Using Peltier elements in low temperature district heating networks

by

L. kleyn Winkel

to obtain the degree of Master of Science,
in Mechanical Engineering,
in the Energy, Flow and Process Technology track,
at Delft University of Technology,

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

Abstract

By placing multiple Peltier elements in a linear arrangement while two water flows run past the elements, a temperature increase can be realised in one flow while the other flow is cooled down. In this study the heating of domestic hot water with Peltier elements as solid state heat pumps, and a heating network was investigated. A numerical model that solves the thermal energy balance within the Peltier elements was derived to describe the internal temperature distribution of the Peltier element, and its interaction with the domestic hot water and the heating network. The model was used to simulate the performance of 40 Peltier elements in a custom designed Peltier Heat Exchanger. Experiments were run to validate the numerical model. The numerical simulation of the temperature distribution within a Peltier Heat Exchanger and the temperature distributions observed in the experiments were not in agreement. The model input parameters Seebeck coefficient, resistance, thermal conductivity and a relation for the Nusselt number were re-evaluated using the experimental results. After the adjustment of the model input parameters, the new simulation results were able to accurately describe the temperature distribution with the Peltier Heat Exchanger. The Peltier Heat Exchanger was able to deliver domestic hot water with a COP between 1.2 and 1.8 depending on the flow speed of the domestic hot water and the heating network. The COP can potentially be increased by using Peltier elements with a higher Seebeck coefficient.

L. kleyn Winkel
Leiden, June 21, 2022

Contents

<div align="right">

1

</div>

Introduction

In this research, the use of Peltier elements as solid state heat pumps to increase the temperature of low temperature district heating networks to 60 °C is investigated. This chapter provides background information as to inform why this research is necessary.

1.1. Heating networks

A heating network (or heat distribution network) is an energy source, a hot fluid, that can provide energy to other systems like domestic hot water or central heating systems. Examples of heating networks are an electric boiler, a central heating system for an entire apartment complex or even district heating networks like the one in Rotterdam [38] where thermal energy from the port of Rotterdam is used for domestic heating. The temperature of the district heating network in Rotterdam is designed to be 130 °C. The temperatures of heating networks are set to reduce, as mentioned by Schmidt et al. [31], to 40 °C. Heating networks at these temperatures are considered a *low temperature heating networks*. The reduction in temperature allows for better integration of renewable energy and contributes significantly to a more efficient use of energy resources [31, sec. 2.1] as there is less heat loss compared to networks with higher temperatures. Figure 1.1 schematically shows the heat sources and heat sinks in the fifth generation district heating networks. A downside to the temperature reduction is the possible growth of the legionella bacteria.
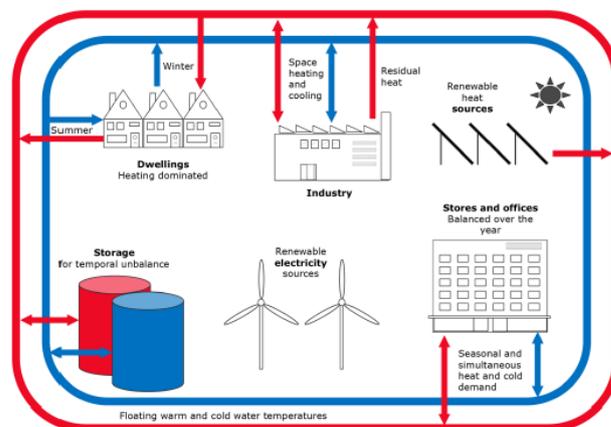
Figure 1.1: Schematic visualisation of a the fifth generation district heating networks by Boesten et al. [4]. The two loops on the outside represent a hot loop and a cold loop.

1.2. Legionella

At temperatures above 20 °C and below 51 °C in domestic water lines, the legionella bacteria can settle and grow according to the World Health Organisation [44]. Dutch law prescribes specific conditions for thermal disinfection of domestic water lines to prevent legionella growth [26]. The minimum temperature demand for thermal disinfection of water is 60 °C. If the district heating has a supply temperature of 45 °C, then an other source of energy is needed to for fill this demand. The government of The Netherlands is stimulating the use of district heating networks [30]. Therefore, the utility of district heating networks will likely rise and with

that the demand for the additional energy source. If combustion heating is undesirable, then two obvious candidates for a sustainable heat source are:

1. Electric heating elements

2. (Air-water) heat pumps

While electric heaters can be cheap, small and as simple as an electric heating element in the water flow, their Coefficient of Performance (COP, Section 1.3) will never exceed 1. Conversely, heat pumps can have a relatively high COP (>4, [20, sec. 1.4]) but can contain harmful, unwanted chemicals and can be relatively large system that are not suitable in every domestic living situation.

1.3. Coefficient of performance

The coefficient of performance (COP) is an import value in heating systems as it is the ratio of the useful heat delivered per power input [20, sec. 1.4]. A higher COP means less input power is required to obtain a similar heat delivery as compared to a lower COP. The previously mentioned electric heating elements and the heat pump use electricity as the input power. A heat pump with COP 4 thus uses only a quarter of the electric input power of an electric heater to provide the same heat delivery. This gives the impression that a higher COP is always desirable. However, due to size, high COP systems are not always a practical solution. The company Flamco proposes to use Peltier elements as solid state heat pumps to necessary thermal energy to increase the domestic water to 60 °C.

1.4. Flamco

Flamco B.V. is a company that develops, produces and sells components for use in HVAC (Heating, Ventilation and Air Conditioning) systems [6]. Among others they produce Heat Interface Units for domestic hot water, HIU for short, as shown in Figure 1.2:

Figure 1.2: A Heat Interface Unit as produced by Flamco [1].

The current products are able to transfer heat from a heat distribution network (like an electric boiler), to tap water and central heating via a heat exchanger, keeping the systems separated. The downside to this method is that the temperature of the leaving water streams can never exceed the temperature of the in-coming water stream. This means, for the Dutch market, a heat source of *at least* 60 °C must be provided to comply with the Dutch law concerning legionella (Section 1.2).

As low temperature heating networks provide lower temperatures than 60 °C, additional heating is needed. In this project a low temperature heating network is considered to deliver water at 45 °C at a flowrate of 20 L/min. The solution proposed by Flamco is to pre-heat the domestic water to 40 °C using the heating network, and to use Peltier elements to extract additional heat from the heating network to increase the temperature of domestic water to 60 °C.

1.5. Current, past and future use of thermoelectric devices

Peltier elements have been widely used since the development of bismuth-telluride room-temperature semi-conductor material in the late 1950s, according to Bell [2]. Applications ranged from creating the temperature of -80 °C needed for infrared guided missile technology where the Peltier elements are used in cooling mode, to generating power in deep-space probe from the temperature difference between nuclear fissile material and the space temperature where the Peltier elements are used in power generation mode [2, sec. Present Applications].

Modern-day applications of Peltier devices are: controlling the temperature cycle of enzymatic reactions to multiply specific sequences of DNA in PCR systems [7], and keeping laser diodes at constant temperature to stabilize operating wavelengths [2, sec. Present Applications]. Yilmazoglu [45] has run experiments where the performance of a prototype thermoelectric heating/cooling unit was investigated for air-air heating and cooling systems. The reported COP for heating was between 2.5 and 5.

Bell [2] claims that as thermoelectric devices become more efficient, they will replace harmful refrigerants in heating and cooling systems, improve electronics' life time by providing better temperature control and improve the mileage of internal combustion engined vehicles by converting waste heat to electricity.

1.6. Project scope

The goal of this project is to investigate, design and verify the use of Peltier elements and heating networks to increase the temperature of domestic hot water. Limitations on the system are the flow speeds of the water flows, the temperature availability of the heating network and the maximum power draw by the system. The proposed solution should be able to run on single phase 230 Volt, 16 amp fused current. The goal is to propose a solution that is able to increase the domestic hot water from 40 °C to 60 °C with a COP of 2.

1.7. Report structure

The relevant physics needed to understand end explain the thermoelectric effect is presented in Chapter 2, as well as relevant information on heat transfer. Chapters 3 and 4.2 discuss the numerical model and the numerical results. The experimental setup and experimental results are presented in Chapters 5 and 6 together with a discussion on the apparent differences in the numerical simulation and the experimental observations. The experimental data is used to re-evaluate relevant numerical simulation input parameters in Chapter 7. The new numerical model is validated in chapter 8. The solution to the research question is presented in Chapter 9 and the conclusions are drawn in Chapter 10.

2

Theory

2.1. Solid state physics

The thermoelectric effect finds its explanation in solid state physics. To understand where the Seebeck and Peltier effect come from, the concepts *band structures, Fermi level/Energy, Fermi distribution, semiconductors, conduction band* and *valence band* need to be introduced.

2.1.1. Fermi energy, Fermi level and Fermi distribution

Atoms consist of a nucleus, surrounded by electrons [9, Section 5.2]. The electrons move in orbits around the nucleus, each orbit corresponding to a certain energy. Suppose the electrons are in the ground state, that is, the energy states are filled from the lowest energy state until the electrons have run out. Since the temperature is at absolute zero, there can be no thermal excitation and the highest occupied energy level is a sharp cutoff. The *Fermi energy* is the difference between the highest and the lowest energy state. Taking the lowest energy level to be equal to $E = 0$ J, the Fermi energy is then equal to the highest energy state, that is denoted with E_f. The Fermi energy of materials of typically in the order of electron volts [33, Somerfeld model].

The *Fermi level*, confusingly also denoted by E_f is defined at finite temperature $T > 0$ K as equal to the energy state that has a chance of 50% of being occupied. The important difference between Fermi energy and Fermi level is that the Fermi energy is only defined at $T = 0$ and is an energy *difference* while the Fermi *level* is the total energy of an energy state, including kinetic and potential energy. It is therefor likely, but not a necessity, that the Fermi level is larger than the Fermi Energy. At finite temperatures $T > 0$ K, the probability that a state with Energy E is occupied by an electron is given by the Fermi distribution [33, Somerfeld model] displayed in Figure 2.1:

$$f(E, E_f, T) = \frac{1}{\exp\left((E - E_f)/k_B T\right) + 1} \tag{2.1}$$
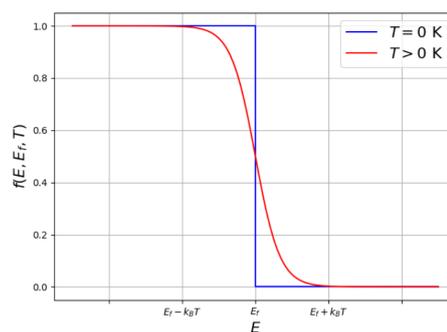
Figure 2.1: Fermi distribution for electrons (actually fermions) at temperature T.

Note that as $T \rightarrow 0$, every state with $E \leq E_f$ will have a probability of 1 of begin occupied, and states with $E > E_f$ have a zero probability of being occupied, confirming the idea of a sharp cutoff of occupied energy states. Furthermore, E_f in Equation 2.1 will always denote the Fermi *level*, since the Fermi energy is only defined for $T = 0$ K and Equation 2.1 is valid for $T > 0$ K.

Equation 2.1 gives that at $T > 0$ K, there is a non-zero change that energy levels with an energy above the Fermi level are occupied. As mentioned before, materials often have a Fermi energy in the order of electron volts. Thermal excitations come with an energy of $k_B T$, with $k_B = 1.38 \cdot 10^{-23}$ J/K the Boltzmann constant,

5

or an energy of roughly 0.1 meV per Kelvin. This means that at T = 300 K the system gained only 30 meV hence the distribution of electrons changed only slightly and locally around the Fermi energy, due to thermal excitations.

2.1.2. Band structures

Free electrons in a box can be described by their (plane) wave function that solves the Schrödinger Equation [9, Eq. 2.20]:

$$E\Psi = -\frac{\hbar^2}{2m}\nabla^2\Psi \tag{2.2}$$

Using the boundary conditions that the wave function must vanish at the boundaries, the solution is know to be $\Psi(x) = \exp(i\mathbf{k}\cdot\mathbf{r})$ with corresponding eigenvalues: $E = \frac{\hbar^2\mathbf{k}^2}{2m}$ (Figure 2.2a) with $\mathbf{k}$ the wave number of the electron. $\mathbf{k}$ is related to the electron momentum by the de Broglie equation $\hbar\mathbf{k} = \mathbf{p}$ with $\mathbf{p}$ the momentum. The assumption made here is that electrons are particles that are not interacting with atomic nuclei [33, Sommerfeld model].

Within metals, the electrons do in fact interact with other nuclei. The 1-dimensional Schrödinger Equation can be solved, assuming the on-side energy (the energy of the electron at the atom it is residing) to be E_0 and assuming an electron can tunnel to the neighbouring nucleus at a cost of energy $-t$, the hopping integral [33, Tight-binding model]. The sign of the hopping integral is due to the negative coulomb potential. Since a monolithic chain of atoms is considered, the hopping integral is equal for both neighbours. The corresponding Schrödinger equation is:

$$E\phi_n = E_0\phi_n - t\phi_{n-1} - t\phi_{n+1} \tag{2.3}$$

To find the available energy states, Equation 2.3 needs to be solved. The proposed solution is $\phi_n = A\exp(ikx_n)$ with $x_n = na$, the locations of the nuclei. $A = 0$ gives the trivial solution is therefor it is assumed $A \neq 0$. Substituting the proposed solution and dividing Equation 2.3 by $A\exp(ikna)$, the energy dispersion relation, Equation 2.4, is obtained:

$$E = E_0 - 2t\cos(ka) \tag{2.4}$$

The available energy states for the system are now limited between $E_0 - 2t$ and $E_0 + 2t$, instead of 0 and ∞ like for the free-electron model (Figure 2.2b). The range of energies that can be occupied by electrons is know as the *band structure*. These band structures arise because bring similar atoms, with similar electron distributions, are brought close to each other. When they are brought close enough some energy states will overlap, meaning that two fully filled states with two electrons (four electrons total) try to occupy the same state. By the Pauli exclusion principle, only two electrons can be in the same state hence what is tried to be achieved, four electrons in a single state, is not possible. The accommodate the four electrons both energy states shift slightly in energy, one state increases while the other decreases [43, Section 37.3, Band Theory]. If another two atoms are put close by the same problem arises, hence the energy states slightly shift their energies again. This means there are four available energy states, where a single atom only has one available energy state. If $2N$ atoms are put together, $2N$ states are available where a single atom would have only one state. The range of energies that correspond to the same state are know as a band, hence the name *band structure*.

From Peierls theorem [17] it is know that an equally spaced monolithic chain is unstable, the material undergoes a Peierls transition and effectively there is a different hopping between adjacent nucleus sites ($-t_1$ and $-t_2$). The Schrödinger equation for this system can be obtained and solved [33, Many atoms per unit cell]. The obtained energy dispersion is now given by:

$$E = E_0 \pm \sqrt{t_1^2 + t_2^2 + 2t_1 t_2 \cos(ka)} \tag{2.5}$$

The available energy states (assuming $t_1 > t_2$) are now located between $E_0 - t_1 - t_2$ and $E_0 - t_1 + t_2$, but also between $E_0 + t_1 - t_2$ and $E_0 + t_1 + t_2$. There are no available states between $E_0 - t_1 + t_2$ and $E_0 + t_1 - t_2$! The difference between the latter two, $2(t_1 - t_2)$, is known as the *band gap* (Figure 2.2c). It is the minimal energy electrons need to gain to go from the low-energy band to the high-energy band. Materials where the Fermi level is located in a band are known as metals. Materials where the Fermi level lays within the band gap, that are not insulators, are known as *semiconductors*. The distribution of electrons therefor has a major influence on the behavior of materials, metal like, insulator or semiconductor like. The splitting of the bands arises when (similar) atoms are brought together close enough that the orbitals of the electrons overlap.

Real materials have much more complicated band structures, and band gaps often arise. Common everyday semiconductors are for instance Silicon and Germanium (alloys). Materials with a large band gap are often referred to as *insulators*, simply because only at very large potential differences electrons can gain enough energy to jump the band gap. The different band structures as discussed before are shown in Figure 2.2:
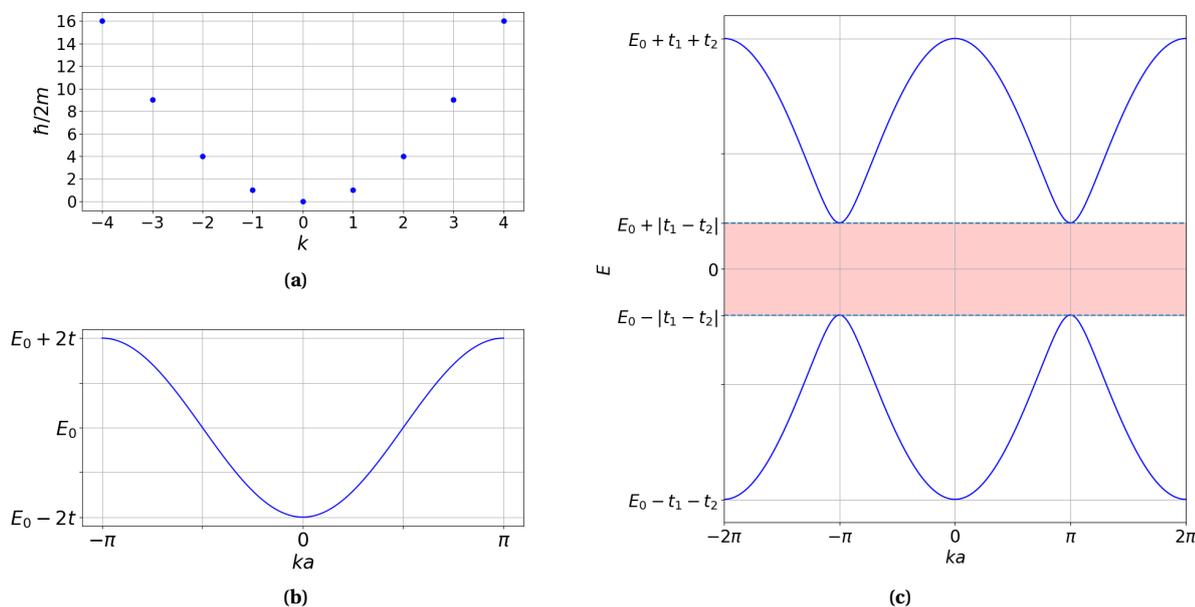
Figure 2.2: Electron dispersions for; Figure 2.2a, free electrons, x-axis shows the product of the wave number and the lattice constant ka. y-axis shows the energy in terms of $\hbar/2m$; Figure 2.2b, a monolithic chain, x-axis shows the product of the wave number and the lattice constant ka. y-axis shows the energy of the highest energy electrons; Figure 2.2c, Peierls transition dispersion, x-axis shows the product of the wave number and the lattice constant ka. y-axis shows the energy of the electrons. The highlighted area shows the forbidden zone known as the band gap.

A typical band structure consists of multiple bands with multiple band gaps. Let us consider a system of N atoms. In the ground state ($T = 0$ K), the orbitals of the electrons are filled from the lowest energy to the point where there are no electrons left. If the outer most shell is not completely filled with electrons, then the electrons in the outer shell are called *valence electrons*. Every atom has q electrons. Every atom contributes one state to a band in a band structure, that can be occupied by two electrons. Therefor, atoms with an odd number of electrons per unit cell have halve filled electron bands (at $T = 0$ K, and roughly too at $T > 0$). The Fermi energy therefor lays inside an electron band, these materials are called metals [33, Band structures in 2D]. Atoms with an even number of electrons in a unit cell are either insulators or semiconductors, depending on the size of the band gap.

Now the bands for $T > 0$ K are considered. The highest band that is not fully filled is called the *conduction band* while the band below is called the *valence band*. There can still be bands below the valence band, but the electrons in those states are unlikely to interact with other particles since there are very few states still available in the bands below the valence band.

2.1.3. Semiconductor doping

Semiconductors are often solid state (compound) materials that have an even number of electrons in their outer orbital. Such as Silicon (2 electrons), Germanium (2 electrons) or Gallium Arsenic (4 electrons). Previously mentioned elements are group III, IV or V elements, meaning they have either 3, 4 or 5 electrons in their outer shell (or 1, 2 or 3 in their outer orbital). Furthermore, semiconductors have the Fermi level in a band gap. Only the relative position of the Fermi level can be changed, by altering the band structure.

Suppose a group V element (like Phosphorus, 5 electrons in the outer shell) is added to the bulk of a Silicon semiconductor. Phosphorus has one electron more in its outer orbital than Silicon. The extra electron is only weakly bound to the Phosphorus atom, and simple thermal excitations can pull the electron away from the nucleus and an applied electric field can then cause the electron to move through the material [33, Doping and devices]. For an electron to move through the material bulk, it has to be in the conduction band since the current flows at (the bottom of) the conduction band (Section 2.1.4). Since this electron was promoted

to the conduction band by thermal excitations, this means the extra state added by the Phosphorus has to be close, but below the conduction band. Thus, a state just below the conduction band has been added to the system, which means the state with a chance of being occupied of 50% is now increased hence the Fermi level is increased. Since an electron is added to the system, the charge is carried by negatively charged particles. These type of doped semiconductors are called *N-type semiconductors*. Semiconductors with a group III dopant are called *P-type semiconductors* for a similar reason.

2.1.4. Current carrying electrons

As discussed before, the Fermi level lays within a band of the band structure for metals. This means that electrons can easily be excited by electric fields and carry an electronic current. The current will therefor flow at the Fermi level.

For semiconductors, the Fermi level lays within the band gap. This means that the states below the valence band are likely filled, and the valence band only has a few available states that are simply denoted as holes. There is a chance that some electrons are located in the conduction band as predicted by Equation 2.1. The electron(s) in the conduction band have a lot of states available since that band is only occupied by a few electrons, at least there are more and easier accessible states than the states in the valance band. The current in a semiconductor will therefor be carried by the electrons in the conduction band. The conduction band only hosts a few electrons, hence the current flows at the energy of the bottom of the conduction band, denoted by E_c.

2.2. Thermoelectric effect

The thermoelectric effect sits at the basis of the working principles of thermocouples [29, section 7.2.4.1] and solid state heat pumps and generators [22, page 73]. Ludstrom and Jeong [22] describe four thermoelectric transport coefficients that govern two thermoelectric transport equations given by:

$$J = \sigma E - \sigma S \frac{dT}{dx} \tag{2.6}$$

$$q = \Pi J - (\kappa_n + \kappa_p) \frac{dT}{dx} \tag{2.7}$$

In which J is the electric current density in A/m^2, T the temperature in K, E is the electric field in N/C and q the heat flux in W. The four thermoelectric coefficients are σ, S, Π, κ_n which are the thermal conductivity, Seebeck coefficient, Peltier coefficient and the electronic conductivity respectively. κ_p in Equation 2.7 is the phononic thermal conductivity, which is not a thermoelectric phenomena. Equation 2.6 (Section 2.2.1) is essentially a modified version of Ohm's law, in which a temperature gradient across a (semi)conductor reduces the electric field within the material (Seebeck effect). Equation 2.7 (Section 2.2.2) is essentially Fourier's law of cooling, but with an additional term ΠJ which implies that a heat current is accompanying an electric current (Peltier effect).

2.2.1. Seebeck effect

The Seebeck effect states that, in conductors, a thermal gradient is accompanied by an electric potential difference. This effect can be represented by Equation 2.8:

$$\Delta V = -S \Delta T \tag{2.8}$$

Physically, it is easy to understand where the Seebeck effect comes from. Let us consider an electric conductor that is heated on one side only. A thermal gradient will form across the length of the conductor. As long as the materials are not melting or evaporating, the atoms are confined to their position in the conductor but the electrons can move easily through the material since the material is a conductor. At higher temperatures the electrons will therefor diffuse towards the colder side creating a charge differential across the conductor. The hot side will become slightly positively charged and the cold side slightly negative, creating a potential difference over the conductor.

This seems to be a contradiction. There is an electric field within the conductor, but the charge carriers do not move to cancel the field. In fact, they let a potential difference build up. Whereas an electric potential difference is often pointed towards as the driving force behind current flow, it is actually the difference in Fermi levels that causes a current to flow [22, Section 2.2]. This explains why at first, when there is only a temperature difference between the two ends, the electrons flow down the energy slope since they feel a force

$F = -\nabla E$ as the energy of the hot side is $\sim k_B \Delta T$ higher than the cold side. Eventually, there is an electrostatic force $q\Delta V$, with q the electric charge of the electrons, that counters the thermal energy giving an equilibrium distribution of electrons. This suggests:

$$k_B \Delta T = qV = -qS\Delta T \implies S = -\frac{k_B}{q} \approx -100\mu V/K$$

for all materials. From measurements it is known that this is not true, it especially seems semiconductors have a higher Seebeck coefficient than ordinary metals. This explanation is given by Ludstrom and Jeong [22, Section 4.2]:

Whenever the two sides of the conductor are in equilibrium, the Fermi functions at the sides should be equal. One side is at a temperature T_1 while the other side at a higher temperature T_2 with $T_2 > T_1$, and an electrostatic builds up within the conductor giving Fermi functions:

$$f_1(E, E_f, T_1) = \frac{1}{\exp\left((E - E_f)/k_B T_1\right) + 1}$$
$$f_2(E, E_f, T_2) = \frac{1}{\exp\left((E + qV - E_f)/k_B T_2\right) + 1}$$

(2.9)

For the conductor to be in equilibrium the two functions must be equal. This must mean that:

$$\frac{E - E_f}{k_B T_1} = \frac{E + qV - E_f}{k_B T_2}$$

(2.10)

Which after some rearranging gives:

$$V = \frac{E - E_f}{q T_1}(T_1 - T_2) = -S\Delta T \implies S = -\frac{E - E_f}{q T_1}$$

(2.11)

Which states that the Seebeck coefficient is proportional to the difference between the energy of the charge carriers and the Fermi level. For metals this difference is nearly 0 since current roughly flows at the Fermi level, but for semiconductors the current flows just above the bottom of the conduction band which can be a significant amount of energy above the Fermi level. N-type semiconductors, where the current flows above the Fermi level, the Seebeck coefficient must be negative and for p-type semiconductors it must be positive since the charge is carried by the holes that flow below the Fermi level. As measured by Hyun et al. [14], the Seebeck coefficient of a particular n-type Silion is -170 $\mu V/K$ and for p-type Silicon 153 $\mu V/K$.

A practical use of the Seebeck effect is measuring temperature using thermocouples of thermopiles. Since different materials have different electron densities and band structures, different materials will have a different potential difference corresponding to the same temperature difference. If two wires of different materials (the top and bottom wire of Figure 2.3a) have one junction in common (the black dot in the right of Figure 2.3a) with the other ends at the same temperature but different than the junction temperature, both wires will develop potential differences that are unequal in magnitude. This means there now is an electric potential difference between the two ends of the total wire. Placing multiple junctions of alternating wire in series, a *thermopile* (Figure 2.3b) is obtained. The total potential difference between the two ends of the thermopile is equal to the temperature difference over a single thermocouple, multiplied my the amount of thermocouples. If one of the two temperautres in Figure 2.3b is known, the other can be determined using the Seebeck effect.

2.2.2. Peltier effect

As mentioned in Section 2.2.1, a temperature difference between two points in a conductor can cause an electric potential difference to build up within the conductor, know as the Seebeck effect. A similar but opposite effect happens too: when an electric potential difference is applied to a (semi)conductor, a temperature difference manifests itself across the the (semi)conductor. To be more precise, an electrical current is accompanied by a heat current since the current carrying electrons also carry heat across the conductor. Effectively there is an electric potential difference that causes an electric current, that is accompanied by a heat current, that causes a temperature difference. The Peltier effect can be captured by Equation 2.12:

$$Q = \Pi I$$

(2.12)

The Peltier effect is closely related to the Seebeck effect. This is easily shown by considering a horizontal conductor with metal contacts at both ends. If there is an applied potential difference ΔV over the conductor
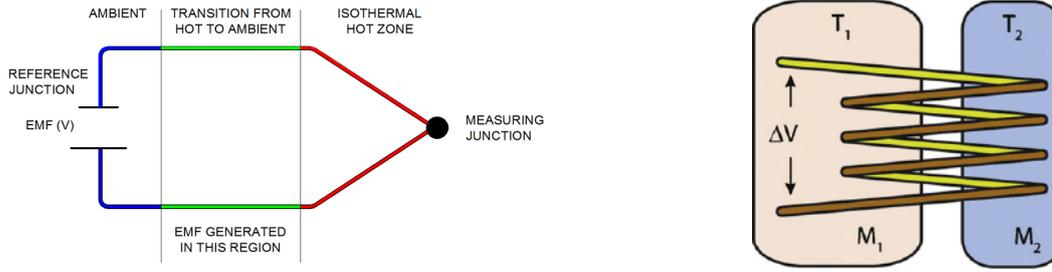
(a) Schematic of a thermocouple. Figure taken from Litteaur [21, Figure 7]

(b) Schematic of a thermopile consisting of four thermocouples. Figure taken from Thapaa et al. [34, Fig 1].

Figure 2.3: Figure 2.3a shows a schematic of a thermocouple. Figure 2.3b shows the schematic of a thermopile, an array of thermocouples.

such that the current flows from left to right, then there is also a heat current flowing from left to right (by the Peltier effect), cooling down the left contact, and heating up the right contact. Now there is a temperature gradient from right to left, creating a potential difference due to the Seebeck effect, that counters the applied potential lowering the current. This indicates that the Peltier and Seebeck effect are interconnected, as explicitly showed by Ludstrom and Jeong [22, Section 4.3]. It is important to note that altough the Peltier effect is a property of the semi-conductor, it requires a metal junction.

Ludstrom and Jeong [22] provide a way to calculate the Peltier coefficient by means of a physical insight. Let us consider an n-type semiconductor, where the bottom of the conduction band is above the Fermi level, that is on both ends connected to metal contacts that have a potential difference between them as depicted schematically in Figure 2.4a.
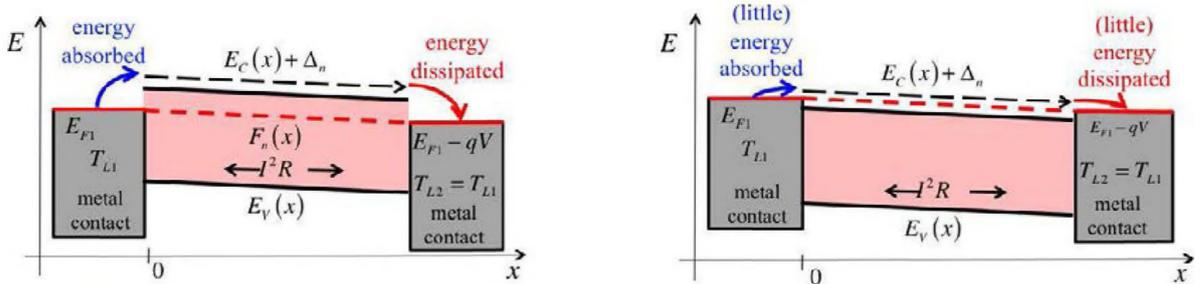
(a) Band structure of a (lightly doped) semiconductor with metal contacts.

(b) Band structure of a heavily doped semiconductor, essentially acting like a metal, with metal contacts.

Figure 2.4: Band structures of different metal contacts. The dashed red line indicates the Fermi level throughout the contacts. The red zones indicate the band gaps. E_{Fi} are the respective Fermi levels in the metal contacts, $E_c + \Delta_n$ is the energy of the bottom of the conduction band where the electrons flow, for a lightly doped semiconductor $\Delta_n = 0$, E_v is the energy of the top of the valence band, $F_n(x)$ is the electron flux. Figures taken from Ludstrom and Jeong [22].

The metal contact on the right in Figure 2.4a is at a higher potential, which lowers its energy due to the electron charge being negative. This causes the electrons to flow from the left contact to the right contact via the semiconductor. But how do the electrons in the left metal contact, where the current flows at the Fermi level, enter the conduction band of the semiconductor where the current flows at the energy of the bottom of the conduction band? The answer is that at finite temperatures, the electron distribution is given by the Fermi function as given by Equation 2.1 and Figure 2.5 indicating that there is a nonzero chance that there is an electron that occupies an energy state that is higher than the conduction band energy of the semiconductor. Given the vast amounts of electrons in metals and semiconductors (10^{18} cm^{-3} [33, Doping and devices]), it is very likely that an electron will occupy a state with an energy high enough to enter the semiconductor[1].

The electron that has entered the semiconductor now leaves a hole in the electron distribution in the metal contact creating a highly non-equilibrium distribution of the electrons. Now there is inelastic scattering between electrons and lattice vibrations (phonons), which essentially means electrons take up thermal energy

---

[1] Ludstrom and Jeong [22] mention additional *band bending* in the metal-semiconductor contact, creating an additional (Schottky [40]) energy barrier for the electrons to overcome.
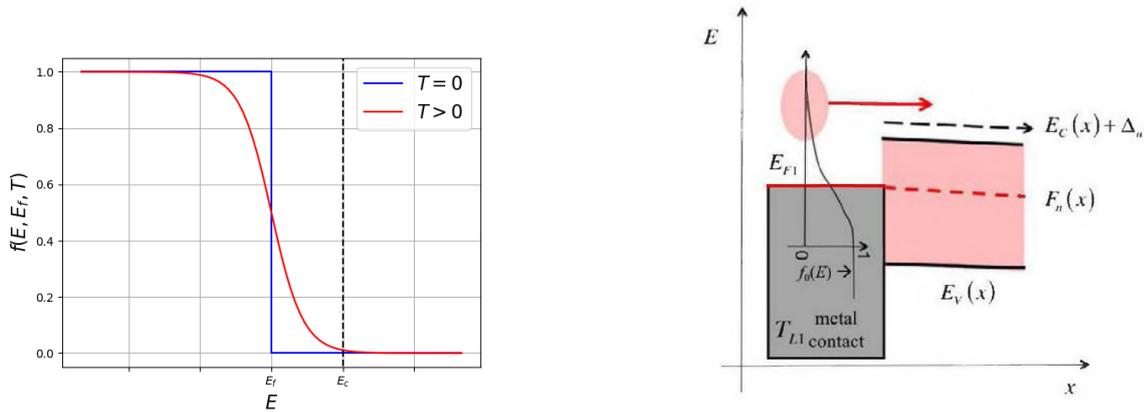
(a) Electron distribution within the metal contact. Both Fermi level and the bottom of the conduction band are indicated.

(b) Fermi function projected on top of the metal contact. Showing high energy electrons enter the semiconductor.

Figure 2.5: Fermi distribution at finite temperature and in the metal contact. Figures taken from Ludstrom and Jeong [22].

from the lattice to move lower energy electrons into the created electron vacancy. Since this energy is coming from the lattice vibrations, the entire metal contact loses some thermal energy which cools the contact down [22, Section 4.3]. Figure 2.4a shows that at the other metal contact, electrons with an energy $E_c (> E_f)$ enter the metal contact, again creating a non-equilibrium electron distribution causing the electrons to give up some of their energy to the lattice vibrations due to the energy-phonon scattering, heating up the contact!

In p-type semiconductors this process is the other way around. The charge carriers are (positively charged) holes, that move with the electric current, from the valence band into the conduction band. The jump from a hole to the conduction band is accompanied, simultaneously, by a high energy electron giving of some of its energy to the lattice to repopulate the vacancy. This means the metal contact where the charge carriers flow into the semiconductor is now heating up (instead of cooling down like in the n-type semiconductor). At the other metal contact the hole takes a place in the Fermi distribution corresponding to its energy E_c which is accompanied, simultaneously, by a high energy electron flowing into the semiconductor, cooling down the metal contact. Table 2.1 gives an overview of the Peltier effect:

Table 2.1: Characteristics of charge flow and temperature in n-type and p-type semiconductors.

n-type semiconductors	low potential contact	high potential contact
Temperature	Increases	Decreases
Charge carriers flow	into the semiconductor	out of the semiconductor

p-type semiconductors	low potential contact	high potential contact
Temperature	Decreases	Increases
Charge carriers flow	out of the semiconductor	into the semiconductor

An electric current is now accompanied by a heat current that is proportional to the movement of the electrons (or holes), which is know as the electric current. The proportionality constant, Π, is know as the Peltier element and is calculated as:

$$Q = (\text{Energy carried per electron}) \cdot (\text{electron rate}) = (E_c - E_f) \cdot (I/q) = \Pi I \implies \Pi = \frac{E_c - E_f}{q} \qquad (2.13)$$

Where $\Pi = ST$ which is the fundamental Kelvin relation [22, Section 4.3] and it explains the connection between the Peltier effect and the Seebeck effect.

Just as with the thermopile, that enhances the effect of thermocouples, pairs of metal-semiconductor contacts can be placed in series to enhance the Peltier effect. These devices are known as Peltier elements and have a structure similar to Figure 2.6.

From Figure 2.6 it is clear that a Peltier element consists of a long array of metal-semiconductor pairs, where the semiconductor is alternating between n-type and p-type. This alternating is necessary since if two
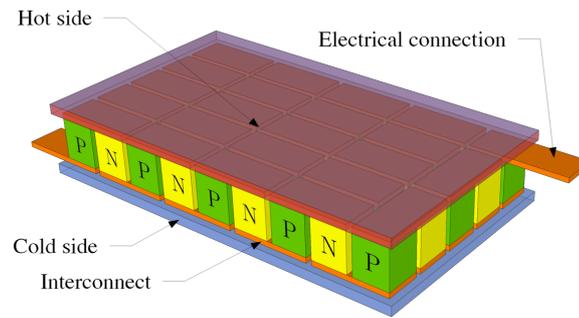
Figure 2.6: Schematic of a Peltier element [24].

n-type semiconductors are connected to a metal contact (*"Electrical connection"* in Figure 2.6) then charge will flow out of one semiconductor, and via the metal into the other semiconductor hence there is both the Peltier heating and Peltier cooling effect on the same piece of metal. If the two semiconductors connected to the same piece of metal are dissimilar (n-type and p-type) the metal contact is either heated or cooled depending on the direction of the current.

The pieces of metal connecting two semiconductors should be electrically insulated from other pieces of metal. On top of all the pieces of metal is another piece of (ceramic) material that covers the total surface area of the Peltier element on either side. The Peltier element is now a piece of semiconductor technology that, when a potential difference is applied, heats up on one side and cools down on the other. Essentially, a Peltier element is a high energy density (i.e. 42 kW/m^2 [28]), small size heat pump.

Peltier element naming convention
Peltier elements often have a red and black wire, and a piece of text written on the side that heats up of the red wire is connected to a positive potential and the black wire to ground. The text has the form TEC1-xxxyy or TEG1-xxxyy where *TEC* is short for *thermoelectric cooler* and *TEG* is short for *thermoelectric generator*, 1 indicates the number of stages (essentially how many Peltier elements are stacked on top of each other), *xxx* is the number of semi-conductor pairs in the element and *yy* is the rated current that can flow through the device.

2.2.3. Thomson effect
When both an electric current and a temperature difference are present, a third thermoelectric effect known as the Thomson effect [20, Section 8.3.1.3] is present. The Thomson effect states that in the case of an electric current and a temperature difference the heat flow is given as:

$$\dot{Q} = -k\left(I \cdot \frac{dT}{dx}\right) \tag{2.14}$$

with $k = \frac{d\Pi}{dT} - S$, using the second Kelvin relation $\Pi = ST$ it is found that $k = T\frac{dS}{dT}$. The Seebeck coefficient S is assumed to be constant in temperature range of operation, giving a Thomson coefficient of 0. In practise, the Thomson effect is much smaller than the Peltier effect and is therefor often neglected [20, Section 8.3.1.3].

2.3. Heat transfer
Heat transfer can be modelled using the analogy with electrical circuits as done in Mills and Coimbra [25, Section 1.4]. In this analogy, temperature difference ΔT is analogous with electric potential difference ΔV, heat current Q is analogous to electric current I and the product of the overall heat transfer coefficient and total heat transfer are $\frac{1}{UA}$ is analogous to electric resistance R:

$$I = \frac{\Delta V}{R} \iff Q = (UA)\Delta T \tag{2.15}$$

This means that multiple resistances in series, like two different pieces of material, have a total resistance that adds up. The heat transfer resistance of a plate with area A, thickness d and thermal conductivity k is given by:

$$(UA)^{-1} = R = \frac{d}{kA}$$

The resistance of two different materials in series is then given by:

$$R_{tot} = \frac{d_1}{k_1 A_1} + \frac{d_2}{k_2 A_2}$$

which gives a total heat transfer coefficient:

$$UA = R^{-1} = \frac{k_1 k_2 A_1 A_2}{d_1 k_2 A_2 + d_2 k_1 A_1}$$

The heat transfer resistance of a circular symmetric piece of material can be found by integrating the radial heat equation and is given by:

$$R = \frac{\ln(r_o/r_i)}{2\pi L k}$$

with r_o the outside radius, r_i the inside radius and L the length of the material. The resistance to heat transfer of a convective boundary with heat transfer coefficient h is given by.

$$R = \frac{1}{hA}$$

A circular pipe carrying a hot fluid, exposed to cooler air has a total heat trasnfer resistance:

$$UA = \left(\sum_i R_i\right)^{-1} = \frac{1}{\frac{1}{h_{inside}2\pi r_i} + \frac{\ln(r_o/r_i)}{2\pi k} + \frac{1}{h_{outside}2\pi r_o}} \tag{2.16}$$

2.3.1. Heat transfer coefficients from dimensionless numbers

In order to determine how much energy is transferred due to a temperature difference, the product of the total heat transfer coefficient U and the total heat transfer area A needs to be known. Whereas for solids, the thermal conductivity is readily available or can be easily calculated from experiments, the convective heat transfer coefficient in fluids depends on more than temperature (and pressure).

Heat transfer coefficents can be determined from *Nusselt relations* [25, Table 4.3]. The Nusselt number is defined as the conductive heat transfer resistance over the convective heat transfer resistance:

$$\text{Nu} = \frac{\frac{d}{kA}}{\frac{1}{hA}} = \frac{hd}{k} \tag{2.17}$$

in which h is the heat transfer coefficient in W/m^2, k the thermal conductivity in W/mK of the fluid and d a characteristic length scale in m. Thus the Nusselt number is a dimensionless quantity. If the Nusselt number for a certain situation is known, the heat transfer coefficient h can be calculated from Equation 2.17 using $h = \text{Nu}\frac{k}{d}$.

Other dimensionless number that are often used in heat transfer are the Reynolds number, the Prandtl number and the Rayleigh number:

$$\text{Re} = \frac{\rho U^2}{\mu U/D} = \frac{\rho U D}{\mu} \tag{2.18}$$

$$\text{Pr} = \frac{\mu/\rho}{k/\rho c_p} = \frac{c_p \mu}{k} \tag{2.19}$$

$$\text{Ra} = \frac{l^2/\alpha}{\eta/\rho\beta\Delta T l g} = \frac{\rho\beta\Delta T l^3 g}{\mu\alpha} \tag{2.20}$$

With ρ the fluid density in kg/m^3, U the velocity in m/s, D the hydraulic diameter in m, μ the dynamic viscosity in Pa·s, c_p the specific heat of the fluid in J/kgK, k the thermal conductivity in W/mK, β the coefficient of thermal expansion in K^{-1}, ΔT the temperature difference in K over a characteristic distance l in m, g the gravitational acceleration in m/s^2, $\alpha = k/\rho c_p$ the thermal diffusivity in m^2/s. Reynolds number is used to characterise the flow in terms of laminar of turbulent flow. The Prandtl number gives the ratio of the momentum transfer rate to the heat transfer rate. The Rayleigh number gives the ratio of the time scale for thermal transport via diffusion to the time scale for thermal transport via convection. The Prandtl number can be calculated using temperature dependant relations for μ [41] and k [18], the specific heat capacity of water is taken constant at $c_p = 4.18$ kJ/kg [36].

The hydraulic diameter is used to characterise non-circular channels as in Equation 2.21 [37, eq. 5.26]:

$$D = \frac{4A}{P} \tag{2.21}$$

Such that the hydraulic diameter of a circular tube is $D = 4\pi r^2/2\pi r = 2r$.

3

Numerical Model

The goal of this project is to investigate if Peltier elements can be used to increase the temperature of a water flow by extracting thermal energy from another flow. This will be done by placing 10 Peltier elements in line in a *Peltier Heat Exchanger* (PeHEX) schematically shown in Figure 3.1. Water flows on both side of the Peltier elements, one of the flows will be heated while the other is cooled.
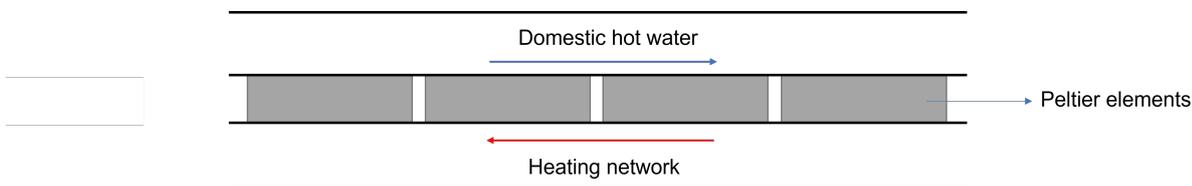
Figure 3.1: Schamtic of the Peltier Heat Exchanger as shown with 4 Peltier elements. The domestic hot water is heated using thermal energy from the heating network, extracted by the Peltier elements. The flows operate in counter current conditions.

To model the Peltier Heat Exchangers, the performance of the individual Peltier elements has to be modelled. More precisely, the behavior of the individual Peltier elements, when an electric potential difference is applied, has to be modelled. This is done in three steps:

1. The internal temperature distribution for the Peltier Elements will be derived by considering a thermal energy balance on the interior of the Peltier element. From there, the heat flux at the element boundaries will be determined.

2. Using the heat flux at the element boundaries, the interaction between the Peltier element and the environment (heat absorption/rejection, electric potential) will be derived.

3. The individual Peltier element models are added together to analyse the performance of the Peltier Heat Exchangers.

Additionally the two dimensional heat equation was solved on the cross sectional area of the Peltier heat exchanger to get an idea for the heat transfer area.

3.1. Internal temperature analysis

The Peltier element is essentially a heat pump where the heat is carried by the electric charge carriers and the atomic lattice vibrations (phonons [33, Chap. Debye model]). The heat flux can be calculated using Equation 2.7. The Seebeck coefficient of the material is assumed to be constant with temperature in the temperature range of interest. Also, the second Kelvin relation $\Pi = ST$ will be substituted in Equation 2.7 and it will be multiplied with the cross sectional area A to obtain the thermal current in Equation 3.1:

$$\dot{Q} = IST - kA\frac{dT}{dx} \tag{3.1}$$

The Peltier elements will be used as a simple heat pump, moving energy from the district heating networks to the domestic (hot) water lines. Only where the ceramic coatings take up or reject heat, at the Peltier element's boundaries, the heat flux has to be evaluated. To this end the Peltier element is analyzed via a slice of the interior of the Peltier element as depicted in Figure 3.2. Figure 3.2a also indicates the boundary conditions of the temperature distribution on the outside interfaces of the Peltier element.

(a)

(b)
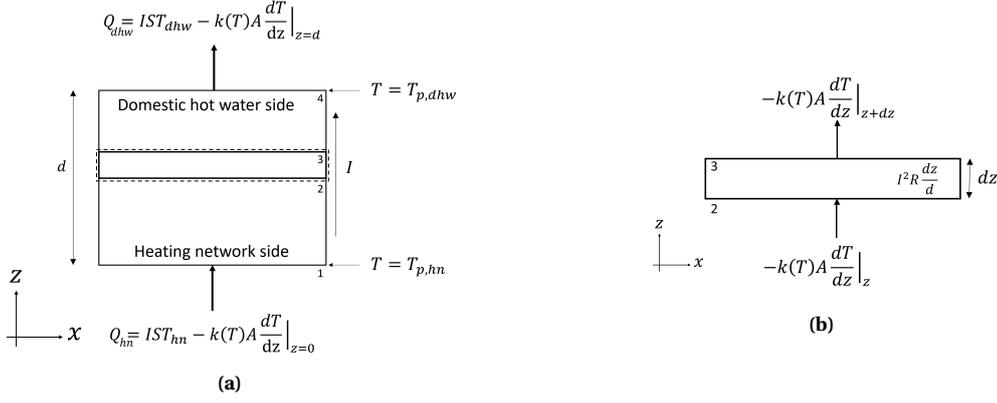
Figure 3.2: Figure 3.2a shows the schematic of a horizontal Peltier element as they are shown in Figure 3.1. The interfaces 1 and 4 in Figure 3.2a represent the boundary conditions $T(z=0) = T_{p,hn}$ and $T(z=d) = T_{p,dhw}$ respectively. Interfaces 2 and 3 correspond to the interfaces in Figure 3.2b. Figure 3.2b shows the energy balance from the dashed box from Figure 3.2a in detail.

Balancing the slice from Figure 3.2 gives:

$$-\left(k(T)A\frac{dT}{dz}\right)\Big|_z + \left(k(T)A\frac{dT}{dz}\right)\Big|_{z+dz} + I^2R\frac{dz}{d} = 0$$

Collecting likewise terms and dividing by dz the differential equation that governs the temperature distribution within a Peltier element is given by Equation 3.2:

$$\frac{d}{dz}\left(k(T)A\frac{dT}{dz}\right) + \frac{I^2R}{d} = 0 \tag{3.2}$$

As mentioned by Ludstrom and Jeong [22], the thermal conductivity in semiconductors is rather low, $1 \sim 10$ W/mK, when compared to metals and can vary significantly with temperature. Experimental data of the temperature dependence of the thermal conductivity of the Peltier element used in this research is given by Figure 3.3.
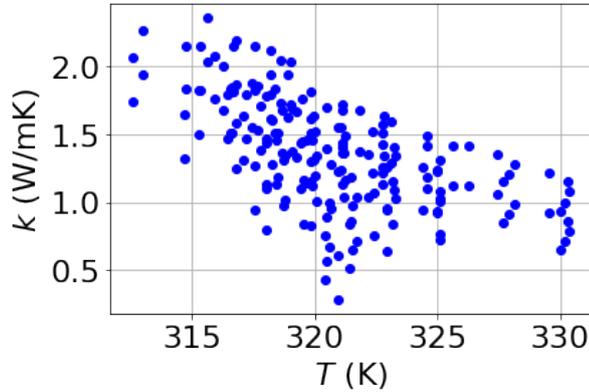
Figure 3.3: Thermal conductivity of the Peltier element as function of temperature. The thermal conductivity has halved from 2 W/mK to 1 W/mK over the range of 15 K.

The total thermal conductivity is the sum of the electronic and phononic thermal conductivity. The thermal conductivity due to charge carriers is linearly proportional to temperature, as described by Ludstrom and Jeong [22, Equation 4.7], whereas the effect of lattice vibrations is inversely proportional to temperature according to Holland [12, Equation 8]. The total thermal conductivity can be modelled by Equation 3.3:

$$k(T) = k_1 T + \frac{k_2}{T} \tag{3.3}$$

However, Formula 3.3 has a root at $T_{k=0} = \sqrt{-k_2/k_1}$. Typical values for k_1 and k_2 (from the experimental observations, Chapter 6) are $k_1 = -0.033$ and $k_2 = 3830$ giving $T_{k=0} = 67$ °C. This is an issue since it is very

likely that the Peltier elements will actually attain a temperature higher than 67 °C to get the domestic hot water to the desired temperature of 60 °C. This problem could be solved by using a simple power law to model the thermal conductivity $k \sim \alpha T^\beta$, as proposed by Palankovski [27, sec. 3.2.3]. The downside to this method is that due to the parameter fitting of an exponent (β), which at only 10% relative error can cause big differences in the value for $k(T)$. The proposed solution is to use a thermal conductivity of the form given by Equation 3.4:

$$k(T) = k_1 \exp(k_2 T) \tag{3.4}$$

Formula 3.4 has the advantage of never giving a negative thermal conductivity, as well as never giving a thermal conductivity of 0 W/mK. A comparison of the two models for the thermal conductivity is given in Figure 3.4.
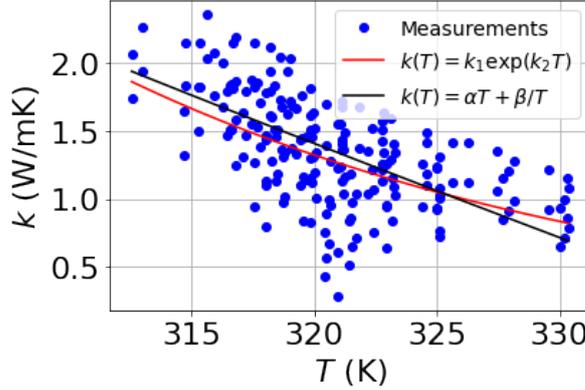
Figure 3.4: Thermal conductivity with fitted equations 3.3 and 3.4.

The ODE given by Equation 3.2 is subject to the boundary conditions $T(z = 0) = T_{p,hn}$ and $T(z = d) = T_{p,dhw}$ where $T_{p,hn}$ indicates the temperature of the Peltier element at the heating network side and $T_{p,dhw}$ indicates the temperature of the Peltier element at the domestic hot water side. The goal is to find an expression for $-\left(k(T)A\frac{dT}{dz}\right)$ such that the the total heat transfer rate can be calculated at the boundaries of the Peltier element. The heat transfer rates at the boundaries can be calculated using Equation 3.1 with $\dot{Q}_{hn} > 0$ for $T_{hn,out} < T_{hn,in}$.

$$\begin{aligned}
\dot{Q}_{dhw} &= IST_{p,dhw} - k(T)A\frac{dT}{dz}\Big|_{z=d} \\
\dot{Q}_{hn} &= IST_{p,hn} - k(T)A\frac{dT}{dz}\Big|_{z=0}
\end{aligned} \tag{3.5}$$

The full derivation of the solution can be found in Appendix B. The final solution is given by:

$$-(k(T)A\frac{dT}{dz}) = -\frac{I^2 R}{2}\left(1 - \frac{2z}{d}\right) - k\left(\frac{T_{p,dhw} + T_{p,hn}}{2}\right)\frac{A}{d}(T_{p,dhw} - T_{p,hn}) \tag{3.6}$$

Such that the heat fluxes at the Peltier element's boundaries are given by Equation 3.7:

$$\begin{aligned}
Q_{dhw} &= IST_{p,dhw} + \frac{I^2 R}{2} - k\left(\frac{T_{p,dhw} + T_{p,hn}}{2}\right)\frac{A}{d}(T_{p,dhw} - T_{p,hn}) \\
Q_{hn} &= IST_{p,hn} - \frac{I^2 R}{2} - k\left(\frac{T_{p,dhw} + T_{p,hn}}{2}\right)\frac{A}{d}(T_{p,dhw} - T_{p,hn})
\end{aligned} \tag{3.7}$$

From a total energy balance equation the input power can be calculated to be:

$$P_{in} = Q_{dhw} - Q_{hn} = IS(T_{p,dhw} - T_{p,hn}) + I^2 R \tag{3.8}$$

Since the electric power input is given by $P_{in} = IV$, it is clear that the total applied electric potential difference is distributed over the resistive load of the Peltier element and the *Seebeck load* ($= S(T_{p,dhw} - T_{p,hn})$).

3.2. Estimating thermoelectric parameters

From previous analysis it is clear that the parameters S, R and $k(T)$ are important for the determination of the performance of a Peltier element. A datasheet is usually provided with a Peltier element, informing what the specification of the device are. Figure 3.5 shows the graphs on the datasheet for the type of Peltier element used in this research (TEC1-12706 by Hebei I.T. (Shanghai) Co., Ltd. [10]). To the day of writing the author has not found a datasheet yet, where the relevant thermoelectric parameters, S, R and $k(T)$ are simply listed. The parameters have to be extracted from the datasheets as they are implicitly represented in the graphs of the datasheets.

(a) Datasheet for TEC1-12706 at $T_{p,dhw} = 25\,°C$

(b) Datasheet for TEC1-12706 at $T_{p,dhw} = 50\,°C$
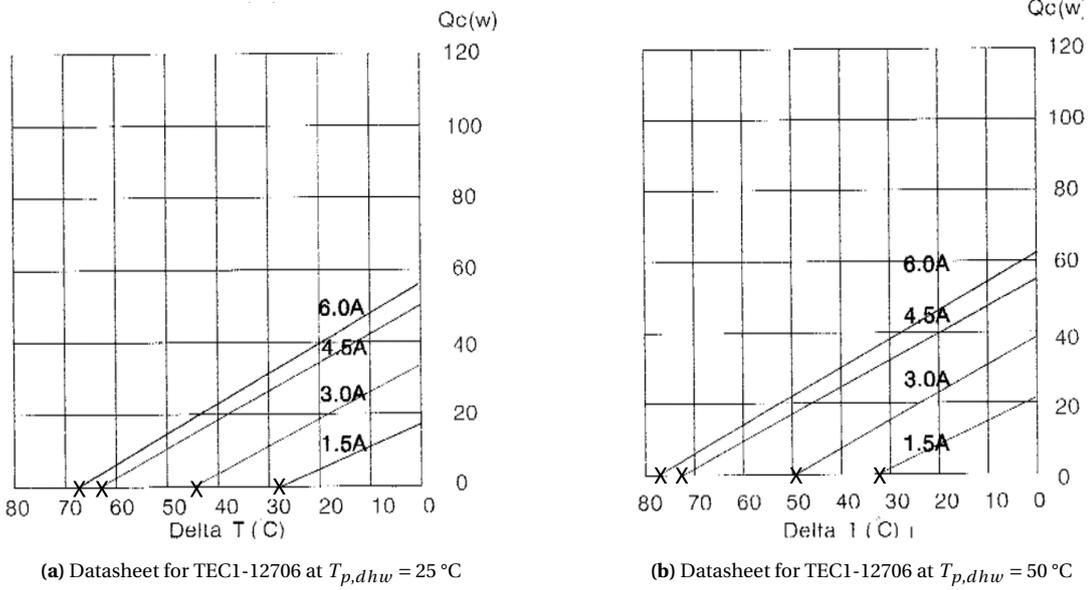
Figure 3.5: Performance of Peltier element TEC1-12706 for two different temperatures. The horizontal axis shows the temperature difference over the Peltier element, the vertical axis shows the heat absorption by the Peltier element on the heating network side $Q_c = \dot{Q}_{hn}$. [11].

3.2.1. Seebeck coefficient and resistance

According to Equation 3.7 the heat transfer rate at $\Delta T_p = 0$ (the horizontal axis in Figure 3.5) at the heating network side of the Peltier element is given by:

$$Q_{hn} = SIT_{p,hn} - \frac{I^2 R}{2}$$

From the Figures 3.5a and 3.5b eight datapoints (Q_{hn}, I) are obtained (marked with an X in figure 3.5) to which a least squares fit of the form of Equation 3.2.1, with S and R as coefficients, is performed. Four data points have two temperature $T_1 = 298$ K and four data points have temperature $T_2 = 323$ K.

If the four different values for the current from Figure 3.5 are denoted as $I_1 = 1.5, I_2 = 3, I_3 = 4.5, I_4 = 6$ A and the heat transfer rate values Q_c corresponding to the values for the current as $Q_{1,1},...,Q_{1,4},Q_{2,1},...,Q_{2,4}$ with the first subscript indicating the correspondence to the temperature T_1 or T_2, and the second subscript indicating the correspondence to the current $I_1,...,I_4$, then the linear least squares system $cX = b$ to be solved is:

$$X = \begin{bmatrix} \frac{I_1}{T_1} & -\frac{I_1^2}{2} \\ \vdots & \vdots \\ \frac{I_4}{T_1} & -\frac{I_4^2}{2} \\ \frac{I_1}{T_2} & -\frac{I_1^2}{2} \\ \vdots & \vdots \\ \frac{I_4}{T_2} & -\frac{I_4^2}{2} \end{bmatrix} \qquad b = \begin{bmatrix} Q_{1,1} \\ \vdots \\ Q_{1,4} \\ Q_{2,1} \\ \vdots \\ Q_{2,4} \end{bmatrix}$$

The solution to the least squares problem $\min_{S,R} \sum_i ||X^T \begin{bmatrix} S \\ R \end{bmatrix} - b||^2$ is given by:

$$\begin{bmatrix} S \\ R \end{bmatrix} = (x^T x)^{-1}(x^T b)$$

Using Figure 3.5 the obtained values for S and R are $S = 0.0477$ V/K and $R = 1.57\ \Omega$. The value of S seems a factor of 100 too large compared to the values mentioned in Section 2.2.1. This is because the values mentioned in Section 2.2.1 are the Seebeck coefficients for a metal-semiconductor *junction*. Peltier elements consist of 127 n-type and p-type semiconductors in series (Figure 2.6). There are four junctions per n-type/p-type couple giving a value of $S_n = \frac{0.0477}{4 \cdot 127} = 94\ \mu$V/K for the used semi-conductor material, which is in the expected order of magnitude.

3.2.2. Temperature dependence of thermal conductivity

Using the Seebeck coefficient $S = 0.0477$ V/K and resistance $R = 1.57\Omega$, the thermal conductivity of the Peltier element can be evaluated. Using Figures 3.5a and Figures 3.5b and the temperature difference over the Peltier element at $\dot{Q}_c = 0$, the relation between $k(T)$ and T can be determined since:

$$Q_c = 0 = IST_{p,hn} - k(\frac{T_{p,dhw} + T_{p,hn}}{2})\frac{A}{d}(T_{p,dhw} - T_{p,hn}) - \frac{I^2 R}{2}$$

$$k(\frac{T_{p,dhw} + T_{p,hn}}{2}) = \frac{IST_{p,hn} - \frac{I^2 R}{2}}{\frac{A}{d}\Delta T_p}$$

With ΔT_p the temperature difference over the Peltier element. Using the data from Figure 3.5, eight datapoints can be determined. The initial fit resulted in $k_1 = 0.79$ W/mK and $k_2 = 0.002$ 1/K, with k_1 and k_1 as defined in Equation 3.3, which is an increasing exponential. The thermal conductivity of the semiconductor material is expected to decrease with temperature. This is because as the temperature rises, more electrons occupy the conduction band making it increasingly harder for electrons in the valance band to occupy the conduction band. Electrons in the valance band cannot move since there are no, or little, available states available, by assumption. Since the electrons have a large contribution to the thermal conductivity of semiconductors (explaining the difference in magnitude of Peltier effect between metals and semi-conductors) the thermal conductivity decreases as temperature increases. A a certain point, a substantial amount of electrons have moved from the valence band to the conduction band giving all the electrons available states to move around in, causing the thermal conductivity to theoretically rise again. The shear amount of electrons needed to move from the valance band to the conduction band for this phenomena to happen is means an energy of several $k_B T$ has to added to the system, making it like that the system has melted before the thermal conductivity has risen again.

To obtain a decreasing thermal conductivity, the datapoints with $T_h = 25°$C and the datapoints with $T_h = 50°$C are averaged, the resulting values for the parameters k_1 and k_2 of the thermal conductivity model are $k_1 = 2.14$ W/mK and $k_2 = -0.0011$ 1/K.

3.3. Environment interaction

As mentioned in Section 3.1, the effective potential difference over the Peltier element's resistive load is reduced, when compared to the applied potential difference, by an amount $S\Delta T_p$ where ΔT_p indicates the temperature difference over the Peltier element. Such that:

$$V_{eff} = V_0 - S\Delta T_p \tag{3.9}$$

With R the resistance of the Peltier element. This is due to the Seebeck effect as mentioned in Section 2.2.1. In practice this means a reduced current will flow, as compared to the initial current that would flow at the applied electric potential difference. The effective potential determines the current through the Peltier element, and thus the power draw of the Peltier element according to Equations 3.10 and 3.11:

$$I_p = \frac{V_0 - S(T_{p,dhw} - T_{p,hn})}{R} \tag{3.10}$$

$$P_{\text{element}} = V_0 I = V_0 \left(\frac{V_0 - S\Delta T_p}{R} \right) \tag{3.11}$$

The reduced current will reduce the heat flow due to the Peltier Effect and therefor will lower the temperatures of the boundaries of the Peltier elements. The reduced temperature difference ΔT_p will increase the effective electric potential over the Peltier element. In essence, the Peltier effect and Seebeck effect work together in a negative feedback loop as depicted in Figure 3.6.
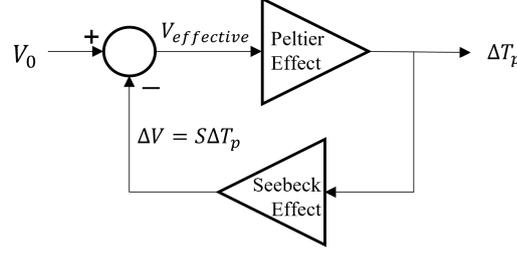
Figure 3.6: Negative feedback loop in which the Peltier effect and the Seebeck effect work together.

From Equation 3.11 it is clear that for $\Delta T_p < \frac{V_0}{S} \approx \frac{10}{0.05} = 200$ K (or °C) the power draw per element is reduced due to the Seebeck effect compared to an electric resistor of the same resistance. The maximum allowable temperature difference $\Delta T_{p,max}$, in terms of system boundaries, is determined by the difference between the maximum allowable temperature of 130 °C (melting point of the solder in the Peltier element), and the minimum allowable temperature difference of 0 °C (freezing point of water) resulting in a maximum allowable temperature difference of 130 °C. Therefor, $\Delta T_{p,max} = 130 < 200$ °C and the power per element will be below $P_0 = \frac{V_0}{R}$ whenever the system is running.

In the Peltier heat exchanger the Peltier elements will be placed between the domestic hot water flow and the heating network flow. The hot side of the Peltier element will be engaged in heat transfer to the domestic hot water and the cold side of the Peltier element will be engaged in heat transfer with the heating network flow. The temperature difference between the Peltier element's boundaries and the water flows will determine the amount of heat transferred to each respective flow. The interaction of a single Peltier element can be described by the four equations of System of Equations 3.12:

$$U_{dhw}A_{dhw}(T_{p,dhw} - \frac{T_{dhw,in} + T_{dhw,out}}{2}) = \dot{m}_{dhw}c_p(T_{dhw,out} - T_{hn,in})$$

$$\dot{m}_{dhw}c_p(T_{dhw,out} - T_{dhwwater,in}) = IST_{p,dhw} + \frac{I^2R}{2} - k(\frac{T_{dhw} + T_{hn}}{2})\frac{A_p}{d}(T_{p,dhw} - T_{p,dhw})$$

$$\dot{m}_{hn}c_p(T_{hn,in} - T_{hn,out}) = IST_{p,hn} - \frac{I^2R}{2} - k(\frac{T_{p,dhw} + T_{p,hn}}{2})\frac{A_p}{d}(T_{p,dhw} - T_{p,hn})$$

$$U_{hn}A_{hn}(\frac{T_{hn,in} + T_{hn,out}}{2} - T_{p,hn}) = \dot{m}_{hn}c_p(T_{hn,in} - T_{hn,out})$$

(3.12)

The four equations from System 3.12, in order, represent:

1. An energy balance over the domestic hot water.

2. A surface thermal energy flux on the domestic hot water side of the Peltier element.

3. A surface thermal energy flux on the heating network side.

4. An energy balance over the heating network.
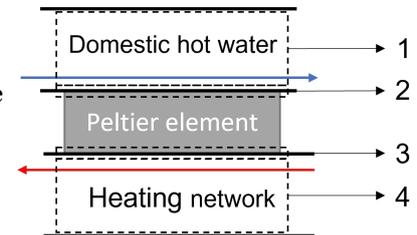
Figure 3.7: Schematic representation of the energy balances from system 3.12.

Figure 3.7 shows where the four equation from System 3.12 are relevant. The first and fourth equation contain a $\sim UA\Delta T$ term where the ΔT term represents the logarithmic average temperature difference. The formulas in System of Equations 3.12 use the *arithmetic mean*. When the temperature gradient of the domestic water per Peltier element is small ($\Delta T/n = 0.5$ °C per element) compared to the temperature differences between the domestic hot water and the boundaries of the Peltier elements the difference between the logarithmic mean and the arithmetic mean becomes negligible.

Notice that the first two equations consist of three different formulas, any permutation of the three formulas over the two equations gives a valid system of equations that describes the behavior of the Peltier element with its environment as long as the two formulas in an equation are different. The same holds for the last two equations. Since the Equations of System of Equations 3.12 are quasi-linear (the thermal conductivity of the Peltier elements $k(T)$ is non-linear) they can be put in matrix-vector form as done in Equation 3.13:

$$
\begin{bmatrix}
\dot{m}_{dhw}c_p + \frac{U_{dhw}A_{dhw}}{2} & -U_{dhw}A_{dhw} & 0 & 0 \\
-\frac{U_{dhw}A_{dhw}}{2} & \left(\begin{smallmatrix} U_{dhw}A_{dhw} \\ +k(\frac{T_{p,dhw}+T_{p,hn}}{2})\frac{A_p}{d}-IS \end{smallmatrix}\right) & -k(\frac{T_{p,dhw}+T_{p,hn}}{2})\frac{A_p}{d} & 0 \\
0 & k(\frac{T_{p,dhw}+T_{p,hn}}{2})\frac{A_p}{d} & \left(\begin{smallmatrix} -U_{hn}A_{hn} \\ -k(\frac{T_{p,dhw}+T_{p,hn}}{2})\frac{A_p}{d}-SI \end{smallmatrix}\right) & \frac{U_{hn}A_{hn}}{2} \\
0 & 0 & U_{hn}A_{hn} & -\dot{m}_{hn}c_p - \frac{U_{hn}A_{hn}}{2}
\end{bmatrix}
$$

$$
\cdot
\begin{bmatrix}
T_{dhw,out} \\
T_{p,dhw} \\
T_{p,hn} \\
T_{hn,out}
\end{bmatrix}
=
\begin{bmatrix}
(\dot{m}_{dhw}c_p - \frac{U_{dhw}A_{dhw}}{2})T_{dhw,in} \\
\frac{I^2R}{2} + \frac{U_{dhw}A_{dhw}}{2}T_{dhw,in} \\
-\frac{I^2R}{2} + -\frac{U_{hn}A_{hn}}{2}T_{hn,in} \\
-(\dot{m}_{hn}c_p - \frac{U_{hn}A_{hn}}{2})T_{hn,in}
\end{bmatrix}
\tag{3.13}
$$

The input parameters to System 3.13 are:

- $\dot{m}_{dhw}, \dot{m}_{hn}$ Mass flow rate of the respective side of the Peltier element.

- U_{dhw}, U_{hn} Total heat transfer coefficient at the respective sides.

- I Electric current through the element.

- $T_{dhw,in}, T_{hn,in}$ Inflow temperature of the respective sides.

The constant model parameters are given by (see Section 5.1):

- c_p Domestic hot water specific heat capacity (= 4180 J/kg).

- A_{dhw}, A_{hn} Heat transfer area at the respective sides (=0.00152 m^2).

- A_p Cross sectional area of the Peltier element (=0.0016 m^2).

- d Thickness of the Peltier element (= 0.0039 m).

- S Seebeck coefficient (=0.0477 V/K).

The solution to System 3.13 is a vector containing; the temperature of the domestic hot water after the Peltier element, the temperature of the hot side of the Peltier element, the temperature of the cold side of the Peltier element, the temperature of the heating network after the Peltier element. The final solution to System 3.13 is obtained by iteration, the iteration terminates if the temperature has not changed by more than 0.001 K over a single iteration and if the electric current flow is below the allowed current by the power supply. After every iteration the electric current has to be re-evaluated according to Equation 3.14:

$$
I = \frac{V_0 - S(T_{p,dhw} - T_{p,hn})}{R}
\tag{3.14}
$$

It is likely that the value for the thermal conductivity $k((T_{p,dhw} - T_{p,hn})/2)$ has changed (slightly) after an iteration. Additionally, since a power supply will limit the applied electric potential difference when too much current is drawn, a check has to be done to prevent a too high current draw, i.e.: $I < I_{max,ps}$ where $I_{max,ps}$ is the maximum current that the power supply can provide. If the calculated current from Equation 3.14 is larger than $I_{max,ps}$, the applied electric potential difference will be (numerically) reduced to $V_{0,new} = \frac{1}{2}(V_{0,old} + I_{max,ps}R)$. Since the solution to the quasi-linear system is found by iteration, a starting point for the Peltier element's boundary temperature has to be chosen. The initial guesses are chosen to be:

$$
\begin{aligned}
T_{p,dhw,0} &= T_{dhw,in} \\
T_{p,hn,0} &= T_{hn,in}
\end{aligned}
\tag{3.15}
$$

The iteration schemes used to solve System of equations 3.12 are shown schematically in Appendix D.

The heat transfer coefficient from the Peltier element's boundary to the respective flow has to be calculated. The design of the PeHEX that will be used to run the experiments (Chapter 5) was made in parallel with the numerical model. The Peltier elements will not be in direct contact with the flow, but will be separated by a 1 mm thick plate of aluminium with a thermal conductivity of $k = 205$ W/mK. Therefor the total heat trasnfer coefficient will be $U^{-1} = h^{-1} + d/k$ with $d = 0.001$ m the thickness of the aluminium plate and h the convective heat transfer coefficient from the wall to the water flow. The convective heat transfer coefficient is calculated using a correlation for the Nusselt number from Mills and Coimbra [25, Eq. 4.50] for laminar flows (Re < 2300):

$$\text{Nu}_D = 3.66 + \frac{0.065(D/L)\text{RePr}}{1 + 0.04((D/L)\text{RePr})^{(2/3)}} \tag{3.16}$$

With D the hydraulic diameter of the channel, L the length of the channel, Re the Reynolds number and Pr the Prandtl number. For turbulent flows (Re > 3000) the Gnielinski correlation [8] is used:

$$\text{Nu}_D = \frac{(f/8)(\text{Re}_D - 1000)\text{Pr}}{1 + 12.7(f/8)^{1/2}(\text{Pr}^{2/3} - 1)} \tag{3.17}$$

$$f = \frac{1}{(0.79\ln(\text{Re}) - 1.64)^2} \tag{3.18}$$

with f the friction factor as defined in Incropera and DeWitt [15, p. 490]. For Reynolds numbers in the range 2300 < Re < 3000 a Hermite cubic polynomial (Appendix C) is used to interpolate between Equations 3.16 and 3.17. Code snippet E.1 shows how the final solution to System 3.13 is found.

3.4. Coupling of Peltier element models

System of equations 3.12 describe the temperature distribution of a single Peltier element. To analyse the performance of the PeHEX all the individual models for the Peltier elements have to be coupled together since the temperature of the domestic hot water as it leaves one element will be the input temperature of another element. Therefor, to find the temperature distribution within the entire PeHEX, iterations over all the Peltier elements have to be performed.

Before the iterations can be done a flow type, co-current, cross-current or counter-current, has to be chosen. In an ordinary heat exchanger, a counter-current flow is chosen to accommodate a maximum logarithmic temperature difference between the two flows in order to maximize the heat transfer. However, there is a major difference between an ordinary heat exchanger and a PeHEX. In an ordinary heat exchanger the temperatures of the two flows approach each other, such that the temperatures at which the flows leave the heat exchanger have to be in between the two temperatures that the flows entered the heat exchanger with. In a PeHEX the temperatures *diverge* from one another as visible in Figure 3.8, meaning the temperatures at which the two flows leave the PeHEX do no necessarily have to be in between the temperatures at which the two flows entered the PeHEX. Therefor a co-current arrangement is not desirable, as the temperature difference over the Peltier element would grow from the beginning to the end of the heat exchanger and a large temperature difference over the Peltier element can severely limit their performance according to equation 3.7.

The advantage of a cross-current arrangement is that the heating network sides of all the Peltier elements feel the hottest water at temperature $T_{hn,in}$ when compared to only the last element feeling $T_{hn,in}$ for a counter-current arrangement as the heating network will cool down throughout the PeHEX. The disadvantage of a cross-current arrangement is the lower mass flow rate as the water has to travel to a larger (transversal) surface area. The choice was made to use a counter-current arrangement for this research. The PeHEX now consists of two identical water channels, limiting the cost of the test setup (Chapter 5).

To find the temperature distribution throughout the PeHEX, the System of Equations 3.12 has to be solved for each individual Peltier element. The value $T_{dhw,out}$ from the solution vector becomes $T_{dhw,in}$ for the next element, and the value $T_{hn,out}$ from the *next* element becomes $T_{hn,in}$ for the current Peltier element. Since the solution vector for the next Peltier element is not found yet, an initial guess of the temperature distribution has to made. This is a consequence of using a counter-current arrangement. The solution vectors for the Peltier elements are solved sequentially, meaning the model *sweeps* over the elements. After all the solution vectors have been found, the model sweeps back to the first element. The final solution is said to be found if the temperature profiles before and after a sweep is smaller than 0.01 for every Peltier element. The code used for determining the temperature distribution in the PeHEX is shown in Figure E.2.
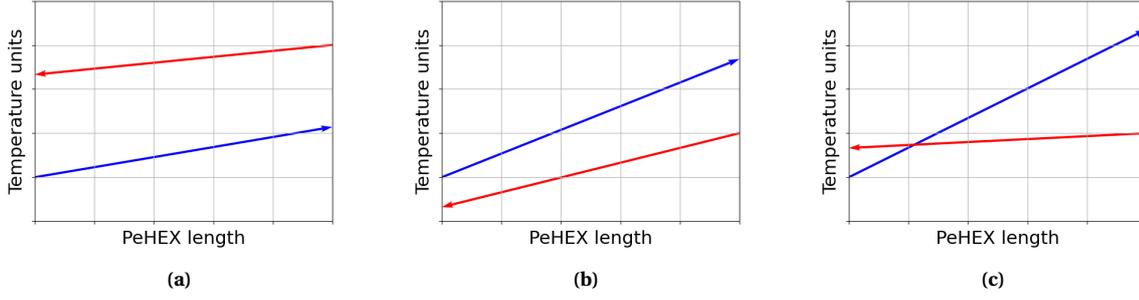
Figure 3.8: Different temperature profiles for the PeHEX. The rising arrows represent the domestic hot water temperature, while the falling arrows represent the heating network temperature. Figure 3.8a shows the only temperature profile that is possible in an ordinary heat exchanger and in a PeHEX and indicates an inefficient use of the Peltier effect. Figure 3.8b shows the diverging temperature difference within the PeHEX. The relatively large temperature decrease in the heating network indicates a medium efficient use of the Peltier effect. Figure 3.8c indicates a very efficient use of the Peltier effect in a PeHEX. Figures 3.8b and 3.8c clearly require an extra source of power.

3.5. Maximum coefficient of performance

The Coefficient of Performance (COP) of the PeHEX is given by COP $= \frac{Q_{dhw}}{P_{in}} = 1 + \frac{Q_{hn}}{P_{in}}$. The mass flow rate of the domestic hot water is not controllable (it is a demand by the user). The only variables that affect the COP are the applied electric potential V_0 and the mass flow rate of the heating network $\dot{m}_{hn}$. As the temperature difference over the Peltier elements limits the current and therefor the heat transfer it is desirable to set the mass flow rate of the heating network to the maximum value. This reduces the temperature drop over the Peltier element (for a given power absorption) and thus an increased power rejection at the *domestic hot water* side. Therefor, the maximum COP at a given state of operation is determined by the applied electric potential difference and given by:

$$\max \text{COP} = \max_{V_0} \frac{Q_{dhw}}{P_{in}} \tag{3.19}$$

where V_0 is the electric potential difference that is applied to the Peltier element. The maximum COP is found by differentiating $\frac{Q_{dhw}}{P_{in}}$ with respect to V_0 and finding the root of the derivative. The derivative is given by:

$$\frac{d}{dV_0} \text{COP} = \frac{d}{dV_0} \left[\frac{Q_{dhw}}{P_{in}} \right]$$
$$= \frac{P_{in} \frac{dQ_{dhw}}{dV_0} - Q_{dhw} \frac{dP_{in}}{dV_0}}{P_{in}^2} = 0 \tag{3.20}$$

Using the formulas for $\dot{Q}_{dhw}$ and P_{in}:

$$Q_{dhw} = \Delta H_{dhw} = \dot{m}_{dhw} c_p (T_{dhw,out} - T_{dhw,in}) \implies \frac{d}{dV_0}(Q_{dhw}) = \dot{m}_{dhw} c_p \frac{d}{dV_0}(T_{dhw,out}) = \frac{d}{dV_0}(H_{dhw,out})$$

$$P_{in} = \sum_n V_0 \left[\frac{V_0 - S\Delta T_{p,n}}{R} \right] \implies \frac{d}{dV_0}(P_{in}) = \sum_n \left[\frac{2V_0 - S\Delta T_p - V_0 S \frac{d\Delta T_p}{dV_0}}{R} \right]$$

with ΔH_{dhw} the enthalpy change over the entire entire PeHEX and $H_{dhw,out}$ the enthalpy of the domestic hot water as it leaves the entire PeHEX. The subscript n in the formula for the power draw represents the sum over all the individual elements. The root of the numerator of Equation 3.20 is found by solving Equation 3.21:

$$0 = \left(\sum_n V_0 \left[\frac{V_0 - S\Delta T_{p,n}}{R} \right] \right) \frac{d}{dV_0}(H_{dhw,out}) - \Delta H_{dhw}(V_0) \sum_n \left[\frac{2V_0 - S\Delta T_{p,n} - V_0 S \frac{d\Delta T_{p,n}}{dV_0}}{R} \right] \tag{3.21}$$

It is not immediately clear if there are any solutions to Equation 3.21 with non-zero finite potential difference $0 < V_0 < \infty$. These bounds are necessary on V_0 since for $V_0 = 0$ the input power is zero and the Peltier elements act like a simple heat exchanger which is not the goal of the research, and $V_0 \to \infty$ is an impossible scenario. The quantitative solution to Equation 3.21 is best found incrementally using numerical methods.

A practical limit on the applied electric potential difference V_0 can be obtained by observing that COP = $\frac{Q_{dhw}}{P_{in}} = 1 + \frac{Q_{hn}}{P_{in}}$. Therefor, if $Q_{hn} < 0$ the COP will be less than 1. The formula for $\dot{Q}_{hn}$ is given by Equation 3.22:

$$\dot{Q}_{hn} = I S T_{p,hn} - \frac{1}{2} I^2 R - k(\frac{T_{p,dhw} + T_{p,hn}}{2})\frac{A}{d}(T_{p,dhw} - T_{p,hn}) \tag{3.22}$$

If the third term on the right hand side (the thermal conduction $k\frac{A}{d}\Delta T_p$) is regarded as linear with respect to current ($\approx \alpha I$) for the sake of obtaining an estimate for the maximum electric potential difference to be applied, then Equation 3.22 can be regarded as a quadratic equation with roots $I = 0$ and:

$$I = 2\frac{S T_{p,hn} - \alpha}{R} = \frac{V_0 - S\Delta T_p}{R} = \frac{V_0 - \frac{\alpha S d I}{kA}}{R}$$
$$\implies V_0 = 2(S T_{p,hn} - \alpha)\left(1 + \frac{S\alpha d}{kAR}\right) \tag{3.23}$$

Typical values for the unknown parameters are $T_{p,hn} = 303$ K, $\alpha = 8.5$ V, $S = 0.05$ V/K, $k = 2$ W/mK and $R = 3$ such that the maximum potential is roughly $V_{0,max} = 15$ V.

As discussed above, to reach the highest possible COP the heating network has to be run at maximum flow speeds to reduce the temperature difference across the Peltier elements. In practice, the only variable that can still be altered is the electric potential difference. Since the output temperature of the domestic hot water is set fixed at 60 °C, there are no more free variables and thus the COP is set.

After solving the coupled system of equations for the entire PeHEX (Code snippet E.2) and using the formulas for $Q_{dhw} = \Delta H_{dhw}$ and $P_{in} = \sum_n V_0 \left[\frac{V_0 - S\Delta T_p}{R}\right]$, the COP is calculated as:

$$\text{COP} = \frac{\Delta H_{dhw}}{\sum_n V_0 \left[\frac{V_0 - S\Delta T_{p,n}}{R}\right]} \tag{3.24}$$

<div align="right">

4

</div>

Numerical Results

Numerical simulations using the model from Chapter 3 were run to determine if time and money should be invested to build a physical test setup and experimentally validate the numerical model.

4.1. Temperature distribution

As discussed in Section 3.4 there are three different possibilities for the temperature profile within the entire PeHEX. Figure 4.1 shows the three different temperature distributions as a result of numerical simulations for different domestic hot water flow speeds ϕ_{dhw}, heating network flow speeds ϕ_{hn} and applied electric potential differences V. This indicates that the temperature profiles from Figure 4.1 are not just hypothetical, but are also the results of numerical simulations.

<center>(a)   (b)   (c)</center>
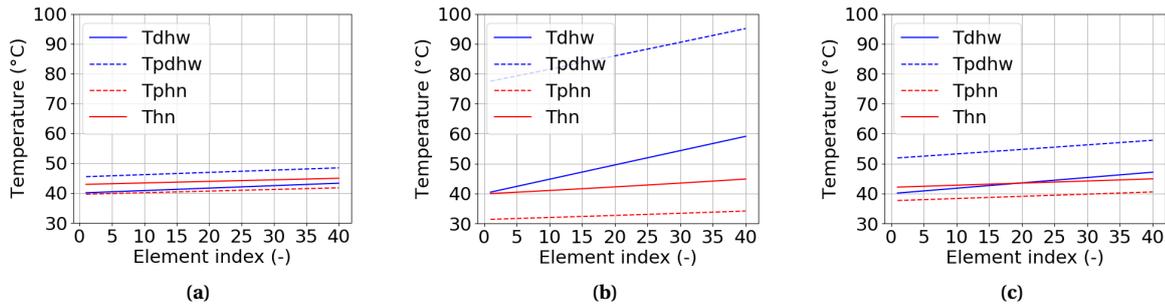
Figure 4.1: Figure 4.1a show the temperature distribution for high flow speeds and low potential differences $\phi_{dhw} = 4$ L/min, $\phi_{hn} = 4$ L/min, $V = 5$ V. Figure 4.1b show the temperature distribution for low flow speeds and high potential differences $\phi_{dhw} = 1$ L/min, $\phi_{hn} = 1$ L/min, $V = 9.6$ V. Figure 4.1a show the temperature distribution for high flow speeds and high potential differences $\phi_{dhw} = 4$ L/min, $\phi_{hn} = 4$ L/min, $V = 9.6$ V.

Figure 4.1b shows an increase in domestic hot water temperature from 40 °C to 60 °C while the heating network temperature gets reduced from 45 °C to 40 °C. From Figure 4.1b it is clear that the temperature difference between the hot side of the Peltier element and the domestic hot water is much larger than the temperature difference of the heating network and the cold side of the Peltier element, even though they have the same flow speed. This is because at low flow speeds the thermal energy supplied to the hot side of the Peltier element is much larger than the heat absorbed at the cold side due to the difference in sign in the $\frac{1}{2} I^2 R$ term. At low flow speeds, the difference in power absorption and rejection of the Peltier element can influence the temperatures of the domestic hot water and the heating network significantly. At low flow speeds, the total heat transfer coefficient is relatively small $U \sim 1000$ W/m^2. A large temperature difference is then sustained to balance the energy equation that governs the heat flow from the Peltier element to the wall $(UA(T_{p,dhw} - \bar{T}_{dhw}) = \dot{Q}_{dhw})$. At larger flow speeds the thermal energy is given of easier to the domestic hot water, and taken up easier from the heating network due to the larger heat transfer coefficient.

4.2. Modelling the experimental outcome

The simulations that were run are given by Table 4.1. All simulations have a potential per element of $V_0 = 9.6$ V, (unless the potential is current limited to 5 amps, then it is lowered). The maximum flow speed of $V_{max} = 5$ L/min is chosen to ensure that the water channels and connection hoses in the physical PeHEX (Section 5.5) do not suffer damage from a too high flow speed.

Table 4.1: Overview of flow setting per measurement. Vertical bars indicate an increase in the domestic hot water flow.

Experiment index	1	2	3	4	5	6	7	8	9	10	11	12	13
ϕ_{dhw} (L/min)	1.0	1.0	1.0	1.0	1.0	2.0	2.0	2.0	2.0	2.0	3.0	3.0	3.0
ϕ_{hn} (L/min)	1.0	2.0	3.0	4.0	5.0	1.0	2.0	3.0	4.0	5.0	1.0	2.0	3.0

Experiment index	14	15	16	17	18	19	20	21	22	23	24	25	
ϕ_{dhw} (L/min)	3.0	3.0	4.0	4.0	4.0	4.0	4.0	5.0	5.0	5.0	5.0	5.0	
ϕ_{hn} (L/min)	4.0	5.0	1.0	2.0	3.0	4.0	5.0	1.0	2.0	3.0	4.0	5.0	

The full results of the simulations are given in Table A.1, and are graphically represented in Figure 4.2 and Figure 4.3.

(a)

(b)

(c)

(d)
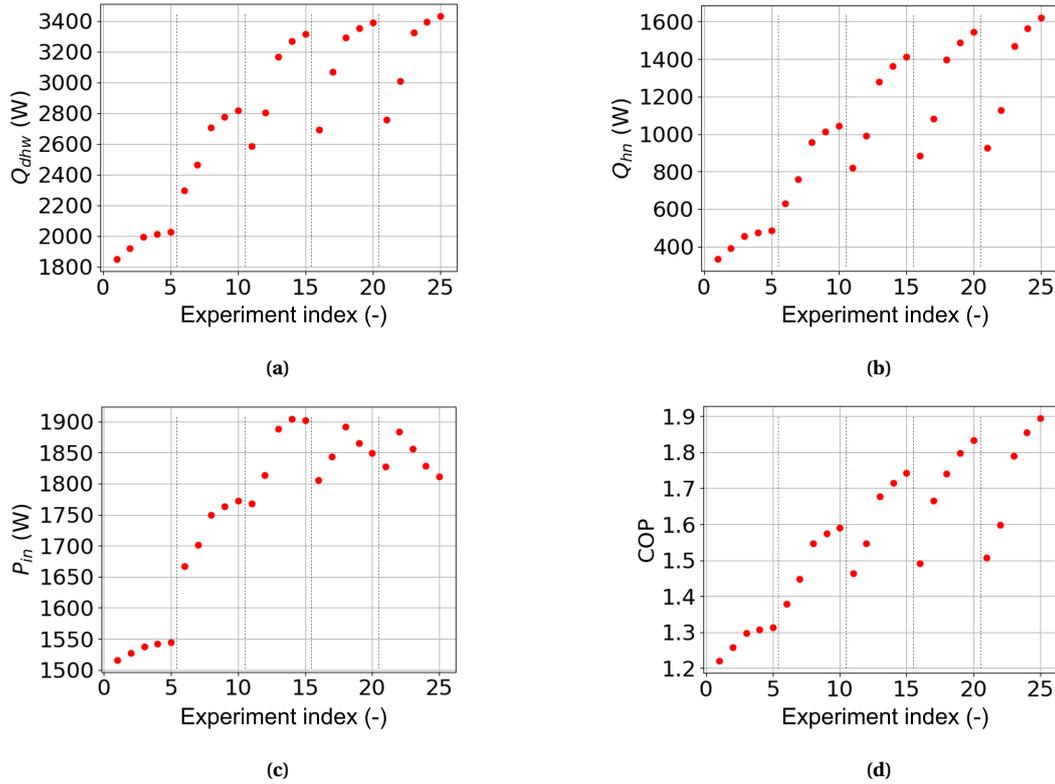
Figure 4.2: Figure 4.2a: Domestic hot water power absorption; Figure 4.2b: Heating network power rejection. Figure 4.2c: Power input; Figure 4.2d: COP. The vertical dashed lines correspond to the vertical lines in Table 4.1.

An interesting observation from Figure 4.2 is the general increase in $\dot{Q}_{dhw}$, $\dot{Q}_{hn}$, P_{in}, and the COP as the domestic hot water flow speed and the heating network flow speed (increasing experiment index) increase. This is an artifact from the Thermoelectric effect. As the heating network flow speed and the domestic hot water flow speed increase (increasing Experiment index) the total heat transfer coefficient increases, which, as discussed in section 4.1, decreases the temperatures of the boundaries of the Peltier element. As the temperature differences over the Peltier elements decrease, the effective electric potentials (Formula 3.9) increase causing larger currents to run and therefore larger thermal currents will be supplied to the domestic hot water by the Peltier effect. The same argument holds for the increasing power rejection of the heating network.

The power input P_{in} is the only variable that decreases as the heating network flow speed increases. This is because the applied electric potential gets limited as the current draw would otherwise exceed the maximum 5 amps the power supply is able to deliver to the elements (as visible in table A.1). As discussed before, the temperature difference over the Peltier element is reduced as the Experiment index increases causing an increasing effective electric potential. The increasing electric potential causes an increasing current until it exceeds 5 amps, when it is reduced by limiting the applied electric potential.

Figure 4.2d suggests that a COP of 2.0 is possible with higher flow speeds (higher experiment index). The

downside of higher flow speeds is a lower output temperature of the domestic hot water, as visible from Table A.1 and Figure 4.3. Generally, it is advantageous to have $\phi_{hn} > \phi dhw$ as visible from Figure 4.3. A different type of Peltier element with a higher Seebeck coefficient S, and lower electric resistance R might reach higher output temperatures for higher flow speeds whilst also achieving a high (>2.0) COP.
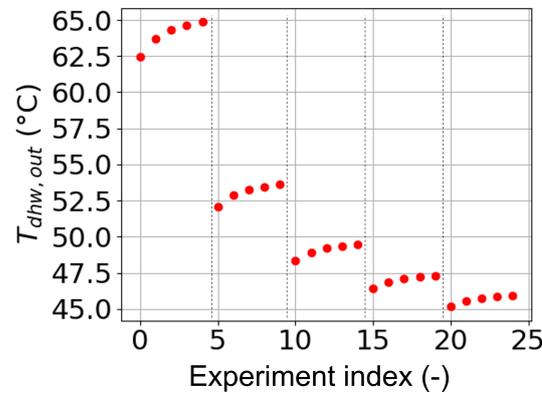
Figure 4.3: Output temperature of the domestic hot water $T_{dhw,out}$ as a function of Experiment index. The output temperature decreases with increasing domestic hot water flow speed, and increases with increasing heating network flow speeds.

The numerical results presented in this chapter show evidence that a Peltier Heat Exchanger is able to heat domestic hot water ,using a heating network, with an acceptable performance (COP between 1.2 and 1.9). Therefore, a PeHEX was designed and build. The experimental setup is discussed in Chapter 5. The experimental results are presented and discussed in Chapter 6.

5

Experimental setup

5.1. Peltier Heat Exchanger design

The Peltier Heat Exchanger used to run the experiments was designed from scratch by the author. The final design is depicted in Figure 5.1 with a cross sectional view in Figure 5.2a. The PeHEX was produced by DEMO [23], at Delft University of Technology.

(a)   **(b)**   **(c)**

(d)   **(e)**
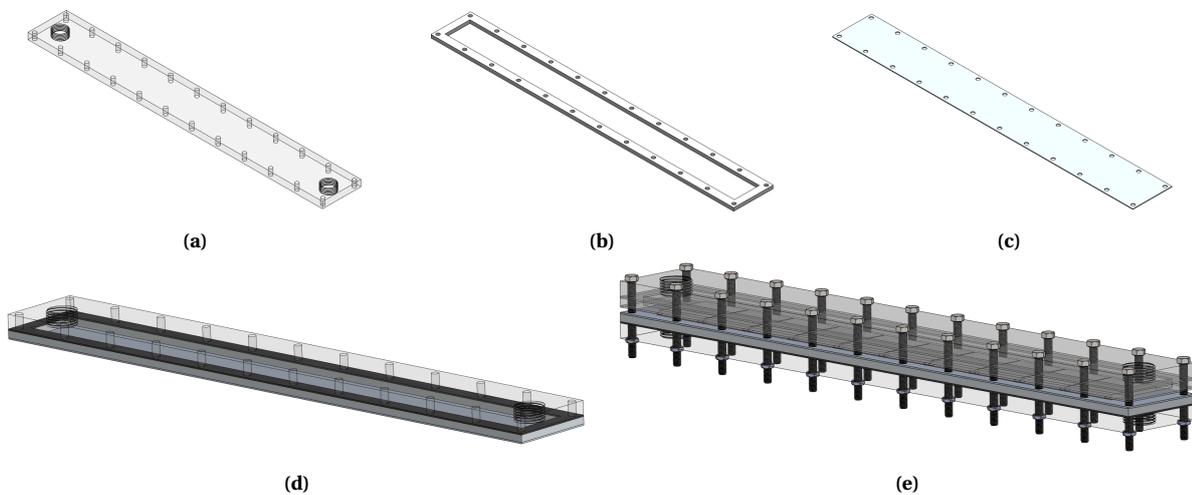
Figure 5.1: Figure 5.1a: 5 mm thick perspex lid of a single channel with 1/2" tapped holes for hose connectors; Figure 5.1b: shape of the gasket (1 mm thick silicon) and the channel shape (3mm thick aluminium); Figure 5.1c: 1 mm thick aluminium base plate; Figure 5.1d: assembly of the base plate, channel, gasket and perspex lid, the base plate and the channel are glued together with metal glue. The length of the water channel is 44 cm, the total length of the assembly is 46 cm; Figure 5.1e: Peltier Heat Exchanger (PeHEX) consisting of tho opposite facing channels (Figure 5.1d) with Peltier elements in between.

The width of the channel is $w = 0.038$ m, which is narrower smaller than the Peltier elements. This was to ensure the Peltier elements do not push through the base plate (Figure 5.1c) when the entire assembly was bolted together. The Peltier elements are located between two channels. Thermal adhesive was used to ensure a proper thermal contact between the Peltier elements and the base plates.

The heat transfer area from Section 3.3 (A_{dhw}, A_{hn}) is less than the total metal surface area that is in contact with the water. This is because due to the thin base plate, the heat (or cold) provided by the Peltier element travels directly trough the aluminium to the water rather than through the aluminium around the water, as depicted in Figure 5.2b. Therefor, the heat transfer area is taken as the area of the base plate in contact with the water $A_{dhw} = A_{hn} = 0.04 \cdot 0.038 = 0.00152$ m^2. The boundary conditions for Figure 5.2b are:

- A prescribed heat flux where the metal touches the Peltier element, 50 W for the domestic hot water side, 40 W for the heating network side.

- A convective boundary condition where the metal is in contact with the domestic hot water or the heating network.

- An insulation boundary condition on the remainder of the boundary: signifying no heat to the environment.

The code used to solve the 2D-heat equation can be found in Appendix F.

(a)

(b)
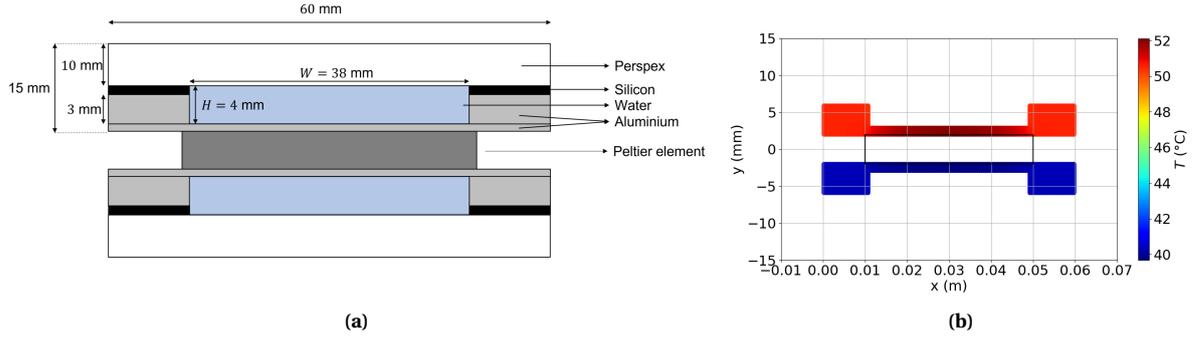
Figure 5.2: Figure 5.2a: Schematic cross sectional view of the Peltier Heat Exchanger of Figure 5.1e. Figure 5.2b: Heat flow through the Aluminium parts of the assembly in Figure 5.2a. The black box indiates the location of the Peltier element. The water temperatures are $T_{dhw} = 43.9\,°C$ and $T_{hn} = 43.0\,°C$.

The total number of Peltier elements used is $n = 40$. As every PeHEX has ten elements, the number of PeHEX used in this research is four.

5.2. Electric connections in a PeHEX

The PeHEX build for the measurements consists of four arrays of ten Peltier elements, each consisting of two parallel arrays of five Peltier elements in series (Figure 5.3a). This configuration is chosen such that, in combination with the 48 V_{DC}, 10 A Mean Well HLG-480H-48AB power supplies [39] per array, each element receives a potential difference of ~ 9.6 V and can draw a current of 5 amps meaning the power draw per element is limited to 48 Watts. The modules used (TEC1-12706) are rated for 100 Watts so there should be no damage to the modules caused by an electric overloading. The resistance of the Peltier elements is $R_p = 2.85 \pm 0.04\ \Omega$, giving a resistance of the parallel string of Peltier elements of 7.13 Ω giving a maximum current of 6.7 A.

(a)

(b)
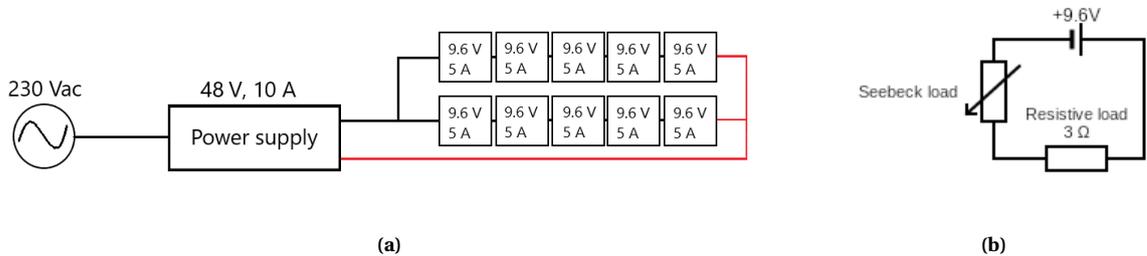
Figure 5.3: Figure 5.3a: Electrical wiring of the Peltier elements in a PeHEX with the two strings of Peltier elements. The two strings are placed in the heat exchanger such that the first 5 elements and last 5 elements are part of a separate string of elements. Figure 5.3b: Model of the electric characteristics of a Peltier element.

Figure 5.3b shows how a Peltier element can be modelled in an electric circuit. The variable load of the Seebeck load is given by $P_{\text{Seebeck}} = IS\Delta T_p$ whereas the resistive load is given by $P_{\text{resistive}} = I^2 R$. The experimental setup consists of four PeHEX in series, each with their own power supply. Since every power supply is limited to drawing 480 Watts, the four power supply can be powered from a single 230 V_{AC}, 16 amps fused socket.

5.3. Heating network simulation

For the test setup, the Flamco workbench (used to test the Flamco HIU's 1.4) was used. The workbench can hold a water tank at a specific temperature, which was set to 45 °C for the measurements. The workbench can also provide a certain flowrate of the heated tank, and of the non heated water flow. As discussed in Section 1.1, the incoming domestic water can at least be pre-heated to 40 °C. To obtain the 40 °C water from the hot water tank the water flow from the hot water tank was split into two flows as visible from Figure 5.4, one flow was used to pre heat the domestic water to 40 °C and the other was used as the heating network.

The pre-heat of the domestic water is done by the HIU in Figure 5.4 by a PID controller. The PID controlled system will affect the differential pressures on the two split flows from the hot water tank. A differential pressure control valve (DPCV in Figure 5.4) was added between the supply and the return flow of the heating network that flows to the PeHEX to prevent oscillations in the flow rate.

5.4. Potential and current measurements

To measure the current running through the Peltier elements in the PeHEX, the Isabellenhütte ISA-plan Precision Resistor [19] of 0.02 Ω was used in series with the strings of Peltier elements and the power supply for every PeHEX (4 shunts in total). The potential difference over the shunt resistor provides the total current running through the system. Both strings of Peltier elements receive roughly half of the total current that is running. Variations are possible due to differences in temperature gradients over the Peltier elements along the length of the PeHEX. The potentials and resistances were measured with a 4 digits digital multimeter.

The total applied potential $V_{0,i}$ with subscript i indicating the corresponding PeHEX is measured directly from the terminals of the power supplies. Using the average resistance R_i of Peltier elements in PeHEX i and the current running through the elements in PeHEX i, the effective potential V_{eff} over the elements can be estimated and can therefor be used to calculate the temperatures of the boundaries of the Peltier elements $T_{p,dhw}$ and $T_{p,hn}$. These temperatures are needed to calculate the Nusselt number of the flow to correlate the measured values to the numerical values of the Nusselt number to verify the model (Chapter 7).

5.5. Total assembly

Along the length of the PeHEX eight temperature sensors are placed as indicated in figure 5.4. The temperature sensors are simple J-type thermocouples. The flow sensors (F1 and F2 in Figure 5.4) send 950 pulses per liter. The data acquisition (DAQ) is done by a DATAQ DI-2008 [16] with a custom Python interface (Code snippet E.2).
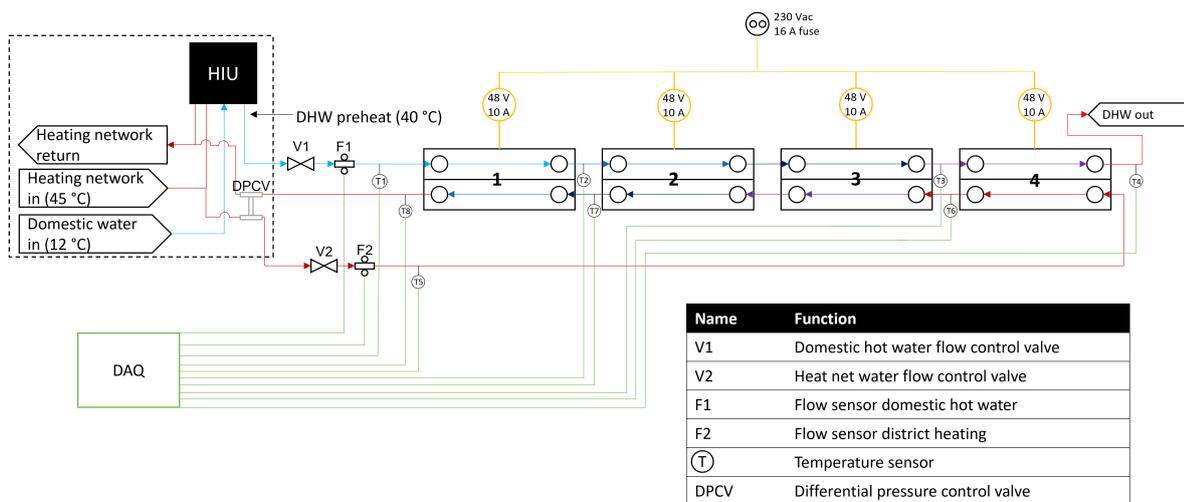
Figure 5.4: Schematic resemblance of the test setup. The dashed box around the HIU indicates the pre-heat of the domestic hot water. The green lines (going to the DAQ) represent data lines. The yellow lines (from the 230 Vac) represent power wires. The numbers in the PeHEXs are used for future reference.

Due to a limit on the available ports on the DAQ, no temperature sensors are placed between the second and third PeHEX. The temperatures in between will be estimated using a weighted average of T_2 and T_3 for the domestic hot water side, and T_6 and T_7 for the heating network side. The weights are the reciprocal of the power per element P_e in the respective PeHEX, since a higher power draw means a larger temperature differential over that PeHEX. The raw data from the measurements can be found in Table A.2. The mid-point temperatures are thus calculated as indicated in Equation 5.1.

Although the workbench mentioned in Section 5.3 is able to provide water at certain flow speeds, two extra manual valves (V1 and V2 in Figure 5.4) were added to have a more precise control over the flow speeds. The threaded holes in the perspex lid from Figure 5.1a are fit with 1/2" BSP to 1/4" nickel brated brass hose connectors, and PVC tubes are used to connect the four PeHEX together. Figure 5.5b shows two pictures of the PeHEX test setup.

$$T'_{dhw} = \frac{\frac{T_2}{P_{e,2}} + \frac{T_3}{P_{e,3}}}{\frac{1}{P_{e,2}} + \frac{1}{P_{e,3}}} = \frac{T_2 p_{e,3} + T_3 p_{e,2}}{p_{e,2} + p_{e,3}}$$

$$T'_{hn} = \frac{\frac{T_6}{P_{e,2}} + \frac{T_7}{P_{e,3}}}{\frac{1}{P_{e,2}} + \frac{1}{P_{e,3}}} = \frac{T_6 p_{e,3} + T_7 p_{e,2}}{p_{e,2} + p_{e,3}}$$

(5.1)

(a)   (b)
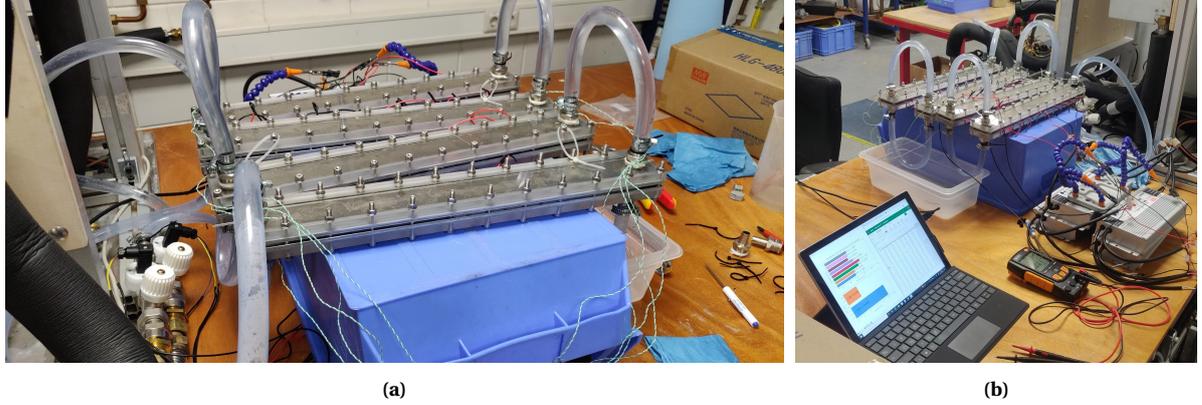
Figure 5.5: Test setup for the Peltier heat exchanger. Figure 5.5a: The four PeHEX are shown, on the left the two manual valves (white) V1 and V2 are shown, and the two flow sensors (black) F1 and F2 are shown. Figure 5.5b: PeHEX setup setup with the power supply, multimeter and data acquisition station.

5.6. Measurement decimation

The measurement devices, eight J-type thermocouples and two BIO-TECH FCH-midi-POM [3] flow sensors, provide a measurement by supplying a voltage or pulses respectively. The DI2008 Data acquisition box has 8 thermocouple ports, a digital rate channel and a digital count channel that are used. The thermocouples provide a voltage due to the Seebeck effect (Section 2.8). The voltage-temperature characteristic of K-type thermocouples is known and therefor the measured voltage can be converted to a temperature difference between the probe and the DI2008. Since the DI2008 has an internal reference temperature, the temperature at the probes can be determined.

The thermocouple probes are squeezed between the 2 mm thick PVC hose and the 1 mm thick brass hose connector as shown in Figure 5.6. This means the probe essentially measures the outside temperature of the brass hose connector. The thermal resistances to radial heat conduction of the brass and the PVC are calculated using Formula 2.3 and are $R_{brass} = 0.0003$ mK/W ($k_{brass} \approx 100$) and $R_{PVC} = 0.2$ mK/W ($k_{PVC} \approx 0.2$) with $r_0 = 6, r_1 = 7, r_2 = 9$ mm. The contribution to the temperature difference across the water and the outside temperature of the brass is given by $\frac{R_{brass}}{R_{brass} + R_{PVC}}$ and is roughly 1% indicating that the temperature difference across the brass will be relatively small and therefor the temperature as measured by the thermocouple probe will be used as the temperature of the water.

r_2 r_1 r_0
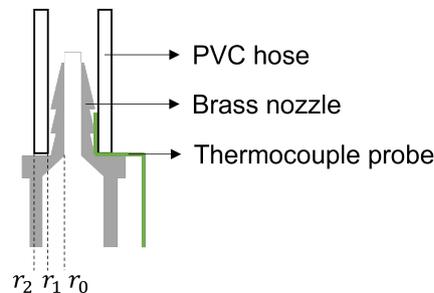
Figure 5.6: Schematic of the squeezed thermocouple probe between the PVC hose and the brass hose connector.

Due to random noise, thermal oscillations and possible PID conroller effects from the HIU, the voltage that the thermocouples provide can oscillate giving an unreliable measurement. To circumvent this problem the

measurement are averaged over time, set by the *decimation* variable in Code snippet E.2. The sampling rate of the DI2008 is 200 Hz divided over 8 thermocouple channels giving a sampling rate of 25 Hz per thermocouple channel. The measurements are averaged over 1500 samples, or 60 seconds of data.

The flow sensor of the heating network was connected to the *rate* channel of the DI2008. The flow sensor of the domestic hot water was connected to the digital *count* channel. The flow sensors sent square waves with 50% duty cycle (essentially pulses) to the DI2008 at a frequency of 950 pulses per liter. At 5 L/min the frequency at which data was sent to the DI2008 is 80 Hz. The sampling frequency of the DI2008 for the digital count channel is 200 Hz, so the Nyquist criterion for no aliasing sampling was satisfied. The data from the flow sensor connected to the rate channel is automatically converted to a frequency by the DI2008 python interface. The data from the flow sensor connected to the count channel had to be converted to a rate value. The DI2008 increments the counter when a potential is applied. This was done by simply dividing the difference in count value by the difference in time between when two data packages were sent from the DI2008 to the computer. The decimation of the flow sensors data was set to 10 samples, or 0.4 seconds, to reduce measurement noise but maintain an accurate reading of the current flow speed.

5.7. Measurement error

All sensors have a certain measurement uncertainty. The measurement uncertainties for the sensors used in this research are given in Table 5.1.

Table 5.1: Measurement errors due to sensors and measurement techniques.

Quantity	V_{dhw} (L/min)	V_{hn} (L/min)	V_e (V)	I_e (A)	T (°C)
Measurement error	0.05	0.01	0.005	0.025	0.5

The data from the flow sensors is decimated by keeping track of the last ten measurements and taking the average. From the last ten measurements, a standard deviation can be calculated. The calculated standard deviations for the flowsensors were 0.05 L/min for the domestic hot water (connected to the count channel of the DI2008) and 0.01 L/min for the heating network (connected to the rate channel of the DI2008). The count channel has a higher uncertainty than the rate channel. This is likely explained by random noise on causing the counts to increment randomly.

The measurement error for the potential V_e and I_e come from the multimeter. The multimeter is a 4 digit digital multimeter. The total PeHEX applied electric potential difference measurements are done with a precision of two decimals, the uncertainty is therefore 0.005 V. The current measurements are done by calculating the potential over the shunt resistor, the potential over the shunt is with an precision of three decimals meaning the uncertainty is 0.0005 V. The uncertainty in the current is therefor $dI = dV/R = 0.0005/0.02 = 0.025$ A (Section 5.7.1).

The uncertainty in the temperature measurements is provided by the manufacturer's datasheet and is 1.5 °C. However, due to the decimation of 1000 samples this error is reduced to $1.5/\sqrt{1000} = 0.05$ °C. The 1000 samples from the decimation can be used to obtain the measurement error. The uncertainty in the temperature measurements was found to be 0.5 °C.

5.7.1. Measurement uncertainty propagation

The error propagation of (multi-variable) functions $f(X,Y,...)$ is given by Hughes and Hase [13, Chap. 4] as:

$$u\big(f(X,Y,...)\big) = \sqrt{\left(\frac{\partial f(X,Y,...)}{\partial X}u(X)\right)^2 + \left(\frac{\partial f(X,Y,...)}{\partial Y}u(Y)\right)^2 + ...} \tag{5.2}$$

where $u(f)$ indicates the total error in $f(X,Y,...)$ due to the uncertainties $u(X), u(Y),...$ in the variables. In terms of the uncertainty in the current $I = I(V) = V/R_{shunt}$ this gives:

$$u(I) = \sqrt{\left(\frac{dI}{dV}u(V)\right)^2}$$
$$= \sqrt{\left(\frac{1}{R}\cdot 0.0005\right)^2}$$
$$= \frac{0.0005}{0.02} = 0.025 \text{ A}$$

6

Experimental Results

In this section the experimental results will be presented and discussed. The experimental results will be compared to the model predictions from Chapter 4. Using the experimental observations, the model parameters S and $k(T)$ will be re-evaluated in Chapter 7.

A total of 25 measurements were done (Table A.2) with increasing domestic hot water and heating network flow speeds (ϕ_{dhw} and ϕ_{hn} respectively) ranging from 1 L/min to 5 L/min as indicated in Table 4.1.

The electric potential supplied to each PeHEX was set to the maximum value of the power supplies of 49.92 V, 48.88 V, 46.11 V and 50.3 V respectively. The results from Table A.2 are summarized in Figures 6.1c, 6.1b, 6.1a and 6.1d.

6.1. Power absorption and rejection

The supply and return temperatures of the PeHEX were measured. Using these temperatures the power absorbtion of the domestic hot water, and the power rejection of the heating network can be calculated. Figure 6.1a shows the power absorption by the domestic hot water as calculated by $\dot{Q}_{dhw} = \dot{m}_{dhw} c_p (T_4 - T_1)$ with $\dot{m}_{dhw} = \rho \phi_{dhw}$ where $\rho = 1$ kg/L, $c_p = 4180$ the specific heat capacity and T_1 and T_4 as defined in Figure 5.4. Figure 6.1b shows the power rejection by the heating network as calculated by $\dot{Q}_{hn} = \dot{m}_{hn} c_p (T_5 - T_8)$ with $\dot{m}_{hn} = \rho \phi_{dhw}$ and T_1 and T_4 as defined in Figure 5.4.

(a) Calculation of the power absorption of the domestic hot water $\dot{Q}_{dhw}$.

(b) Calculation of the power rejection of the heating network $\dot{Q}_{hn}$.
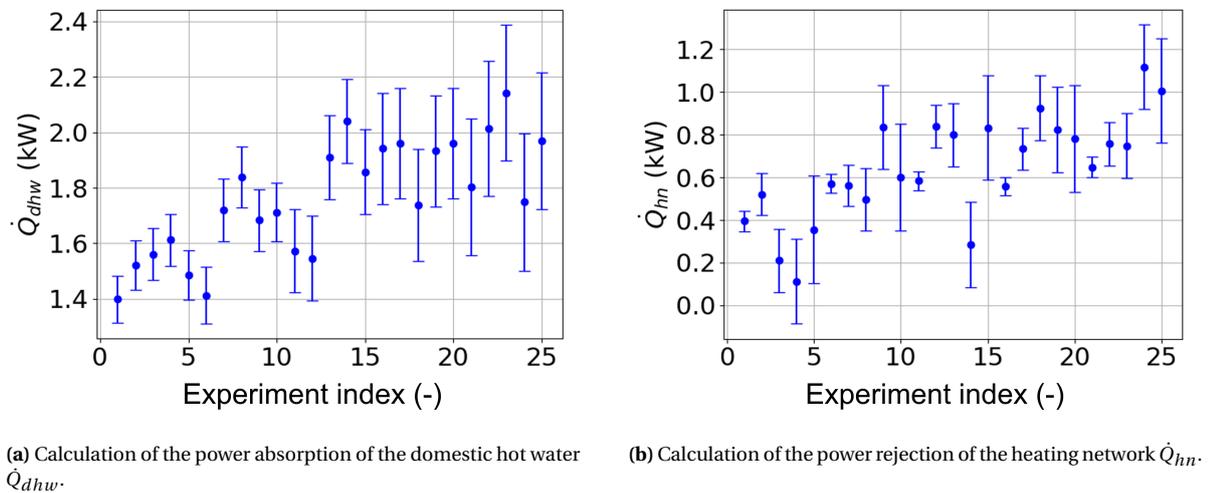
Figure 6.1: Visualised measurement data from Table A.2.

The measured domestic hot water flow rates and the heating network flow rates do not exactly match the values used in the numerical simulation, although they differ only by 1%. Therefor, the experimental results can still be compared to the numerical simulations, and an increasing experiment index still corresponds to increasing water flow speeds. Figures 6.1a and 6.1b suggest that both the domestic hot water power absorption and the heating network power rejection increase as the experiment index increases, as the model predicted.

The error bars in Figures 6.1a increase as the domestic hot water flow speed increases. The error bars in Figures 6.1b increase as the domestic hot water flow speed increases. The uncertainty in the flow speed is 5%

maximum for the domestic hot water and 1% for the heating network, explaining the differences in the error bars between the domestic hot water and the heating network at the same experiment index.

As the flow speed increases, the temperature difference across the flow decreases. Since the measurement error in the temperature measurements is fixed, the effects of the uncertainties in the temperature on the total power becomes larger for smaller temperature differences. The uncertainty in a temperature difference is $\sqrt{2}dT \approx 0.7$ °C. At a temperature difference of 20 °C the uncertainty is 3.5%. At a temperature difference of 6 °C, the uncertainty is 12 %!. Therefore, at higher flow speeds (lower temperature differences) the relative uncertainty in the temperature difference is much larger than the temperature difference in the flow speed, causing the total uncertainties in $\dot{Q}_{dhw}$ and $\dot{Q}_{hn}$ to increase.

6.2. Electric power draw and heat loss

The electric power draw by the Peltier elements increases as the experiment index increases, shown in Figure 6.1c, as the simulation results from Chapter 4 predicted.

(c) Calculation of the effective electrical input power P.   **(d)** Calculation of the heat loss of the system $\dot{Q}_{loss}$.
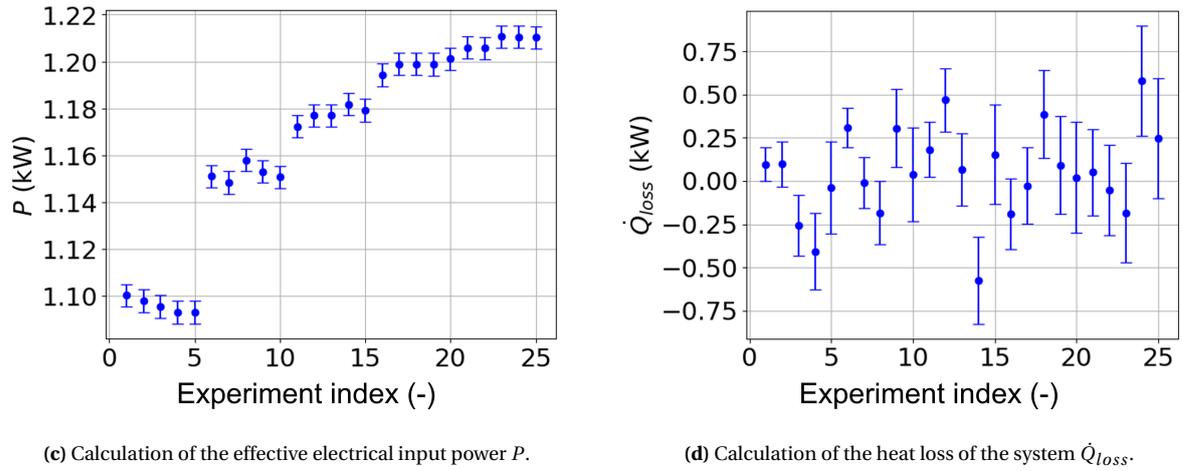
Figure 2.1 continued: Visualised measurement data from Table A.2.

The heat lost by the PeHEX is due to natural convection to the air and due to radiative heat losses. The convective heat loss is modelled by using Formula 4.85 from Mills and Coimbra [25]:

$$
\begin{aligned}
\text{Nu} &= 0.68 + 0.67(\text{Ra}\Psi)^{(1/4)} \\
\text{Ra} &= \frac{\beta \Delta T g H^3}{\nu \alpha} \\
\Psi &= \left[1 + \left(\frac{0.492}{\text{Pr}}\right)^{9/16}\right]^{-16/9}
\end{aligned}
\tag{6.1}
$$

With Ra the Rayleigh number [25, eq. 4.29], Ψ a Prandtl number function from Mills and Coimbra [25, eq. 4.84], $\beta \approx 0.003 \text{ K}^{-1}$ the thermal expansion coefficient of air, $\Delta T \approx 40$ the temperature difference between the PeHEX and the air, $g = 9.81$ the gravitational acceleration, $H = 0.01$ m the combined height of the metal parts of the PeHEX, $\nu = 1.5 \cdot 10^{-5} \text{ m}^2/\text{s}$ the kinematic viscosity of air, $\alpha \approx 20 \cdot 10^{-6} \text{ m}^2/\text{s}$ the thermal diffusivity of air and Pr= 0.7 the Prandtl number of air. The Nusselt relation holds for Ra $\leq 10^9$. Combining these values the Rayleigh number is found to be 7100, and a value of 5.4 was found for the Nusselt number. The convective heat transfer coefficient is calculated as $h = \text{Nu}\frac{k}{H} = 11 \text{ W/m}^2\text{K}$. The heat transfer area is $A = 0.01 \text{ m}^2$. The convective heat loss is therefor calculated as $Q_c = hA\Delta T = 7$ W for each heat exchanger.

The radiative heat loss is calculated by $Q = h_r A\Delta T$ [25, e1. 1.19] where $h_r = 4\epsilon\sigma T_m^3$ with $\epsilon = 0.5$ the radiative emission coefficient, $\sigma = 5.67 \cdot 10^{-8} \text{ W/m}^2\text{K}^4$ and $T_m \approx 324$ K such that the radiative heat loss is $Q = hA\Delta T = 2$ W. The total heat loss per heat exchanger is therefor 9 watts which is a relative heat loss of $9/350 \approx 2\%$ where 350 watts is the nominal power of the power supply of a single PeHEX.

A potential power gain of 2 % means an extra potential temperature increase of 0.4 °C. This temperature increase is in the order of magnitude of the temperature measurement uncertainty. The temperature loss due to convective and radiative heat losses so small that they can be neglected. From Figure 6.1d it is clear that in some measurements there was a significant imbalance in the conservation of energy: $P_{in} = \dot{Q}_{dhw} - \dot{Q}_{hn}$.

6.2.1. Heat loss

Figure 6.1d indicates that in some measurements there was a gain of energy ($\dot{Q}_{loss} < 0$) rather than an energy loss to the environment. The energy loss is given as $\dot{Q}_{loss} = \sum_{n=1}^{4} \left(P_n - (\dot{Q}_{dhw,n} - \dot{Q}_{hn,n}) \right)$ where the sum is over the Peltier Heat Exchangers. The negative energy loss is an indication that the system was not stabilized when the measurement was taken. As mentioned before, the heat loss is negligible. Therefore a regression was performed on the domestic hot water power absorption and the heating network power rejection to reduced the effects of the energy imbalance for later analysis of the power absorption and rejection (Chapter 7). The chosen regression variables are the *Reynolds number* for the domestic hot water flow Re_{dhw}, the *Reynolds number* for the heating network flow Re_{hn} and the *relative power number* given by Equation 6.2:

$$P_{relative} = P_{effective}/P_{nominal} = (I^2 R)/(V_0^2/R) = (IR/V_0)^2 \tag{6.2}$$

The domestic hot water power absorption was non-dimensionalized with the nominal power, which essentially is the COP of the system. This gives:

$$\frac{\dot{Q}_{dhw}}{P_{nominal}} = \beta_1 \cdot Re_{dhw} + \beta_2 \cdot P_{relative} + \beta_3 \cdot Re_{hn} \tag{6.3}$$

By the conservation of energy, $\dot{Q}_{\text{dhw}} = \dot{Q}_{\text{hn}} + P_{\text{effective}}$, therefore

$$\frac{\dot{Q}_{hn} + P_{effective}}{P_{nominal}} = \beta_1 \cdot Re_{dhw} + \beta_2 \cdot P_{relative} + \beta_3 \cdot Re_{hn} \tag{6.4}$$

Combining equations 6.3 and 6.4 gives a system of equations 6.5 that must hold for each measurement:

$$\begin{bmatrix} Re_{dhw} & P_{effective}/P_{nominal} & Re_{hn} \\ Re_{dhw} & P_{effective}/P_{nominal} & Re_{hn} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} = \begin{bmatrix} \dot{Q}_{dhw}/P_{nominal} \\ (\dot{Q}_{hn} + P_{effective})/P_{nominal} \end{bmatrix} \tag{6.5}$$

Since each measurement provides four Equation 6.5-like system of equations (one for each Peltier Heat Exchanger), where the regressions parameters β_1, β_2 and β_3 are the same for each system, this provides 200 equations for performing the regression on. Table 6.1 displays the calculated values for the regression parameters as a the least squares solution to the total system of equations 6.5. The uncertainties in the regression parameters are determined according to Hughes and Hase [13, Sec. 7.2].

Table 6.1: Values for the regression parameters β_1, β_2 and β_3.

Parameter	β_1	β_2	β_3
Value	$5.4 \cdot 10^{-5}$	1.27	$3.0 \cdot 10^{-5}$
Standard deviation	$2.9 \cdot 10^{-5}$	0.14	$2.2 \cdot 10^{-5}$
Relative error	0.40	0.11	0.73

An eye catching observation from Table 6.1 is that parameters β_1 and β_3 are five orders of magnitude smaller than β_2. Typical value for the variables corresponding to the regression parameters are $\bar{x}_1 = 3000$ (Reynolds number), $\bar{x}_2 = 0.8$ (relative power number) and $\bar{x}_3 = 3000$ (Reynolds number) for β_1, β_2 and β_3 respectively. The relative influence of each variable is given by: $\bar{x}_1 \beta_1 : \bar{x}_2 \beta_2 : \bar{x}_3 \beta_3 = 0.16 : 1.0 : 0.09$. These ratios give the impression that in order to increase the power absorption of the domestic hot water flow it is seven times more effective to increase the power draw by the elements than increasing the Reynolds number of the domestic hot water flow, and fifteen times more effective than increasing the Reynolds number of the heating network.

The relative errors in the parameters β_1 and β_3 are $d\beta_1/\beta_1 = 0.40$ and $d\beta_3/\beta_3 = 0.73$. Although these relative errors may seem substantial, their relative contributions to the total error are $\bar{x}_1 d\beta_1 : \bar{x}_2 d\beta_2 : \bar{x}_3 d\beta_3 = 0.85 : 1.0 : 0.65$ which indicates that the uncertainties in β_1 and β_3 do not dominate the total error.

It is important to note that the regression only indicates a correlation between $\dot{Q}_{dhw}$, Re_{dhw}, $P_{\text{effective}}/P_{\text{nominal}}$ and Re_{hn} and not an analytic relation. In fact, all the different variables are dependant on each other. For example: The relative power number indicates the power drawn by a single element, which is dependant on the temperature difference across the Peltier element, which in turn is connected to the domestic hot water power absorption and the in-going temperature of the domestic hot water and the Reynolds number of the domestic hot water.
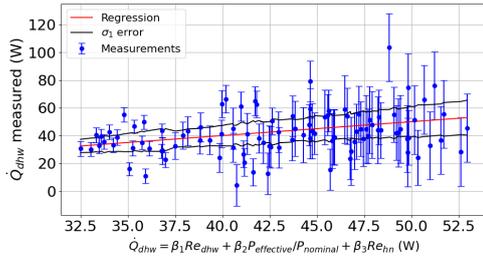
(a) Comparison of the measured domestic hot water power absorption with the regression values.
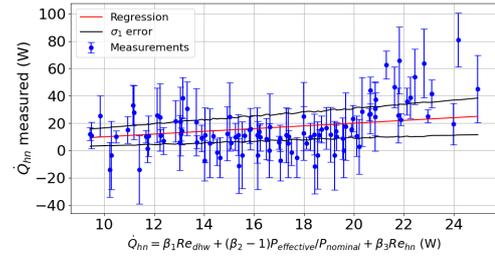
(b) Comparison of the measured heating network power rejection with the regression values.

Figure 6.2: Comparison between the Measured and calculated (with the regression parameters) power absorption and rejection.

Figure 6.2 shows the comparison between the measured values (dots with error bars) for $\dot{Q}_{dhw}$ and $\dot{Q}_{hn}$ on the one hand, and the values for $\dot{Q}_{dhw}$ and $\dot{Q}_{hn}$ as calculated with the regression parameters (linear line with σ_1 error bars). Figures A.1a, A.2a, A.3a and A.4a show how the regression compares to the measurements on the respective arrays of the Peltier elements. The domestic hot water power absorption and the heating network power rejection as calculated with the regression parameters will be used in Chapter 7 to re-evaluate the model parameters.

6.3. Comparison with simulation results

Figure 6.3 shows a comparison between the measurement results and the simulations from Section 4. Figure 6.3d shows that the COP value is between 1.2 and 1.8 for the measurements.
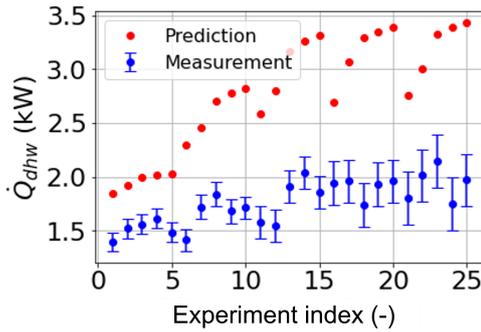
(a) Comparison of measurements and simulated predictions for domestic hot water power absorption.
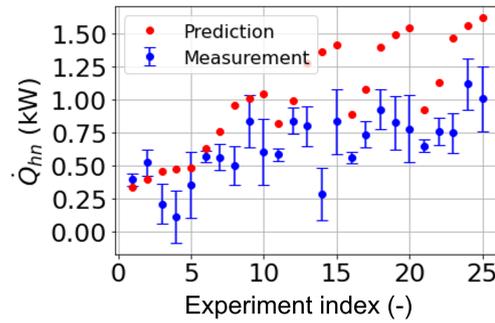
(b) Comparison of measurements and simulated predictions for heating network rejection.
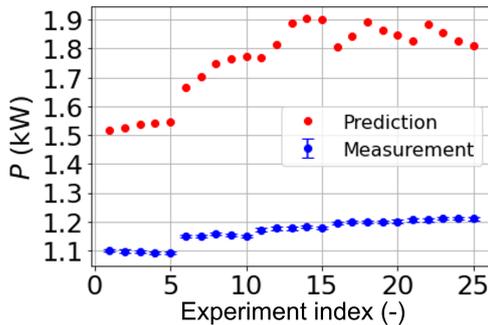
(c) Comparison of measurements and simulated predictions for electric power use.
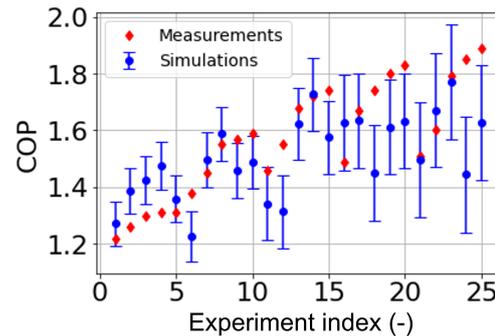
(d) Calculated COP from the measurement data.

Figure 6.3: Results of the simulation of the measurements.

An immediate observation from Figures 6.3a, 6.3b and 6.3c is that the prediction does not accurately represent the measurements. The differences between the simulations and the observations of the power absorption and rejection are between 30% and 50%. The power input differs between 50% and 75%. Although

the COP calculations in Figure 6.3d do not seem to follow the 30-50% difference between measured and calculated value, it seems there is a larger dependence on experiment index in the simulated results than the experimental observations show.

An interesting observation from Figure 6.1c is that the qualitative increase in the total power draw as the experiment index increases matches the model prediction. The model simulations were done using a resistance of $R = 1.57$ Ω per Peltier element. The measured values for the average Peltier element resistance per PeHEX are given in Table 8.1. The average Peltier element resistance is $R = 2.85$ Ω. This difference in resistance lowers the power draw by 45%. The resistance and the Seebeck coefficient of a Peltier element were obtained from the least squares fit in Chapter 3. Since the measured resistance value does not match the fitted value, this suggests that the fitted Seebeck coefficient is also different from the actual value. These parameter S and $k(T)$ will be re-evaluated in Section 7.1 using the measurement data.

7

Numerical Adjustment

As mentioned Chapter 6, the numerical simulations and the experimental observations differ by 30% to 75%. To improve relation between the numerical simulations and the the experimental observation the model parameters S and $k(T)$ will be re-evaluated. Also a new correlation for the Nusselt number will be fitted.

7.1. Computation of Seebeck coefficient

The electric potential difference reduction ($S\Delta T_p$) will be used to re-evaluate the Seebeck coefficient S. The reduction is calculated using Formula 3.9. The measured electric potential reductions are shown graphically in Figure 7.1a:
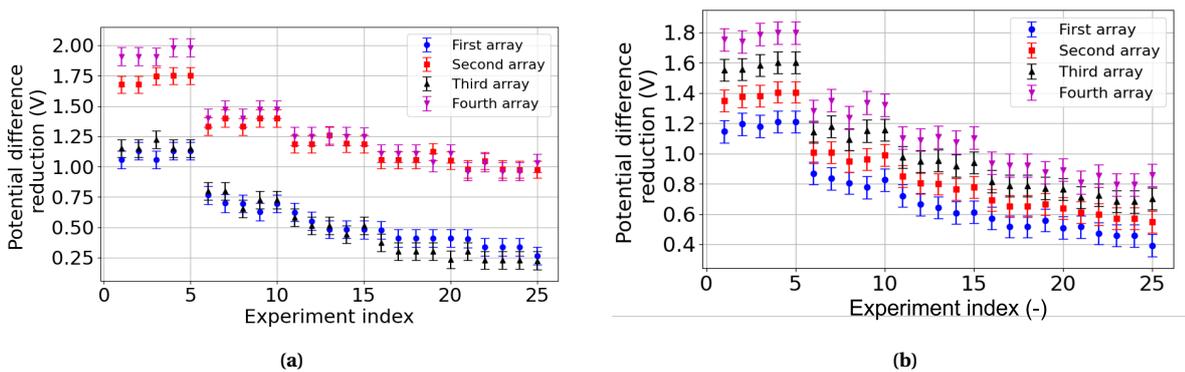
Figure 7.1: Figure 7.1a: Reduction in potential per array for each measurement. Figure 7.1b: Adjusted reduction in potential per array for each measurement.

The second and fourth PeHEX seem to have a substantially higher potential reduction (~ 0.75 V) than the first and third PeHEX. The temperature difference over the Peltier elements is expected to rise from first to fourth PeHEX, and therefore the potential difference reduction is expected to increase from first to fourth PeHEX. The extra potential reduction probably arose due to poor soldering joints, or poor connections to the power supplies.

To try to eradicate the discrepancies in the potential reduction, the potential difference reductions for every experiment index were distributed linearly from PeHEX 1 to PeHEX 4 such that the electric potential reduction increases from PeHEX 1 to PeHEX 4 as expected. The new adjusted potential difference reduction are shown in Figure 7.1b.

To calculate the new value for S, the temperature differences over the Peltier elements need to be calculated as will be discussed in Section 7.1.1. The final value of S is calculated by iteration over S, taking the fitted value from the datasheet $S_{datasheet} = 0.0477$ V/K as a starting point.

7.1.1. Calculating Peltier element temperatures

The reduced electric potential difference is used to calculate the temperature difference over the Peltier element according to $T_{p,dhw} - T_{p,hn} = \Delta V / S$. The actual values for $T_{p,dhw}$ and $T_{p,hn}$ are determined by distributing the Peltier element temperature difference $T_{p,dhw} - T_{p,hn}$ in such a way that the ratio of the temperature differences between the Peltier element's boundary and the respective flows $(T_{p,dhw} - T_{dhw}) : (T_{hn} - T_{p,hn})$ is equal to the ratio $\dot{Q}_{dhw} : \dot{Q}_{hn}$. The power rejection at the domestic hot water side is usually larger than the

power absorption on the heating network side, this is because the dissipation I^2R is pumped towards the hot side of the Peltier element relative to the cold side.

The new value for S was recalculated by minimizing the square norm $\mathbf{J} \cdot \mathbf{J}$ of System of non-linear equations 7.1 using a bisection algorithm on S. The elements of vector $\mathbf{J}$ represent:

- The power difference at the domestic hot water side between the measured power absorption and the modelled power absorption.

- The power difference at the heating network side between the measured power rejection and the modelled power rejection.

- The difference between the measured electrical power input and the modelled electrical power draw.

$$\mathbf{J}(S, T_{p,dhw}, T_{p,hn}) = \begin{bmatrix} IST_{p,dhw} + \frac{1}{2}I^2R - k\frac{A}{d}(T_{p,dhw} - T_{p,hn}) - \dot{Q}_{dhw} \\ IST_{p,hn} - \frac{1}{2}I^2R - k\frac{A}{d}(T_{p,dhw} - T_{p,hn}) - \dot{Q}_{hn} \\ IS(T_{p,dhw} - T_{p,hn}) + I^2R - P \end{bmatrix} \qquad (7.1)$$

The value for S has to be found by iteration. After each iteration, the new value for S was taken as the average between the old value for S and the value S_{min} that minimizes the square norm $\mathbf{J} \cdot \mathbf{J}$. The new value for S was then used to re-evaluate the temperature difference over the Peltier elements $T_{p,dhw} - T_{p,hn} = \Delta V / S$, and $T_{p,dhw}$ and $T_{p,hn}$ were determined for the new S. This iteration continues until the values of S converge to 0.1% of the previous value of S. The standard deviation of S can be approximated by $\sigma_S = \frac{\mathbf{J} \cdot \mathbf{J}}{(\mathbf{J} \cdot \mathbf{J})^T (\mathbf{J} \cdot \mathbf{J})\sqrt{N}} = \frac{1}{\sqrt{100}(\mathbf{J} \cdot \mathbf{J})}$ (Standard deviation from a population). Figure 7.2 shows the calculated value of S over several iterations.
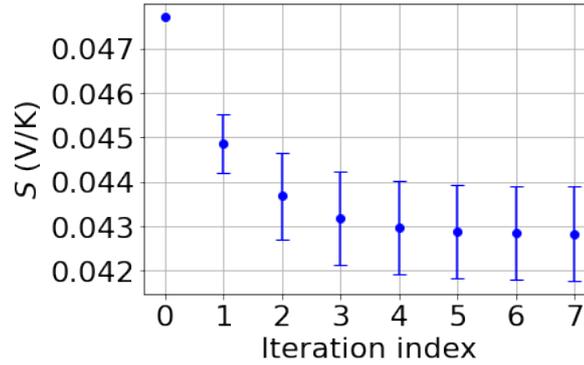
Figure 7.2: Calculated value of S after several iterations

The final calculated value is $S = 0.0428 \pm 0.0011$ V/K. The value for S as fitted to the datasheet is $S_{\text{fit}} = 0.0477$ V/K. The value for S as calculated from the measurements is surprisingly close to the fitted value for S which suggests that the fitted value from the datasheet could be correct. However, the value fitted from the datasheet is 4 standard deviations away. This means that the probability that the calculated value for $S = 0.0428$ is due to uncertainty in the measurement data and the real value is actually $S = 0.0477$ is very small.

7.2. Re-evaluation of the thermal conductivity of the Peltier element

Using the new value of S the thermal conductivity of the Peltier element can be re-evaluated using Equation 7.2:

$$k(\overline{T}_p) = \frac{IST_{p,dhw} + \frac{1}{2}I^2R}{\frac{A}{d}\Delta T_p}$$

$$k(\overline{T}_p) = \frac{IST_{p,hn} - \frac{1}{2}I^2R}{\frac{A}{d}\Delta T_p}$$

$$(7.2)$$

where $\Delta T_p = T_{p,dhw} - T_{p,hn}$, $\overline{T}_p = (T_{p,dhw} - T_{p,hn})/2$, I the current, $S = 0.0428$ the Seebeck coefficient, R the Peltier element's resistance, A the Peltier element's area and d the thickness of the Peltier element. Fitting

the model 3.3, the parameters A and B are $A = \exp(15)$ W/mK and $B = -0.046$ 1/K. Formula 3.4 can thus be rewritten as:

$$k(T) = \exp\left(B(T - T_0)\right)$$

with $T_0 = -A/B$. Figure 7.3 shows the relation between the measured thermal conductivities and the fitted model.
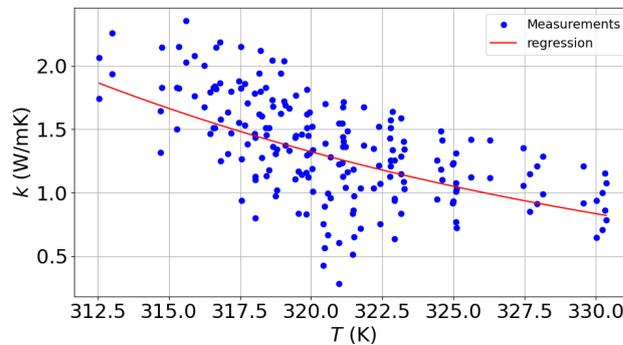
Figure 7.3: Relation between the measured thermal conductivities and the fitted model

The uncertainties in $k(T)$ are not shown as they are several orders of magnitude bigger than the actual values for $k(T)$. This is because the regression parameter A is the argument of an exponential. A measurement error of 10% in A causes an uncertainty in $k(T)$ of $\exp(1.1) \approx 3$ times the actual value of $k(T)$. As mentioned in Chapter 3 the exponential relation for the thermal conductivity was purposely chosen to prevent zero and negative thermal conductivities. Therefore the uncertainty in k(T) is accepted, as the calculated values for $k(T)$ match the measurement results to a degree similar to the theoretical relation.

7.3. Calculation of Nusselt number

Figure 7.4 shows the poor correlation between measured Nusselt number and the Gnielinski correlation for the Nusselt number. This is probably caused by the relation used for the friction factor in Equation 3.18 which seems not applicable for the PeHEX. Therefore, a new Nusselt relation is determined.
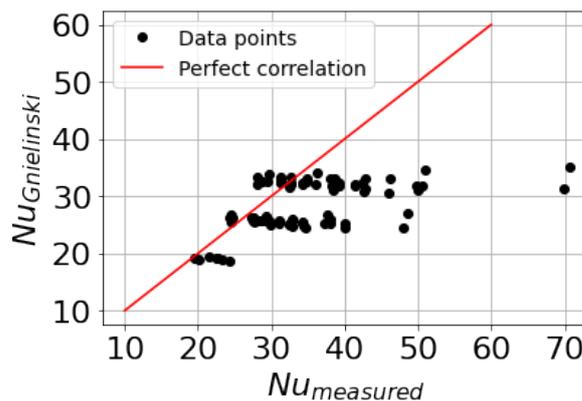
Figure 7.4: Comparison between the Gnielinsky correlation and the measured Nusselt numbers for Re > 3000.

In order to calculate the Nusselt numbers corresponding to the measurements, the heat transfer coefficient h needs to be determined from the total heat transfer coefficient U. The total heat transfer coefficient is, according to the theory from Section 3.3), expected to be a strong function of the Reynolds number (Re) and the Prandtl number (Pr) for turbulent flows (Re > 2300), and a weak function of the Reynolds number and Prandtl number for laminar flows (Re < 2300). The relation between U, Re and Pr can be determined by obtaining U from Equation 7.3:

$$
\begin{aligned}
\dot{m}c_p\Delta T_{dhw} &= \int_0^L UP(T_{p,dhw} - T_{dhw})dx \\
&= \overline{U}(\mathrm{Re},\mathrm{Pr})P\int_0^L (T_{p,dhw} - T_{dhw})dx \\
&= \overline{U}(\mathrm{Re},\mathrm{Pr})A\overline{(T_{p,dhw} - T_{dhw})} \\
\dot{m}c_p\Delta T_{hn} &= \int_0^L UP(T_{hn} - T_{p,hn})dx \\
&= \overline{U}(\mathrm{Re},\mathrm{Pr})P\int_0^L (T_{hn} - T_{p,hn})dx \\
&= \overline{U}(\mathrm{Re},\mathrm{Pr})A\overline{(T_{p,hn} - T_{hn})}
\end{aligned}
\tag{7.3}
$$

In which the bar indicates a spatial average over a single PeHEX. The only unknown in Equation 7.3 is $\overline{U}(\mathrm{Re},\mathrm{Pr})$.

7.3.1. Calculating the convective heat transfer coefficient

The only unknowns in Equation 7.3 are the total heat transfer coefficients U and can therefor easily be calculated. From U, the values of h, the heat transfer coefficient of the water, can be calculated as given in Equation 7.4:

$$
\frac{1}{h} = \frac{1}{U} - \frac{d_{Al}}{k_{Al}}
\tag{7.4}
$$

Where $d_{Al} = 0.001$ m is the thickness of the aluminium base plate, and $k_{Al} = 238$ W/mK is the thermal conductivity of the aluminium base plate [35]. The Nusselt number is now easily calculated as $\mathrm{Nu} = \frac{hD}{k_{water}}$ with $D = 0.00556$ m the hydraulic diameter of the channels and k_{water} as defined in E.1. From the theory, the expected relation between the Nusselt number, Nu, and the Reynolds number, Re, and the Prandtl number, Pr, are non trivial and the relation to be used depends on the type of flow. However, even though the Reynolds number is in the laminar flow regime (Re < 2300) entrance effects will cause turbulent mixing throughout the heat exchanger since the entrance length for laminar flow is $L_{laminar} = 0.0575\mathrm{Re}_D D \approx 0.7$ m for Re = 1250 and the length of the heat exchanger is only 0.4 m [32]. This indicates that entrance effects cause turbulent mixing in the PeHEX for Re > 1250. A model of $\mathrm{Nu} = f\mathrm{Re}^{\alpha}\mathrm{Pr}^{\beta}$ was fitted to the measured values of Nu. Table 7.1 shows the fitted values for the parameters f, α, β. Figure 7.5 shows a plot of the calculated Nusselt number from the measurements, and the fitted values for Nu.
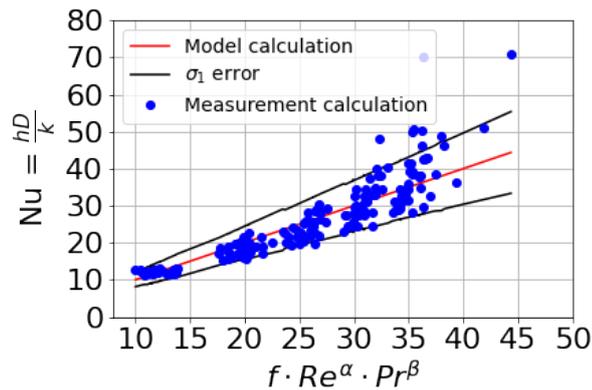
Figure 7.5: Plot of measured Nusselt numbers versus fitted Nusselt numbers.

This relation was fitted to data with $738 \le \mathrm{Re} \le 5005$ and $3.1 \le \mathrm{Pr} \le 4.9$. What is interesting about Figure 7.5 is the magnitude of the error in the fitted Nusselt numbers, compared to the relative errors in the fitted parameters. This difference in order of magnitude can be explained by considering that error $\partial\alpha$ in parameter α propagates through as $\mathrm{Re}^{\partial\alpha}$ which is roughly a factor of 1.7 for the range of Reynolds numbers of interest. A same reasoning holds for the error in the Prandtl number. Since the errors do not add linearly, a total error of one times the Nusselt number instead of four times the Nusselt number is found. Since the Nusselt number is tool for engineers to get an estimate of the heat transfer coefficient of a fluid, the calculated values for the

Table 7.1: Values for the regression parameters f, α and β.

Parameter	f	α	β
Value	$2.71 \cdot 10^{-2}$	0.608	1.4
Standard deviation	$0.03 \cdot 10^{-2}$	0.02	0.13
Relative error	0.01	0.03	0.09

Nusselt number will be used without the errors. A p-value (the probability that the fitted relation for Nu is actually due to chance) of $1.3 \cdot 10^{-7}$ is found for the fit, indicating that the model seems adequate to predict the Nusselt number for the given Peltier Heat Exchangers.

<div align="right">

8

</div>

Model Validation

8.1. Re-evaluation of numerical results

Using the fitted value for $S = 0.0428$, the measured values for R, and the model for the Nusselt number, the model simulations can be re-calculated and compared with the measurements. The new simulation results are shown in Figures 8.1a, 8.1b and 8.1c.

(a) Measurement and prediction of the power absorption of the domestic hot water $\dot{Q}_{dhw}$.

(b) Measurement and prediction of the power rejection of the heating network $\dot{Q}_{hn}$.

(c) Measurement and prediction of the effective electrical input power P.
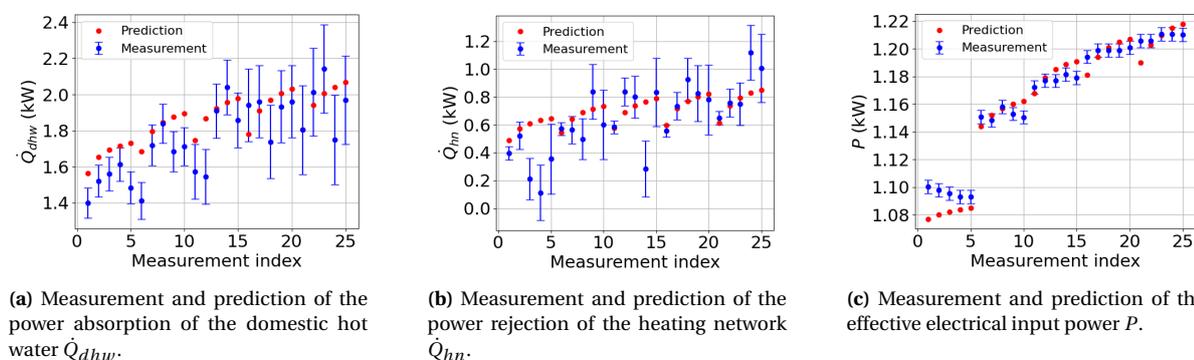
Figure 8.1: Comparison of the simulations with new parameters and measured results.

The simulated results for $\dot{Q}_{dhw}$, $\dot{Q}_{hn}$ and P are much more in line with the experimental observations than they were in Figure 6.3. This indicates that the model for the PeHEX, in combination with the values for S and R could be used to model the used Peltier Heat Exchangers. The value of R on the datasheet of $R = 2.2\Omega$ was not an accurate value for the resistance of an element, but an average Peltier element resistance of $R = 2.852\Omega$ was measured.

8.2. Coefficient of performance

Figure 8.2 shows the comparison between the calculated COP from the measurements and simulated COP:
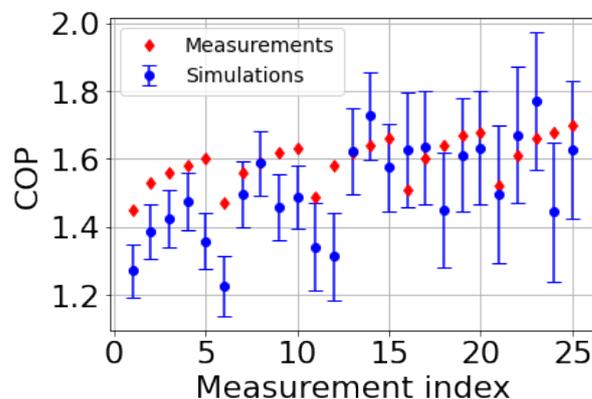
Figure 8.2: Comparison between calculated COP from the measurements, and simulated COP.

47

Figure 8.2 shows that the adjusted model predicts a COP that is close to the measured COP. Figure 6.3d shows a COP simulation that is a stronger function of the experiment index than the COP simulation from Figure 8.2. Figure 6.3d gives the impression that the domestic hot water flow speed ϕ_{dhw} has a larger influence on the COP than the heating network flow speed ϕ_{hn}. The domestic hot water flow speed is not a variable that can be controlled in a final product, it is demanded by the user. As mentioned in Section 6.2.1 it is two times more effective to increase the domestic hot water flow speed ϕ_{dhw} than it is to increase the heating network flow speed ϕ_{hn}, in order to increase the COP. This is visible from Figure 8.2 by comparing experiment index 1 and 2 (increase in ϕ_{dhw}) to 1 and 5 (increase in ϕ_{hn}). The increase from experiment index 1 to 5 is about 2% (1.45 to 1.47), while the increase from experiment index 1 to 2 is about 4% (1.45 to 1.53).

8.2.1. Return on investment

A major difference between a PeHEX and an air-water heat pump for domestic use is that the heat extracted from the outside air is free of charge. This means that a heat pump with a COP of 3, hypothetically cuts the domestic heating bill by a factor 3. The PeHEX used the heat from a heating network, for which it is likely that extracted heat from the network has to be paid for. Therefor, it is not immediately clear what the potential savings of a PeHEX could be. The Return On Investment (ROI) gives an idea for the potential savings:

$$\text{ROI} = \frac{c_0 Q_{dhw}}{c_0 \cdot E_{in} + c_1 Q_{hn}} \tag{8.1}$$

Where Q_{dhw} is the energy usage for domestic heating, E_{in} is the total electric energy input, Q_{hn} is the total heat extraction from the heating network, $c_0 = €0.25$ the cost of electric per kWh and $c_1 = € 0.10$ [5] the costs of extracted heat from the heating network per kWh. Using that $E_{in} = Q_{dhw} - Q_{hn}$ and COP $= Q_{dhw}/E_{in}$ the ROI becomes:

$$\text{ROI} = \frac{\text{COP}}{1 + \frac{c_1}{c_0}(\text{COP} - 1)} \tag{8.2}$$

Using COP = 1.5 a value of ROI=1.25 is found. This means that for every €1 spent on domestic heating using a PeHEX, €0.25 is saved compared to using simple electric heating. As mentioned in Section 1.2 a conventional air-water heat pump might have a higher COP of 3, but a PeHEX can be a much smaller and easier to fit device.

8.3. Recalculation of model prediction

To validate the model, all the measurements are now simulated using the new model and the simulated output temperatures of the domestic hot water $T_{dhw,out}$ and the heating network $T_{hn,out}$ are compared to the measurements. The results of the simulation and the observations are in Table A.4. The result is that the the model predicts the actually measured return temperatures with an accuracy of 0.7 K, the maximum temperature difference between a measured an predicted results is 2.7 K at experiment index 5 ($V_{dhw} = 5$ L/min, $V_{hn} = 1$ L/min). From Figure 8.3 it can be seen that the average cumulative error decays as the experiment index increases, indicating that the model simulations come more in line with the measurements as the flow speeds increase.
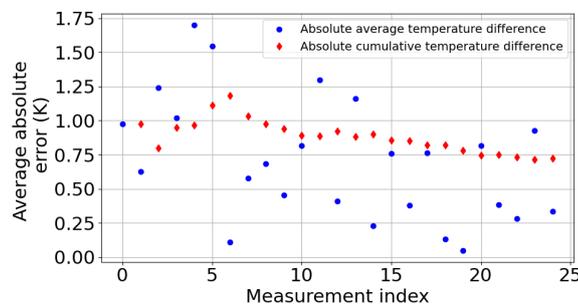
Figure 8.3: Error between simulations and experimental observations.

Figure 8.3 suggests that for low flow speeds (experiment index ≤ 10) the model is not accurately representing the temperature distribution within the PeHEX. This is likely due to the flow having a low Reynolds

number ≤ 2300, but entrance effects disturbing the laminar flow such that the correlation for the Nusselt number is not accurate for low Reynolds numbers.

For transitional to turbulent Reynolds numbers (≥ 3000) the error in the temperature simulation averaged roughly to 0.8 K and approaches the final value of 0.7 K average absolute temperature difference between the simulations and the observations. This indicates that as the flow speeds increase, the model gets better at predicting the behavior of the PeHEX.

To fully validate the model. A comparison between model simulations and corresponding measurements that were **not** used in the fitting of the new model parameters was made. The simulations consist of different flow speeds that were not used in the fitting of the model. The unused measurements' input parameters are are given in Table 8.1. The comparison between the experimental observations ans the simulation results are given in Table 8.2.

Table 8.1: Unused measurement input parameters.

#	V_{dhw} (L/min)	V_{hn} (L/min)	$T_{dhw,in}$ (°C)	$T_{hn,in}$ (°C)
1	2.67	3.49	40.4	45.4
2	3.87	0.92	39.7	44.2
3	4.50	2.60	38.8	43.8
4	3.34	4.76	39.3	44.4
5	1.74	3.84	38.6	44.1

Table 8.2

#	$T_{dhw,out,exp}$ (°C)	$T_{dhw,out,sim}$ (°C)	$T_{hn,out,exp}$ (°C)	$T_{hn,out,sim}$ (°C)	$P_{in,exp}$ (W)	$P_{in,sim}$ (W)
1	49.2	50.7	41.7	42.3	1174	1179
2	46.7	46.2	35.7	35.4	1184	1178
3	45.3	45.1	39.9	39.7	1205	1206
4	47.5	48.49	42.3	42	1184	1199
5	52	53.8	41.5	41.5	1124	1153

Comparing the measured return temperatures with the simulated return temperatures again gives an average deviation of 0.6 K. The simulated power usage is within 3% of the measured power usage. This indicates that the proposed model with the new model parameters gives an accurate description of the temperature behavior and the power draw of a PeHEX.

9
Optimal layout of a PeHEX

Now that a model has been derived that accurately represents the temperature distribution behavior of a PeHEX, the best configuration in terms of the layout of the Peltier elements in the PeHEX can be derived.

To determine to optimal layout, initially three simulations were run with a total of 120 elements in the total PeHEX. The simulations differ in the channel width in terms of how many Peltier elements are placed side by side perpendicular to the flow direction to determine if it is desirable to place multiple Peltier elements side by side. Figure 9.1a has a single line of Peltier elements in the PeHEX (one element channel width), Figure 9.1b has two element perpendicular to the flow (two element channel width) and Figure 9.1c has three element perpendicular to the flow (three element channel width).
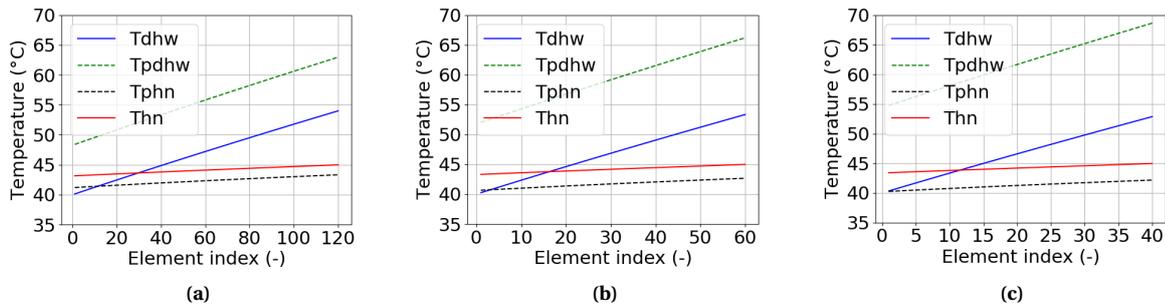
Figure 9.1: Comparison between different layouts of the PeHEX for 120 elements with $V_{dhw} = 6$ L/min, $V_{hn} = 20$ L/min, $V_0 = 10$ V. Figure 9.1a: one element channel width. Figure 9.1b: two element channel width. Figure 9.1c: three element channel width. Figures 9.1b and 9.1c show the average temperature distributions over the 2 and 3 elements respectively per element index.

Using the Nusselt relation Nu $= 0.0271\text{Re}^{0.608}\text{Pr}^{1.4}$ from Table 7.1, it can be deduced that by doubling the width of the channel, and placing two rows of side by side Peltier elements, the Reynolds number of the flow will halve causing a reduction in Nusselt number of a factor $(\frac{1}{2})^{0.608} = 0.65$. This affects the total convective heat transfer coefficient U that governs the heat transfer from the Peltier element to the flows.

An interesting observation from Figure 9.1 is that the output temperatures of the domestic hot water $T_{dhw,out}$ is reduced as the channel width increases (54 °C, 53.3 °C and 52.9 °C respectively). Also, the temperature difference across the Peltier elements increases. This is easily understood. The number of Peltier elements does not change between Figures 9.1a, 9.1b and 9.1c so all the elements feel the same electric potential difference and will therefore, initially, draw the same current $I = V_0/R$ indicating all the elements initially move the same amount of heat. As the total convective heat transfer coefficient U reduces as the channel width increases the temperature of the Peltier element on the domestic hot water side has to increase to accommodate the heat pumped from the heating network side to the domestic hot water side. A similar argument shows why the temperature on the heating network side has to decrease. The increased temperature difference across the Peltier element means that the effective electric potential V_{eff} is lower than for smaller channel widths and the thermal conduction term $k\frac{a}{d}\Delta T_p$ is larger. Therefor, a smaller current will run for larger channel widths and thus less heat is pumped from the heating network side to the domestic hot water side, causing a lower output temperature. Unsurprisingly this also has a negative effect on the COP (1.78, 1.73, and 1.7 respectively). It is therefor, as a rule of thumb, not desirably to place multiple Peltier elements side by side.

The exception to this rule is when an increased channel width is accompanied by an increased Seebeck

coefficient for the Peltier element. A larger Peltier element could increase the output temperature. If the Seebeck coefficient of a certain Peltier element is modelled as in Equations 9.1 and 9.2:

$$S = \frac{0.0428}{4 \cdot 127} \cdot 4 \cdot 127 \left(\frac{L}{L_0}\right)^2 = \sigma_S \cdot \frac{4L^2}{n_0} = 4\sigma_S n \tag{9.1}$$

$$R = \frac{\rho(2nd)}{\frac{L^2}{2n}} = \frac{\rho(2n)^2 d}{L^2} = \frac{\rho d}{L_0^2} \cdot 4(127)n^2 \tag{9.2}$$

with $\sigma_S = 84 \cdot 10^{-6}$ the Seebeck coefficient of the semi-conductor material in V/K per junction, L the dimension of the (square) Peltier element in m, $L_0 = 0.04$ the dimension of the reference Peltier element in m, $n_0 = L_0^2/127 = 12.6 \cdot 10^{-6}$ square meter per semiconductors, $\rho = 18 \cdot 10^{-6}\Omega^{-1}m^{-1}$ the resistivity of the Peltier element, d the spacing of the Peltier element and 127 is the number of semiconductor pairs in the Peltier element, then a Peltier element of size 0.05 x 0.05 m^2 has a Seebeck coefficient of $S = 0.067$ V/K and a resistance $R = 4.46\Omega$. The single element channel width is now 25% larger than an element with size 0.04 x 0.04 m^2. Figure 9.2 shows a comparison for $L = 0.04$m, $L = 0.05$m and $L = 0.06$m.
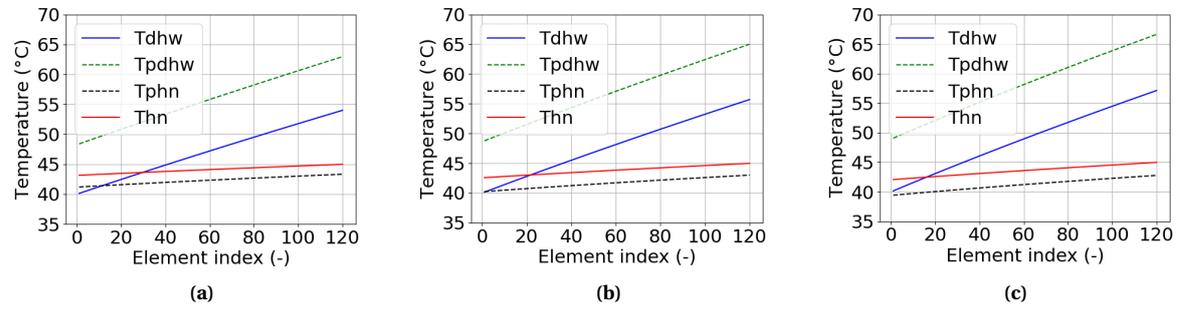
Figure 9.2: Comparison between the effects of different sizes of Peltier elements, on the PeHEX for 120 elements with $V_{dhw} = 6$ L/min, $V_{hn} = 20$ L/min, $V_0 = 10$ V. Figure 9.1a: $L = 0.04$m . Figure 9.1b: $L = 0.05$m, $S = 0.067$ V/K, $R = 4.46\Omega$. Figure 9.1c: $L = 0.06$m, $S = 0.096$ V/K, $R = 6.42\Omega$. Figures 9.1b and 9.1c show the average temperature distributions over the 2 and 3 elements respectively per element index.

The output temperatures are 54 °C, 55.7 °C and 57.1 °C for Figures 9.2a, 9.2b and 9.2c respectively indicating that the increased Seebeck coefficient out weights the reduced total convective heat transfer coefficient. The COP has also increased from 1.78, to 2.06 and 2.32 respectively. Therefore, if the channel width increase is compensated for by an increased Seebeck coefficient a larger channel width might be favorable.

The number of Peltier elements in a single PeHEX placed in sequence affects the heat transfer in the entrance area of the PeHEX for low Reynolds number. Entrance effects for low Reynolds number are not modelled in this research and therefor no conclusion can be drawn on the amount of Peltier element in a single PeHEX. This parameter could be adjusted to get the desired dimensions of a PeHEX in terms of length.

It is important to note that with $S = 0.067$ V/K, $R = 4.46\Omega$, $V_{dhw} = 6$ L/min, $V_{hn} = 20$ L/min and $V_0 = 10$ V the desired 60 °C to stop legionella growth is not yet reached. A larger Peltier element with a higher Seebeck coefficient might be used to achieve this goal, or more elements have to be used.

9.1. Second law analysis

Kiss and Infante Ferreira [20, sec. 3.1] mention that in order to increase the efficiency of thermal machines (such as heat pumps), the entropy production in the devices should be limited. However, in this application the thermal energy is moved against the temperature gradient which is only possible if a large amount of entropy is produced in the PeHEX as otherwise the second law of thermodynamics would be violated [20, eq. 2.14]. Using Equation 9.3 [20, eq. 310] as the second law efficieny:

$$\eta_{II} = \frac{\text{COP}_{\text{real}}}{\text{COP}_{\text{carnot}}} \tag{9.3}$$

with $\text{COP}_{\text{carnot}} = \frac{T_h}{T_h - T_c} = \frac{323}{323 - 313} = 32.3$ with T_h the thermodynamic average temperature of the domestic hot water and T_c the thermodynamic average temperature of the heating network. The second law efficiency of the PeHEX is then only $\eta = \frac{2.0}{32.3} = 6\%$. This very low efficiency is caused by the entropy production of

the Peltier elements. An exergy analysis Kiss and Infante Ferreira [20, sec. 3.4] of the PeHEX gives similar efficiencies.

The author of this report is uncertain if Equation 9.3 is a proper definition of the second law efficiency of a PeHEX. For example: if the PeHEX is unable to heat the domestic hot water to a temperature above the heating network's temperature, then $T_h < T_C$ and $\eta < 0$, even though the domestic hot water is heating up.

To the day of writing, the author has not found an other definition of the efficiency of the PeHEX.

10
Conclusion

Peltier elements can be used to increase the temperature of a water flow (domestic hot water) by using the temperature of another water flow (a heating network). A Peltier Heat Exchanger (PeHEX) as proposed in this research is able to elevate the temperature of a water flow above the temperature of the heat source, which is not possible in an ordinary heat exchanger or without any input power.

The model as proposed by Ludstrom and Jeong [22] was fitted to the datasheet of the used Peltier element, and re-fitted to the data from the measurements. The thermal conductivity of the Peltier element was fitted with an exponential function. The choice to use an exponential was made based on the empirical data for the thermal conductivities.

Initially, the model was not able to accurately simulate the experimental observations with differences between simulated and observed results up to 75%. The experimental data was used to re-evaluate the model parameters S, R, $k(T)$ and the relation for the Nusselt number. The Seebeck coefficient of the Peltier elements was determined to be $S = 0.0428 \pm 0.0011$ V/K, the average resistance of the Peltier elements was $R = 2.85 \pm 0.02\Omega$.

The model with the new parameters was successfully used to accurately simulate the temperature distribution within a PeHEX as a function of domestic hot water flow speed, heating network flow speed and input power. The model that describes how the Peltier effect and the Seebeck effect work together in a single Peltier element was successfully used to determine the power usage of a single Peltier element. The model of a PeHEX consisting of coupled Peltier element models predicts the temperature distribution within a PeHEX to an accuracy of 0.7 K.

The model for the PeHEX was able to simulate new measurements to an accuracy of 0.6 K, indicating that the model and correlations as proposed in this research can be used to accurately describe the behavior of a PeHEX for at least the range $738 \leq Re \leq 5005$ and $3.1 \leq Pr \leq 4.9$.

The optimal layout for a PeHEX as proposed in this research is a channel that has the width of the used Peltier elements. Increasing the channel width by placing more Peltier elements side by side limits the efficiency of the PeHEX. Peltier elements with a higher Seebeck coefficient can allow a wider channel without the cost of reduced efficiency.

The desired 60 °C at a flow speed of 6 L/min was not reached with the used Peltier elements, but a Peltier element with a higher Seebeck coefficient might to able to increase the temperature output significantly.

A possible future research idea is to model how a heating network can be used to (pre)heat a domestic water flow before and after a PeHEX, or if a parallel placement is advantageous.

Bibliography

[1] Flamco B.V. *LogoEco Split E H-HW - Afleversets voor collectieve warmte / koeling | Flamco Group*. URL: https://flamcogroup.com/nl/catalog/afleversets/afleversets-voor-collectieve-warmte-koeling/logoeco-split-e-h-hw/logoeco-split-e-h-hw/groups/g+c+p+a+view (visited on 06/15/2022).

[2] L. E. Bell. "Cooling, Heating, Generating Power, and Recovering Waste Heat with Thermoelectric Systems". In: *Science, Vol 321, Issue 5895, pp. 1457-1461* (2008).

[3] BIO-TECH. *97478940-FCH-midi-POM*. URL: https://www.btflowmeter.com/fileadmin/PDF/Flowmeter/97478940-FCH-midi-POM.pdf (visited on 06/08/2022).

[4] S. Boesten et al. "5th generation district heating and cooling systems as a solution for renewable urban thermal energy supply". In: *Adv. Geosci., 49, 129–136* (2019). DOI: https://doi.org/10.5194/adgeo-49-129-2019.

[5] ACM Consuwijzer. *Hoeveel betaal ik voor mijn warmte? | ACM ConsuWijzer*. URL: https://www.consuwijzer.nl/stadsverwarming-en-blokverwarming/warmtetarieven (visited on 06/13/2022).

[6] Flamco B. V. *About Flamco | Flamco Group*. URL: https://flamcogroup.com/uk-en/page/about-flamco (visited on 10/19/2021).

[7] GMI. *MJ Research PTC-100 Thermal Cycler*. URL: https://www.gmi-inc.com/product/mj-research-ptc-100-thermal-cycler/ (visited on 06/17/2022).

[8] V. Gnielinski. "Neue Gleichungen für den Wärme und den Stoffübergang in turbulent durchströmten Rohren und Kanälen". In: *Forschung im Ingenieurwesen, vol. 41, no. 1* (1975).

[9] D. J. Griffiths. *Introduction to Quantum Mechanics, second edition*. Cambridge University Press, 2017.

[10] Ltd. Hebei I.T. (Shanghai) Co. *Contact Us*. URL: https://peltiermodules.com/?p=z.contact (visited on 06/06/2022).

[11] Ltd. Hebei I.T. (Shanghai) Co. *TEC1-12706*. June 6, 2022. URL: https://peltiermodules.com/peltier.datasheet/TEC1-12706.pdf.

[12] M. G. Holland. "Analysis of Lattice Thermal Conductivity". In: *Physics review, volume 132, number 6* (Dec. 15, 1963).

[13] I. G. Hughes and T. P. A. Hase. *Measurements and their uncertainties*. Oxford University Press, 2010.

[14] Y. Hyun et al. "Evaluation of Seebeck coefficients in n- and p-type silicon nanowires fabricated by complementary metal–oxide–semiconductor technology". In: *Nanotechnology, Volume 23, Number 40* (Sept. 20, 2012).

[15] F. P. Incropera and D. P. DeWitt. *Fundamentals of Heat and Mass Transfer, sixth edition*. Wiley, 2007.

[16] DATAQ Instruments. *DI-2008 Thermocouple and Voltage Data Acquisition System*. URL: https://www.dataq.com/resources/pdfs/datasheets/di-2008-usb-daq-datasheet.pdf (visited on 06/07/2022).

[17] IUPAC. "Definitions of terms relating to phase transitions of the solid state (IUPAC Recommendations 1994)". In: *Pure and Applied Chemistry, Vol. 66, No. 3* (1994).

[18] W. Kays, M. Crawford, and B. Weigand. *Convective Heat and Mass Transfer, fourth edition*. McGraw-Hill. URL: 2005.

[19] ISABELLENHÜTTE HEUSLER GMBH & CO. KG. *PBV.PDF*. URL: https://www.isabellenhuette.de/fileadmin/user_upload/PBV.PDF.

[20] A. A. Kiss and C. A. Infante Ferreira. *Heat Pumps in Chemical Process Industry*. CRC Press, 2017.

[21] R. Litteaur. "In Situ Verification Techniques for Multipoint Thermocouples in Pressure Vessels". In: *Daily Thermetrics Corp.* (2018).

[22] M. Ludstrom and C. Jeong. *Near-Equilibrium Transport Fundamentals and Application*. World Scientific Publishing Co. Pte. Ltd., 2013.

[23] Dienst Elektronische en Mechanische Ontwikkeling. URL: `https://www.tudelft.nl/demo`.

[24] michbich. *Peltierelement.png*. May 30, 2022. URL: `https://nl.wikipedia.org/wiki/Peltiereffect#/media/Bestand:Peltierelement.png`.

[25] A. F. Mills and C. F. M. Coimbra. *BASIC HEAT AND MASS TRANSFER, third edition*. Temporal Publishing LLC, 2015.

[26] overheid.nl. *wetten.nl - Regeling - Regeling legionellapreventie in drinkwater en warm tapwater - BWBR0030166*. URL: `https://wetten.overheid.nl/BWBR0030166/2019-01-01` (visited on 10/19/2021).

[27] Vassil Palankovski. "Simulation of Heterojunction Bipolar Transistors". PhD thesis. Technischen Universität Wien. URL: `https://www.iue.tuwien.ac.at/phd/palankovski/`.

[28] Quick-cool. *Technical data*. URL: `https://asset.conrad.com/media10/add/160267/c1/-/en/000189182DS02/datasheet-189182-quickcool-qc-71-14-85m-thermoelectric-cooler-high-performance-86-v-85-a-40-w-a-x-b-x-c-x-h-30-x-30-x-x-34-mm.pdf` (visited on 10/28/2021).

[29] P. P. L. Regtien. *Electronic instrumentation*. VSSD, 2005.

[30] Rijksoverheid. *Overheid stimuleert nuttige toepassingen van restwarmte | Duurzame energie | Rijksoverheid.nl*. URL: `https://www.rijksoverheid.nl/onderwerpen/duurzame-energie/overheid-stimuleert-nuttige-toepassingen-van-restwarmte` (visited on 10/19/2021).

[31] D. Schmidt et al. "Low Temperature Disctrict Heating for Future Energy Systems". In: *Energy Procedia 116* (2017). DOI: `https://doi.org/10.1016/j.egypro.2017.05.052`.

[32] T. L. T L Bergman and F. P. Incropera. *Fundamentals of heat and mass transfer*. Hoboken, 2011.

[33] Delft University of Technology. *Open Solid State Notes*. URL: `https://solidstate.quantumtinkerer.tudelft.nl/` (visited on 10/24/2021).

[34] S. Thapaa et al. "The thermopile - An anisotropic temperature sensor". In: *Sensors and Actuators A 187 (2012) 132–140* (2012).

[35] The Engineering Toolbox. *Metals, Metallic Elements and Alloys - Thermal Conductivities*. URL: `https://www.engineeringtoolbox.com/thermal-conductivity-metals-d_858.html` (visited on 06/12/2022).

[36] The Engineering Toolbox. *Water - Specific Heat vs. Temperature*. URL: `https://www.engineeringtoolbox.com/specific-heat-capacity-water-d_660.html` (visited on 06/13/2022).

[37] H. Van den Akker and R. Mudde. *Fysische transportverschijnselen, vierde druk*. Delft Academic Press, 2014.

[38] WarmtelinQ. *Veelgestelde vragen › WarmtelinQ*. URL: `https://www.warmtelinq.nl/veelgestelde-vragen` (visited on 10/20/2021).

[39] Mean Well. *HLG-480H.cdr*. URL: `https://www.meanwell-web.com/content/files/pdfs/productPdfs/MW/HLG-480H/HLG-480H-48AB.pdf`.

[40] Wikipedia. *Schottky barrier - Wikipedia*. URL: `https://en.wikipedia.org/wiki/Schottky_barrier` (visited on 10/28/2021).

[41] Wikipedia. *Temperature dependence of viscosity - Wikipedia*. URL: `https://en.wikipedia.org/wiki/Temperature_dependence_of_viscosity` (visited on 10/31/2021).

[42] L. kleyn Winkel. *Coupled reduced order model and adjoint methodologies for proton therapy*. 2019. URL: `https://repository.tudelft.nl/islandora/object/uuid%5C%3A2e932bb0-98b7-43ac-b6f4-3da4bb869fb4?collection=education`.

[43] R. Wolfson. *Essential University Physics, volume 2, third edition*. Pearson Education Limited, 2016.

[44] World Health Organisation. *Legionella and the prevention of legionellosis*. 2007. URL: `https://apps.who.int/iris/bitstream/handle/10665/43233/9241562978_eng.pdf?sequence=1&isAllowed=y`.

[45] M. Z. Yilmazoglu. "Experimental and numerical investigation of a prototype thermoelectric heating and cooling unit". In: *Elsevier, Energy and Buildings 113, pp. 51-60* (2016). DOI: https://doi.org/10.1016/j.enbuild.2015.12.046.

A

Results

Table A.1: Numerical results graphically represented in Chapter 4.

Measurement index	V_{dhw} (L/min)	V_{hn} (L/min)	V_0 (V)	I (A)	P_{in} (W)	$T_{dhw,out}$ (C)	$T_{hn,out}$ (C)	COP
1	1	1	9.60	3.95	1516	66.57	40.20	1.22
2	1	2	9.60	3.98	1527	67.56	42.18	1.26
3	1	3	9.60	4.01	1538	68.63	42.82	1.30
4	1	4	9.60	4.02	1542	68.93	43.30	1.31
5	1	5	9.60	4.02	1544	69.11	43.61	1.31
6	2	1	9.60	4.34	1667	56.49	35.97	1.38
7	2	2	9.60	4.43	1701	57.67	39.54	1.45
8	2	3	9.60	4.56	1749	59.41	40.43	1.55
9	2	4	9.60	4.59	1763	59.92	41.37	1.57
10	2	5	9.60	4.61	1772	60.23	42.00	1.59
11	3	1	9.60	4.61	1768	52.38	33.24	1.46
12	3	2	9.60	4.72	1814	53.42	37.89	1.55
13	3	3	9.60	4.92	1888	55.15	38.88	1.68
14	3	4	9.58	4.97	1904	55.63	40.11	1.72
15	3	5	9.54	4.99	1902	55.86	40.95	1.74
16	4	1	9.60	4.70	1805	49.66	32.28	1.49
17	4	2	9.60	4.80	1843	51.01	37.24	1.67
18	4	3	9.46	5.00	1891	51.81	38.31	1.74
19	4	4	9.33	5.00	1865	52.03	39.66	1.80
20	4	5	9.25	5.00	1849	52.17	40.57	1.83
21	5	1	9.60	4.76	1827	47.91	31.70	1.51
22	5	2	9.60	4.90	1883	48.64	36.91	1.60
23	5	3	9.29	5.00	1856	49.54	37.98	1.79
24	5	4	9.15	5.00	1828	49.74	39.39	1.85
25	5	5	9.06	5.00	1811	49.85	40.35	1.89

Table A.2: Raw measurement data

M.I.	$\dot{V}_{dhw}$ (L/min)	$\dot{V}_{hn}$ (L/min)	V1 (V)	$V1_e$ (V)	V_{I1} (V)	$I1_e$ (A)	V2 (V)	$V2_e$ (V)	V_{I2} (V)	$I2_e$ (A)	V3 (V)	$V3_e$ (V)	V_{I3} (V)	$I3_e$ (A)
1	1.03	0.96	46.93	9.39	0.116	2.90	48.87	9.77	0.116	2.90	46.11	9.22	0.114	2.85
2	1.05	1.97	46.93	9.39	0.115	2.88	48.87	9.77	0.116	2.90	46.11	9.22	0.114	2.85
3	0.97	3.01	46.93	9.39	0.116	2.90	48.87	9.77	0.115	2.88	46.11	9.22	0.113	2.83
4	1.02	4.04	46.93	9.39	0.115	2.88	48.88	9.78	0.115	2.88	46.11	9.22	0.114	2.85
5	1.01	5.1	46.93	9.39	0.115	2.88	48.88	9.78	0.115	2.88	46.11	9.22	0.114	2.85
6	1.95	0.9	46.91	9.38	0.12	3.00	48.89	9.78	0.121	3.03	46.11	9.22	0.119	2.98
7	2.11	1.97	46.93	9.39	0.121	3.03	48.88	9.78	0.12	3.00	46.11	9.22	0.119	2.98
8	2	2.97	46.92	9.38	0.121	3.03	48.88	9.78	0.121	3.03	46.11	9.22	0.121	3.03
9	2.12	4	46.93	9.39	0.122	3.05	48.88	9.78	0.12	3.00	46.11	9.22	0.12	3.00
10	1.95	5.07	46.92	9.38	0.121	3.03	48.88	9.78	0.12	3.00	46.11	9.22	0.12	3.00
11	3.01	0.92	46.92	9.38	0.122	3.05	48.87	9.77	0.123	3.08	46.1	9.22	0.122	3.05
12	3.04	2.04	46.92	9.38	0.123	3.08	48.87	9.77	0.123	3.08	46.1	9.22	0.123	3.08
13	2.98	3.02	46.93	9.39	0.124	3.10	48.87	9.77	0.122	3.05	46.1	9.22	0.123	3.08
14	2.99	4.07	46.93	9.39	0.124	3.10	48.88	9.78	0.123	3.08	46.1	9.22	0.124	3.10
15	3.03	4.98	46.92	9.38	0.124	3.10	48.86	9.77	0.123	3.08	46.11	9.22	0.123	3.08
16	4.04	0.88	46.91	9.38	0.124	3.10	48.9	9.78	0.125	3.13	46.13	9.23	0.125	3.13
17	4.02	1.99	46.92	9.38	0.125	3.13	48.9	9.78	0.125	3.13	46.12	9.22	0.126	3.15
18	4.09	3.09	46.92	9.38	0.125	3.13	48.9	9.78	0.125	3.13	46.12	9.22	0.126	3.15
19	4.02	4.08	46.92	9.38	0.125	3.13	48.89	9.78	0.124	3.10	46.12	9.22	0.126	3.15
20	4.02	5.09	46.92	9.38	0.125	3.13	48.89	9.78	0.125	3.13	46.12	9.22	0.127	3.18
21	4.98	0.95	46.91	9.38	0.125	3.15	48.87	9.77	0.126	3.13	46.11	9.22	0.126	3.15
22	4.9	2.05	46.92	9.38	0.126	3.15	48.87	9.77	0.125	3.13	46.11	9.22	0.127	3.18
23	4.96	3.07	46.92	9.38	0.126	3.15	48.87	9.77	0.126	3.15	46.1	9.22	0.127	3.18
24	5.02	4.01	46.92	9.38	0.126	3.15	48.86	9.77	0.126	3.15	46.1	9.22	0.127	3.18
25	4.96	4.98	46.92	9.38	0.127	3.18	48.86	9.77	0.126	3.15	46.09	9.22	0.127	3.18

continuation of Table A.2.

M.I.	$\dot{V}_{dhw}$ (L/min)	$\dot{V}_{hn}$ (L/min)	V4 (V)	V4$_e$ (V)	V$_f$4 (V)	I4$_e$ (A)	P (W)	T_1 (°C)	T_2 (°C)	T_3 (°C)	T_4 (°C)	T_5 (°C)	T_6 (°C)	T_7 (°C)	T_8 (°C)
1	1.03	0.96	50.31	10.06	0.112	2.80	1100.20	38.9	44.4	54	58.4	43.7	41.4	39.6	37.8
2	1.05	1.97	50.31	10.06	0.112	2.80	1097.86	39.4	45.2	55.3	60.2	44.5	42.8	41.5	40.7
3	0.97	3.01	50.3	10.06	0.112	2.80	1095.40	38	44.9	56.8	61.1	43.5	43	43.7	42.5
4	1.02	4.04	50.31	10.06	0.111	2.78	1092.96	38.3	45.3	55.2	61	43.8	43.7	42.9	43.4
5	1.01	5.1	50.31	10.06	0.111	2.78	1092.96	38.8	45	54.9	59.9	44.5	43.7	43.4	43.5
6	1.95	0.9	50.32	10.06	0.119	2.98	1151.00	39.8	42.2	47	50.2	44.8	40.6	37.3	35.7
7	2.11	1.97	50.31	10.06	0.118	2.95	1148.39	39.2	41.7	48.4	50.9	44.4	41.2	41.4	40.3
8	2	2.97	50.32	10.06	0.119	2.98	1157.96	37.9	42.4	49.2	51.1	43.5	42.2	42.7	41.1
9	2.12	4	50.31	10.06	0.118	2.95	1153.04	38.8	43.2	48.1	50.2	44.4	43.3	42.4	41.4
10	1.95	5.07	50.31	10.06	0.118	2.95	1150.64	38.8	43.4	47.9	51.4	44.3	43.6	42.2	42.6
11	3.01	0.92	50.3	10.06	0.121	3.03	1172.29	40.3	40.5	45.4	47.8	45	39.2	37.5	35.9
12	3.04	2.04	50.3	10.06	0.121	3.03	1176.94	40.7	41.3	45.1	48	45.7	41.3	39.9	39.8
13	2.98	3.02	50.3	10.06	0.121	3.03	1176.90	39.6	42.2	45.7	48.8	44.7	42.5	41.4	40.9
14	2.99	4.07	50.3	10.06	0.121	3.03	1181.71	38	41.8	47.1	47.8	43.4	42.5	43.3	42.4
15	3.03	4.98	50.29	10.06	0.121	3.03	1179.22	39.3	41.9	46.4	48.1	44.7	42.8	43	42.3
16	4.04	0.88	50.32	10.06	0.123	3.08	1194.25	39.2	40.8	44.1	46.1	44	40.3	36.7	34.9
17	4.02	1.99	50.32	10.06	0.123	3.08	1198.90	39.4	41.3	44.3	46.4	44.8	42.2	40	39.5
18	4.09	3.09	50.32	10.06	0.123	3.08	1198.90	39.8	41.7	44.6	45.9	45	43.2	41.7	40.7
19	4.02	4.08	50.31	10.06	0.124	3.10	1198.85	39.7	41.3	44.6	46.6	45	43.1	42.3	42.1
20	4.02	5.09	50.31	10.06	0.123	3.08	1201.08	39.5	41.4	44.9	46.5	44.9	43.1	43.3	42.7
21	4.98	0.95	50.3	10.06	0.125	3.13	1205.94	39.8	41.5	43.4	45	44.4	40.6	36.4	34.6
22	4.9	2.05	50.3	10.06	0.124	3.10	1205.69	39.5	41	44	45.4	44.7	41.8	40.4	39.4
23	4.96	3.07	50.3	10.06	0.125	3.13	1210.59	38.7	41.7	44.7	44.9	44	43.1	42.3	40.5
24	5.02	4.01	50.3	10.06	0.125	3.13	1210.52	39.9	40.7	44.1	44.9	45.2	42.3	42.3	41.2
25	4.96	4.98	50.29	10.06	0.124	3.10	1210.23	39.6	41.8	44	45.3	44.9	43.6	42.6	42

Table A.3: Total and elemental resistance for the Peltier Heat Exchangers

Peltier Heat Exchanger number	1	2	3	4
Total resistance (Ω)	7.2	7.0	7.1	7.3
Element resistance (Ω)	2.87	2.79	2.83	2.91

Figures for DHW regression:
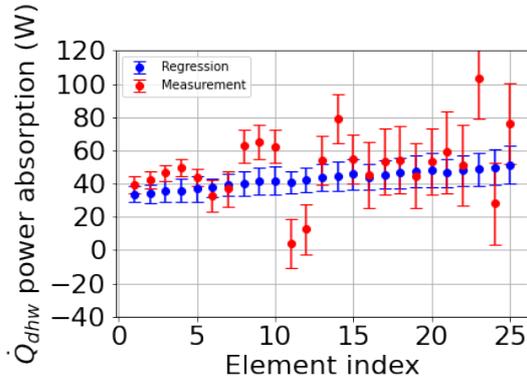
(a) Regression on domestic hot water power absorption for the first array.
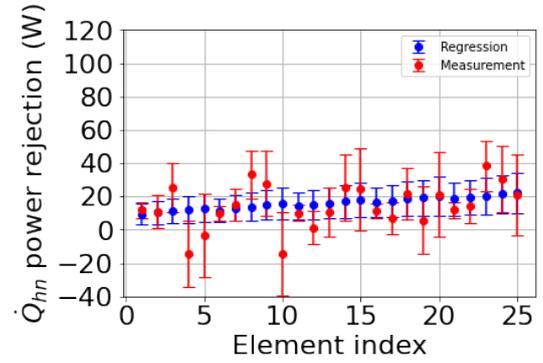
(b) Regression on heating network power rejection for the first array.

Figure A.1: Regression on the first array of the test setup.
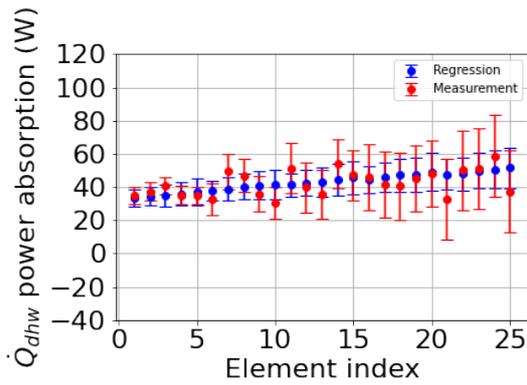
(a) Regression on domestic hot water power absorption for the second array.
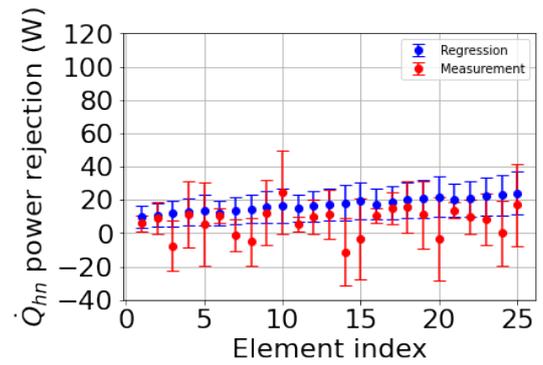
(b) Regression on heating network power rejection for the second array.

Figure A.2: Regression on the second array of the test setup.
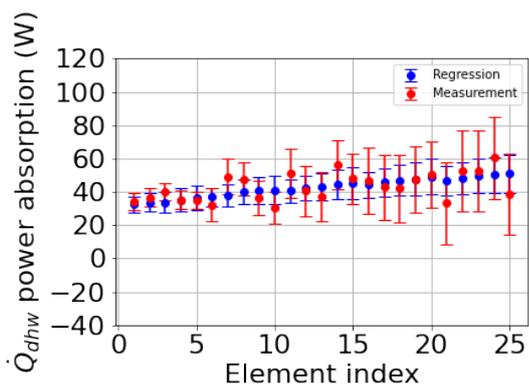
(a) Regression on domestic hot water power absorption for the third array.
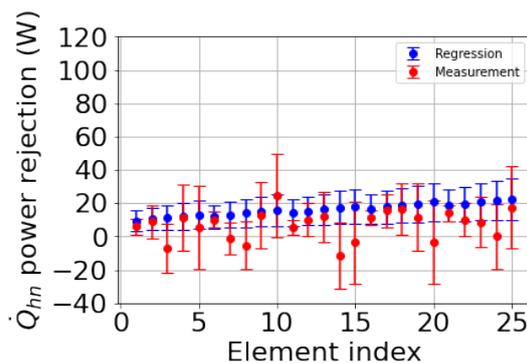
(b) Regression on heating network power rejection for the third array.

Figure A.3: Regression on the third array of the test setup.
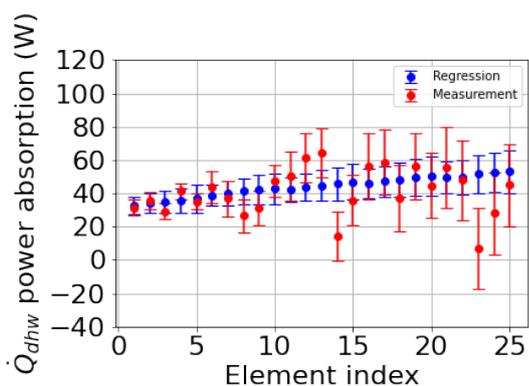
(a) Regression on domestic hot water power absorption for the fourth array.
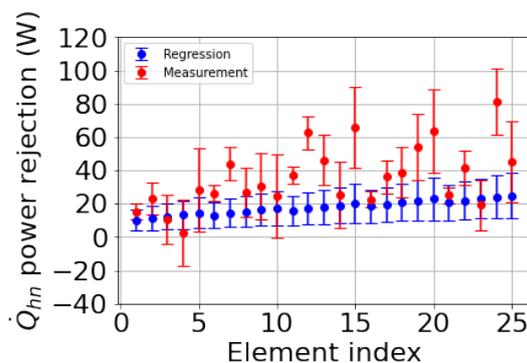
(b) Regression on heating network power rejection for the fourth array.

Figure A.4: Regression on the fourth array of the test setup.

Table A.4: Model simulations compared with measurement results.

Measurement Index	1	2	3	4	5	6	7	8	9	10
$T_{dhw,meas}$ (K)	331.4	333.2	334.1	334	332.9	323.2	323.9	324.1	323.2	324.4
$T_{dhw,sim}$ (K)	333.03	334.43	335.03	334.43	335.62	324.73	323.88	323.56	323.98	325.08
$T_{hn,meas}$ (K)	310.8	313.7	315.5	316.4	316.5	308.7	313.3	314.1	314.4	315.6
$T_{hn,sim}$ (K)	310.48	313.72	313.95	314.79	315.82	310.26	313.10	313.49	314.99	315.38

Measurement Index	11	12	13	14	15	16	17	18	19	20
$T_{dhw,meas}$ (K)	320.8	321	321.8	320.8	321.1	319.1	319.4	318.9	319.6	319.5
$T_{dhw,sim}$ (K)	321.31	322.30	321.51	319.97	321.31	318.28	319.07	319.56	319.67	319.56
$T_{hn,meas}$ (K)	308.9	312.8	313.9	315.4	315.3	307.9	312.5	313.7	315.1	315.7
$T_{hn,sim}$ (K)	310.02	314.10	314.42	313.90	315.55	308.60	312.93	314.57	315.29	315.66

Measurement Index	21	22	23	24	25
$T_{dhw,out,meas}$ (K)	318	318.4	317.9	317.9	318.3
$T_{dhw,out,sim}$ (K)	317.88	318.06	317.35	318.61	318.42
$T_{hn,out,meas}$ (K)	307.6	312.4	313.5	314.2	315
$T_{hn,out,sim}$ (K)	309.11	312.82	313.49	315.34	315.54

B

Solution to the Peltier Element's ODE

B.1. Solving Equation 3.2

Substituting the model for the thermal conductivity from Equation 3.3 into Equation 3.2 and integrate twice gives:

$$\frac{k_1}{k_2}\exp(k_2 T)A + \frac{I^2 R z^2}{2d} + C_1 z + C_2 = 0 \tag{B.1}$$

The constants C_1, C_2 can be determined by the boundary conditions $T(z=0) = T_{p,hn}$ and $T(z=d) = T_{p,dhw}$ and are given by:

$$C_1 = -\frac{I^2 R}{2} - \frac{k_1 A}{k_2 d}\Big[\exp(k_2 T_{p,dhw}) - \exp(k_2 T_{p,hn})\Big]$$

$$C_2 = \frac{k_1}{k_2} A \exp(k_2 T_{p,hn})$$

Such that:

$$
\begin{aligned}
-k(T)A\frac{dT}{dz}\Big|_z &= \frac{I^2 R}{2}\Big(\frac{z}{2d} - 1\Big) - \frac{k_1 A}{k_2 d}\Big[\exp(k_2 T_{p,dhw}) - \exp(k_2 T_{p,hn})\Big] \\
&= \frac{I^2 R}{2}\Big(\frac{z}{2d} - 1\Big) - \frac{k_1 A}{k_2 d}\Big[\frac{\exp(k_2 T_{p,dhw}) - \exp(k_2 T_{p,hn})}{T_{p,dhw} - T_{p,hn}}\Big](T_{p,dhw} - T_{p,hn}) \\
&= \frac{I^2 R}{2}\Big(\frac{z}{2d} - 1\Big) - \frac{A}{k_2 d}\Big(\frac{dk(T)}{dT}\Big|_{T=(T_{p,hn}+T_{p,dhw})/2}\Big)(T_{p,dhw} - T_{p,hn}) \\
&= \frac{I^2 R}{2}\Big(\frac{z}{2d} - 1\Big) - \frac{A}{d}k\Big(\frac{T_{p,dhw} + T_{p,hn}}{2}\Big)(T_{p,dhw} - T_{p,hn})
\end{aligned}
\tag{B.2}
$$

<div align="right">

C

</div>

Hermite interpolation

Hermite interpolation uses function values, as well as derivatives to fit a third order cubic polynomial between two data points. If the two data points are $(x_0, f(x_0))$ and $(x_1, f(x_1))$, with corresponding derivatives $(x_0, f'(x_0))$ and $(x_1, f'(x_1))$, then the coefficient os the polynomial $p_3(x) = ax^3 + bx^2 + cx + d$ are found using:

$$
\begin{bmatrix}
1 & x_0 & x_0^2 & x_0^3 \\
1 & x_1 & x_1^2 & x_1^3 \\
0 & 1 & 2x_0 & 3x_0^2 \\
0 & 1 & 2x_1 & 3x_1^2
\end{bmatrix}
\begin{bmatrix}
d \\ c \\ b \\ a
\end{bmatrix}
=
\begin{bmatrix}
f(x_0) \\ f(x_1) \\ f'(x_0) \\ f'(x_1)
\end{bmatrix}
\tag{C.1}
$$

Any interpolation polynomial based on function values as well as derivatives, or sensitivities, is called an *Hermite Interpolation Polynomial*. An example of how Hermite interpolation can improve a fit is given in figure C.1.
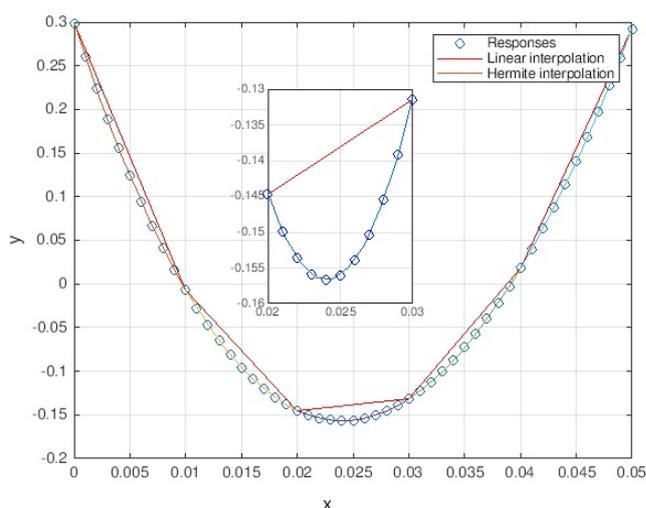
Figure C.1: From [42]. An example of a set of arbitrary data points, a linear interpolation polynomial and a Hermite interpolation polynomial. Interpolation points are $x = [0, 0.01, ..., 0.05]$ The figures shows the linear interpolation simply connecting the interpolation points, whilst Hermite interpolation follows the curvature of the responses. The zoom on $0.02 \leq x \leq 0.03$ especially shows how the linear interpolation is unaware of the curvature.

D

Iteration schemes.

Cell Equations Solver

Model inputs:
Temperatures:
 - *Tcin, Thin*
Velocities:
 - *vc, vh*
Geometry:
 - *Ac, Awc*
 - *Ah, Awh*
Heat transfer:
 - *Uc, Uh*
Peltier:
 - *S, R, k(T), A, d*
Peltier driving:
 - *I*
Initial guess:
 - *Twc, Twh*

Build cell matrix *A* from equation:

$A\mathbf{x} = \mathbf{b}$

$$\begin{bmatrix} v_cA_c\rho c_p + \frac{U_cA_{wc}}{2} & -U_cA_{wc} & 0 & 0 \\ -\frac{U_cA_{wc}}{2} & U_cA_{wc} + k\left(\frac{T_{wh}+T_{wc}}{2}\right)\frac{A}{d} - IS & -k\left(\frac{T_{wh}+T_{wc}}{2}\right)\frac{A}{d} & 0 \\ 0 & k\left(\frac{T_{wh}+T_{wc}}{2}\right)\frac{A}{d} & -U_hA_{wh} - k\left(\frac{T_{wh}+T_{wc}}{2}\right)\frac{A}{d} - SI & \frac{U_hA_{wh}}{2} \\ 0 & 0 & U_hA_{wh} & v_hA_h\rho c_p - \frac{U_hA_{wh}}{2} \end{bmatrix}$$

Solve the system using *numpy.linalg.solve(A, **b**)*

$$\begin{bmatrix} (v_cA_c\rho c_p - \frac{U_cA_{wc}}{2})T_{cin} \\ \frac{I^2R}{2} + \frac{U_cA_{wc}}{2}T_{cin} \\ -\frac{I^2R}{2} + -\frac{U_hA_{wh}}{2}T_{hin} \\ -(v_hA_h\rho c_p - \frac{U_hA_{wh}}{2})T_{hin} \end{bmatrix}$$

Build right hand side, **b**, of the cell equations:

$A\mathbf{x} = \mathbf{b}$

Return solution:
Tcout, Twc, Twh, Thout

Twh and *Twc* are initially an estimate!

Figure D.1: Matrix vector equation for solving the temperature distribution of a Peltier element.

Single Cell Solver

Model inputs:
Temperatures:
 - *Tcin*
 - *Thin*
Velocities:
 - *vc*
 - *vh*
Geometry:
 - *Wc, Lc, Hc, CLc*
 - *Wh, Lh, Hh, CLh*
Peltier:
 - *S, R, k(T)*
Peltier driving:
 - *V0*

Calculate

Awc = LcWc
Awh = LhWh

Ac = WcHc
Ah = WhHh

Dc = 2WcHc/(Wc + Hc)
Dh = 2WhHh/(Wh + Hh)

Calculate heat transfer coefficient

Calculate Nu_x = Nu(*v_x,D,T_x*) and *h_x = Nu_x × k_w/D*

Calculate overall heat transfer coefficients:

1/U = 1/h + d/k_m

Calculate current flow:
I = V0/R

Awx: Area of the water channel on 'x'-side
Ax: Heat transfer area on 'x'-side
Dx: Hydraulic diameter on 'x'-side

Nu_x: Nusselt number on 'x'-side
T: Temperature of the fluid
h_x: heat transfer coefficient on 'x'-side
k_w: Thermal conductivity of water

d: Thickness of the channel
k_m: Thermal conductivity of the material of the channel

Use:
Tpdhw = Tcin
Tphn = Thin

Solve the Cell Equations

Returns:
Tcout, Tpdhw, Tphn, Thout

Use new *Tpdhw* and *Tphn*

Solve the Cell Equations

Returns:
Tcout, Tpdhw, Tphn, Thout

Is the difference between the new and old *Tpdhw* and *Tphn* larger than 0.001 K? Or is *I* larger than *Iallowed*?

Set *V0_new = V0_old - 0.01*
Set *Veff = V0 - S(Tpdhw - Tphn)*
Set *I = Veff/R*

No

Is *I* larger than *Iallowed*?

yes

Yes

Essentially we are looking for a 'fixed point' of the 'Cell Equations' with respect to *Twc* and *Twh*

Use:
Tpdhw = (Tpdhw_old + Tpdhw_new)/2
Tphn = (Tphn_old + Tphn_new)/2

No

Return:
Tcout, Twc, Twh, Thout, I, V0

Iteration of the Cell Equations

Figure D.2: Iteration scheme to find solution to system 3.12.

E
Python code

Code snippet E.1: Supporting functions used for the PeHEX model

```python
import numpy as np
from scipy import stats
import matplotlib.cm as cm
import matplotlib.pyplot as plt

def z(S, R, n, k_T, A = 0.0016, d = 0.0039):
    s = (n*d)/(R*A/n)
    Sn = S/n
    return (Sn**2)*s/k_T

def mu_w(T):
    return (1.856*10**-14)*np.exp(4209.0/T + 0.04527*T - 3.376*10**-5*T**2)

def k_w(T):
    return -0.5981 + 6.53*10**-3*T - 8.354*10**-6*T**2

def Pr_f(T):
    return 4180*mu_w(T)/k_w(T)

def Re_f(v,D,T):
    rho = 1000
    return v*rho*D/mu_w(T)

def k(T):
    global k_coef
    try:
        T_v = np.zeros((len(k_coef),len(T)))
    except:
        T_v = np.zeros((len(k_coef),1))
    T_v[0,:] = T
    T_v[1,:] = 1/T
    return np.dot(k_coef, T_v)

def Nu(v,D,T,L):
    rho = 1000
    Re = Re_f(v,D,T)
    Pr = Pr_f(T)

    if Re < 2300:   # Mills, non circular laminar flow 4.50
        return 3.66 + (0.065*(D/L)*Re*Pr)/(1 + 0.04*((D/L)*Re*Pr)**(2.0/3))
    elif Re < 3000:
        f0 = lambda Re : (3.66 + (0.065*(D/L)*Re*Pr)/(1 + 0.04*((D/L)*Re*Pr)**(2.0/3)))

        f = ((0.79*np.log(Re) - 1.64)**-2)
        f1 = lambda Re : ((f/8)*(Re - 1000)*Pr/(1 + 12.7*((f/8)**0.5)*(Pr**(2.0/3) - 1)))

        Nu, c = spline(Re, 2300, 3000, f0, f1)
        return Nu
    else:    # Gnielinski
        f = ((0.79*np.log(Re) - 1.64)**-2)

        return (f/8)*(Re - 1000)*Pr/(1 + 12.7*((f/8)**0.5)*(Pr**(2.0/3) - 1))
```

```python
def spline(x, x0, x1, f0, f1):
    df0 = (f0(x0) - f0(x0 - 1e-3))/(1e-3)
    df1 = (f1(x1 + 1e-3) - f1(x1))/(1e-3)

    A = np.zeros((4,4))
    A[0,:] = [x0**3, x0**2, x0, 1]
    A[1,:] = [x1**3, x1**2, x1, 1]
    A[2,:] = [3*x0**2, 2*x0, 1, 0]
    A[3,:] = [3*x1**2, 2*x1, 1, 0]

    b = np.zeros((4,1))
    b[0, 0] = f0(x0)
    b[1, 0] = f1(x1)
    b[2, 0] = df0
    b[3, 0] = df1
    c = np.linalg.solve(A, b)
    X = np.array([x**3, x**2, x, 1])
    return np.dot(X, c), c

def solveCellEqs(vc, vh, Ac, Ah, Uc, Uh, Awc, Awh, Tcin,
     Thin, k, A, d, S, I, R, Tp):
    B = np.zeros((4,4))
    b = np.zeros((4,1))

    B[0,0] = vc*Ac*1000*4180 + Uc*Awc/2
    B[0,1] = -Uc*Awc
    b[0,0] = vc*Ac*1000*4180*Tcin - Tcin/2*Uc*Awc

    B[1,0] = -Uc*Awc/2
    B[1,1] = Uc*Awc + k((Tp[1]+Tp[0])/2)*A/d - S*I
    B[1,2] = -k((Tp[1]+Tp[0])/2)*A/d
    b[1,0] = Tcin/2*Uc*Awc + R/2*I**2

    B[2,1] = k((Tp[1]+Tp[0])/2)*A/d
    B[2,2] = -Uh*Awh - k((Tp[1]+Tp[0])/2)*A/d - S*I
    B[2,3] = Uh*Awh/2
    b[2,0] = -Thin/2*Uh*Awh - R/2*I**2

    B[3,2] = Uh*Awh
    B[3,3] = -vh*Ah*1000*4180 - Uh*Awh/2
    b[3,0] = -vh*Ah*1000*4180*Thin + Thin/2*Uh*Awh

    T = np.squeeze(np.linalg.solve(B,b))

    return T
```

```python
def singleCell(Tcin, Thin, vc, vh, Wc, Hc, Lc, Wh, Hh, Lh,
    S, R, k, Tp, V = -1, I = -1, A = -1, Ac = -1, Ah = -1,
    CLh = 1, CLc = 1, n = 1, Awc = -1, Awh = -1, Iallowed
    = 5.0):
    global k_w
    B = np.zeros((4,4))
    b = np.zeros((4,1))
    d_al = 0.001
    k_al = 205
    Rf = 0.0002

    if A == -1:
        A = Lc**2

    if Awc == -1:
        Awc = Lc*Wc

    if Awh == -1:
        Awh = Lh*Wh

    if Ac == -1:
        Ac = Wc*Hc

    if Ah == -1:
        Ah = Wh*Hh

    Dc = 4*Wc/n*Hc/(2*(Wc/n+Hc))
    Dh = 4*Wh/n*Hh/(2*(Wh/n+Hh))

    h_h = Nu(vh, Dh, Thin, CLh)*k_w(Thin)/Dh
    h_c = Nu(vc, Dc, Tcin, CLc)*k_w(Tcin)/Dc

    Uh = 1/(1/h_h + d_al/k_al)
    Uc = 1/(1/h_c + d_al/k_al)

    if I == -1:
        I = V/R
    else:
        V = I*R

    T = solveCellEqs(vc, vh, Ac, Ah, Uc, Uh, Awc, Awh, Tcin
        , Thin, k, A, d, S, I, R, [Tcin, Thin])
    Tp = T[1:3]
    T = solveCellEqs(vc, vh, Ac, Ah, Uc, Uh, Awc, Awh, Tcin
        , Thin, k, A, d, S, I, R, Tp)

    V_eff = V - S*(Tp[0] - Tp[1])
    I = V_eff / R

    while np.linalg.norm(T[1:3] - Tp) > 0.001 or I >
        Iallowed:
        Tp = (Tp + T[1:3])/2
        T = solveCellEqs(vc, vh, Ac, Ah, Uc, Uh, Awc, Awh,
            Tcin, Thin, k, A, d, S, I, R, Tp)
        V_eff = V - S*(T[1] - T[2])
        I = V_eff / R
        if I > Iallowed:
            V = (V + Iallowed * R)/2
            V_eff = V - S*(T[1] - T[2])
            I = V_eff / R

    return (T, I, V)
```

Code snippet E.2: Code for determining the entire temperature distribution in the PeHEX.

```python
# Initialize parameters and arrays
Nw = 1                  # number of Peltier elements placed side by side
NL = 10                 # number of Peltier elements in a single PeHEX
Ntot = 40               # total number of elements used
V = 48.0/5              # potential per element
Tdhwin = 273 + 45       # input temperature of the domestic hot water
Thnin = 273 + 45        # input temperature of the heating network
W = 0.038               # width of the water channel
Hc = 0.004              # Height of the water channel
Debdhw = 5              # domestic hot water flow rate in L/min
vc = (((Debdhw/60)/1000)/(Nw*W*Hc))        # /(l/s)/(m3/s)/(m/s)
Debhn = 15.0            # heating network water flow rate in L/min
vh = (((Debhn/60)/1000)/(Nw*W*Hc))         # /(l/s)/(m3/s)/(m/s)


T = np.zeros((N,Nw,4))
T2 = np.zeros((N,Nw,4))
I_v = np.zeros((N,Nw))          # Current through element
Velem = np.zeros((N,Nw))        # Potential over element


for w in range(Nw):
    T[:,w,3] = np.linspace(Thnin - ((V**2)/R*N)/(Debhn/60*4180), Thnin, N)


(T[0,:,:], I_v[0,:], Velem[0,:]) = singleCell(Tdhwin, T[1,0,3], vc, vh, W, Hc, L, W, Hh, L, S, R, k, [], V = V, A = W*L,
    CLc = NL*L + 0.04, CLh = NL*L + 0.04)
(T[-1,:,:], I_v[-1,:], Velem[-1,:]) = singleCell(273 + 60, Thnin, vc, vh, W, Hc, L, W, Hh, L, S, R, k, [], V = V, A = W*L
    , CLc = NL*L + 0.04, CLh = NL*L + 0.04)


# Start iteration
while (np.abs(T - T2) > 0.01).any():
    T2 = np.copy(T)
    for n in range(N):
        for w in range(Nw):
            if n == 0:
                (T[n,w,:], I_v[n,w], Velem[n,w]) = singleCell(Tdhwin, T[1,w,3], vc, vh, W, Hc, L, W, Hh, L, S, R, k, [],
                    V = V, A = W*L, CLc = NL*L + 0.04, CLh = NL*L + 0.04)
            elif n == N-1:
                (T[n,w,:], I_v[n,w], Velem[n,w]) = singleCell(T[-2,w,0], Thnin, vc, vh, W, Hc, L, W, Hh, L, S, R, k, [],
                    V = V, A = W*L, CLc = NL*L + 0.04, CLh = NL*L + 0.04)
            else:
                (T[n,w,:], I_v[n,w], Velem[n,w]) = singleCell(T[n-1,w,0], T[n+1,w,3], vc, vh, W, Hc, L, W, Hh, L, S, R, k
                    , [], V = V, A = W*L, CLc = NL*L + 0.04, CLh = NL*L + 0.04)

    for n in range(N):
        for w in range(Nw):
            if n == 0:
                (T[N-n-1,w,:], I_v[N-n-1,w], Velem[N-n-1,w]) = singleCell(T[-2,w,0], Thnin, vc, vh, W, Hc, L, W, Hh, L, S
                    , R, k, [], V = V, A = W*L, CLc = NL*L + 0.04, CLh = NL*L + 0.04)
            elif n == N-1:
                (T[N-n-1,w,:], I_v[N-n-1,w], Velem[N-n-1,w]) = singleCell(Tdhwin, T[1,w,3], vc, vh, W, Hc, L, W, Hh, L, S
                    , R, k, [], V = V, A = W*L, CLc = NL*L + 0.04, CLh = NL*L + 0.04)
            else:
                (T[N-n-1,w,:], I_v[N-n-1,w], Velem[N-n-1,w]) = singleCell(T[-(n+2),w,0], T[-(n),w,3], vc, vh, W, Hc, L, W
                    , Hh, L, S, R, k, [], V = V, A = W*L, CLc = NL*L + 0.04, CLh = NL*L + 0.04)

# Calculate Pin, Qdhw and COP
Pin = np.sum(I_v*Velem)
Tdhwout = np.mean(T[-1,:,0])
Qdhw = 4180 * 1000 * W * Nw * Hc * vc * (Tdhwout - Tdhwin)
COP = Qdhw/Pin
```

Code snippet E.2: Code used to acquire data from the test setup with the DATAQ DI-2008.

```python
import serial
import serial.tools.list_ports
import keyboard
import time
import matplotlib
import matplotlib.pyplot as plt
import numpy as np

matplotlib.use('Tkagg')

TCs = 8
Flows = 2

measHis = np.zeros((TCs, 1))
rateHis = np.zeros((Flows, 1))

slist = []
for i in range(TCs):
    slist.append(0x1300 + i)

if Flows > 0:
    slist.append(0x080A)
    if Flows == 2:
        slist.append(0x0809)

analog_ranges = [.5, 0.25, 0.1, .05, .025, .01, 0, 0, 50
    ,25, 10, 5, 2.5, 1, 0, 0]
rate_ranges = tuple((50000,20000,10000,5000,2000,1000,
    500,200,100,50,20,10))
tc_m = [0.023956,0.018311,0.021515,0.023987,0.022888,
    0.02774,0.02774,0.009155]
tc_b = [1035,400,495,586,550,859,859,100]
range_table = list(())

# Define flag to indicate if acquiring is active
acquiring = False
ser=serial.Serial()

def discovery():
    # Get a list of active com ports to scan for possible
    #     DATAQ Instruments devices
    available_ports = list(serial.tools.list_ports.comports
        ())
    # Will eventually hold the com port of the detected
    #     device, if any
    hooked_port = ""
    for p in available_ports:
        # Do we have a DATAQ Instruments device?
        if ("VID:PID=0683" in p.hwid):
            # Yes! Dectect and assign the hooked com port
            hooked_port = p.device
            break
    if hooked_port:
        print("Found a DATAQ Instruments device on",
            hooked_port)
        ser.timeout = 0
        ser.port = hooked_port
        ser.baudrate = '115200'
        ser.open()
        return (True)
    else:
        # Get here if no DATAQ Instruments devices are
        #     detected
        print("Please connect a DATAQ Instruments device")
        input("Press ENTER to continue...")
        return(False)

# Sends a passed command string after appending <cr>
def send_cmd(command):
    ser.write((command+'\r').encode())
    time.sleep(.1)
    if not(acquiring):
        # Echo commands if not acquiring
        while True:
            if(ser.inWaiting() > 0):
                while True:
                    try:
                        s = ser.readline().decode()
                        s = s.strip('\n')
                        s = s.strip('\r')
                        s = s.strip(chr(0))
                        break
                    except:
                        continue
                if s != "":
                    print(s)
                    break
```

```python
# Configure the instrment's scan list
def config_scn_lst():
    # Scan list position must start with 0 and increment
    #     sequentially
    position = 0
    for item in slist:
        send_cmd("slist "+ str(position) + " " + str(item)
            )
        position += 1
        # Update the Range table
        if (item & 0xf < 8) and (item & 0x1000 == 0):
            # This is a voltage channel.
            range_table.append(analog_ranges[item >> 8])

        elif (item & 0xf < 8) and (item & 0x1000 != 0):
            # This is a TC channel. Append 0 as a
            #     placeholder
            range_table.append(0)

        elif item & 0xf == 8:
            # This is a dig in channel. No measurement
            #     range support.
            # Append 0 as a placeholder
            range_table.append(0)

        elif item & 0xf == 9:
            """
            This is a rate channel
            Rate ranges begin with 1, so subtract 1 to
                maintain zero-based index
            in the rate_ranges tuple
            """
            range_table.append(rate_ranges[(item >> 8)-1])

        else:
            """
            This is a count channel. No measurement range
                support.
            Append 0 as a placeholder
            """
            range_table.append(0)

while discovery() == False:
    discovery()
# Stop in case Device was left running
send_cmd("stop")
# Keep the packet size small for responsiveness
send_cmd("ps 1")
# Configure the instrument's scan list
config_scn_lst()
# set filter mode to average
send_cmd("filter * 1")
# obtain divided for rate troughput
send_cmd("info 9")
# Define sample rate = 10 Hz (refer to protocol:)
# 800/(srate * dec) = 800/(4 * 20) = 10 Hz
send_cmd("dec 1")
send_cmd("srate 4")

k = 0
decimation = 1500
TCresults = []
TC1results = []
c1, c2 = 0, 0
t1, t2 = 0, 0
cv = np.array([])
rateValue1, rateValue2 = 0, 0

plotLoop = 0
# reset the counter
send_cmd("reset 1")
plotted = True
Creset = False

plt.ion()
fig = plt.figure(1)
plt.plot([1],[1])
plt.draw()
plt.pause(0.001)
```

```python
        slist_pointer = 0
        output_string = ""

        while True:
            # If key 'SPACE' start scanning
            if keyboard.is_pressed('g' or 'G') and acquiring ==
                    False:
                tic = time.time()
                keyboard.read_key()
                acquiring = True
                send_cmd("start")
                t1 = time.time()
            # If key 'esc' stop scanning
            if keyboard.is_pressed('s' or 'S'):
                keyboard.read_key()
                send_cmd("stop")
                time.sleep(1)
                #ser.flushInput()
                print ("stopped")
                ser.flushInput()
                acquiring = False
            # If key 'q' exit
            if keyboard.is_pressed('q' or 'Q'):
                keyboard.read_key()
                send_cmd("stop")
                break
            # If key 'r' reset counter
            if keyboard.is_pressed('r' or 'R'):
                keyboard.read_key()
                send_cmd("reset_1")

            # clearing the storage
            TCresults = []
            while (ser.inWaiting() >= (2 * len(slist))):
                if not Creset:
                    Creset = True
                plotted = False
                for i in range(len(slist)):
                    # The four LSBs of slist determine measurement
                            function
                    function = slist[slist_pointer] & 0xf
                    mode_bit = slist[slist_pointer] & 0x1000
                    # Always two bytes per sample...read them
                    bytes_received = ser.read(2)
                    if (function < 8) and (not(mode_bit)):
                        # Working with a Voltage input channel.
                                Scale accordingly.
                        result = range_table[slist_pointer] * int.
                            from_bytes(bytes_received,byteorder='
                            little', signed=True) / 32768
                    elif (function < 8) and (mode_bit):
                        result = int.from_bytes(bytes_received,
                            byteorder='little', signed=True)
                        if result == 32767:
                            output_string = output_string + "cjc_
                                error,_"
                        elif result == -32768:
                            output_string = output_string + "open,_
                                "
                        else:
                            # Get here if no errors, so isolate TC
                                    type
                            tc_type = slist[slist_pointer] & 0x0700
                            # Move TC type into 3 LSBs to form an
                                    index we'll use to select m & b
                                    scaling constants
                            tc_type = tc_type >> 8
                            result = tc_m[tc_type] * result + tc_b[
                                tc_type]
                            TCresults.append(result)
                    elif function == 8:
                        # Working with the Digital input
                                channel
                        result = (int.from_bytes(bytes_received
                            ,byteorder='big', signed=False)) &
                            (0x007f)
                    elif function == 9:
                        # Working with the Rate input channel
                        result = (int.from_bytes(bytes_received
                            ,byteorder='little', signed=True)
                            + 32768) / 65535 * (range_table[
                            slist_pointer])
                        rateValue1 = result
                    else:
                        # Working with the Counter input channel
                        result = (int.from_bytes(bytes_received,
                            byteorder='little', signed=True)) +
                            32768
                        c2 = result
                        t2 = time.time()
                        #print(c2, t2)

                    # Get the next position in slist
                    slist_pointer += 1

                    if (slist_pointer + 1) > (len(slist)):
                        # End of a pass through slist items...
                                output, reset, continue
                        output_string = ""
                        slist_pointer = 0

            if acquiring and not(plotted):

                if len(measHis[0,:]) >= decimation:
                    measHis = measHis[:,-decimation:]
                if len(rateHis[0:1]) >= 10:
                    rateHis = rateHis[:,-10:]

                measHis = np.c_[measHis, np.zeros((TCs, len(
                    TCresults[0::TCs])))]
                rateHis = np.c_[rateHis, np.zeros((Flows,1))]
                rateValues = [rateValue1]
                if Flows > 1:
                    if t2 - t1 == 0.0:
                        t2 = t1 + 1e-6
                    if (c2 - c1) < 0:
                        rateValue2 = (c2 + 2**16 - c1)/(t2 - t1)
                    else:
                        rateValue2 = (c2 - c1)/(t2 - t1)
                    if len(cv) == 10:
                        cv = cv[1:]
                    cv = np.append(cv, rateValue2)
                    c1, t1 = c2, t2
                    rateValues.append(np.mean(cv))
                    rateHis[:,-1] = rateValues
                output_string = ""
                plt.clf()
                if TCs > 0 and Flows > 0:
                    plt.subplot(211)
                    plt.xlim((15, 70))
                if TCs > 0:
                    avgRes = []
                    for i in range(TCs):
                        measHis[i,-len(TCresults[0::TCs]):] =
                            TCresults[i::TCs]
                    for i in range(TCs):
                        plt.barh(i+1, np.mean(measHis[i,:]))
                        plt.text(np.mean(measHis[i,:]), i+1, str(np
                            .round(np.mean(measHis[i,:]), 1)) + "_
                            +_" + str(np.round(np.std(measHis[i
                            ,:]),3)), ha="left", va="center",
                            fontsize=14)
                    plt.xlabel("Temeprature_(C)")
                    plt.ylabel("Analogue_inpute")
                    plt.title("Decimation_is_over_%d_samples,_or_
                        %.2f_seconds" % (decimation, decimation
                        /(200/8)))
                if TCs > 0 and Flows > 0:
                    plt.subplot(212)
                    plt.xlim(0, 6)
                if Flows > 0:
                    for i in range(Flows):
                        plt.barh(i+1, np.mean(rateHis[i,-10:])
                            /950*60)
                        plt.text(0.1*np.mean(rateHis[i,-10:])
                            /950*60, i+1, str(np.round(np.mean(
                            rateHis[i,-10:])/950*60,2)) + "_+_" +
                            str(np.round(np.std(rateHis[i,-10:])
                            /950*60,3)), ha="left", va="center",
                            fontsize=18)
                    plt.xlabel("Flowrate_(L/min)")
                    plt.ylabel("DHN_-_DHW")
                if not (plotLoop - 0):
                    plotLoop = 0
                    plt.draw()
                    plt.pause(0.0000001)
                else:
                    plotLoop += 1
                plotted = True
                Creset = False

SystemExit
```

F

2D heat equation

Figure F.1: Supporting functions used for the PeHEX model

```python
import matplotlib.pyplot as plt
import numpy as np
from mpl_toolkits import mplot3d
import matplotlib.patches as mpatches


def createGrid(x0, x1, dx, y0, y1, dy):
    Nx = int((x1-x0)/dx)
    Ny = int((y1-y0)/dy)

    x = np.linspace(x0 + dx/2, x1-dx/2, Nx)
    y = np.linspace(y0 + dy/2, y1-dy/2, Ny)

    [X, Y] = np.meshgrid(x, y)
    X = np.reshape(X,(-1))
    Y = np.reshape(Y,(-1))
    return X, Y


def removePoints(X, Y, x0, x1, y0, y1):
    X_n = []
    Y_n = []
    for n in range(len(X)):
        if (X[n] > x0) and (X[n] < x1) and (Y[n] > y0) and (Y[n] < y1):
            continue
        X_n = np.append(X_n, X[n])
        Y_n = np.append(Y_n, Y[n])
    return X_n, Y_n


def findNeighbours(X, Y, dx, dy):
    Nb = -np.ones((len(X), 4))
    for n in range(len(X)):
        i = 0
        for m in range(len(X)):
            if m == n:
                continue
            if i == 4:
                break
            px = (X[m] > X[n] - 1.5*dx) * (X[m] < X[n] + 1.5*dx)
            py = (Y[m] > Y[n] - 1.5*dy) * (Y[m] < Y[n] + 1.5*dy)
            if px*py:
                if np.linalg.norm([X[n]-X[m], Y[n]-Y[m]]) > max(dx,dy):
                    continue
                Nb[n,i] = int(m)
                i += 1
    return Nb.astype(int)


def plotNeighbours(X, Y, nb):
    for i in range(len(nb)):
        if nb[i] == -1:
            break
        plt.plot(X[nb[i]], Y[nb[i]], 'ro')


def classifyGridPoints(Nb):
    pointType = 4*np.ones(len(Nb))
    pointType = 4 - np.sum(Nb == -1, 1)
    return pointType
```

```python
def plotClassifiedGrid(X, Y, gridPointClass):
    plt.plot(X[gridPointClass == 4], Y[gridPointClass ==
        4], 'bo')
    plt.plot(X[gridPointClass == 3], Y[gridPointClass ==
        3], 'ro')
    plt.plot(X[gridPointClass == 2], Y[gridPointClass ==
        2], 'ko')
    plt.plot(X[gridPointClass == 1], Y[gridPointClass ==
        1], 'mo')
    return None

def findBoundaryNeighbours(Nb, gridPointClass):
    isBoundary = (gridPointClass != 4)
    BNb = -np.ones((len(Nb),3))
    for n in range(len(Nb)):
        k = 0
        if isBoundary[n]:
            for m in range(len(Nb[n])):
                if Nb[n][m] == -1:
                    break
                BNb[n,k] = Nb[n][m]
                k += 1
    return BNb.astype(int)

def classifyBoundaryPoints(X, Y, BNb):
    boundaryClass = np.zeros((len(X)))
    for n in range(len(X)):
        for nb in BNb[n]:
            if nb == -1:
                break
            if np.abs(X[n] - X[nb]) < 0.001:
                boundaryClass[n] -= 1
            if np.abs(Y[n] - Y[nb]) < 0.001:
                boundaryClass[n] += 1
    return boundaryClass

def findBoundaryInteriorPair(X, Y, BNb, CBP, B):
    boundaryPair = np.zeros((len(B)))
    for n in range(len(B)):
        if B[n]:
            if CBP[n] == -1:
                for m in range(len(BNb[n])):
                    if np.abs(X[n] - X[BNb[n,m]]) > 0.001:
                        boundaryPair[n] = BNb[n,m]
                        break
            elif CBP[n] == 0:
                for m in range(len(BNb[n])):
                    if np.abs(Y[n] - Y[BNb[n,m]]) > 0.001:
                        boundaryPair[n] = BNb[n,m]
                        break
                    elif np.abs(X[n] - X[BNb[n,m]]) >
                        0.001:
                        boundaryPair[n] = BNb[n,m]
                        break
            else:
                for m in range(len(BNb[n])):
                    if np.abs(Y[n] - Y[BNb[n,m]]) > 0.001:
                        boundaryPair[n] = BNb[n,m]
                        break
    return boundaryPair.astype(int)

def setBoundaryType(X, Y, B):
    boundaryType = np.zeros((len(B)))
    boundaryType[(X > 10) * (X < 50) * (Y < 0.5) * B] = 1
    boundaryType[(X > 11 - 1.5*dx) * (X < 49 + 1.5*dx) * (Y
        > 0.5) * B] = 2
    return boundaryType

def classifyNeighbour(x, nbx):
    return (np.abs(x - nbx) < 0.5*dx)
    if np.abs(x - nbx) < 0.5*dx:  # Check if neighbour
        point is vertical
        return 1
    else:
        return 0

def findCellTemp(X, cx, T, CNb):
    rhs = 0
    lhs = 0
    for n in range(len(CNb)):
        if CNb[n] == -1:
            break
        if classifyNeighbour(cx, X[CNb[n]]):
            rhs += T[CNb[n]] * (k*dx/dy)
            lhs += k*dx/dy
        else:
            rhs += T[CNb[n]] * (k*dy/dx)
            lhs += k*dy/dx
    return lhs, rhs
```

```python
def finddT(T0, f):
    Tn = np.zeros((T0.shape))
    dT = np.zeros((T0.shape))
    Tdm = np.ones((len(T0),1)) * T0 + 1e-3 * np.diag(np.
        ones(T0.shape))
    J = np.zeros((len(T0), len(T0)))
    Jinv = np.zeros((J.shape))
    for n in range(len(T0)):
        J[:,n] = (g(Tdm[n,:], f) - g(T0, f))/(Tdm[n,n] - T0
            [n])
##         print(J[CBP[n].astype(int), n])
    Tn = T0 - np.linalg.solve(J, g(T0, f))
    return (Tn - T0), Tn, J

def g(T0, f):
    return T0 - f(T0)

def finddT2(f, X, Y, T0, Nb, B, boundaryPair, boundaryType,
    CBP):
    Tn = np.copy(T0)
    dT = np.zeros((T0.shape))
    Tdm = np.ones((len(T0),1)) * T0 + 1e-3 * np.diag(np.
        ones((T0.shape)))
    J = np.zeros((len(T0), len(T0)))
    Jinv = np.zeros((J.shape))
    g0 = T0 - f(X, Y, T0, Nb, B, boundaryPair, boundaryType
        , CBP)
    for n in range(len(T0)):
        checkCells = np.append([n], Nb[n][Nb[n] != -1])
        for j in range(len(checkCells)):
            cell = checkCells[j]
            span = Nb[cell][Nb[cell] != -1]
            fullspan = np.append([cell], span)
            func = lambda T : singleSweep(X[fullspan], Y[
                fullspan], T, [np.linspace(1,4,4) - np.
                linspace(2,5,4) * (Nb[cell] == -1)], B[
                fullspan], [], boundaryType[fullspan], CBP[
                fullspan], Xn = [X[cell]])
            J[cell, n] = ((g(Tdm[n, fullspan], func) - g0[
                fullspan])/(Tdm[n, n] - T0[n]))[0]
    Tn = T0 - np.linalg.solve(J, g0)
    return (Tn - T0), Tn, J

def singleSweep(X, Y, T, Nb, B, boundaryPair, boundaryType,
    CBP, Xn = -1):
    if Xn == -1:
        Xn = X
    Ttemp = np.copy(T)
    for n in range(len(Xn)):
        if B[n]:
            lhs, rhs = findCellTemp(X, X[n], T, Nb[n].
                astype(int)) # build [Ttemp * (lhs) = (rhs
                )]
            if int(boundaryType[n]) == 0:
                if CBP[n] == 1:
                    lhs += dx * 4 * (5.67 * 1e-8) * (
                        emissivity) * Ttemp[n]**3
                    rhs += dx * 4 * (5.67 * 1e-8) * (
                        emissivity) * (Ttemp[n]**3) * Tenv
                else:
                    lhs += dy * 4 * (5.67 * 1e-8) * (
                        emissivity) * (Ttemp[n]**3)
                    rhs += dy * 4 * (5.67 * 1e-8) * (
                        emissivity) * (Ttemp[n]**3) * Tenv
                pass
            if int(boundaryType[n]) == 1:
                if CBP[n] == 1:
                    rhs += q*dx
                else:
                    rhs += q*dy
            if int(boundaryType[n]) == 2:
                if CBP[n] == 1:
                    rhs += h*dx*Tw
                    lhs += h*dx
                else:
                    rhs += h*dy*Tw
                    lhs += h*dy
            Ttemp[n] = rhs/lhs
    for n in range(len(Xn)):
        if B[n]:
            continue
        lhs, rhs = findCellTemp(X, X[n], T, Nb[n].astype(
            int)) # build [Ttemp * (lhs) = (rhs)]
        Ttemp[n] = rhs/lhs
    return Ttemp
```

```python
def params(z):
    Qh = 55.3 + (51.1 - 55.3)/1600*z
    Qc = 23.3 + (20.0 - 23.3)/1600*z
    Twc = 313.2 + (320.7 - 313.2)/1600*z
    Twh = 314.1 + (317.9 - 314.1)/1600*z
    return Qh, -Qc, Twc, Twh

def solveTotalSystem(X, Y, T, Nb, B, boundaryPair,
    boundaryType, CBP, z = -1):
    global q
    global Tw
    dz = 1600/N
    Tc = np.zeros((len(T), N))
    Th = np.zeros((len(T), N))
    if z == -1:
        for i in range(N):
            Qh, Qc, Twc, Twh = params(dz/2 + i*dz)
            if i == 0:
                Tcold = np.copy(T)
                Thold = np.copy(T)
            else:
                Tcold = np.copy(Tc[:,i-1])
                Thold = np.copy(Th[:,i-1])

            dT = 1
            q = Qh/(0.04**2)
            Tw = Twc
            while (np.linalg.norm(dT) > 1e-5):
                dT, _, _ = finddT2(singleSweep, X, Y, Tcold
                    , Nb, B, boundaryPair, boundaryType,
                    CBP)
                Tcold += dT

            dT = 1
            q = Qc/(0.04**2)
            Tw = Twh
            while (np.linalg.norm(dT) > 1e-5):
                dT, _, _ = finddT2(singleSweep, X, Y, Thold
                    , Nb, B, boundaryPair, boundaryType,
                    CBP)
                Thold += dT

            Tc[:,i] = Tcold
            Th[:,i] = Thold

            if np.mod(i, N/10) == 0:
                print('another_10_percent_done')
    else:
        Qh, Qc, Twc, Twh = params(z)
        q = Qh/(0.04**2)
        Tw = Twc
        dT = 1
        Tcold = np.copy(T)
        while (np.linalg.norm(dT) > 1e-5):
            dT, _, _ = finddT2(singleSweep, X, Y, Tcold, Nb
                , B, boundaryPair, boundaryType, CBP)
            Tcold += dT

        Qh, Qc, Twc, Twh = params(z)
        q = Qc/(0.04**2)
        Tw = Twh
        dT = 1
        Thold = np.copy(T)
        while (np.linalg.norm(dT) > 1e-5):
            dT, _, _ = finddT2(singleSweep, X, Y, Thold, Nb
                , B, boundaryPair, boundaryType, CBP)
            Thold += dT

        Tc = Tcold
        Th = Thold

    return Tc, Th

def plotResults(Tc, Th):
    cmin, cmax = np.min(Th - 273), np.max(Tc - 273)
    try:
        L = Tc.shape
        N = L[1]
    except:
        N = 1
    if N > 1:
        plt.ion()
        plt.plot([0.01, 0.05, 0.05, 0.01, 0.01], [0.0039/2,
            0.0039/2, -0.0039/2, -0.0039/2, 0.0039/2], 'k
            ')
        plt.scatter([X, X], [Y + 0.0039/2, - Y - 0.0039/2],
            c = [Tc[:,0] - 273, Th[:,0] - 273])
        plt.colorbar()
        plt.clim(cmin, cmax)
        plt.draw()
```

```python
        dz = (400/N)/10
        for i in range(N-1):
            for j in range(10):
                plt.clf()
                plt.plot([0.01, 0.05, 0.05, 0.01, 0.01],
                    [0.0039/2, 0.0039/2, -0.0039/2,
                    -0.0039/2, 0.0039/2], 'k')
                plt.scatter([X, X], [Y + 0.0039/2, - Y -
                    0.0039/2], c = np.array([(10-j)*Tc[:,i
                    ] + (j)*Tc[:,i+1], (10-j)*Th[:,i] + (j
                    )*Th[:,i+1]])*1/10 - 273, cmap = "jet"
                    )
                plt.colorbar()
                plt.clim(cmin, cmax)
                plt.title('z_=_%f' % (1600/(2*N) + (1600/N)
                    *i + j*dz))
                plt.draw()
                plt.pause(0.00000001)
        i += 1
        plt.clf()
        plt.plot([0.01, 0.05, 0.05, 0.01, 0.01], [0.0039/2,
            0.0039/2, -0.0039/2, -0.0039/2, 0.0039/2], 'k
            ')
        plt.scatter([X, X], [Y + 0.0039/2, - Y - 0.0039/2],
            c = np.array([Tc[:,i], Th[:,i]]) - 273, cmap
            = "jet")
        plt.colorbar()
        plt.clim(cmin, cmax)
        plt.title('z_=_%f' % (1600/(2*N) + (1600/N)*i))
        plt.draw()
        plt.pause(0.00000001)

    else:
        Tc, Th = np.squeeze(Tc), np.squeeze(Th)
        plt.plot([0.01, 0.05, 0.05, 0.01, 0.01], [3.9/2,
            3.9/2, -3.9/2, -3.9/2, 3.9/2], 'k')
        plt.scatter([X, X], [1000*Y + 3.9/2, - 1000*Y -
            3.9/2], c = [Tc - 273, Th - 273], cmap = "jet"
            )
        cbar = plt.colorbar()
        plt.clim(cmin, cmax)
        cbar.set_label(r'$T$_($\degree$C)', fontsize=22)
        cbar.ax.tick_params(labelsize=22)
        plt.xlabel('x_(m)', fontsize=22)
        plt.xticks(fontsize=22)
        plt.ylabel('y_(mm)', fontsize=22)
        plt.yticks(fontsize=22)
        plt.grid()
        plt.xlim(-0.01, 0.07)
        plt.ylim(-15, 15)
        plt.show()

def plotCrossResults(X, Y, T, x = -1, y = -1):
    if x == -1:
        p = (Y == y)
    else:
        p = (X == x)

    Xslice = []
    Yslice = []
    Tslice = []

    L = T.shape[1]

    for i in range(L):
        Xslice = np.append(Xslice, (1600/(2*L) + i*1600/L)*
            np.ones(X[p].shape))
        if x == -1:
            Yslice = np.append(Yslice, X[p])
        else:
            Yslice = np.append(Yslice, Y[p])
        Tslice = np.append(Tslice, T[p,i])

    plt.scatter(Xslice, Yslice, c = Tslice, cmap = "jet")

x0 = 0
x1 = 60
dx = 0.5
y0 = 0
y1 = 4
dy = 0.1

k = 250 # in W/mK
h = 4000 # W/m2
emissivity = 0.3

Tenv = 273 + 20

# Creating mesh
X, Y = createGrid(x0, x1, dx, y0, y1, dy)
X, Y = removePoints(X, Y, 11, 49, 1, 5)
```

```python
Nb = findNeighbours(X, Y, dx, dy)
gridPointClass = classifyGridPoints(Nb)
B = (gridPointClass != 4)
Bx = X[B]
By = Y[B]
BNb = findBoundaryNeighbours(Nb, gridPointClass)
CBP = classifyBoundaryPoints(X, Y, BNb)
boundaryPair = findBoundaryInteriorPair(X, Y, BNb, CBP, B)
boundaryType = setBoundaryType(X, Y, B)

X /= 1000
Y /= 1000
dx /= 1000
dy /= 1000


#'''
# Plotting the gridpoints
plotClassifiedGrid(X, Y, gridPointClass)
# Plotting grid edges
x_edges = np.array([0, 0, 11, 11, 49, 49, 60, 60, 0])*1/1000
y_edges = np.array([0, 4, 4, 1, 1, 4, 4, 0, 0])*1/1000

for j in range(len(x_edges)-1):
    plt.plot(x_edges[j:j+2], y_edges[j:j+2], 'k')

plt.grid()
plt.show()

##plt.plot(X[B], Y[B], 'bo')
plt.plot(X[(boundaryType == 0) * B], Y[(boundaryType == 0) * B], 'ro')
plt.plot(X[(boundaryType == 1) * B], Y[(boundaryType == 1) * B], 'ko')
plt.plot(X[(boundaryType == 2) * B], Y[(boundaryType == 2) * B], 'mo')

plt.grid()
plt.show()
#'''

N = 1
Tinit = 293 * np.ones((X.shape))
Tc, Th = solveTotalSystem(X, Y, Tinit, Nb, B, boundaryPair, boundaryType, CBP)
plotResults(Tc, Th)
plt.show()
```