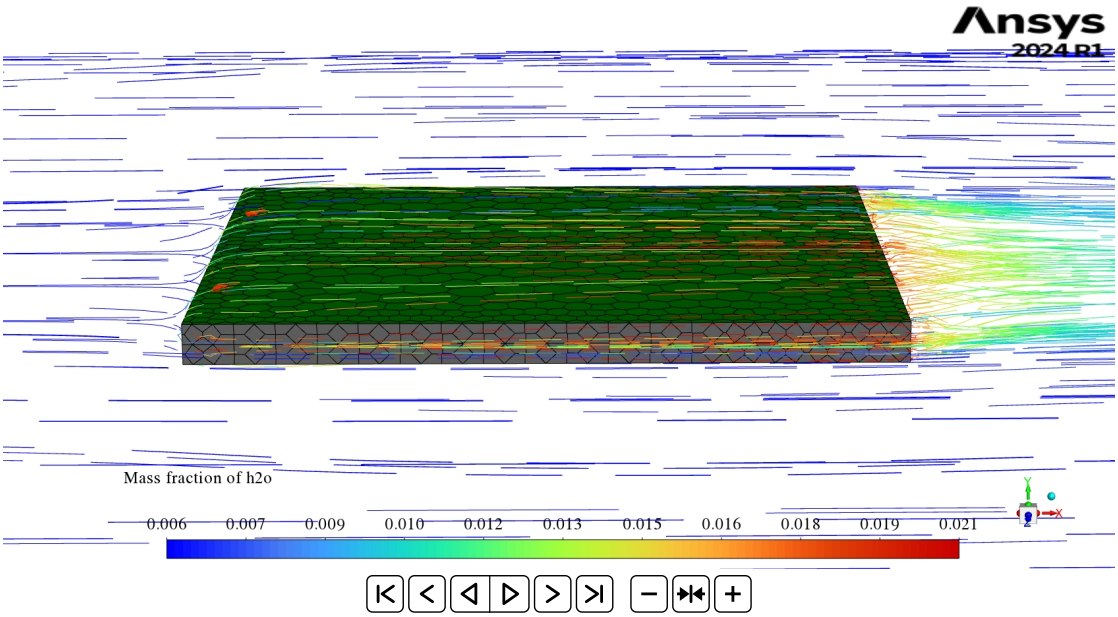


Figure 5.9: Pathlines of mass fraction of H_2O for the steady-state three-dimensional simulation.

The animation reveals a small recirculation zone at the front of the paper on the top, while a larger recirculation zone appears at the paper's end. In contrast, the flow along the length of the paper remains laminar.

To ensure that the turbulence models can accurately capture the turbulence generated at the surface, the y^+ values are plotted in Figure 5.10. The "Wall-West" refers to the front wall perpendicular to the inlet flow direction, "Wall-Side" refers to the side of the drying paper, "Wall-Top" refers to the top of the drying paper, and "Wall-East" refers to the back end of the paper.

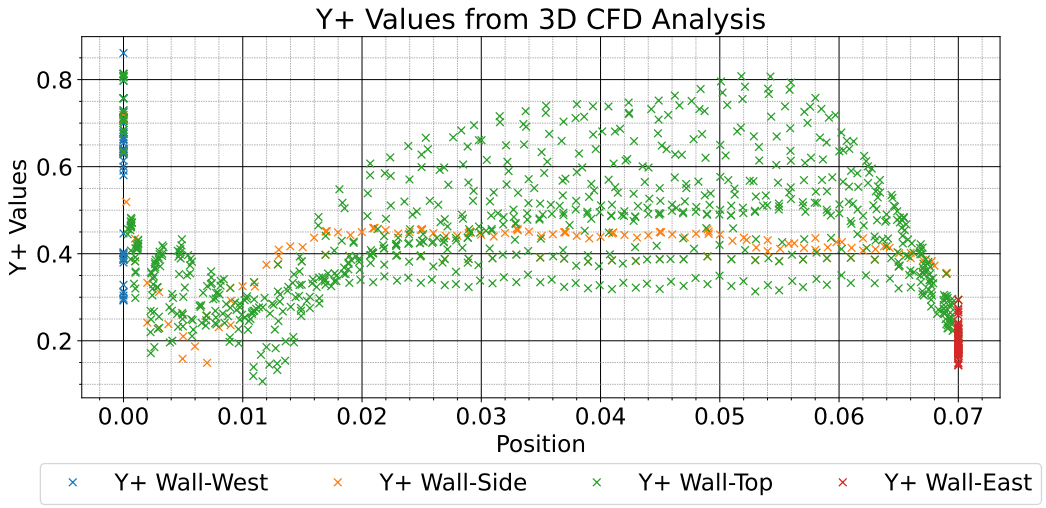


Figure 5.10: Distribution of y^+ values across different walls of the drying paper.

The highest y^+ value observed is 0.861, which occurs at the front of the drying paper. The general range of y^+ values spans from 0.2 to 0.8, indicating that the k-ω SST turbulence model is applicable. These y^+ values are appropriate for the k-ω SST, as they suggest that the near-wall region is adequately resolved, ensuring accurate turbulence modeling near the surface.

The steady-state drying model results from the two-dimensional and three-dimensional simulations have predicted an evaporation rate. Notably, the three-dimensional simulation offers a more accurate representation compared to experimental results. Furthermore, it has been demonstrated that the k-ω SST model is particularly well-suited for modeling the surface diffusion of water vapor. Consequently, the transient model will be developed based on a three-dimensional simulation using the k-ω SST turbulence model.

5.3 Validation of the transient drying model

The transient drying model is validated with experimental results obtained in [5], as shown in figure 5.1. The ambient conditions are the same as the steady-state drying model, as shown in table 5.1. The transient drying model requires the following constants: $D_0, \beta, U_{FSP}, \psi$, and γ. The values are shown in Table 5.4.

Table 5.4: Constant values used in the transient drying model.

Input	D_0	β	U_{FSP}	χ	γ
Value	1e-7	2	0.3	5	2

The three-dimensional mesh with 82.9k cells is used to evaluate the transient drying model's heat and mass transfer coefficients. The simulation is run with and without the radiation S2S. Table 5.5 shows the heat and mass transfer coefficients utilized in the Python model.

Table 5.5: Heat and mass transfer coefficients from the CFD simulation of the three-dimensional drying.

Simulation	h_{eff} (W/(m^2 K))	h_{mass} (m/s)
Without radiation	35.8	0.029
With radiation	55.2	0.0447

The results from the Python drying model with the inputs from tables 5.4 and 5.5 are shown in figure 5.11.

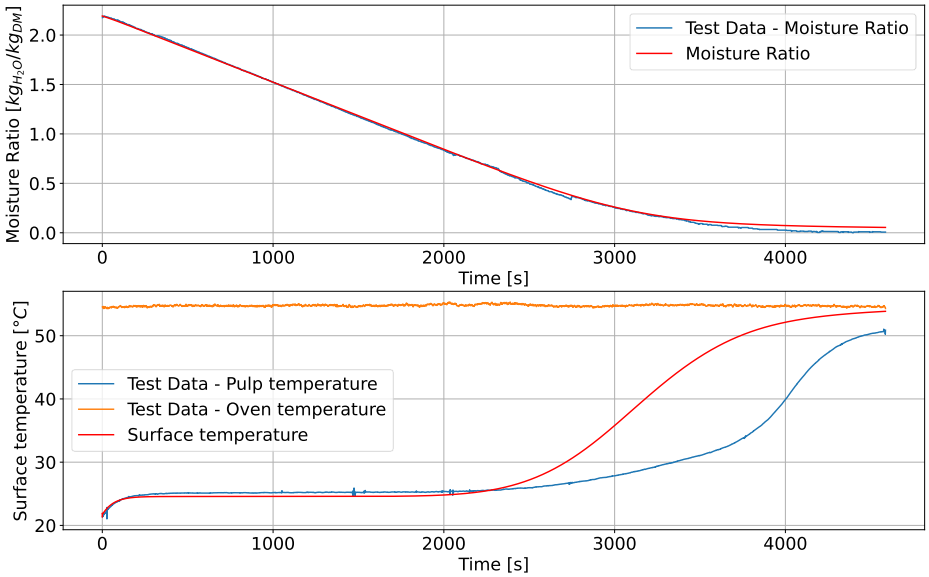


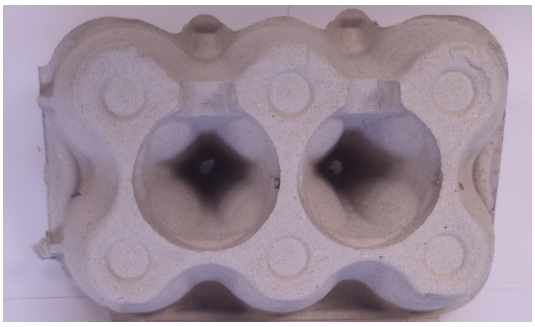
Figure 5.11: Moisture content and temperature evolution for the transient drying model.

The transient drying model aligns well with the moisture ratio throughout the drying period, with only a slight deviation near when the paper is completely dried. The temperature model aligns well with the first two stages of the drying process. However, as the final drying phase begins, the temperature increases faster than measured.

6 Results

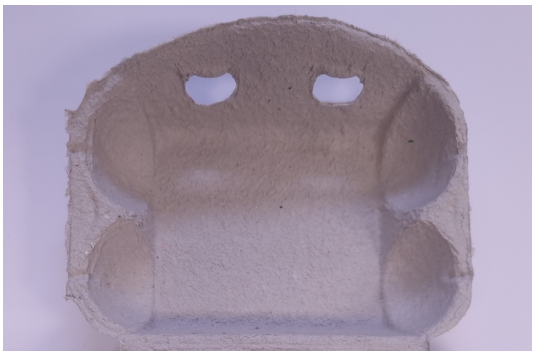
This chapter shows the steady-state and transient drying model results compared to experimental results. The two models are applied to an egg-packaging geometry. The mesh used for the simulation is also described in this chapter.

To align with the named sections, the egg-packaging has been divided into four sections, which are shown in figure 6.1:

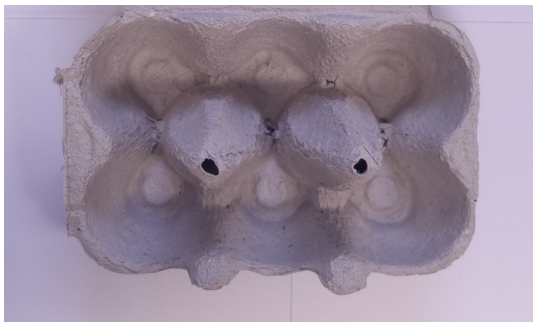


(a) Bottom-bottom view

(b) Bottom-lid view



(c) Top-lid view



(d) top-bottom view

Figure 6.1: Figures of an actual egg packaging, illustrated from the four different angles

6.1 Mesh of egg packaging

An external company provided the egg packaging geometry. The CAD file was prepared in SpaceClaim 2024 R1, where bad surfaces were merged or removed. Ansys Fluent Mesh was used to mesh the egg packaging and enclosure geometry. The mesh was constructed with a refined mesh on the egg packaging and a refined surface mesh. This resulted in 320k surface cells. Figure B.3 shows the top-bottom surface mesh.

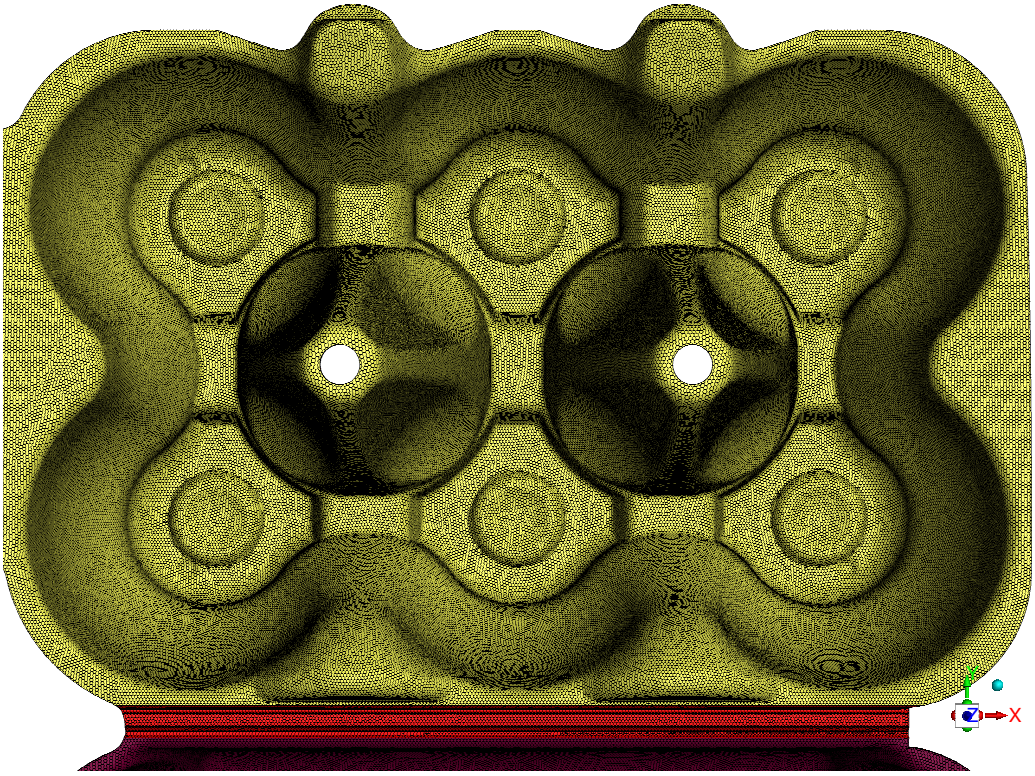


Figure 6.2: Surface mesh of the top-bottom of the egg packaging

The mesh of the other three parts of the egg packaging can be seen in appendix B.

The volume mesh includes five inflation layers, and to limit the total number of cells, the volume mesh is constructed with poly-hexcore. The volume mesh is shown in figure 6.3.

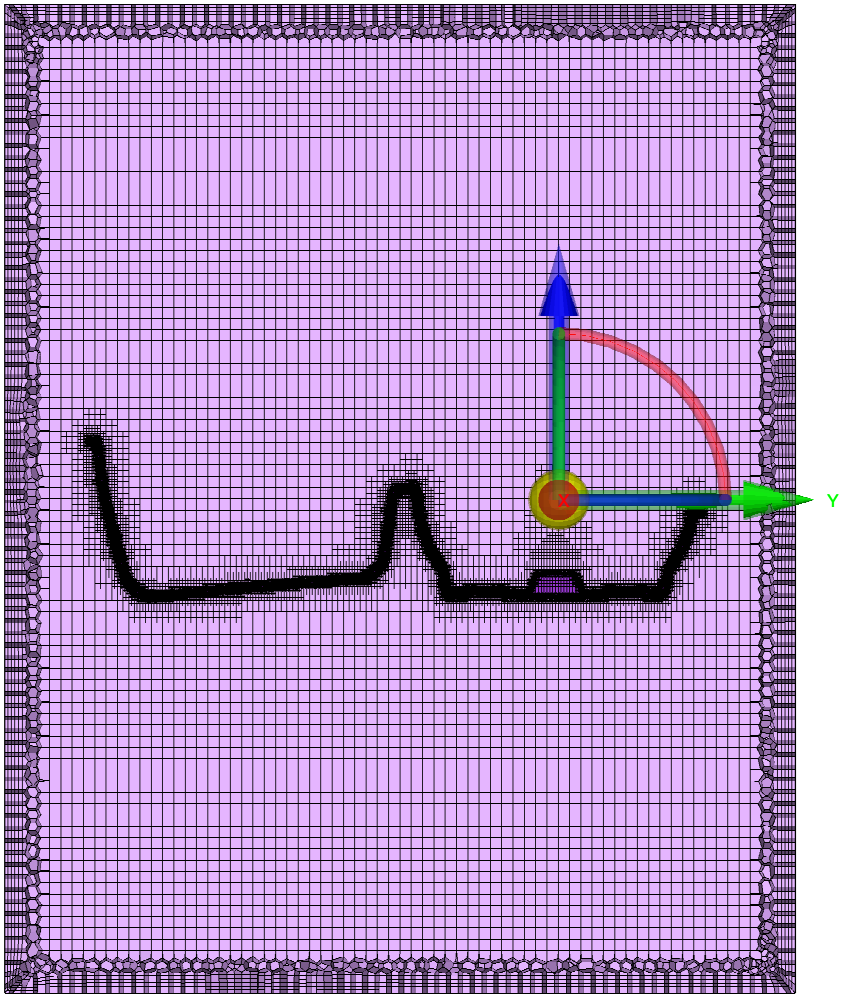


Figure 6.3: Volume mesh of the egg packaging

This resulted in 1000k cells, with a minimum orthogonal quality of 0.103 and a maximum aspect ratio of 5.60. ANSYS Fluent checks the mesh quality without any warnings.

The inlet length for the CFD simulation has been extended to 1 meter instead of the previous 0.5 meters. This change ensures a fully developed flow close to the walls, as the egg packaging edges are relatively close compared to the small square used in previous simulations, as shown in figure 6.3 and figure 5.4.

6.2 Experimental results

The drying process of the egg packaging was conducted by rewetting a dry egg packaging. This was done by submerging the egg packaging in a bucket of water until a moisture content of approximately two was obtained. The dry mass of the egg packaging is 23 g. Table 6.1 shows the oven conditions and starting moisture content.

Table 6.1: The operating condition for the drying oven

Input	Oven temperature	Oven humidity	Oven velocity	Moisture content
Value	51.8 °C	0.0062 kg_{H_2O}/kg_{air}	0.6 m/s	2.1 kg_{dm} / kg_{H_2O}

Figure 6.4 shows the results of the moisture content evolution during the drying.

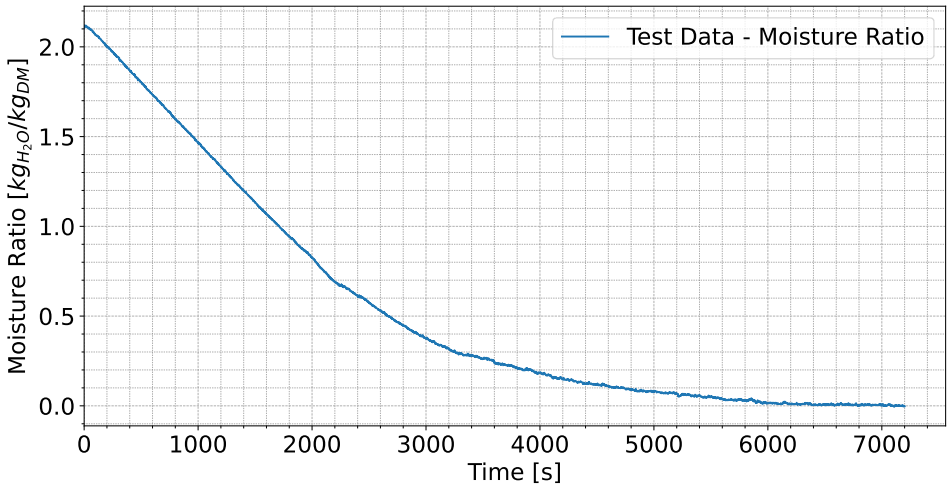


Figure 6.4: The moisture content of the egg packaging during drying.

Figure 6.4 indicates the second and third stages of the drying process, whereas the first stage is quite hard to identify. The second stage of the drying process starts at 200 seconds and ends at approximately 2000 seconds. The third stage occupies over 70% of the total drying time. The drying process took approximately two hours (7200 seconds).

With a starting moisture content of 2.1 kg_{dm} / kg_{H_2O}, the total starting mass is 71.3 g. At 200 seconds, the total mass is 69 g, and after 1800 seconds, the total mass is 41.9 g, resulting in an evaporation rate of $1.505 \cdot 10^{-5}$ kg_{H_2O} / s.

The local temperature is captured with a FLIR camera. The images are processed using the testo Thermography app, where an area surrounding the egg packaging is used to determine the minimum, maximum, and average temperatures. Figure 6.5 shows the minimum and average temperatures and the uncertainty related to the FLIR camera. Due to the complex geometry, the maximum temperature is not depicted, as the maximum temperature is located in places other than the egg packaging.

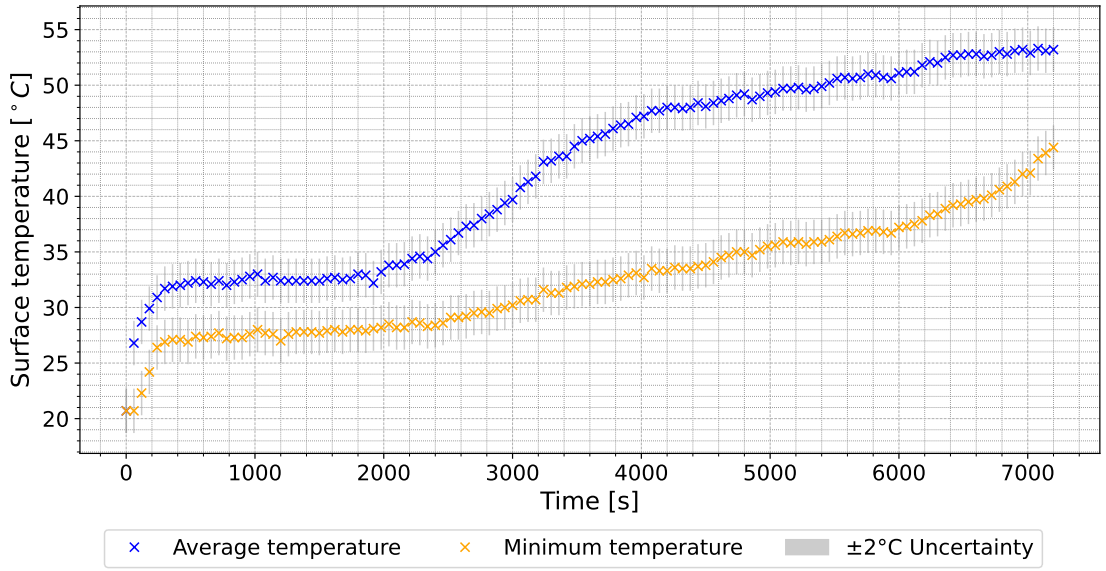


Figure 6.5: The temperature of the egg packaging during drying.

Figure 6.5 highlights the first stage of the drying process more clearly than the moisture content in figure 6.4. The starting temperature is approximately 25°C. The constant temperature in stage two is around 30°C. The temperature evolution in the third stage aligns well with the decreasing evaporation rate, as seen in Figure 6.4.

Figure 6.6 shows an animation of the drying process, where 120 pictures are displayed (one per minute of the drying process). This also illustrates where the maximum, minimum, and average temperatures are captured. The direction of the hot air flows from the bottom of the figure towards the top.

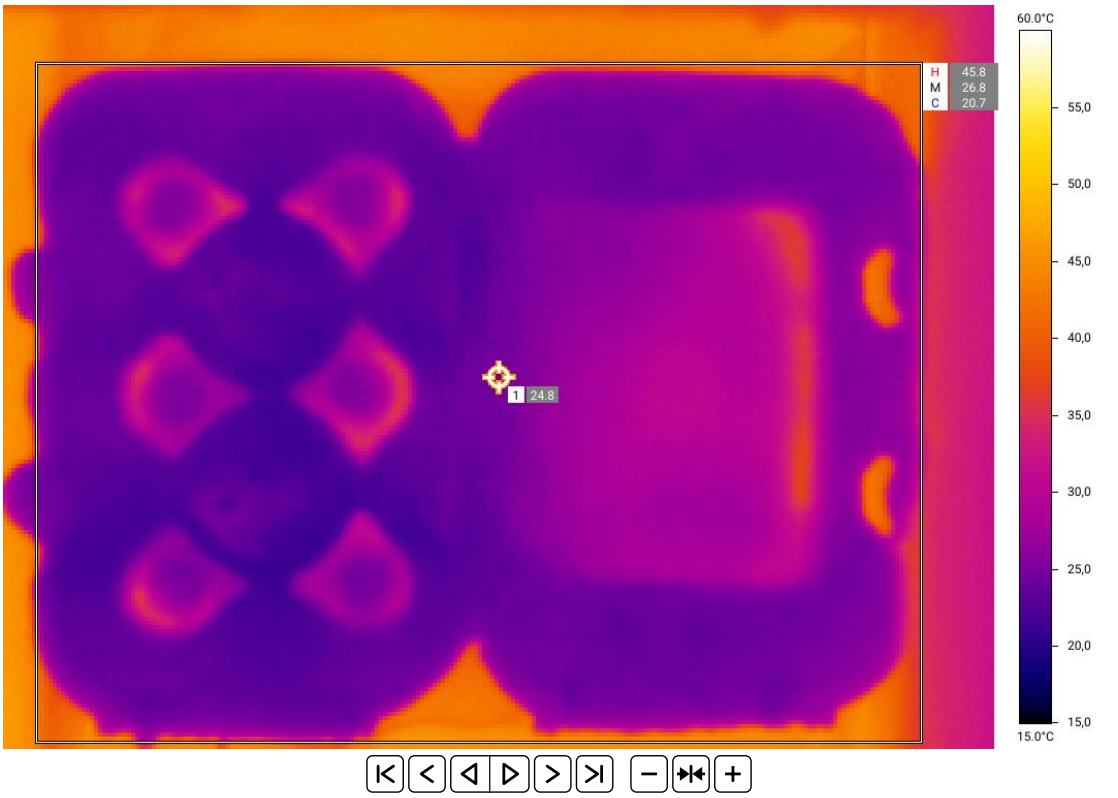


Figure 6.6: Temperature development over time of the egg packaging

The animation of the drying egg packaging clearly shows that the drying rate is not equally distributed. This can be seen by the uneven temperature distribution, which indicates different evaporation rates. As the surface temperature approaches the oven temperature, the area is assumed to be completely dry. Therefore, the animation also highlights the areas of the slowest and fastest drying rates.

6.3 Steady-state drying model results

The steady-state drying model requires the oven inputs from table 6.1 and additional boundary conditions for the egg packaging for the CFD simulation. The boundary conditions required are the temperature at steady-state conditions and the mass fraction of water leading to a relative humidity of 100%.

The temperature at stage two is based on figure 6.5. The steady-state temperature is set to 30°C. This value is at the lower value of the uncertainty of the average value

measured by the FLIR camera. This is to account for the high temperatures outside the egg packaging that influenced the average temperature. The mass fraction is calculated with equation 2.10 and 2.11. Table 6.2 shows all the steady-state drying model inputs.

Table 6.2: The operating condition for the drying oven

	Oven conditions			Boundary conditons	
Input	Temperature	Relative humidity	Velocity	Temperature	Mass fraction
Value	51.8 °C	7.3%	0.6 m/s	30°C	0.024

The evaporation rate is calculated as the difference in the H_2O flow rate at the inlet and outlet. This value is determined by the surface integral of the area-weighted average report in Ansys Fluent. Table 6.3 shows the results.

Table 6.3: Steady state drying model results

Inlet flow rate	Outlet flow rate	Evaporation rate
$3.03 \cdot 10^{-4}$ kg_{H_2O}/s	$3.24 \cdot 10^{-4}$ kg_{H_2O}/s	$2.14 \cdot 10^{-5}$ kg_{H_2O}/s

The results from the CFD simulation show that the evaporation rate exceeds the measured evaporation rate from the experimental results. This discrepancy might be caused by the selected steady-state temperature of 30°C. Figure 6.7 illustrates the results where the wall temperature is set to 25°C and 26°C.

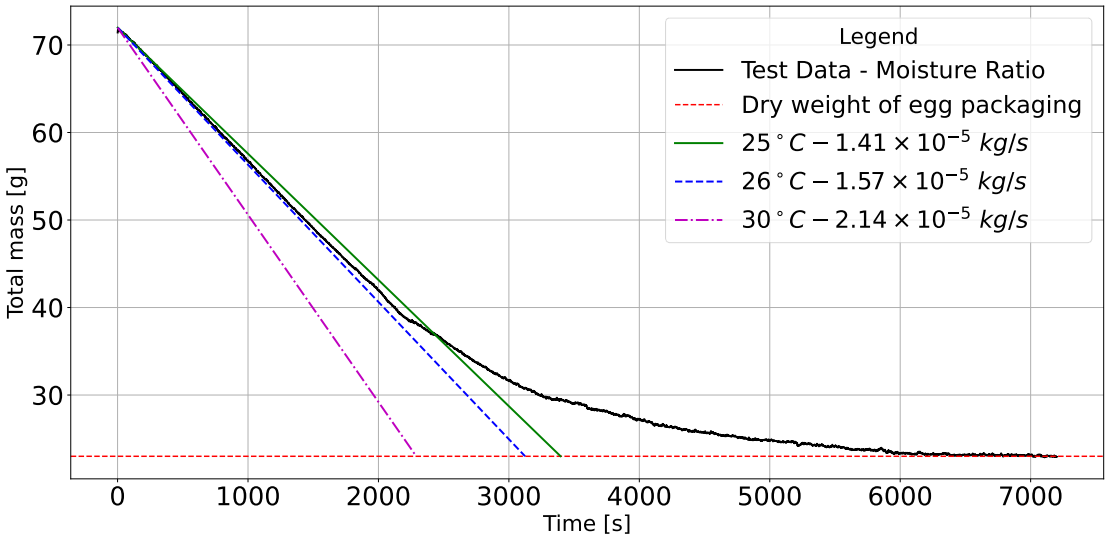


Figure 6.7: Evaporation rates for the steady-state drying model

Figure 6.7 illustrates that the temperature during stage two is most likely between 25°C and 26°C. Within this temperature range, the evaporation rate aligns well with the experimental results.

The flow across the egg packaging is shown with two animations of velocity pathlines in figures 6.8 and 6.9. The first animation provides a view from the front left, while the second animation offers a view from the front right.

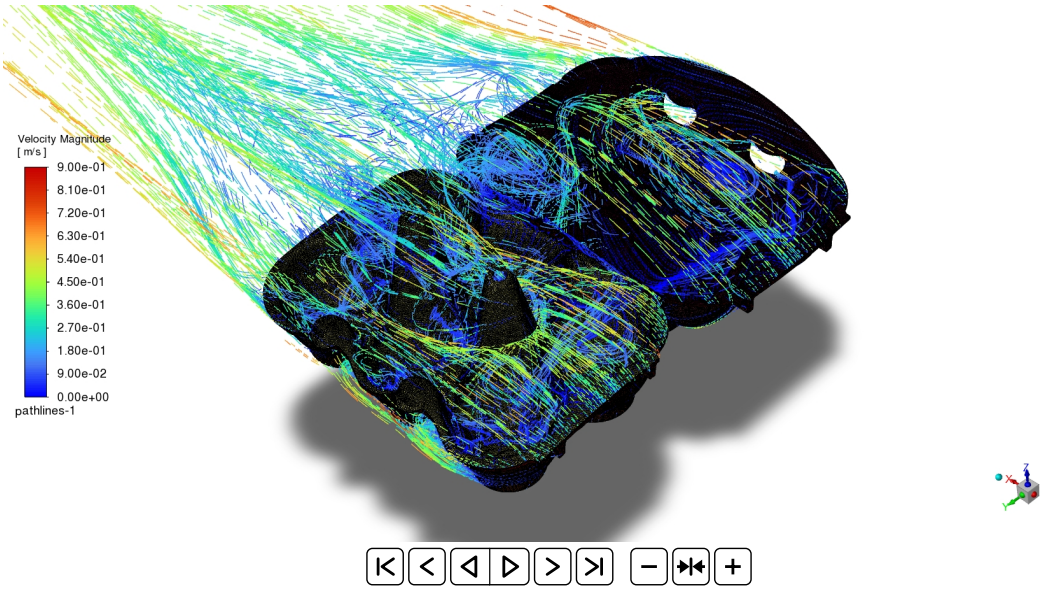


Figure 6.8: Velocity pathlines crossing the egg packaging - front left view.

The animation shows the pathlines in the bottom of the egg packaging recirculating while the pathlines from the top lid move toward the bottom. The velocity range is 0-0.9 m/s, with the lower velocities clearly shown at the bottom of the egg packaging. Pathlines from the bottom of the egg packaging can also be seen rising from the top of the towers, demonstrating a chimney effect.

Figure 6.9 shows the velocity pathlines from the front right view.

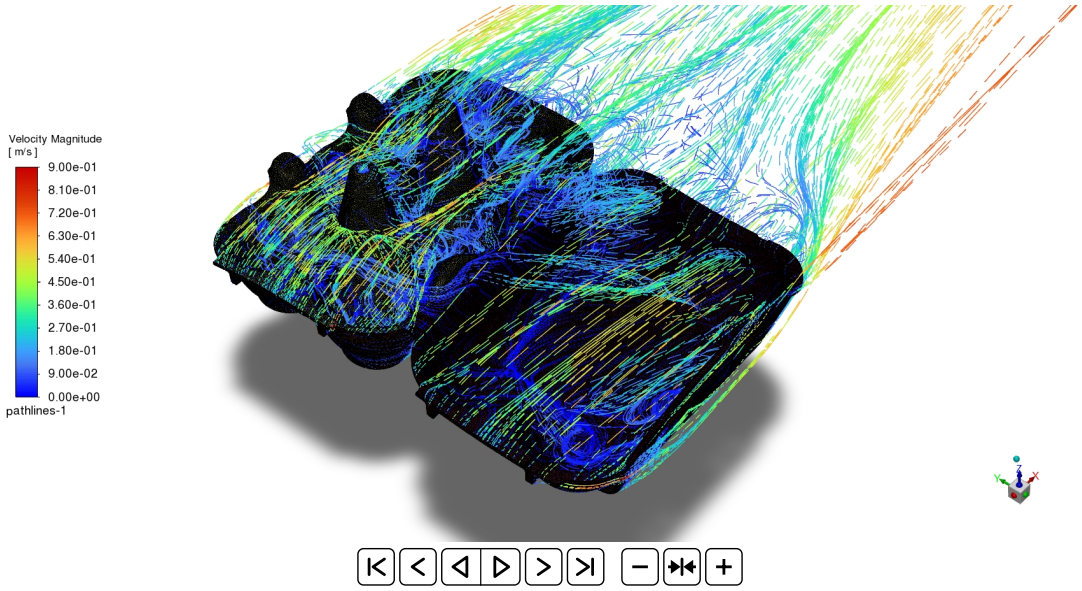
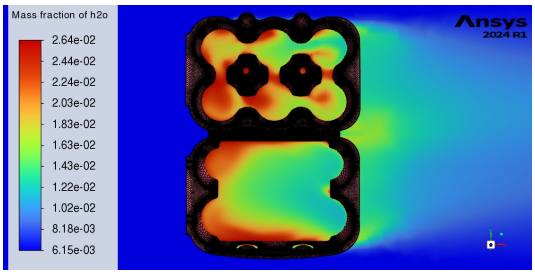


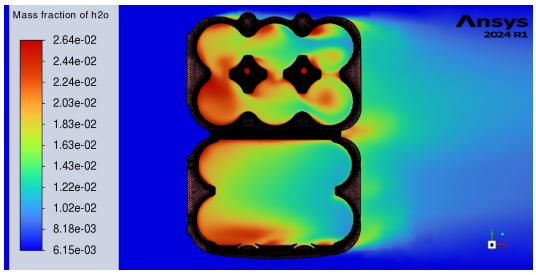
Figure 6.9: Velocity pathlines crossing the egg packaging - front right view.

The animation shows the pathlines hitting the back end of the lid, forcing the flow back into the top lid, where it splits into a recirculation zone and flows toward the bottom of the egg packaging. The first tower at the bottom also influences the flow from the front edge.

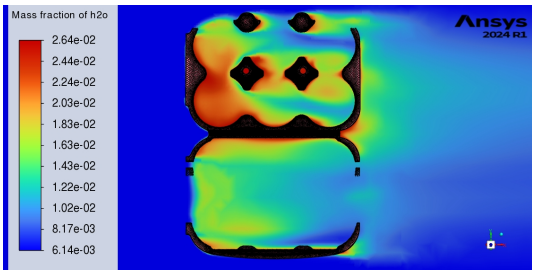
Based on the pathlines from the two animations, four planes 1 cm apart are used to create a contour plot of the mass fraction of water. The first layer is just above the bottom of the top lid. Figure 6.10 shows the four planes of the mass fraction of water.



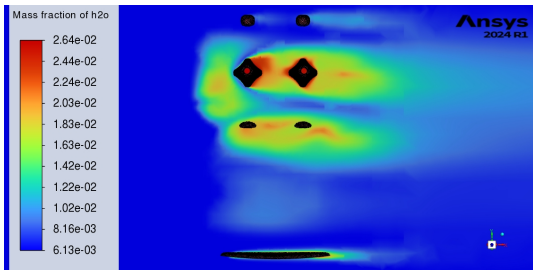
(a) First layer just above the foundation.



(b) Second layer - one cm above foundation.



(c) Third layer - two cm above foundation.



(d) Fourth layer - three cm above foundation.

Figure 6.10: Four plane layers illustrating the mass fraction of water in the air.

The four figures illustrate the locations of high and low water vapor concentrations. The trend from these figures shows that the high water vapor concentration is primarily located at the front of the egg packaging. This also highlights where the lowest evaporation rate is expected.

The locations of low water vapor concentration are similar to the high-velocity pathlines, and conversely, the high concentrations align with the low-velocity pathlines from figures 6.8 and 6.9.

This correlation suggests that areas with high airflow tend to have lower water vapor concentrations due to the enhanced evaporation and moisture removal by the moving air. On the other hand, regions with low airflow, where the pathlines indicate slower movement, tend to accumulate higher concentrations of water vapor, indicating reduced evaporation rates.

The y^+ values is plotted in figure 6.11 and 6.12. In figure 6.11 is the flow direction from the bottom of the figure towards the top, and figure 6.12 is from top towards the bottom.

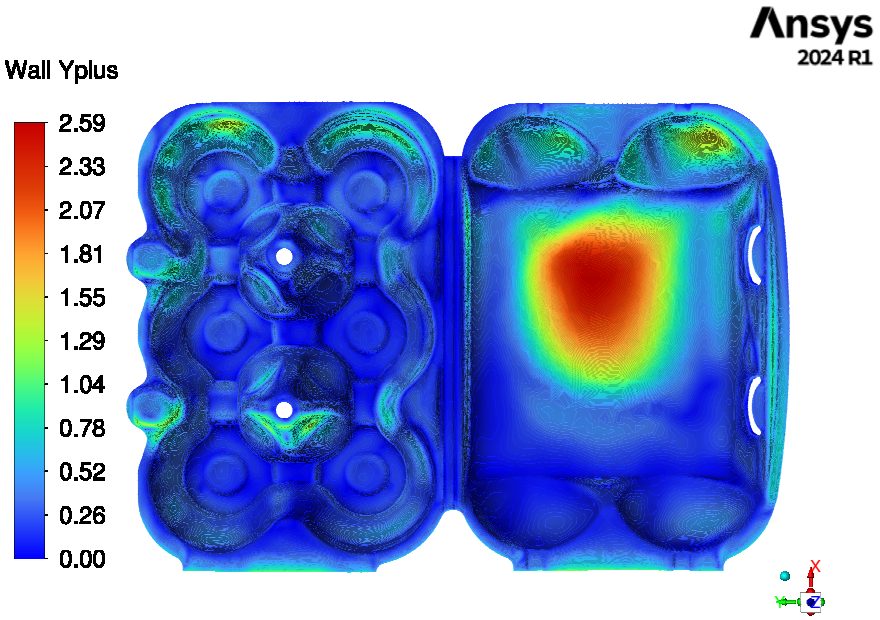


Figure 6.11: Contour plot of y^+ - the top view of the egg packaging.

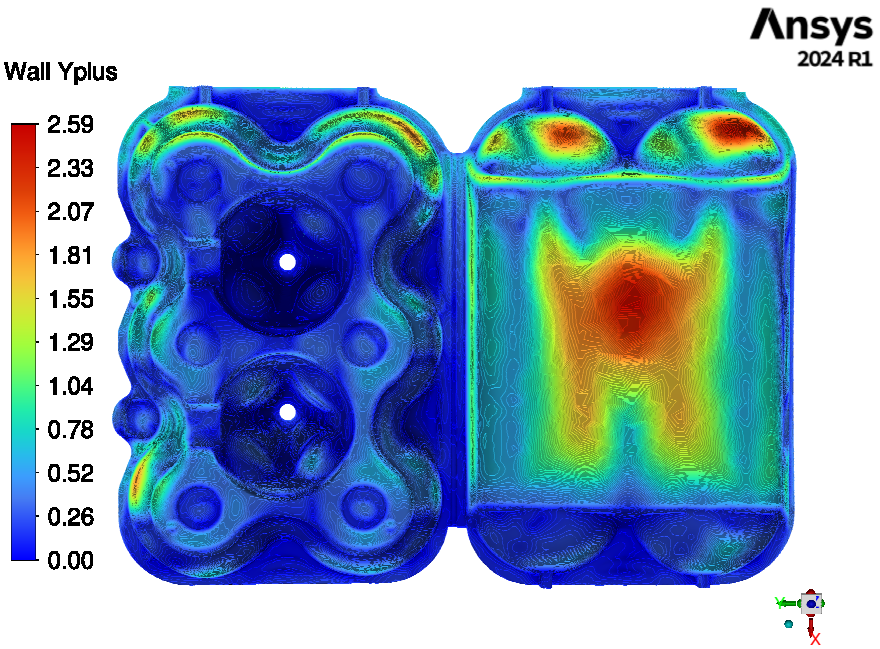


Figure 6.12: Contour plot of y^+ - the bottom view of the egg packaging.

The $y+$ values range between 0 and 2.59, with the highest values observed in the top lid. This aligns well with the velocity pathlines, where the highest velocities where recorded in the top lid, increasing surface shear force. Furthermore, the mesh sizes of the top lid are also larger.

6.4 Transient drying model results

To evaluate the drying process within the paper, a CFD simulation is used to assess the heat and mass transfer coefficients. The boundary conditions employed in the CFD simulation are shown in table 6.4. The wall/egg packaging temperature is set to 25°C, based on the results obtained from the steady-state simulation.

Table 6.4: Transient boundary conditions

	Oven conditions		Boundary conditons	
Input	Temperature	Velocity	Temperature	Emissivity
Value	51.8°C	0.6 m/s	25°C	0.93

The resulting heat and mass transfer coefficients, based on the input parameters from Table 6.4, are shown in table 6.5.

Table 6.5: Heat and mass transfer coefficients for the egg packaging.

Simulation results	h_{eff} (W/(m^2 K))	h_{mass} (m/s)
Value	19.6	0.0159

The heat and mass transfer coefficients from the CFD simulation are used in the Python model. Additionally, the initial temperature of the paper, moisture content, U_{FSP} value, and the fitting parameters are also determined.

The U_{FSP} value is chosen based on the moisture content when the third stage of the drying process starts. Figure 6.4 shows the moisture content from the experimental results, and the U_{FSP} value is set to 0.7.

All the input parameters for the Python model are provided in table 6.6.

Table 6.6: Constant values used in the transient drying model.

Input	T_0	U_0	D_0	β	U_{FSP}	χ	γ
Value	20.7°C	2.12 kg_{H_2O}/kg_{dm}	1e-8	2	0.7	2	2

Using the inputs from tables 6.5 and 6.6, the internal moisture and temperature are evaluated. However, this evaluation is based on the average drying process for the entire egg packaging. Therefore, an additional assumption is made for the transient drying model:

- The thickness is equal throughout the entire geometry of the egg packaging.

With the output parameters from the CFD simulation, initial conditions, and fitting parameters, the moisture ratio and temperature model and experimental results are shown in figure 6.13.

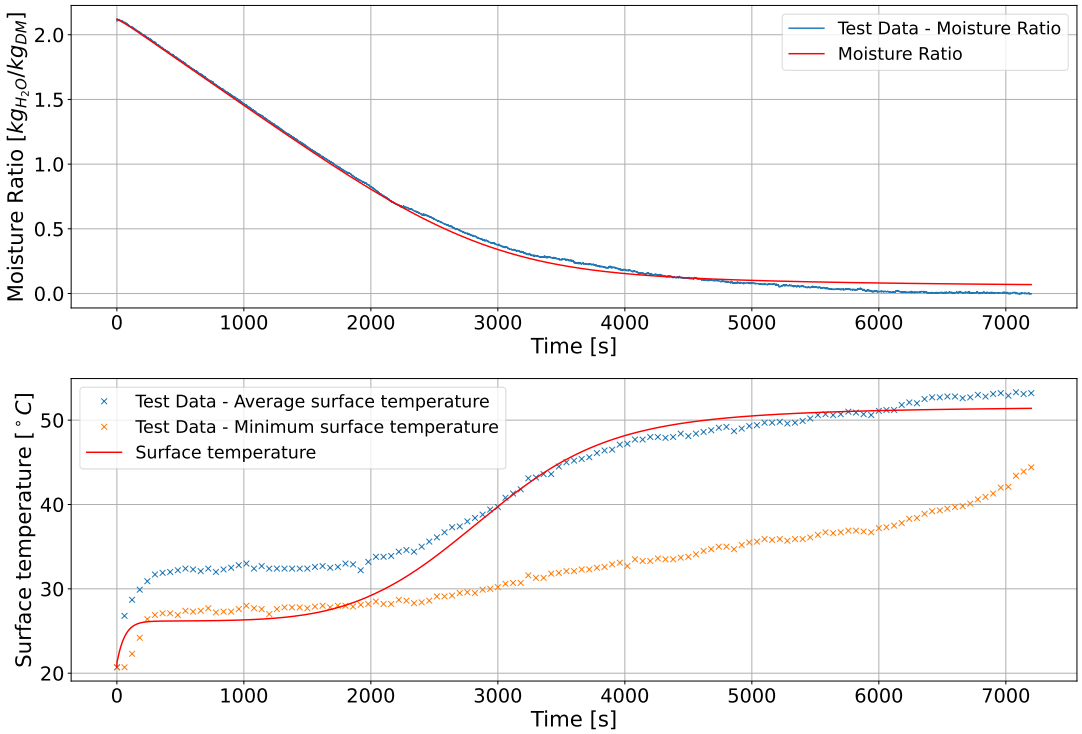


Figure 6.13: Experimental and drying model results for the egg packaging.

Figure 6.13 shows that the drying model aligns well with the moisture ratio in the first and second drying stages. The predicted temperature from the drying model in the second stage is lower than the lowest recorded temperature during the experiment. The temperature in the third stage seems to align better with the recorded temperatures.

Figure 6.14 shows the moisture ratio at the five interior nodes at five different time steps during the drying process to depict the moisture ratio during drying.

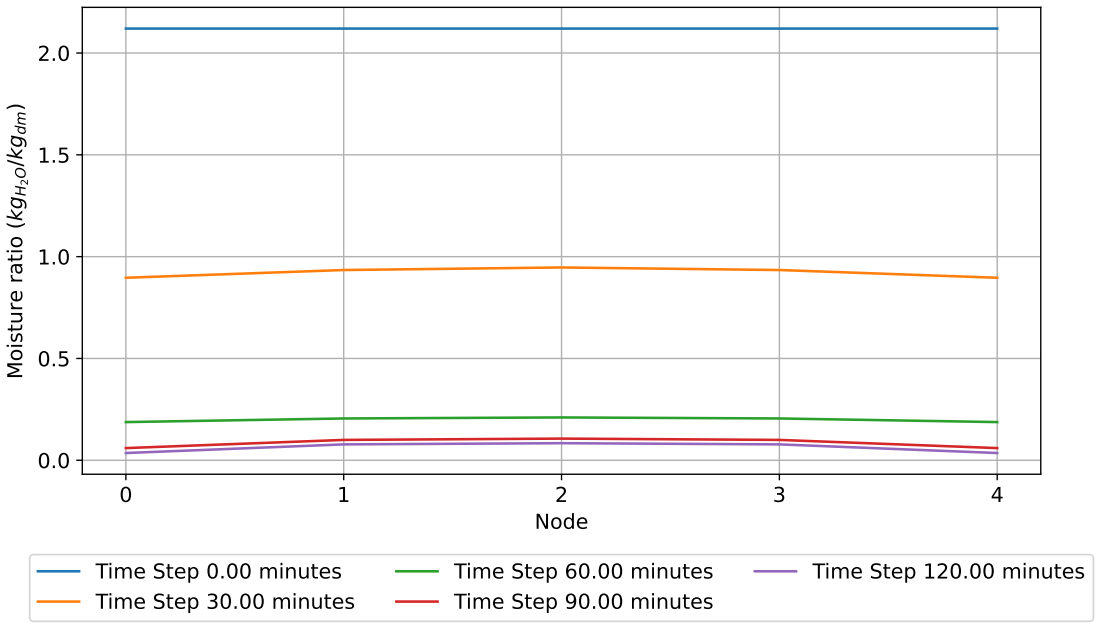


Figure 6.14: Moisture ratio for the interior nodes at five different time steps.

Figure 6.14 shows the moisture ratio for the five discretized nodes from the initial condition until the experimental results were stopped. The time step between the lines is equally disturbed during the entire drying time. This highlights that most water evaporates during the first and second stages as the distance between the lines gets smaller and smaller.

Figure 6.15 shows the evaporation rate for the boundary conditions.

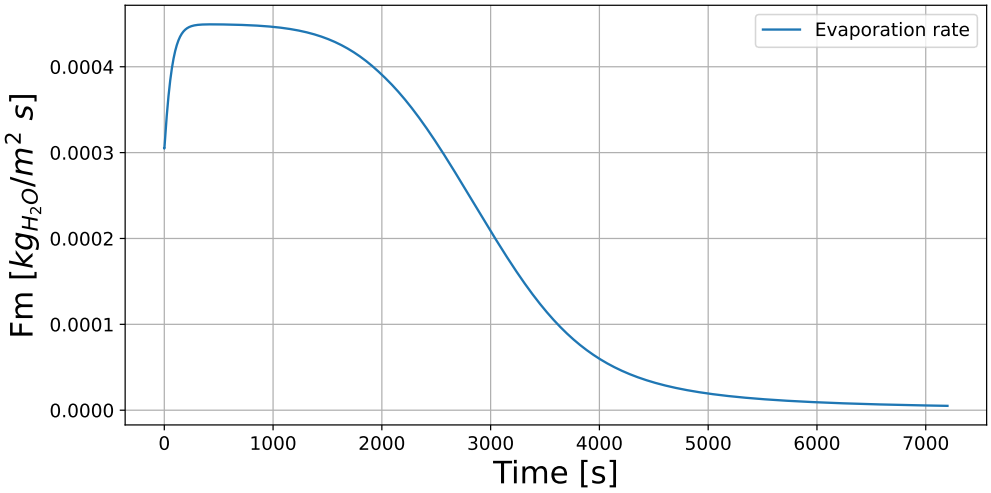


Figure 6.15: Boundary evaporation rate of the egg packaging

Figure 6.15 illustrates the three stages of the drying process:

1. First Stage (0-200 seconds): Characterized by increasing evaporation.
2. Second Stage (200-2000 seconds): Exhibits approximately constant evaporation.
3. Third Stage (2000-7200 seconds): Marked by decreasing evaporation.

Figure 6.16 shows the energy transfer of the convection, radiation, and evaporation over the drying time.

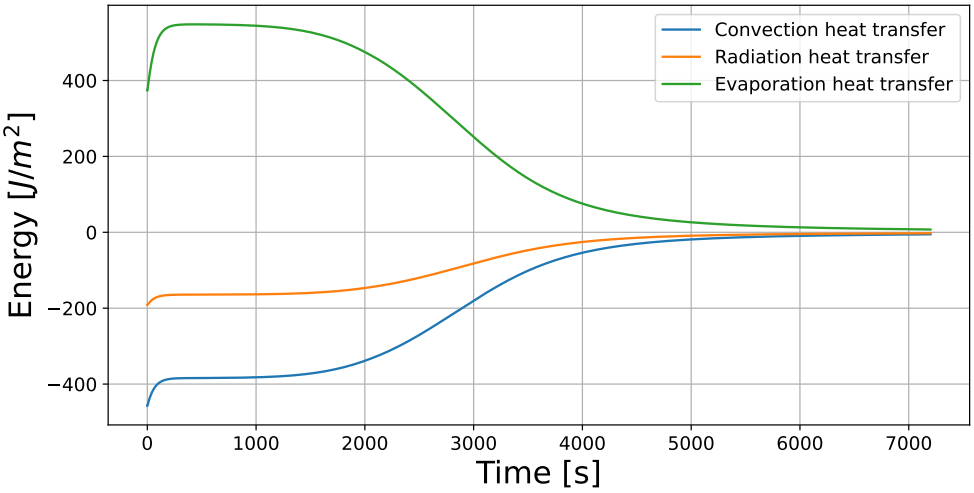


Figure 6.16: Convection, Radiation, and evaporation energy transfer during drying of the egg-packaing.

Figure 6.16 shows the magnitude of the heat transfer at the boundary conditions. The results indicate that radiation heat transfer accounts for 30% of heat transfer towards paper; this shows the importance of including radiation heat transfer.

Figure 6.17 shows the energy distribution between evaporation and sorption energy used during drying.

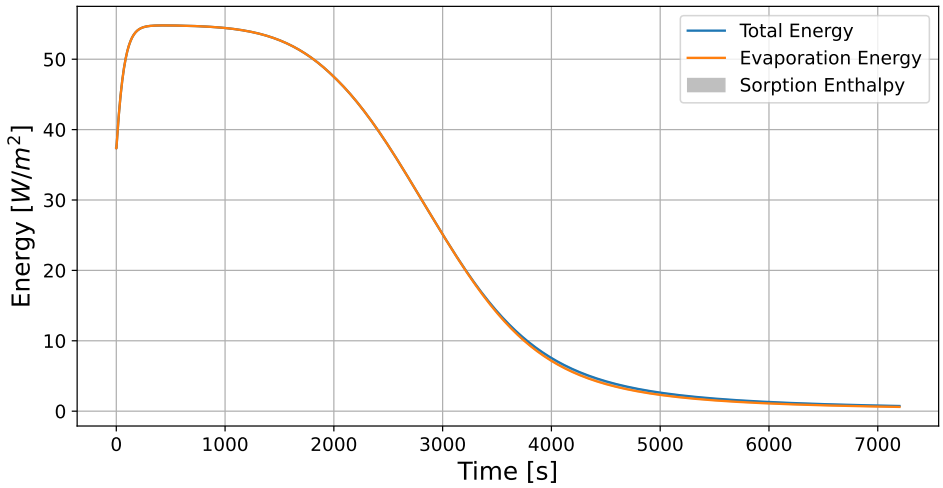


Figure 6.17: Evaporation and Soprtion energy during the drying of the egg packaging

The results show that the primary energy required is to evaporate the water. Total Energy over the entire drying time is 1654 kW/m^2, the sorption energy account for: 0.7% and evaporation er 99.3%

7 Discussion

The discussion evaluates the experimental setup and analyzes the drying model results compared to the experimental data. It also discusses the Ansys Fluent results, focusing on the challenges encountered and potential improvements for the general drying process. Lastly, the discussion addresses the problem formulation for the project.

7.1 Experimental setup

Using a FLIR camera to evaluate the surface temperature of drying egg packaging has proven to be challenging. The animation of the drying egg packaging figure 6.6 shows that the maximum temperature reaches approximately 55°C at the end of the drying process. However, the oven temperature has been recorded at 51°C. The uncertainties listed in table 4.1 for the FLIR camera and humidity probe do not account for this discrepancy. The deviation might be caused by using a plastic layer between the FLIR camera and the drying egg packaging, as this affects the emissivity and reflection properties of the plastic, potentially influencing the temperature readings.

A possible improvement for future measurements is to use a Germanium window designed to operate in the long-wave infrared portion of the spectrum (7.5-13.5 microns), resulting in less performance loss [44]. Another alternative could be to create an opening in the drying oven and include this feature in the CFD simulation. This would allow direct measurement without interference from the plastic layer.

The retrofit of the drying oven, where the flow straightener and flow conditioner have been moved forward and closer to the drying egg packaging, has proven effective. This modification ensures a fully developed flow with a more uniform velocity field. Additionally, this improved uniformity in the flow can lead to more consistent drying results and better overall efficiency of the drying process.

Further studies should consider the impact of the plastic layer and explore additional modifications to enhance measurement accuracy. Implementing these changes could significantly improve the reliability of temperature assessments and overall drying performance.

7.2 Steady-state drying model

The steady-state drying model, computed exclusively in Ansys Fluent, has demonstrated that Ansys Fluent can effectively capture the mass transfer of species transport. The results from the steady-state simulation have been evaluated solely based on the average evaporation by comparing the moisture content at the inlet and outlet.

7.2.1 Two-dimensional simulation results

The two-dimensional simulation results have shown a deviation by a factor of two compared to experimental results, as presented in table 5.2. Figure 5.8 displays the mass fraction distribution from the three-dimensional simulation, and the side effects on the drying paper appear insignificant in explaining the large deviation. Consequently, the cause of the deviation remains unverified.

The turbulence models employed in the two-dimensional simulation indicate that the k-ε model outperforms the k-ω and k-ω SST models in terms of accuracy. The y^+ values, illustrated in figure 5.6, are sufficiently low to ensure the applicability of the k-ω and k-ω SST models for the simulations.

7.2.2 Three-dimensional simulation results

The three-dimensional simulation results from table 5.3 demonstrate a more accurate prediction of constant evaporation. The deviations between the four turbulence models indicate that the general k-ω turbulence model has higher accuracy, with the k-ω SST model providing the most precise results. The higher accuracy of the k-ω SST turbulence model is consistent with the findings of other studies [16] [17] [18]. The differences between the k-ω SST and the transition 4-equation model are negligible, making the k-ω SST model preferable for the rest of the simulations due to its reduced computational requirements.

The flow animation of the three-dimensional simulation, shown in figure 5.9, reveals a small recirculation zone at the top front. This recirculation zone is also indicated by the y^+ values in figure 5.10, where the green dots represent the top layer y^+ values. The highest y^+ value, 0.861, is located on the wall-west/front of the drying paper, confirming the applicability of the k-ω SST turbulence model.

7.3 Drying of egg packaging

The steady-state and transient drying models have been utilized to evaluate the drying dynamics of egg packaging. The steady-state drying model was conducted to rapidly and accurately assess the drying rate. The results depicted in figure 6.7 demonstrate that the CFD simulation effectively evaluates the evaporation rate under a fully saturated surface assumption. However, the simulation appears to be sensitive to surface temperature—a deviation of 5°C in temperature results in a 65% difference in the evaporation rate.

The two flow animations from figure 6.8 and 6.9 and the mass fraction concentration from figure 6.10, highlight the areas where the lowest evaporation rate is expected. The high water vapor concentration in the ambient air would decrease the concentration difference and result in a lower evaporation rate.

By comparing Figure 6.10a and the experimental results shown in Animation 6.6, it is evident that the simulation and experimental results align well in visualizing the varying drying rates. This alignment is particularly clear in the top lid of the egg packaging. The simulation of the velocity pathlines and mass concentration accurately predicted that the velocity and concentration gradient are highest in the top right corner of the lid. This prediction was verified by the experimental results, where the top right corner dried out the fastest.

The transient drying model could simulate the drying process over time. By incorporating dynamic variables, the model provided predictions closely aligned with the observed experimental data as shown in figure 6.13. The model showed the best accuracy in the first and second drying stages, whereas in the last part of the drying process, the model started to deviate from the experimental results.

Despite its utility, the transient drying model is constrained by its reliance on average process parameters. This limitation means that the model primarily captures the

general trends of the drying process rather than the nuanced variations as seen in animations of the drying process 6.6. Consequently, the model might overlook localized deviations and fluctuations in drying rates, leading to a generalized representation that may not fully encompass the complexity of the drying behavior under all conditions.

For the transient drying model to accurately predict drying kinetics, it necessitates the inclusion of several fitting parameters, as shown in table 6.6. These parameters, derived from empirical data, require validation to ensure their reliability and applicability. Without proper validation, the model's predictive capabilities could be compromised, potentially leading to erroneous conclusions.

The dependency on validated fitting parameters inherently limits the model's generalizability. As a result, the model's ability to accurately predict drying behavior under varying environmental factors, such as temperature and humidity, is reduced. This constraint underscores the necessity for developing more adaptive and versatile modeling approaches that can accommodate a broader range of drying scenarios without extensive re-validation.

7.4 Simulating drying behavior in Ansys Fluent

The CFD simulation was conducted using Ansys Fluent. The literature review revealed limited direct comparisons for drying fibers using CFD. Therefore, this study focused on how CFD can predict drying behavior in complex geometries.

Simulations were conducted using a UDF that maintained the egg packaging walls at 100% saturation regardless of temperature. The expected outcome of a transient simulation with an initial patch temperature of 10°C was that the egg packaging temperature would stabilize below the ambient temperature.

Contrary to expectations, the egg packaging temperature in the simulation rapidly increased to match the ambient temperature, indicating a deviation from the anticipated species transport accounting for the energy/enthalpy difference due to evaporation between the 100% saturated surface and ambient relative humidity.

A UDF based on the theoretical mass transfer provided by references [16] and [19] was tested to localize the local mass transfer, which would account for the evaporation energy. However, successful implementation in Ansys Fluent has not yet been achieved.

Due to the challenges with surface mass transfer, the Chilton-Colburn analogy was used to evaluate the mass transfer coefficient in Ansys Fluent. The analogy is described as follows:

The Chilton-Colburn analogy has been observed to hold quite well in laminar or turbulent flow over planes. However, this is not always the case for internal flow and flows over irregular geometries, and in such cases, specific relations should be used [35].

The validation and egg packaging simulation results indicated that this analogy holds well under the given conditions of low temperatures and velocities.

7.5 Problem formulation

The problem formulation for the project was: *How can CFD be effectively applied to predict the drying process of paper with complex geometries?* The results obtained from the CFD simulations demonstrated that the transport mechanism in Ansys Fluent is well-suited for handling species transport in complex geometries such as egg packaging.

Combining CFD to evaluate heat transfer at the surfaces of complex geometries with an external program (Python) to assess the average drying process has yielded promising results. Others have employed a similar approach with notable success [27].

The results have been obtained from a manual handle of the heat and mass transfer coefficient between Ansys Fluent and Python. Two-way coupling between Ansys Fluent and Python is recommended to enhance future calculations.

8 Conclusion

The steady-state and transient drying model results have been validated against experimental data. The results underscore the importance of accurate temperature measurements. Figure 6.7 presents the steady-state model results. The measured average temperature of 30°C used to determine evaporation was found to differ from the experimental results, where an average temperature between 25-26°C aligned well with the experimental data.

The results depicted in figure 6.13 demonstrate that the transient drying model is capable of predicting the first and second stages of the drying process with high accuracy, with only a small deviation observed in the final drying stage.

The simulation results in Ansys Fluent demonstrated effective management of species transport mechanisms. However, achieving an energy balance that aligns with experimental or expected results remains a critical challenge for the practical application of CFD in predicting drying behavior.

Bibliography

[1] M. G. Vieira, L. Estrella, and S. C. Rocha, "Energy efficiency and drying kinetics of recycled paper pulp", *Drying Technology*, vol. 25, no. 10, pp. 1639–1648, Oct. 2007.

[2] M. G. Vieira and S. C. Rocha, "Drying conditions influence on physical properties of recycled paper", *Chemical Engineering and Processing: Process Intensification*, vol. 46, no. 10, pp. 955–963, Oct. 2007.

[3] "European Declaration on Paper Recycling 2021-2030", Tech. Rep.

[4] Baggerud, Erik, "Modelling of Mass and Heat Transport in Paper - Evaluation of Mechanisms and Shrinkage", eng, Ph.D. dissertation, Lund University, 2004.

[5] J. R. Kristensen, "Numerical modelling of moulded egg packaging drying process Project-oriented study in an external organisation 9th Semester Project", Tech. Rep., 2024. [Online]. Available: `http://www.aau.dk`.

[6] M. G. A. Vieira and S. C. S. Rocha, "Mathematical modeling of handmade recycled paper drying kinetics and sorption isotherms", *Brazilian Journal of Chemical Engineering*, vol. 25, no. 2, 299–312, [Online]. Available: `https://doi.org/10.1590/S0104-66322008000200009`.

[7] A. S. Mujumdar, *Handbook of Industrial Drying*. Boca Raton, FL, CRC Press, 2020, ch. Drying of Pulp and Paper.

[8] S. Stenström, "Drying of paper: A review 2000–2018", *Drying Technology*, vol. 38, pp. 825–845, 7 May 2020.

[9] A. Ahsan, *Evaporation, Condensation and Heat transfer*, A. Ahsan, Ed. IntechOpen, 2011.

[10] B. V. Ramarao, A Massoquete, S Lavrykov, and S Ramaswamy, *Drying Technology An International Journal Moisture Diffusion Inside Paper Materials in the Hygroscopic Range and Characteristics of Diffusivity Parameters*. 2003. [Online]. Available: `https://www.tandfonline.com/action/journalInformation?journalCode=ldrt20`.

[11] M Schneeberger, P Leuk, U Hirn, and W Bauer, "The heat of sorption in paper drying - an investigation of measurement methods and influence of pulp parameters", *Trans. of the XVth Fund. Res. Symp. Cambridge*, pp. 469–492, 2013.

[12] P. Chen and D. C. T. Pei, "A mathematical model of drying processes", *J. Heat fm Transfeer*, vol. 32, pp. 297–310, 2 1989.

[13] S. Stenström and E. Baggerud, "Diffusive vapor and liquid transport during convective drying of industrial pulp sheets", *Proceedings of the Progress in Paper Physics Seminar*, pp. 141–144, 2002.

[14] M. G. Vieira, L. Estrella, M. A. Silva, and S. C. Rocha, "Shrinkage of recycled paper sheet during drying", *Drying Technology*, vol. 24, no. 4, pp. 465–474, 2006.

[15] E. Baggerud and S. Stenström, "Further insight in the web consolidation process - part ii. triangular phase diagrams for analysis of paper shrinkage", English, *Nordic Pulp Paper Research Journal*, vol. 20, no. 1, pp. 54–57, 2005.

[16] T. Wen, L. Lu, W. He, and Y. Min, "Fundamentals and applications of CFD technology on analyzing falling film heat and mass exchangers: A comprehensive review", 2020. [Online]. Available: `https://doi.org/10.1016/j.apenergy.2019.114473`.

[17] R. P. Ramachandran, M. Akbarzadeh, J. Paliwal, and S. Cenkowski, *Computational Fluid Dynamics in Drying Process Modelling—a Technical Review*, Feb. 2018.

[18] T. T. Nadew, P. Demissie Tegenaw, and T. S. Tedila, "Mathematical-Based CFD Modelling and Simulation of Mushroom Drying in Tray Dryer", 2023. [Online]. Available: `https://doi.org/10.1155/2023/6128517`.

[19] W Ambrosini, N Forgione, D Mazzini, and F Oriolo, "Computational Study of Evaporative Film Cooling in a Vertical Rectangular Channel", 2010. [Online]. Available: `https://www.tandfonline.com/action/journalInformation?journalCode=uhte20`.

[20] X. Wang, H. Chang, M. Corradini, T. Cong, and J. Wang, "Prediction of falling film evaporation on the AP1000 passive containment cooling system using ANSYS FLUENT code", 2016. [Online]. Available: `http://dx.doi.org/10.1016/j.anucene.2016.05.014`.

[21] Y.-T. Lee, S. Hong, C. Dang, L.-H. Chien, L.-W. Chang, and A.-S. Yang, "Heat transfer characteristics of obliquely dispensed evaporating falling films on an elliptic tube", 2018. [Online]. Available: `https://doi.org/10.1016/j.ijheatmasstransfer.2018.12.031`.

[22] Y.-T. Lee, S. Hong, C. Dang, L.-H. Chien, and A.-S. Yang, "Effect of counter current airflow on film dispersion and heat transfer of evaporative falling film over a horizontal elliptical tube", 2019. [Online]. Available: https://doi.org/10.1016/j.ijheatmasstransfer.2019.07.029.

[23] Y. Wang, L. Zhou, X. Kang, Q. Huang, X. Shi, and C. Wang, "Experimental and numerical optimization of direct-contact liquid film cooling in high concentration photovoltaic system", 2017. [Online]. Available: https://doi.org/10.1016/j.enconman.2017.11.014.

[24] Y. Wang, J. Huo, L. Zhou, and Q. Huang, "Comparative study of high concentrating photovoltaics integrated with phase-change liquid film cooling system", *International Journal of Energy Research*, vol. 43, no. 6, pp. 2108–2122, May 2019. [Online]. Available: https://onlinelibrary-wiley-com.zorac.aub.aau.dk/doi/full/10.1002/er.4414.

[25] N. Chhanwal, A. Tank, K. S. Raghavarao, and C. Anandharamakrishnan, *Computational Fluid Dynamics (CFD) Modeling for Bread Baking Process-A Review*, May 2012.

[26] N. Malekjani and S. M. Jafari, *Simulation of food drying processes by Computational Fluid Dynamics (CFD); recent advances and approaches*, Aug. 2018.

[27] T. Norton and D. W. Sun, "CFD: An Innovative and Effective Design Tool for the Food Industry", *Food Engineering Series*, pp. 45–68, 2011. [Online]. Available: https://link-springer-com.zorac.aub.aau.dk/chapter/10.1007/978-1-4419-7475-4_3.

[28] A. Fluent, *ANSYS FLUENT 12.0 Theory Guide - 1.2 Continuity and Momentum Equations*, Accessed 06/03/2024. [Online]. Available: https://www.afs.enea.it/project/neptunius/docs/fluent/html/th/node11.htm.

[29] A. Fluent, *ANSYS FLUENT 12.0 Theory Guide - 5.2.1 Heat Transfer Theory*, Accessed 06/03/2024. [Online]. Available: https://www.afs.enea.it/project/neptunius/docs/fluent/html/th/node107.htm.

[30] A. Fluent, *ANSYS FLUENT 12.0 Theory Guide - 7.1.1 Species Transport Equations*, Accessed 06/03/2024. [Online]. Available: https://www.afs.enea.it/project/neptunius/docs/fluent/html/th/node128.htm.

[31] A. Fluent, "Ansys Fluent Theory Guide", Tech. Rep., 2024. [Online]. Available: http://www.ansys.com.

[32] W. Wagner, J. R. Cooper, A. Dittmann, *et al.*, "The iapws industrial formulation 1997 for the thermodynamic properties of water and steam", *Journal of Engineering for Gas Turbines and Power*, vol. 122, pp. 150–180, 1 Jan. 2000.

[33] J. Monteith and M. Unsworth, *Principles of Environmental Physics*, eng, 3rd ed. San Diego: Elsevier Science & Technology, 2007.

[34] A. Fluent, *ANSYS FLUENT 12.0 Theory Guide - 5.3.7 Surface-to-Surface (S2S) Radiation Model Theory*, Accessed 03/05/2024. [Online]. Available: https://www.afs.enea.it/project/neptunius/docs/fluent/html/th/node116.htm.

[35] Y. A. Cengel, A. J. Ghajar, and M. Kanoglu, *Heat and mass transfer : fundamentals applications* /, 5. ed. in SI units., Y. A. Cengel, A. J. A. J. Ghajar, and M. Kanoglu, Eds. McGraw-Hill Education, 2015, pp. 841–880.

[36] L Nilsson and S Stenstrom, "A study of the permeability of pulp and paper", *Pergamon Int. J. Multiphase Flow*, vol. 23, pp. 131–153, 1 1997.

[37] Ahlborn, *Operating instructions Digital ALMEMO ® D6 sensor*.

[38] Ahlborn, *Operating instructions*, Accessed 04/04/2024. [Online]. Available: www.ahlborn.com.

[39] KERN, "Kern pcb 3000-2-2023", Accessed 27/10/2023. [Online]. Available: www.kern-sohn.com.

[40] Testo, "testo 865 thermal imager", Accessed 04/04/2024. [Online]. Available: https://www.testo.com/en-TH/testo-865/p/0560-8650.

[41] R Danjoux,

[42] Kayteck, *Mini digital vane anemometer with snap-on head - kayteck precision flow sensors*, Accessed 27/10/2023. [Online]. Available: http://www.kayteck.ca/product/mini-digital-vane-anemometer/.

[43] Rotronic, "Hygroclip2 advanced humidity temperature probes", 2016, Accessed 26/10/2023. [Online]. Available: https://www.rotronic.com.

[44] FLIR, *FLIR Cameras - Placing a Protective Window On Any FLIR Camera*, Accessed 22/05/2024. [Online]. Available: https://flir.custhelp.com/app/answers/detail/a_id/3142/~/flir-cameras---placing-a-protective-window-on-any-flir-camera.

A Appendix: Python - drying model

```
import time
import datetime
import math
import matplotlib.pyplot as plt
import numpy as np
from iapws import IAPWS97
import Rosenkilde
import pandas as pd

def DL(U, U_FSP, D_0, beta): #Valid over plat plate
#Liquid diffusion, based on the moisture ratio U and the defined U_FSP
    if U > 0.0:
        if U <= U_FSP:
            DL = D_0 * (1 - np.exp(-174 * U**3.7))
        elif U > U_FSP:
            DL = D_0 * (1 - np.exp(-174 * U_FSP**3.7)) * np.exp(beta * (U -
            ↪  U_FSP))
    else:
        DL = 0.0
    return DL

def phi_f(U):
    return  (xx + U * xx) #Bulk function from bulk-function python file,
    ↪  [m^3/kg_dm]

def k_g_eff(rho_s):
    return 5.15*10**5 * math.exp(-0.0136*rho_s)/(10**(15))

def diffusionCoefficient(T_film,P):
    #Diffuision function, where film temperature and total pressure has to be
    ↪  specifed
```

```
29      return 1.87 * 10 ** (-10) * T_film **(2.072) / (P/101325) #Valid in the
        ↪  range off 280K < T <450 K, P/101325 due to P is in atm, [m^2/s]
30
31  def psi_f(U,T):
32      return 1-math.exp(-(47.58*U**(1.887) + 0.10085*(T-273)*U**(1.0585))) #Water
        ↪  activity page 123 eq. 3-108
33
34  #Solver
35  if 1 == 1:
36
37      #Pressure solver = 0 : constant atmosperic pressure for interior nodes.
        ↪  Quick solver since timesteps can be larger wihtout convergence
        ↪  problems.
38      #Pressure solver = 1 : variable gas pressure for interior nodes. Slow
        ↪  solver since timesteps needs to be smaller for proper convergence.
39      pressuresolver = 0
40
41  #Inputs
42  if 1 == 1:
43      #Ambient conditions
44      P_atm = xx           #Ambient pressure, [Pa]
45      T_air = xx      #Ambient temperature, [K]
46      omega_ambient = xx  #Absolute humidity of the ambient air, [kg_dm/kg_air]
47
48
49      #Product initial conditions
50      T_0 = xx       #Initial temperature of product, [K]
51      U_0 = xx       #Initials moisture ratio of product, [kg H20 / kg DM]
52
53      #Product parameters
54      Lmm = xx                 #Total thickness of product in bone dry state!,
        ↪  [mm]
55      phi_f0= xx         #Pulp bulk parameter, [m3/kg DM]
56      rho_ss = xx            #Density of pure pulp, [kg/m^3]
57      cp_s = xx             #Specific heat of pulp, [J/(kg*K)] Page 118, table
        ↪  3-9
58
59      #Liquid parameters
60      rho_sl = IAPWS97(T=293.15, x=0).rho #Density of water, [kg/m^3]
61
62      #Gas parameters
63      R=8.314                 #Gas constant, [J/(mol K)]
64      M_H2O = 0.01801528      #Molar weight of water [kg/mol]
```

```
65      M_air = 0.02897          #Molar weight of air [kg/mol]
66
67      #Fitting parameters
68      U_FSP = xx                  #Moisture ratio at FPS, [kg H2O / kg DM] Based
        ↪  on figure 3-11 P. 112
69      beta = xx                     #Based on figure 3-11 P. 112
70      D_0 = xx                    #Based on figure 3-11 P. 112
71      sigma_rad = 5.6696*10**(-8) #Stefan-Boltzmann constant, [W/m^2K^4]
72      e = 0.93                 #Emissivity, []
73
74      #Time
75      simulation_time = xx    #simulated time, [s]
76
77      #Fitting function variables
78      x = xx
79      y = xx
80
81  #Discretization
82  if 1 == 1:
83
84      if pressuresolver == 0:
85          nodes = 5                                    #Number of nodes (choose odd
            ↪  number)
86          dt = 1E-1                                   #Time step, [s]
87          num_time_steps = int(simulation_time/dt) + 1    #Number of time steps
88      elif pressuresolver == 1:
89          nodes = 5                                    #Number of nodes (choose
            ↪  odd number)
90          dt = 1E-6                                    #Time step, [s] (1E-5 for 3
            ↪  nodes)
91          num_time_steps = int(simulation_time/dt) + 1    #Number of time steps
92      else:
93          print("Choose pressure solver!")
94
95      #Discretization calculations
96      N = nodes-1              #Number for last node in matrice (0 ... N)
97      L = Lmm / 1000           #Total thickness of product, [m]
98      dz = L/(nodes-1)         #Distance between nodes in product, [m]
99      dB = dz/phi_f0           #Change in solid-base coordinate (Constant and is
        ↪  calculated based on bone-dry pulp), [kg_DM / m2], Equation B-11 Page
        ↪  B-3 (Appendix B)
100
101 #Initialize matrice for all parameters:
```

```
102 if 1 == 1 :
103     T=np.zeros((num_time_steps,nodes))                #Temperature, [K]
104 
105     U=np.zeros((num_time_steps,nodes))                #Moisture ratio,
        ↪  [kg_H2O/kg_DM]
106 
107     X_a = np.zeros((num_time_steps,nodes))            #Solid-based concentration
        ↪  of air, [kg/kg_dm]
108     X_L = np.zeros((num_time_steps,nodes))            #Solid-based concentration
        ↪  of liquid (Free liquid + bound liquid), [kg/kg_dm]
109     X_v = np.zeros([num_time_steps,nodes])            #Solid-based concentration
        ↪  of vapour, [kg/kg_dm]
110 
111     P_vg = np.zeros((num_time_steps,nodes))           #Saturation pressure, [Pa]
112     P_gg = np.zeros((num_time_steps,nodes))           #Total pressure, [Pa]
113     P_ag = np.zeros((num_time_steps,nodes))           #Pressure of air, [Pa]
114 
115     mu_v=np.zeros((num_time_steps,nodes))            #Dynamic viscosity of water
        ↪  vapor, [Pa*s]
116     mu_a=np.zeros((num_time_steps,nodes))            #Dynamic viscosity of air,
        ↪  [Pa*s]
117 
118     Fm = np.zeros((num_time_steps,nodes))
119 
120     J_as_b = np.zeros((num_time_steps,nodes))         #Boundary Total air flux,
        ↪  [kg/(m^2*s)]
121     J_vs_b = np.zeros((num_time_steps,nodes))         #Boundary Total air flux,
        ↪  [kg/(m^2*s)]
122     J_Ls_b = np.zeros((num_time_steps,nodes))         #Boundary Total air flux,
        ↪  [kg/(m^2*s)]
123 
124     J_as_i = np.zeros((num_time_steps,nodes))         #Interior Total air flux,
        ↪  [kg/(m^2*s)]
125     J_vs_i = np.zeros((num_time_steps,nodes))         #Interior Total air flux,
        ↪  [kg/(m^2*s)]
126     J_Ls_i = np.zeros((num_time_steps,nodes))         #Interior Total air flux,
        ↪  [kg/(m^2*s)]
127 
128 
129     dJ_lsdB = np.zeros((num_time_steps,nodes))        #Liquid flux, [kg/(m^2*s)]
130 
131     dJ_vs1dB = np.zeros((num_time_steps,nodes))       #Vapor flux (Pressure),
        ↪  [kg/(m^2*s)]
```

```
132     dJ_vs2dB = np.zeros((num_time_steps,nodes))       #Vapor flux (Mass
        ↪ fractiopn), [kg/(m^2*s)]
133     dJ_vsdB = np.zeros((num_time_steps,nodes))        #Total Vapor flux (Mass
        ↪ fractiopn), [kg/(m^2*s)]
134
135     dJ_ls_J_vsdB = np.zeros((num_time_steps,nodes))   #Liquid flux, [kg/(m^2*s)]
136
137     dJ_as1dB = np.zeros((num_time_steps,nodes))       #Air flux (Pressure),
        ↪ [kg/(m^2*s)]
138     dJ_as2dB = np.zeros((num_time_steps,nodes))       #Air flux (Mass
        ↪ fractiopn), [kg/(m^2*s)]
139     dJ_asdB = np.zeros((num_time_steps,nodes))        #Total air flux,
        ↪ [kg/(m^2*s)]
140
141     rho_vg = np.zeros((num_time_steps,nodes))         #Water vapor density,
        ↪ [kg/m^3]
142     rho_ag = np.zeros((num_time_steps,nodes))         #air density, [kg/m^3]
143     rho_gg = np.zeros((num_time_steps,nodes))         #Total gas density,
        ↪ [kg/m^3]
144     rho_l = np.zeros((num_time_steps,nodes))          #Water liquid density,
        ↪ [kg/m^3]
145     rho_vs = np.zeros((num_time_steps,nodes))         #Vapor density at the
        ↪ surface, [kg/m^3]
146     rho_air = np.zeros((num_time_steps,nodes))        #Density of air vapor far
        ↪ away from the surface, [kg/m^3]
147
148     ohm_L = np.zeros((num_time_steps,nodes))          #Consant for Liquid flux
149     ohm_v1 = np.zeros((num_time_steps,nodes))         #Consant for vapor flux
        ↪ (pressure)
150     ohm_v2 = np.zeros((num_time_steps,nodes))         #Consant for vapor flux
        ↪ (mass fraction)
151     ohm_a1 = np.zeros((num_time_steps,nodes))         #Consant for vapor flux
        ↪ (pressure)
152     ohm_a2 = np.zeros((num_time_steps,nodes))         #Consant for vapor flux
        ↪ (mass fraction)
153     ohm_c = np.zeros((num_time_steps,nodes))          #Consant for Liquid flux
154
155     omega_v = np.zeros((num_time_steps,nodes))        #Mass fraction of vapour
        ↪ in gas, []
156     omega_a = np.zeros((num_time_steps,nodes))        #Mass fraction of vapour
        ↪ in gas, []
157
```

```
158     epsilon_s = np.zeros((num_time_steps,nodes))      #Volume fraction of solid,
        ↪  []
159     epsilon_l = np.zeros((num_time_steps,nodes))      #Volume fraction of
        ↪  liquid, []
160     epsilon_g = np.zeros((num_time_steps,nodes))      #Volume fraction of gas,
        ↪  []
161
162     phi = np.zeros((num_time_steps,nodes))            #Solid bulk of paper,
        ↪  [m^3/kg_dm]
163
164     time_sim = np.zeros((num_time_steps,1))
165     L = np.zeros((num_time_steps,1))
166     L_sum = 0
167     U_avg = np.zeros((num_time_steps,1))
168     u_sum = 0
169
170     DM = np.zeros((num_time_steps,1))                 #Dry matter fraction, []
171
172     lambda_l = np.zeros((num_time_steps,nodes))       #Heat transfer coefficient
        ↪  for liquid water, [W/(m*K)]
173     lambda_eff = np.zeros((num_time_steps,nodes))     #Effective heat transfer
        ↪  coefficient, [W/(m*K)]
174
175     psi = np.zeros((num_time_steps,nodes))            #Activity of water, []
176
177     cp_a = np.zeros((num_time_steps,nodes))           #Specific heat of dry air,
        ↪  [J/(kg*K)]
178     cp_L = np.zeros((num_time_steps,nodes))           #Specific heat of liquid,
        ↪  [J/(kg*K)]
179     cp_v = np.zeros((num_time_steps,nodes))           #Specific heat of water
        ↪  vapor, [J/(kg*K)]
180
181     h_OS = np.zeros((num_time_steps,nodes))           #Solid bulk of paper,
        ↪  [m^3/kg_dm]
182     sigma = np.zeros((num_time_steps,nodes))          #Mass transfer
        ↪  coefficient, [m/s]
183     Q_conv= np.zeros((num_time_steps,nodes))          #Convection heat transfer,
        ↪  [W/m^2]
184     Q_evap= np.zeros((num_time_steps,nodes))          #Evaporation heat
        ↪  transfer, [W/m^2]
185     Fq = np.zeros((num_time_steps,nodes))             #Combined heat transfer,
        ↪  [W/m^2]
186
```

```
187     Q_rad = np.zeros((num_time_steps, nodes))         #Radiation heat transfer,
        ↪ [W/m^2]
188
189     dq1dB= np.zeros((num_time_steps,nodes))           #1. term in energy
        ↪ balance, [W/m^2]
190     dq2dB= np.zeros((num_time_steps,nodes))           #2. term in energy
        ↪ balance, [W/m^2]
191     dq3dB= np.zeros((num_time_steps,nodes))           #3. term in energy
        ↪ balance, [W/m^2]
192     dq4dB= np.zeros((num_time_steps,nodes))           #4. term in energy
        ↪ balance, [W/m^2]
193     dQdB= np.zeros((num_time_steps,nodes))            #Total energy balance,
        ↪ [W/m^2]
194
195     H_vap = np.zeros((num_time_steps,nodes))          #Enthalpy of vaporization,
        ↪ [J/kg]
196     H_sorp = np.zeros((num_time_steps,nodes))         #Enthalpy of
        ↪ Sorption/desorption, [J/kg]
197
198     Test_1 = np.zeros((num_time_steps,nodes))
199     Test_2 = np.zeros((num_time_steps,nodes))
200     FF_values = []
201
202 #Calculation of initial conditions
203 for i in range(0,1):
204     for j in range(0,nodes):
205
206         #Product temperature
207         T[i,j] = T_0                                  #Initial temperature,
            ↪ [K]
208
209         #Product moisture ratio
210         U[i,j] = U_0                                  #Initial moisture
            ↪ ratio, [kg_H20 / kg_dm]
211
212         #Bulk
213         phi[i,j] = phi_f(U[i,j])                      #Initial bulk of paper,
            ↪ [m^3 / kg_dm]
214
215         #Gas pressures
216         P_gg[i,j] = P_atm                             #Initial total
            ↪ pressure, [Pa]
```

```
217         P_vg[i,j] = IAPWS97(T=T[i,j], x=1).P *10**6      #Initial vapour partial
            ↪ pressure, [Pa]
218         P_ag[i,j] = P_gg[i,j] - P_vg[i,j]                #Initial air partial
            ↪ pressure, [Pa]
219
220         #Gas densities
221         rho_vg[i,j] = (P_vg[i,j]*M_H2O)/(R*T[i,j])
            ↪ #Initial density of water vapor [kg/m^3] #Page 121,   Eq - 3-101
222         rho_ag[i,j] = (P_ag[i,j]*M_air)/(R*T[i,j])
            ↪ #Initial density of air [kg/m^3] #Page 121,   Eq - 3-101
223         rho_gg[i,j] = rho_vg[i,j] + rho_ag[i,j]
            ↪ #Initial total gas density, [kg/m^3]
224         rho_l[i,j]  = IAPWS97(T=T[i,j], x=0).rho
            ↪ #Initial water liquid density, [kg/m^3]
225         rho_vs[i,j] = Rosenkilde.rho_vs(T[i,j])
            ↪ #Initial vapor density at the surface, [kg/m^3]
226         rho_air[i,j] = Rosenkilde.rho_va(T_air, omega_ambient, P_atm)
            ↪ #Initial density of air vapor far away from the surface, [kg/m^3]
227
228         #Gas dynamic viscosity
229         mu_v[i,j] = IAPWS97(T=T[i,j], x=1).mu                    #Dynamic
            ↪ viscosity of water vapor, [Pa*s]
230         mu_a[i,j] = Rosenkilde.mu(T[i,j], omega_ambient)         #Dynamic
            ↪ viscosity of air, [Pa*s]
231
232         #Gas mass fractions
233         omega_v[i,j] = rho_vg[i,j] / rho_gg[i,j]         #Initial mass fraction
            ↪ of water vapour in gas, [kg vapour / kg gas]
234         omega_a[i,j] = rho_ag[i,j] / rho_gg[i,j]         #Initial mass fraction
            ↪ of air in gas, [kg air / kg gas]
235
236         #Gas mass fractions
237         omega_v[i,j] = rho_vg[i,j] / rho_gg[i,j]         #Initial mass fraction
            ↪ of water vapour in gas, [kg vapour / kg gas]
238         omega_a[i,j] = rho_ag[i,j] / rho_gg[i,j]         #Initial mass fraction
            ↪ of air in gas, [kg air / kg gas]
239
240         #Constants for diffusion equations
241         ohm_L[i,j] =  DL(U[i,j], U_FSP, D_0, beta) / phi[i,j]**2
            ↪ #Initial constant for liquid flux, [kg_DM / (m^4 * s)]
242         ohm_v1[i,j] = rho_vg[i,j] * k_g_eff(1/phi[i,j]) / ( mu_v[i,j]*phi[i,j]
            ↪ )              #Initial constant for gas flux for pressure change,
            ↪ [(kg_H2O * kg_DM) / (m^4 * Pa * s)]
```

```
243         ohm_v2[i,j] = rho_gg[i,j] * diffusionCoefficient(T[i,j], P_gg[i,j]) /
            ↪  phi[i,j]      #Initial constant for gas flux for mass fraction
            ↪  change, [(kg_gas * kg_DM) / (m^4 * s)]
244         ohm_a1[i,j] = rho_ag[i,j] * k_g_eff(1/phi[i,j]) / ( mu_a[i,j]*phi[i,j]
            ↪  )              #Initial constant for gas flux for pressure change,
            ↪  [(kg_air * kg_DM) / (m^4 * Pa * s)]
245         ohm_a2[i,j] = rho_gg[i,j] * diffusionCoefficient(T[i,j], P_gg[i,j]) /
            ↪  phi[i,j]      #Initial constant for gas flux for mass fraction
            ↪  change, [(kg_gas * kg_DM) / (m^4 * s)]
246
247         lambda_l[i,j] = IAPWS97(T=T[i,j], x=0).k #Thermal conductivity of
            ↪  liquid water, [W/(m*K)]
248         lambda_s = 0.157 #Thermal conductivity of solid pulp, [W/(m*K)] #Table
            ↪  3-9 page 118
249         lambda_eff[i,j] = (lambda_s + U[i,j] * lambda_l[i,j])/(1 + U[i,j])
            ↪  #Effective Thermal conductivity through the pulp, [W/(m*K)], Page
            ↪  95 table 3-7
250         ohm_c[i,j] = lambda_eff[i,j] / (phi[i,j])
251
252         #Volume fractions
253         epsilon_s[i,j] = 1 / (phi[i,j] * rho_ss)
            ↪  #Initial volume fraction for solid Equation 2-20 + 2-23 page 30
254         epsilon_l[i,j] = U[i,j] / (phi[i,j] * rho_l[i,j])
            ↪  #Initial volume fraction for liquid Equation 2-21 page 30
255         epsilon_g[i,j] = 1 - epsilon_s[i,j] - epsilon_l[i,j]
            ↪  #Initial volume fraction for gas Equation 2-22 page 30
256
257         #Solid-based concentration
258         X_a[i,j] = phi[i,j]*epsilon_g[i,j]*P_ag[i,j]*M_air/(R*T[i,j])
            ↪  #Inital solid-based concentration of air, [kg_air/kg_dm] #Equation
            ↪  3-106 page 122
259         X_v[i,j] = phi[i,j]*epsilon_g[i,j]*P_vg[i,j]*M_H2O/(R*T[i,j])
            ↪  #Inital solid-based concentration of vapour, [kg_H2O/kg_dm]
            ↪  #Equation 3-106 page 122
260         X_L[i,j] = rho_l[i,j] * epsilon_l[i,j] / (1/phi[i,j])
            ↪  #Inital solid-based concentration of Liquid, [kg_H2O/kg_dm]
261
262
263         #Enthalpy of evaporation and sorption
264         H_vap[i,j] = ((IAPWS97(T=T[i,j], x=1.0).h - IAPWS97(T=T[i,j],
            ↪  x=0).h))*10**3 #Entahlpy of evaporation calculated as the
            ↪  difference from vapor to liquid, [J/kg]
```

```
265         psi[i,j] = psi_f(U[i,j], T[i,j]) #Water activity [] #Equation 3-108
            ↪ page 123
266         H_sorp[i,j] = ((R*T[i,j]**2)/(M_H2O)) * ( 1- psi[i,j] )/(psi[i,j]) *
            ↪ 0.10085*U[i,j]**(1.0585) #Enthalpy of sorption/desorption, [J/kg]
            ↪ #Equation 3-109 page 123
267
268         #Fitting function
269         FF = (1 - np.exp(-x * U[i-1,0]**y))
270
271         FF_values.append(FF)
272     L[i] = dB * phi[i,j] * (nodes-1) * 1000      #Initial thickness, [mm]
273     U_avg[i] = U_0                               #Initial U average
274     DM[i] = 1 / (U_avg[i] + 1) * 100             #Initial dry matter %
275     time_sim[i] = 0                              #Initial time
276
277 #Calculation for each time step
278 start = time.time()
279
280 for i in range(1, num_time_steps):
281     #Calculations for the boundary nodes
282     if 1 == 1:
283         #Mass balance of moisture boundary nodes
284         Fm[i-1,0]=h_mass * (rho_vg[i-1,0]-rho_air[i-1,0]) * FF   #Convective
            ↪ mass diffusion in boundary layer 0, [kg H2O / m2 s]
285         Fm[i-1,N]=h_mass * (rho_vg[i-1,N]-rho_air[i-1,N]) * FF   #Convective
            ↪ mass diffusion in boundary layer N, [kg H2O / m2 s]
286
287         #Changes in liquid and vapour flux in boundary
288         dJ_ls_J_vsdB[i-1,0] = (2/dB) * Fm[i-1,0] + (1/dB**2) * ( (ohm_L[i-1,0]
            ↪ + ohm_L[i-1,0+1]) * (U[i-1,0] -  U[i-1,0+1]) + (ohm_v1[i-1,0] +
            ↪ ohm_v1[i-1,0+1]) * (P_gg[i-1,0] - P_gg[i-1,0+1]) + (ohm_v2[i-1,0] +
            ↪ ohm_v2[i-1,0+1]) * (omega_v[i-1,0] - omega_v[i-1,0+1]))   #Change
            ↪ in liquid and vapour flux in boundary 0, [kg_H2O / (kg_DM s)]
289         dJ_ls_J_vsdB[i-1,N] = (2/dB) * Fm[i-1,N] + (1/dB**2) * ( (ohm_L[i-1,N]
            ↪ + ohm_L[i-1,N-1]) * (U[i-1,N] -  U[i-1,N-1]) + (ohm_v1[i-1,N] +
            ↪ ohm_v1[i-1,N-1]) * (P_gg[i-1,N] - P_gg[i-1,N-1]) + (ohm_v2[i-1,N] +
            ↪ ohm_v2[i-1,N-1]) * (omega_v[i-1,N] - omega_v[i-1,N-1]))   #Change
            ↪ in liquid and vapour flux in boundary N, [kg_H2O / (kg_DM s)]
290
291         U[i,0] = U[i-1,0] - dJ_ls_J_vsdB[i-1,0] * dt # Moisture ratio at t=i,
            ↪ [kg_H2O/kg_dm]
292         U[i,N] = U[i-1,N] - dJ_ls_J_vsdB[i-1,N] * dt # Moisture ratio at t=i,
            ↪ [kg_H2O/kg_dm]
```

```
293     # Mass balance air boundary nodes
294     if 1 == 1:
295 
296         #Boundary flux
297         J_as_b[i-1,0] = 0
298         J_as_b[i-1,N] = 0
299 
300         #Changes in air flux in boundary
301         dJ_asdB[i-1,0] = (2/dB) * J_as_b[i-1,0] + (1/dB**2) * ( (ohm_a1[i-1,0]
            ↪  + ohm_a1[i-1,0+1]) * (P_gg[i-1,0] - P_gg[i-1,0+1]) + (ohm_a2[i-1,0]
            ↪  + ohm_a2[i-1,0+1]) * (omega_a[i-1,0] - omega_a[i-1,0+1]))  #Change
            ↪  in air flux in boundary 0, [kg_air / (kg_DM s)]
302         dJ_asdB[i-1,N] = (2/dB) * J_as_b[i-1,N] + (1/dB**2) * ( (ohm_a1[i-1,N]
            ↪  + ohm_a1[i-1,N-1]) * (P_gg[i-1,N] - P_gg[i-1,N-1]) + (ohm_a2[i-1,N]
            ↪  + ohm_a2[i-1,N-1]) * (omega_a[i-1,N] - omega_a[i-1,N-1]))  #Change
            ↪  in air flux in boundary N, [kg_air / (kg_DM s)]
303 
304 
305         #Change in solid-based concentration for air
306         X_a[i,0] = X_a[i-1,0] - dJ_asdB[i-1,0] * dt #Solid-based concentration
            ↪  of air, [kg_air/kg_dm]
307         X_a[i,N] = X_a[i-1,N] - dJ_asdB[i-1,N] * dt #Solid-based concentration
            ↪  of air, [kg_air/kg_dm]
308     #Energy balance boundary nodes
309     if 1 == 1:
310 
311         Q_conv[i-1,0] = h_conv * (T[i-1,0] - T_air)       #W/m^2
312         Q_conv[i-1,N] = h_conv * (T[i-1,N] - T_air)       #W/m^2
313 
314         Q_rad[i-1,0] = sigma_rad * e * (T[i-1,0]**4 - T_air**4) #W/m^2
315         Q_rad[i-1,N] = sigma_rad * e * (T[i-1,N]**4 - T_air**4) #W/m^2
316 
317         Q_evap[i-1,0] = Fm[i-1,0] * (H_vap[i-1,0] + H_sorp[i-1,0]) #W/m^2
318         Q_evap[i-1,N] = Fm[i-1,N] * (H_vap[i-1,N] + H_sorp[i-1,N]) #W/m^2
319 
320         Evap_sorp = Fm[i-1,0] * (H_sorp[i-1,0]) * dt
321         Evap_vap = Fm[i-1,0] * (H_vap[i-1,0]) * dt
322         Evap_tot = Evap_sorp + Evap_vap
323         Total_Q_evap = np.sum(Q_evap)
324 
325         Fq[i-1,0] = Q_conv[i-1,0] + Q_rad[i-1,0] + Q_evap[i-1,0]
            ↪  #W/m^2
```

```
326         Fq[i-1,N] = Q_conv[i-1,N] + Q_rad[i-1,N] + Q_evap[i-1,N]
            ↪  #W/m^2
327
328         J_vs_b[i,0] = -ohm_v1[i-1,0] * (P_gg[i-1,0]-P_gg[i-1,0+1]) / (dB) -
            ↪  ohm_v2[i-1,0] * (omega_v[i-1,0]-omega_v[i-1,0+1])/(dB)
329         J_vs_b[i,N] = -ohm_v1[i-1,N] * (P_gg[i-1,N]-P_gg[i-1,N-1]) / (dB) -
            ↪  ohm_v2[i-1,N] * (omega_v[i-1,N]-omega_v[i-1,N-1])/(dB)
330
331         J_as_b[i,0] = -ohm_a1[i-1,0] * (P_gg[i-1,0]-P_gg[i-1,0+1]) / (dB) -
            ↪  ohm_a2[i-1,0] * (omega_a[i-1,0]-omega_a[i-1,0+1])/(dB)
332         J_as_b[i,N] = -ohm_a1[i-1,N] * (P_gg[i-1,N]-P_gg[i-1,N-1]) / (dB) -
            ↪  ohm_a2[i-1,N] * (omega_a[i-1,N]-omega_a[i-1,N-1])/(dB)
333
334         J_Ls_b[i,0] = -ohm_L[i-1,0] * (U[i-1,0]-U[i-1,0+1])/(dB)
335         J_Ls_b[i,N] = -ohm_L[i-1,N] * (U[i-1,N]-U[i-1,N-1])/(dB)
336
337         dq1dB[i-1,0] = (2/dB) * Fq[i-1,0] + (1/dB**2) * ((ohm_v1[i-1,0] +
            ↪  ohm_v1[i-1,0+1]) * (P_gg[i-1,0] - P_gg[i-1,0+1]) * ((H_vap[i-1,0] +
            ↪  H_sorp[i-1,0])-(H_vap[i-1,0+1] + H_sorp[i-1,0+1])) + (ohm_v2[i-1,0]
            ↪  + ohm_v2[i-1,0+1]) * (omega_v[i-1,0] - omega_v[i-1,0+1]) *
            ↪  ((H_vap[i-1,0] + H_sorp[i-1,0])-(H_vap[i-1,0+1] + H_sorp[i-1,0+1]))
            ↪  + (ohm_c[i-1,0] + ohm_c[i-1,0+1])* (T[i-1,0] - T[i-1,0+1]) ) # [W /
            ↪  kg_DM]
338         dq2dB[i-1,0] = 1/dB * (T[i-1,0] - T[i-1,0+1]) * (cp_a[i-1,0] *
            ↪  J_as_b[i-1,0] + cp_v[i-1,0] * J_vs_b[i-1,0] + cp_L[i-1,0] *
            ↪  J_Ls_b[i-1,0]) # [W / kg_Dm]
339         dq3dB[i-1,0] = J_Ls_b[i-1,0] * 1/dB * (H_sorp[i-1,0] - H_sorp[i-1,0+1])
            ↪  # [W / kg_dm]
340         dq4dB[i-1,0] = J_vs_b[i-1,0] * 1/dB * (H_sorp[i-1,0] - H_sorp[i-1,0+1]
            ↪  + H_vap[i-1,0] - H_vap[i-1,0+1]) # [W / kg_dm]
341         dQdB[i-1,0] = (dq1dB[i-1,0] + dq2dB[i-1,0] + dq3dB[i-1,0] +
            ↪  dq4dB[i-1,0]) / (cp_s + cp_a[i-1,0]*X_a[i-1,0] +
            ↪  cp_v[i-1,0]*X_v[i-1,0] + cp_L[i-1,0]*X_L[i-1,0])
342
343         dq1dB[i-1,N] = (2/dB) * Fq[i-1,N] + (1/dB**2) * ((ohm_v1[i-1,N] +
            ↪  ohm_v1[i-1,N-1]) * (P_gg[i-1,N] - P_gg[i-1,N-1]) * ((H_vap[i-1,N] +
            ↪  H_sorp[i-1,N])-(H_vap[i-1,N-1] + H_sorp[i-1,N-1])) + (ohm_v2[i-1,N]
            ↪  + ohm_v2[i-1,N-1]) * (omega_v[i-1,N] - omega_v[i-1,N-1]) *
            ↪  ((H_vap[i-1,N] + H_sorp[i-1,N])-(H_vap[i-1,N-1] + H_sorp[i-1,N-1]))
            ↪  + (ohm_c[i-1,N] + ohm_c[i-1,N-1])* (T[i-1,N] - T[i-1,N-1]) ) # [W /
            ↪  kg_DM]
```

```
344         dq2dB[i-1,N] = 1/dB * (T[i-1,N] - T[i-1,N-1]) * (cp_a[i-1,N] *
            ↪  J_as_b[i-1,N] + cp_v[i-1,N] * J_vs_b[i-1,N] + cp_L[i-1,N] *
            ↪  J_Ls_b[i-1,N]) # [W / kg_Dm]
345         dq3dB[i-1,N] = J_Ls_b[i-1,N] * 1/dB * (H_sorp[i-1,N] - H_sorp[i-1,N-1])
            ↪  # [W / kg_dm]
346         dq4dB[i-1,N] = J_vs_b[i-1,N] * 1/dB * (H_sorp[i-1,N] - H_sorp[i-1,N-1]
            ↪  + H_vap[i-1,N] - H_vap[i-1,N-1]) # [W / kg_dm]
347         dQdB[i-1,N] = (dq1dB[i-1,N] + dq2dB[i-1,N] + dq3dB[i-1,N] +
            ↪  dq4dB[i-1,N]) / (cp_s + cp_a[i-1,N]*X_a[i-1,N] +
            ↪  cp_v[i-1,N]*X_v[i-1,N] + cp_L[i-1,N]*X_L[i-1,N])
348
349         #Temporary product temperature for boundaries
350         T[i,0] = T[i-1,0] - dQdB[i-1,0] * dt                        #Product
            ↪  temperature at t=i, [K]
351         T[i,N] = T[i-1,N] - dQdB[i-1,N] * dt                        #Product
            ↪  temperature at t=i, [K]
352
353
354         #Update properties for boundary nodes
355         if 1 == 1:
356
357             #Total pressure boundary conditions
358             P_gg[i,0] = P_atm                               #Total pressure, [Pa]
359             P_gg[i,N] = P_atm                               #Total pressure, [Pa]
360
361             #Vapour pressure boundary conditions
362             P_vg[i,0] = IAPWS97(T=T[i,0], x=1).P *10**6 #Saturation pressure,
                ↪  [Pa]
363             P_vg[i,N] = IAPWS97(T=T[i,N], x=1).P *10**6 #Saturation pressure,
                ↪  [Pa]
364
365             #Air pressure boundary conditions
366             P_ag[i,0] = P_gg[i,0] - P_vg[i,0]           #Pressure of air, [Pa]
367             P_ag[i,N] = P_gg[i,N] - P_vg[i,N]           #Pressure of air, [Pa]
368
369             #Bulk boundary conditions
370             phi[i,0] = phi_f(U[i,0])                    #Solid bulk of paper,
                ↪  [m^3/kg_dm]
371             phi[i,N] = phi_f(U[i,N])                    #Solid bulk of paper,
                ↪  [m^3/kg_dm]
372
373             #Volume fractions boundary
```

```
374             epsilon_s[i,0] = 1 / (phi[i,0] * rho_ss)                #Volume
                ↪  fraction solid, [m3_DM / m3]
375             epsilon_l[i,0] = (U[i,0]) / ( phi[i,0] * rho_l[i-1,0])
                ↪  #Volume fraction liquid, [m3_H2O_liquid / m3]
376             epsilon_g[i,0] = 1 - epsilon_s[i,0] - epsilon_l[i,0]    #Volume
                ↪  fraction gas, [m3_gas / m3]
377
378             epsilon_s[i,N] = 1 / (phi[i,N] * rho_ss)                #Volume
                ↪  fraction solid, [m3_DM / m3]
379             epsilon_l[i,N] = (U[i,N]) / ( phi[i,N] * rho_l[i-1,N])
                ↪  #Volume fraction liquid, [m3_H2O_liquid / m3]
380             epsilon_g[i,N] = 1 - epsilon_s[i,N] - epsilon_l[i,N]    #Volume
                ↪  fraction gas, [m3_gas / m3]
381     #Calculations for the internal nodes
382     for j in range(1,N):
383         #Mass balance interior nodes
384         if 1 == 1:
385
386             #Changes in liquid and vapour flux in boundary
387             dJ_lsdB[i-1,j]  = - (1/dB**2) * (  ( ohm_L[i-1,j+1] +
                ↪  ohm_L[i-1,j])/2 *       U[i-1,j+1] - ( ohm_L[i-1,j+1]/2 +
                ↪  ohm_L[i-1,j] +  ohm_L[i-1,j-1]/2) *       U[i-1,j] + (
                ↪  ohm_L[i-1,j] +  ohm_L[i-1,j-1])/2 *       U[i-1,j-1])  #Change
                ↪  in liquid flux in interior nodes, [kg_H2O / (kg_DM s)]
388             dJ_vs1dB[i-1,j] = - (1/dB**2) * (  (ohm_v1[i-1,j+1] +
                ↪  ohm_v1[i-1,j])/2 *    P_gg[i-1,j+1] - (ohm_v1[i-1,j+1]/2 +
                ↪  ohm_v1[i-1,j] + ohm_v1[i-1,j-1]/2) *    P_gg[i-1,j] +
                ↪  (ohm_v1[i-1,j] + ohm_v1[i-1,j-1])/2 *    P_gg[i-1,j-1])
                ↪  #Change in vapour flux in interior nodes (pressure force),
                ↪  [kg_H2O / (kg_DM s)]
389             dJ_vs2dB[i-1,j] = - (1/dB**2) * (  (ohm_v2[i-1,j+1] +
                ↪  ohm_v2[i-1,j])/2 * omega_v[i-1,j+1] - (ohm_v2[i-1,j+1]/2 +
                ↪  ohm_v2[i-1,j] + ohm_v2[i-1,j-1]/2) * omega_v[i-1,j] +
                ↪  (ohm_v2[i-1,j] + ohm_v2[i-1,j-1])/2 * omega_v[i-1,j-1])
                ↪  #Change in vapour flux in interior nodes (concentration force),
                ↪  [kg_H2O / (kg_DM s)]
390             dJ_vsdB[i-1,j] =  dJ_vs1dB[i-1,j] + dJ_vs2dB[i-1,j]     #Change in
                ↪  vapour flux in interior nodes, [kg_H2O / (kg_DM s)]
391             dJ_ls_J_vsdB[i-1,j] = dJ_lsdB[i-1,j] + dJ_vsdB[i-1,j]   #Change in
                ↪  liquid and vapour flux in interior nodes, [kg_H2O / (kg_DM s)]
392
393             U[i,j] = U[i-1,j] - dJ_ls_J_vsdB[i-1,j] * dt #Moisture ratio at
                ↪  t=i, [kg_H2O/kg_dm]
```

```
394         #Mass balance air interior nodes
395         if 1 == 1:
396 
397             #Changes in air flux in interior nodes
398             dJ_as1dB[i-1,j] = - (1/dB**2) * (  (ohm_a1[i-1,j+1] +
                ↪  ohm_a1[i-1,j])/2 *    P_gg[i-1,j+1] - (ohm_a1[i-1,j+1]/2 +
                ↪  ohm_a1[i-1,j] + ohm_a1[i-1,j-1]/2) *    P_gg[i-1,j] +
                ↪  (ohm_a1[i-1,j] + ohm_a1[i-1,j-1])/2 *    P_gg[i-1,j-1])
                ↪  #Change in air flux in interior nodes (pressure force), [kg_air
                ↪  / (kg_DM s)]
399             dJ_as2dB[i-1,j] = - (1/dB**2) * (  (ohm_a2[i-1,j+1] +
                ↪  ohm_a2[i-1,j])/2 * omega_a[i-1,j+1] - (ohm_a2[i-1,j+1]/2 +
                ↪  ohm_a2[i-1,j] + ohm_a2[i-1,j-1]/2) * omega_a[i-1,j] +
                ↪  (ohm_a2[i-1,j] + ohm_a2[i-1,j-1])/2 * omega_a[i-1,j-1])
                ↪  #Change in air flux in interior nodes (concentration force),
                ↪  [kg_air / (kg_DM s)]
400             dJ_asdB[i-1,j] =  dJ_as1dB[i-1,j] + dJ_as2dB[i-1,j]      #Change in
                ↪  air flux in interior nodes, [kg_air / (kg_DM s)]
401 
402             #Change in solid-based concentration for air
403             X_a[i,j] = X_a[i-1,j] - dJ_asdB[i-1,j] * dt #Solid-based
                ↪  concentration of air, [kg/kg_dm]
404         #Energy balance interior nodes
405         if 1 == 1:
406 
407             J_as_i[i,j] = -ohm_a1[i-1,j] * (P_gg[i-1,j+1]-P_gg[i-1,j-1]) / (dB)
                ↪  - ohm_a2[i-1,j] * (omega_a[i-1,j+1]-omega_a[i-1,j-1])/(dB*2)
408             J_vs_i[i,j] = -ohm_v1[i-1,j] * (P_gg[i-1,j+1]-P_gg[i-1,j-1]) / (dB)
                ↪  - ohm_v2[i-1,j] * (omega_v[i-1,j+1]-omega_v[i-1,j-1])/(dB*2)
409             J_Ls_i[i,j] = -ohm_L[i-1,j] * (U[i-1,j+1]-U[i-1,j-1])/(dB*2)
410 
```

```
411             dq1dB[i-1,j] = -((H_vap[i-1,j+1] + H_sorp[i-1,j+1])-2*(H_vap[i-1,j]
                ↪  + H_sorp[i-1,j])+(H_vap[i-1,j-1] + H_sorp[i-1,j-1])) * (-
                ↪  (1/dB**2) * (  (ohm_v1[i-1,j+1] + ohm_v1[i-1,j])/2 *
                ↪  P_gg[i-1,j+1] - (ohm_v1[i-1,j+1]/2 + ohm_v1[i-1,j] +
                ↪  ohm_v1[i-1,j-1]/2) *    P_gg[i-1,j] + (ohm_v1[i-1,j] +
                ↪  ohm_v1[i-1,j-1])/2 *    P_gg[i-1,j-1])  - (1/dB**2) * (
                ↪  (ohm_v2[i-1,j+1] + ohm_v2[i-1,j])/2 * omega_v[i-1,j+1] -
                ↪  (ohm_v2[i-1,j+1]/2 + ohm_v2[i-1,j] + ohm_v2[i-1,j-1]/2) *
                ↪  omega_v[i-1,j] + (ohm_v2[i-1,j] + ohm_v2[i-1,j-1])/2 *
                ↪  omega_v[i-1,j-1])) - (1/dB**2) * (  (ohm_c[i-1,j+1] +
                ↪  ohm_c[i-1,j])/2 * T[i-1,j+1] - (ohm_c[i-1,j+1]/2 + ohm_c[i-1,j]
                ↪  + ohm_c[i-1,j-1]/2) * T[i-1,j] + (ohm_c[i-1,j] +
                ↪  ohm_c[i-1,j-1])/2 * T[i-1,j-1])  #Change in vapour flux in
                ↪  interior nodes (pressure force), [kg_H2O / (kg_DM s)]
412             dq2dB[i-1,j] = 1/(2 * dB) * (T[i-1,j+1] - T[i-1,j-1]) *
                ↪  (cp_a[i-1,j] * J_as_i[i-1,j] + cp_v[i-1,j] * J_vs_i[i-1,j] +
                ↪  cp_L[i-1,j] * J_Ls_i[i-1,j]) # [W / kg_Dm]
413             dq3dB[i-1,j] = J_Ls_i[i-1,j] * 1/ (2 *dB) * (H_sorp[i-1,j+1] -
                ↪  H_sorp[i-1,j-1]) # [W / kg_dm]
414             dq4dB[i-1,j] = J_vs_i[i-1,j] * 1/ (2*dB) * (H_sorp[i-1,j+1] -
                ↪  H_sorp[i-1,j-1] + H_vap[i-1,j+1] - H_vap[i-1,j-1]) # [W /
                ↪  kg_dm]
415
416             dQdB[i-1,j] = (dq1dB[i-1,j] + dq2dB[i-1,j] + dq3dB[i-1,j] +
                ↪  dq4dB[i-1,j]) / (cp_s + cp_a[i-1,j]*X_a[i-1,j] +
                ↪  cp_v[i-1,j]*X_v[i-1,j] + cp_L[i-1,j]*X_L[i-1,j])
417
418             #Temporary product temperature for boundaries
419             T[i,j] = T[i-1,j] - dQdB[i-1,j] * dt                        #Product
                ↪  temperature at t=i, [K]
420         #Update properties for interior nodes
421         if 1 == 1:
422
423             phi[i,j] = phi_f(U[i,j]) #Solid bulk of paper, [m^3/kg_dm]
424
425             epsilon_s[i,j] = 1 / (phi[i,j] * rho_ss)
426             epsilon_l[i,j] = (U[i,j]) / ( phi[i,j] * rho_l[i-1,j])
427             epsilon_g[i,j] = 1 - epsilon_s[i,j] - epsilon_l[i,j]
428
429             #Gas pressures interior nodes
430             if pressuresolver == 0:
431                 P_ag[i,j] = P_atm - IAPWS97(T=T[i,j], x=1).P *10**6 #Pressure
                    ↪  of air, [Pa]
```

```
            elif pressuresolver == 1:
                P_ag[i,j] = (X_a[i,j]*R*T[i,j]) / (phi[i,j]*epsilon_g[i,j]*M_air)
            else:
                print("Choose pressure solver!")
            P_vg[i,j] = IAPWS97(T=T[i,j], x=1).P *10**6 #Saturation pressure, [Pa]
            P_gg[i,j] = P_vg[i,j] + P_ag[i,j] #Total pressure, [Pa]
    #Update common properties for all nodes
    for j in range(0,nodes):

        #Gas dynamic viscosity
        mu_v[i,j] = IAPWS97(T=T[i,j], x=1).mu                          #Dynamic viscosity of water vapor, [Pa*s]
        mu_a[i,j] = Rosenkilde.mu(T[i,j], omega_ambient)               #Dynamic viscosity of air, [Pa*s]

        #Densities
        rho_vg[i,j] = (P_vg[i,j]*M_H2O)/(R*T[i,j]) #Density of of water vapor [kg/m^3] #Page 121,   Eq - 3-101
        rho_ag[i,j] = (P_ag[i,j]*M_air)/(R*T[i,j]) #Density of of air [kg/m^3] #Page 121,   Eq - 3-101
        rho_gg[i,j] = rho_vg[i,j] + rho_ag[i,j] #Total gas density, [kg/m^3]
        rho_l[i,j]  = IAPWS97(T=T[i,j], x=0).rho         #Initial water liquid density, [kg/m^3]
        rho_vs[i,j] = Rosenkilde.rho_vs(T[i,j]) #Initial vapor density at the surface, [kg/m^3]
        rho_air[i,j] = Rosenkilde.rho_va(T_air, omega_ambient, P_atm) #Initial density of air vapor far away from the surface, [kg/m^3]

        omega_v[i,j] = rho_vg[i,j] / rho_gg[i,j] #Mass fraction of water vapour in gas, []
        omega_a[i,j] = rho_ag[i,j] / rho_gg[i,j] #Mass fraction of air in gas, []

        ohm_L[i,j] =  DL(U[i,j], U_FSP, D_0, beta) / phi[i,j]**2 #Constant for liquid flux
        ohm_v1[i,j] = (rho_vg[i,j] * k_g_eff(1/phi[i,j])) / (mu_v[i,j]*phi[i,j]) #Constant for gas flux for pressure change
        ohm_v2[i,j] = rho_gg[i,j] * diffusionCoefficient(T[i,j], P_gg[i,j]) / (phi[i,j]) #Constant for gas flux for mass fraction change
```

```
459         ohm_a1[i,j] = (rho_ag[i,j] * k_g_eff(1/phi[i,j]) ) /
            ↪  (mu_a[i,j]*phi[i,j]) #Constant for gas flux for mass fraction
            ↪  change
460         ohm_a2[i,j] =  rho_gg[i,j] * diffusionCoefficient(T[i,j], P_gg[i,j]) /
            ↪  (phi[i,j]) #Constant for gas flux for mass fraction change
461
462         lambda_l[i,j] = IAPWS97(T=T[i,j], x=0).k #Thermal conductivity of
            ↪  liquid water, [W/(m*K)]
463         lambda_s = 0.157 #Thermal conductivity of solid pulp, [W/(m*K)] #Table
            ↪  3-9 page 118
464         lambda_eff[i,j] = (lambda_s + U[i,j] * lambda_l[i,j])/(1 + U[i,j])
            ↪  #Effective Thermal conductivity through the pulp, [W/(m*K)], Page
            ↪  95 table 3-7
465         ohm_c[i,j] = lambda_eff[i,j] / (phi[i,j])
466
467
468         X_v[i,j] = phi[i,j]*epsilon_g[i,j]*(P_vg[i,j])*M_H2O/(R*T[i,j])
            ↪  #Inital solid-based concentration of vapour, [kg/kg_dm]
469         X_L[i,j] = rho_l[i,j] * epsilon_l[i,j] / (1/phi[i,j])
            ↪  #Inital solid-based concentration of Liquid, [kg_H2O/kg_dm]
470
471         #Specific heats
472         cp_a[i,j] = Rosenkilde.C_pair(T[i,j], omega_ambient) #Specific heat for
            ↪  dry air, [J/(kg*K)]
473         cp_L[i,j] = IAPWS97(T=T[i,j], x=0).cp * 10**3        #Specific heat of
            ↪  Liquid, [J/(kg*K)]
474         cp_v[i,j] = IAPWS97(T=T[i,j], x=1).cp * 10**3        #Specific heat of
            ↪  water vapor, [J/(kg*K)]
475
476
477         L_sum0 = dB * phi[i,0] * 1000           # Thickness, [mm]
478         L_sumN = dB * phi[i,N] * 1000           # Thickness, [mm]
479         L_sum = L_sum + dB * phi[i,j] * 1000    # Thickness, [mm]
480
481         u_sum_0 = U[i,0]
482         u_sum_N = U[i,N]
483         u_sum = u_sum + U[i,j]
484
485         #Enthalpy for evap and sorption
486         H_vap[i,j] = ((IAPWS97(T=T[i,j], x=1.0).h - IAPWS97(T=T[i,j],
            ↪  x=0).h))*10**3 #Entahlpy of evaporation calculated as the
            ↪  difference from vapor to liquid, [J/kg]
```

```
487             psi[i,j] = psi_f(U[i,j], T[i,j]) #Water activity [] #Equation 3-108
                ↪  page 123
488             H_sorp[i,j] = ((R*T[i,j]**2)/(M_H2O)) * ( 1- psi[i,j] )/(psi[i,j]) *
                ↪  0.10085*U[i,j]**(1.0585) #Enthalpy of sorption/desorption, [J/kg]
                ↪  #Equation 3-109 page 123
489
490             FF = (1 - np.exp(-x * U[i-1,0]**y))
491             FF_values.append(FF)
492         #Total thickness
493         if 1 == 1:
494             L[i] = (L_sum - L_sum0/2 - L_sumN/2)
495             L_sum = 0
496             L_sum0 = 0
497             L_sumN = 0
498         #Average moisture ratio U_avg
499         if 1 == 1:
500             U_avg[i] = ( u_sum - u_sum_0/2 - u_sum_N/2 ) / (nodes - 1)
501             u_sum_0 = 0
502             u_sum_N = 0
503             u_sum = 0
504         DM[i] = 1 / (U_avg[i] + 1) * 100
505         time_sim[i] = time_sim[i-1] + dt #Simulation time
506
507 end = time.time()
508 time_elapsed = end - start
509 print("Elapsed time", datetime.timedelta(seconds=time_elapsed))
510
```

B Appendix: Egg-packaging surface mesh

Figure B.1: Surface mesh of the top-lid of the egg-packaging

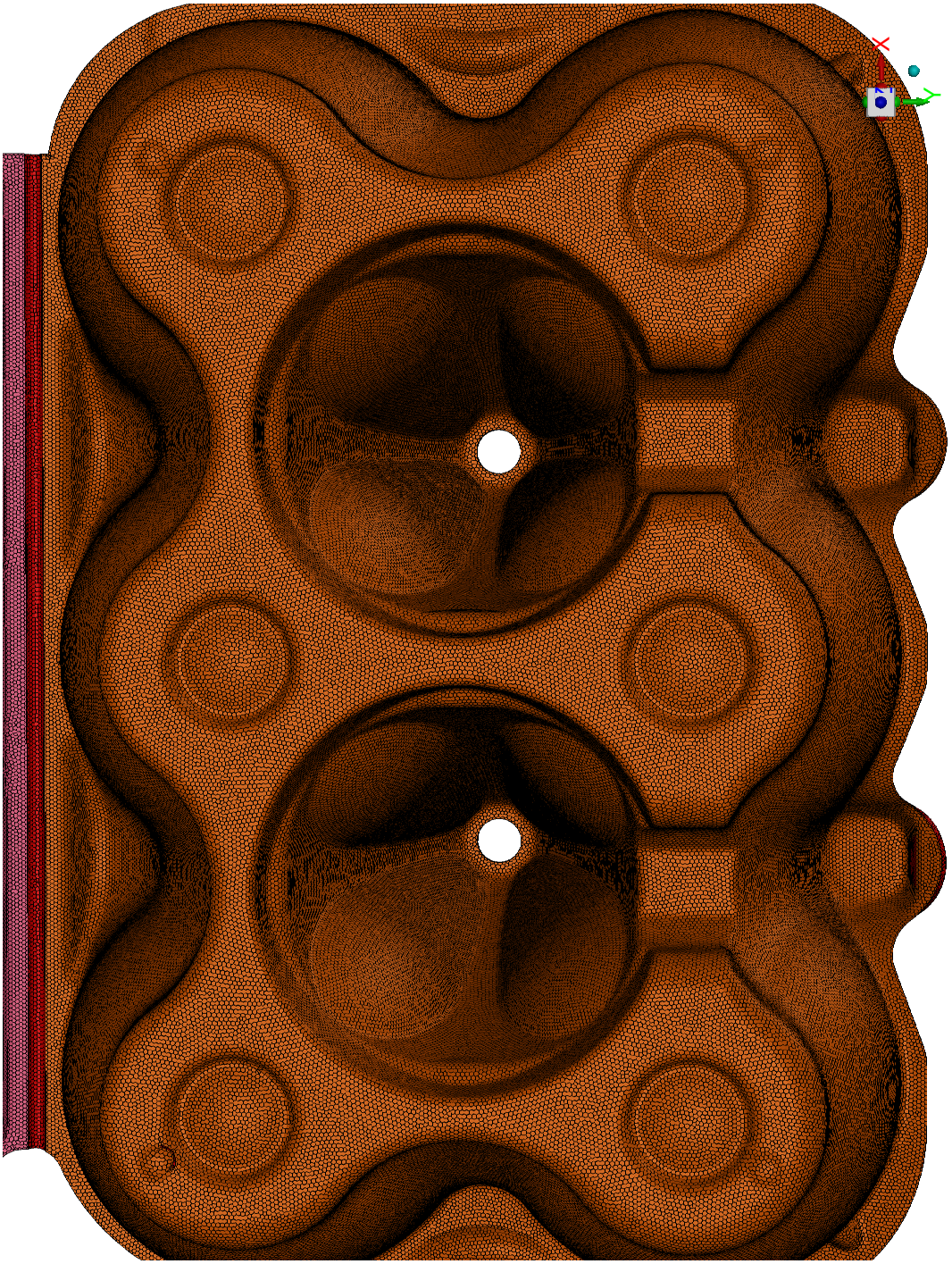


Figure B.2: Surface mesh of the bottom-bottom of the egg-packaging

Figure B.3: Surface mesh of the bottom-lid of the egg-packaging