POLITECNICO DI MILANO

Facoltà di Ingegneria Industriale e dell'Informazione

Dipartimento di Energia

PolyToNuc MSc in Nuclear Engineering

Title: System-level modeling of a Liquid Metal Divertor for the EU DEMO Fusion Reactor

Relatore: Prof. Stefano Lorenzi (PoliMi)

Correlatore: Prof. Fabio Subba (PoliTo)

Correlatore: Antonio Froio (PoliTo)

Correlatore: Giuseppe Francesco Nallo (PoliTo)

Gianluca Palumbo
897901

Ai miei cugini

Luca e Sandro

"Un uomo è forte quando impara di esser fragile"

Contents

List of figures

List of tables

List of Equations

List of Abbreviations

EIA: Energy Information Administration

NPP: Nuclear Power Plant

FPP: Fusion Power Plant

- Different FPP research reactors

ITER: International Thermonuclear Experimental Reactor

JET: Joint European Torus

DTT: Divertor Tokamak Test-facility

Tokamak: to(roidalnj) ka(mera) mak(kina): "macchina (per fusione nucleare) a camera toroidale"

DEMO: DEMOnstration Power Plant

D: Deuterium

T: Tritium

LMD: Liquid Metal Divertor

- Components of a Nuclear Fusion Power Plant in Magnetic Configuration

DP : Divertor Plates

VC: Vacuum Chamber

VV : Vacuum Vessel

SC: SuperConductor

BM: Breeding Module

B: Blanket

NBI: Neutral Beam Injection

PI: Pellet Injection

ELM: Edge Localized Modes

EM: Electromagnetic

PFC: Plasma Facing Component

PFU: Plasma Facing Unit

PFS: Plasma Facing Surface

CPS: Capillary Porous Structure

DC: Differential Chamber

EC: Evaporation Chamber

MC: Main Chamber

SOL: Scrape-Off Layer

CHF: Critical Heat Flux

IT: Inboard Target

OT: Outboard Target

HK: Hertz-Knudsen

BC: Boundary Condition

FEM: Finite Element Modelling

HF: Heat Flux

List of symbols

E_b: binding energy

$t_{1/2}$: half life

λ_q: SOL "power width"

I_p: plasma current

Z_{eff}: effective charge

Γ: mass flow rate

ΔH_{eva}: latent heat of evaporation

P_C: capillary pressure

σ: surface tension of the LM

θ: wetting angle

r_{eff}: effective radius (linked to the width of the wire mesh)

Δp_L: hydrayulic pressure drop

Δp_G: hydrostatic pressure drop (linked to gravitational force)

Δp_L: pressure drop at evaporation surface due to liquid-vapour phase change

Δp_{MHD}: pressure drop due to MHD effect

τ: residence time

ξ: spatial position along the poloidal length inside the VT

Err_{overl}: relative error

A_{overl}: vector collecting the overlapping areas

Q_{overl}: total heat flow array given as input to the units in the model

Q_{tot}: total heat flow coming from the plasma and impinging on each ξ position of the VT

Abstract

Within nuclear engineering world, fission reactors are a widespread technology with a high level of maturity. On the other hand, nuclear fusion is still at an experimental stage, not yet able to produce an economically viable and competitive amount of energy. Among the major challenges to be faced, there is the issue of heat exhaust. If the "First Wall" component of the machine deals with the radiative exchange, in case of advection and conduction the system is formed by a target that can be a limiter or a divertor. The latter represents an interesting and efficient choice. ITER project will use a divertor completely solid, while for the DEMO higher size prototype this may not be enough.

In this thesis, the goal is to investigate the feasibility of a liquid metal divertor. Among the various designs proposed, we will focus on the one presented by ENEA, characterized by a porous structure through which the liquid metal flows. As a preliminary step to the design and testing phase of the component, in this thesis a model of the divertor has been developed using Modelica code and Dymola simulation environment in order to analyze the behavior of divertor under different design conditions. As working fluid, in addition to the lithium option, we wanted to investigate tin that has many advantages over the former, such as less evaporation and therefore less pollution of the plasma.

The results show that, following the evaporation of lithium and tin, the heat coming from plasma at some point starts to become comparable with the one extracted from phase change of the liquid. As a consequence, the metal remained liquid tends to decrease its temperature until it even reaches the point of solidification. It seems counterintuitive that for an increasing input power the target cools down, but these are the results of simulations and therefore in future research activities it is recommended to refine the model and investigate this phenomenon more thoroughly.

Estratto

Nell'ambito dell'ingegneria nucleare, i reattori a fissione sono una tecnologia diffusa e con un livello di maturità elevato. D'altra parte, la fusione nucleare è ancora ad uno stadio sperimentale, non riuscendo ancora a produrre una quantità di energia economicamente sostenibile e competitiva. Tra le grandi sfide da affrontare, c'è la problematica dello smaltimento del calore. Se il componente di "Prima Parete" della macchina si occupa dello scambio radiativo, in caso di convezione e conduzione il sistema è formato da un target che può essere un limiter oppure un divertore. L'ultima opzione rappresenta una scelta interessante e efficiente. Il progetto ITER utilizzerà un divertore interamente solido mentre per il prototipo di taglia superiore DEMO questo potrebbe non bastare.

In questa tesi l'obiettivo è quello di investigare la fattibilità di un divertore a metallo liquido. Tra i vari design proposti, ci si concentrerà su quello presentato da ENEA, caratterizzato da una struttura porosa attraverso cui il metallo liquido scorre. Come fase preliminare alla progettazione e alla fase di test del componente, in questa tesi è stato sviluppato un modello del divertore utilizzando il codice Modelica e l'ambiente di simulazione Dymola in modo da analizzare il comportamento del divertore in diverse condizioni di progetto. Come fluido di lavoro, oltre all'opzione litio, si è voluto investigare lo stagno che ha molti vantaggi rispetto al primo, quale una minore evaporazione e dunque un minore inquinamento del plasma.

Dai risultati si è evinto che, a seguito dell'evaporazione del Litio e dello Stagno, il calore derivante dal plasma a un certo punto diventa confrontabile con quello estratto dal liquido a causa del cambiamento di fase. Come conseguenza il metallo rimasto liquido tende a diminuire la sua temperatura fino a raggiungere addirittura il punto di solidificazione. Sembra controintuitivo che per una potenza in input sempre più grande il target si raffreddi, ma questi sono i risultati delle simulazioni e dunque in ricerche future si consiglia di affinare il modello e investigare più a fondo tale fenomeno.

Gianluca Palumbo: System-level modeling of a Liquid Metal Divertor for the EU DEMO Fusion Reactor

1 INTRODUCTION

According to the Energy Information Administration (EIA) in the next decades the energy need will increase of about 2 times, lead by the big request of Asia and in particular of China [1]. On the other hand there is another fundamental problem that must to be discussed about: the emission of greenhouse gases (CO_2, etc) in atmosphere. Indeed, the exploitation of fossil fuels for power production – according to the artificial global warming theory – is continuing to increase and to influence global temperature raising. Right or wrong it can be the theory (human fault or simply normal Earth era like the glacial one), the necessity to find energy sources which can meet our demand in a sustainable and green way is fundamental.

Renewables are surely the first that comes to mind and they correspond to solar and photovoltaic panels, hydroelectric power plants, wind turbines and so on. Their impact on the outside environment is very low but not 0. In fact, as a professor of Politecnico di Milano always tell, "there is no free meal" and also for this case there is no a final perfect solution.

- For the production and transportation of components of the plants there is a non negligible component of CO_2 emissions
- The waste management of structures (especially for photovoltaic panels with their rare earth elements abundance) is quite difficult and sometimes are not recovered correctly, but disposed in the landfill
- Energy production is not continuous nor modular, except for very few cases like hydroelectric power plants (which on the other hand need a dam, which can influence non negligibly the local marine life)

Among these considerations, the last one is the most troublesome because the industrial and civilian energy request does not follow the natural laws of energy production – related to wind or solar power availability, for example, not constant– and so we need something which can sustain the gap in energy production of renewable sources (e.g. when we have no sun for solar panels or no wind for wind turbines).

For this aim, the only power plants which does not need fossil fuels to run, in its lifetime operation emits CO_2 as much as renewable sources or even less and can run whenever we want is Nuclear Power.

1.1 Nuclear Energy

In general, in Nuclear Power Reactors, nuclear reactions are exploited because of huge energy released, which is therefore splitted through the products in form of kinetic energy[1]. The result is a heating up of the nearby environment by collisions, Bremsstrahlung or similar events, which is then transferred to a thermal cycle where electricity is generated. So far 2 main nuclear reactions have been investigated by humankind: fission of a nucleus and fusion of two different nuclei. In order to have a very brief physical explanation of the problem let us start from a generic nuclear reaction.

$$R1 + R2 => P1 + P2 + \upsilon \cdot x + Q$$
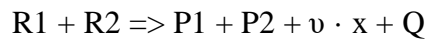
Equation 1.1 – Basic nuclear reaction.

Where R1 and R2 are the reactants, P1 and P2 the products of the reaction, nu the number of a certain particle released– which can be neutrons, protons or alfas – and Q the Q-value. For the sake of simplicity in an exothermic nuclear reaction – of our interest – we go from lower bounded reactants to higher bound products. This difference in binding energy gives birth to the well-known mass defect, which therefore generates a gain of energy represented by the previously introduced Q-value

$$E_b = \Delta_m \cdot c^2 = (m_{components} - m_{nucleus}) \cdot c^2$$

Equation 1.2 – Binding energy definition.

$m_{components} = Z \cdot m_p + (A - Z) \cdot m_n$: mass of all the neutrons and protons of the nucleus kept in consideration [8]

c: speed of light

$$(E_{b,P1} + E_{b,P2}) - (E_{b,R1} + E_{b,R2}) = Q$$

Equation 1.3 – Q-value definition.

The binding energy can be found from the graph below, multiplying the ordinate (E_b per nucleon) of the correspondent element by the number of nucleons of that element. In this way we can easily calculate how much is the gain from each fusion or fission event and why Nuclear Power has such a big energy density.

---

[1] Actually there is also the possibility that part of the energy is stored as excitation energy of the nuclei. For the sake of simplicity this evenience is just ignored since most of the energy becomes just kinetic one.
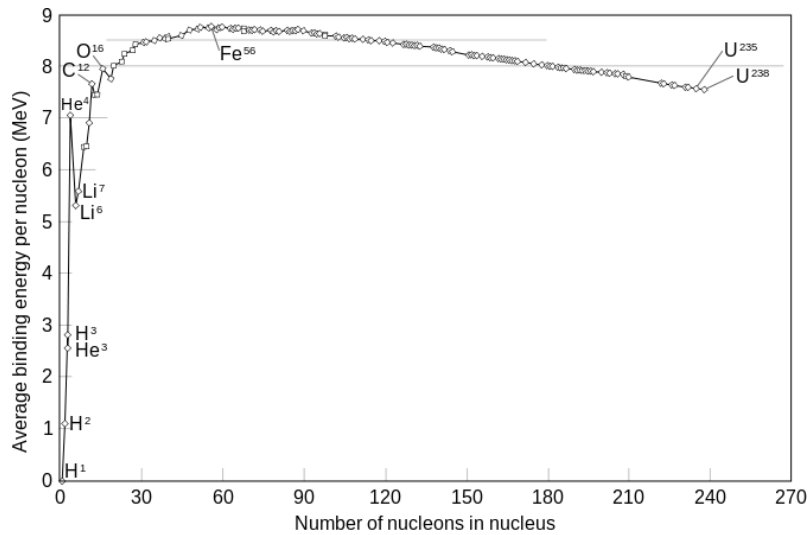
Figure 1.1 - Binding energy of some isotopes present in nature [8].

1.1.1 Fission

From Figure 1.1, our interest focuses on the right side – with respect to the peak of Fe-56 – of the curve. For this case, the reactants are one neutron and a nucleus of Uranium (or Plutonium or other useful Actinides), meanwhile the fission products can be Xe, Sr, Kr, Ba and so on with the adding of a certain number of neutrons

$$^{235}_{92}U + ^{1}_{0}n \rightarrow ^{139}_{54}Xe + ^{95}_{38}Sr + 2^{1}_{0}n + energy$$
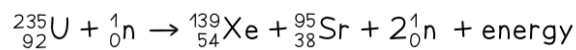
Equation 1.4 – U-235 nuclear reaction.

With this kind of reaction, we produce each time – so for each Uranium nucleus fission – about 211 MeV which is far way larger than the order of magnitude of the energy released in chemical reactions (e.g. burning of fossil fuels (Figure 1.2)).
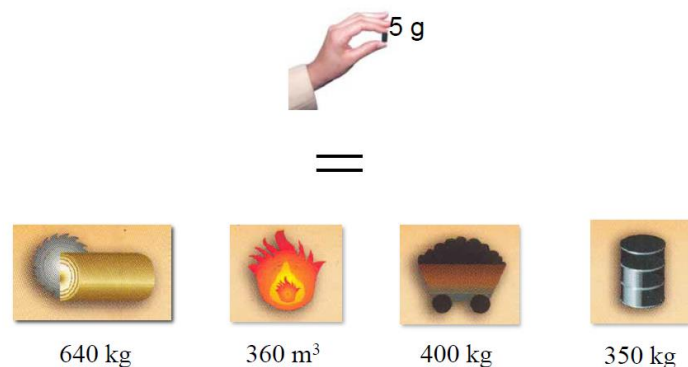
Figure 1.2 - Equivalence - in terms of energy released - of 5g of Uranium with 640 kg of wood, 360 m3 of gas methane, 400 kg of coal and 350 kg of oil [introduction to nuclear engineering, prof. Enrico Ricotti, sl_25].

3

For this reason, Nuclear Fission Power Plants are so compact that they are used also for space mission where we need very small components in order to fit such compact volumes. On the other hand, on Earth we have nowadays more than 400 NPPs (Nuclear Power Plants) with an overall power production installed of 400 GWe. In the near future, there will be nations with an increasing NPPs construction strategy or shutdown approach and this will lead probably in a sort of stationarity in energy production from this source [1].

1.1.2 Fusion

In this case, instead, we are in the region for A < 56, so for lighter nuclei, where the binding energy can be far way smaller than the one of the main characters of fission events. In particular we are talking about Hydrogen and Helium isotopes which combine and emit different particles and Q-values. The possible reactions are the ones below [8], meanwhile their different probabilities calculated with respect to $< \sigma \cdot v >$ in function of the thermal energy of the plasma can be seen in Figure 1.3.

D + T => He-4 (3.52 MeV) + n (14.06 MeV)

Equation 1.5 – D-T nuclear reaction.

D + D => T (1.01 MeV) + p (3.02 MeV)

Equation 1.6 – D-D nuclear reaction (i).

D + D => He-3 (0.82 MeV) + n (2.45 MeV)

Equation 1.7 – D-D nuclear reaction (ii).

D + He-3 => He-4 (3.6 MeV) + p (14.7 MeV)
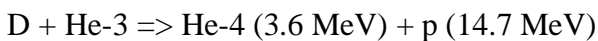
Equation 1.8 – D-He$_3$ nuclear reaction.
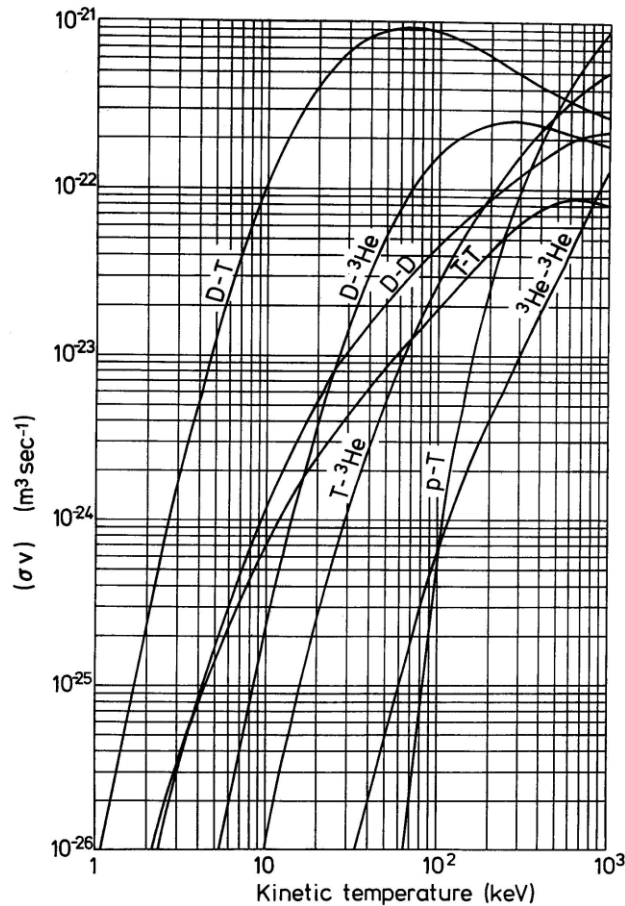
Figure 1.3 - Probability of occurrence (reaction rate) of different reactions in function of kinetic temperature [8].

Knowing the fact that values of reaction rate around $10^{-23} – 10^{-22}$ (reactions$\cdot$m^3/s) are quite good, we can conclude that we must reach really high temperatures[2] in order to have a reasonable number of reactions per unit time. Such exotic conditions push the reactants to a particular state – the so-called "plasma" – in which the particles are fully ionized. The percentage of ionization inside a plasma can be calculated by this simple equation

IP (ionization percentage) = exp (- Δ_E / T) $\cdot$ 100 = exp (- 13.6 eV / 10 keV) $\cdot$100 ~= 99.93 %

Equation 1.9 – IP definition.

Δ_E: Energy spent for the ionization of one Hydrogen atom

T: Temperature of the gas

---

[2] In order to have an idea of the temperatures we are talking about, let us consider the equivalence: T [K] = T [eV] / k_b (= 8.617 ×10−5 [eV/K]). So, if we have T = 10 keV – nearly our case – the result is a temperature even bigger than a hundred million Kelvin

A value like this one can be accepted (we can say with quite good approximation that we have a plasma), but if it starts to decrease too much, recombination of ions can occur and so no plasma state is reached.

In this condition the kinetic energy of the nuclei let them overcome the coulomb field created by the protons in the nucleus and increase the chance to have a fusion reaction. Both inertial and magnetic confinement can help us to succeed in this mission.

However, reaching such large values of temperature is not the only issue in a Fusion Power Plant (FPP). In fact, focusing on the magnetic confinement technology – the one of interest of this thesis – , there are other three main scientific and technological challenges which keeps busy scientist and engineers from all over the world:

1) Magnetic fields and Superconducting Magnets

   If the aim is to maximise the probability of collision of the reactants, one possibility can be to accelerate the fuel particles raising the occurrence of fusion reactions. An idea to perform this task is creating a Tokamak[3] very large magnetic fields which interacts with plasma particles[4]. According to Ampère Law we can see that to reach 10 T or similar values (magnetic field values significant for us) we must use very huge currents which can create abnormal Joule heating if normal conductors are exploited (e.g. Cu). That is why we consider Superconductors (SCs) which minimize this effect. For the particular disposition of these components inside the reactor we refer to Figure 1.4.

---

[3] Tokamak: a Russian acronym for "toroidal'naya kamera s magnitnymi katushkami" (transliterated from Russian "тороидальная камера е магнитыми катушками") translated as "toroidal chamber with magnetic coils"

[4] Remember that plasma is (almost) fully ionized, so it is made by positively (ions, i.e. the nuclei) and negatively (electrons) charged populations
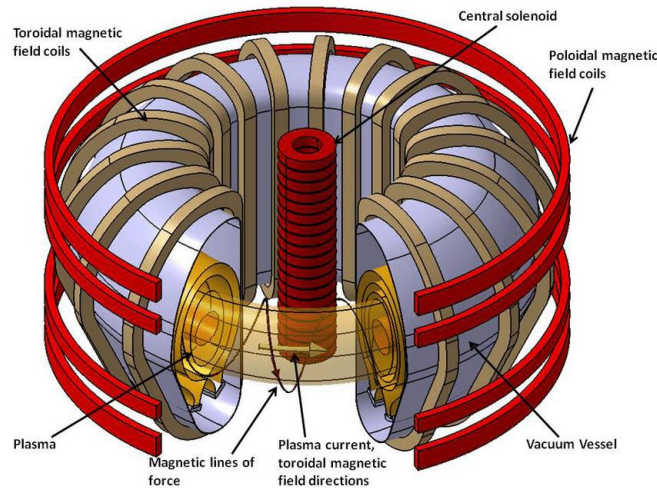
Figure 1.4 - Schematic representation of a tokamak with associated magnetic field lines and plasma current [https://www.researchgate.net/figure/The-scheme-of-tokamak-Figure-taken-from-3_fig2_297725507].

2) Tritium Breeding

Knowing that the most exploited fusion reaction – for now – is Equation 1.5, the fuel needed is Deuterium (D) and Tritium (T). Meanwhile the former is easy to be found (e.g. in ocean water is abundant), the latter is harder to be extracted, especially because it is radioactive ($t_{1/2}$ = about 12 years). HWR (Heavy Water Reactors, like CANDU) can produce some, but it is surely not sufficient and there is always the transportation problem, during which T decays. To overcome this issue, we create Breeding Modules (BMs) in the Blanket (B) which exploit neutrons coming out from the plasma to produce new T atoms after the start-up phase. Reinjecting it in the vacuum chamber (by PI or NBI), we are able to refill the plasma. Here below we can see a possible reaction – between neutron and a Lithium nucleus – which is largely exploited for this aim (Figure 1.5).
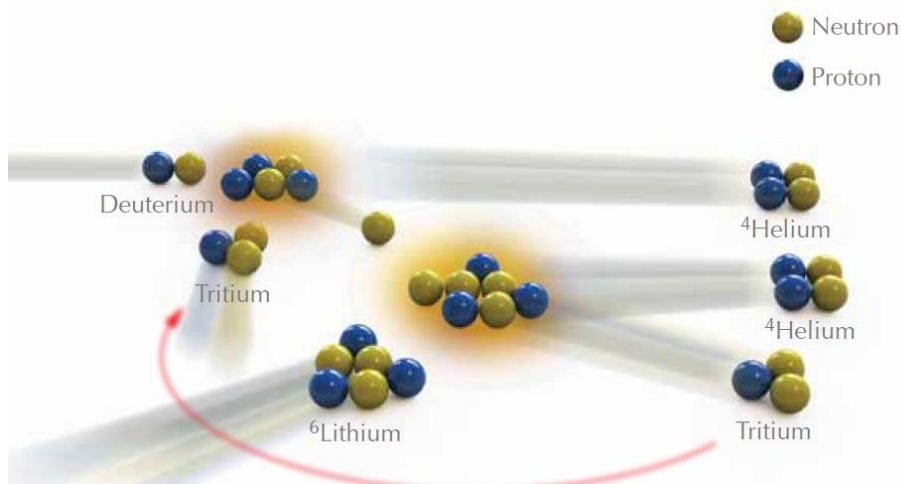
Figure 1.5 - Scheme about the reproduction of new Tritium after the reaction of neutrons, from plasma, and Lithium, from BMs.

7

3) Power Exhaust

This challenge is the one related mostly to the power balance of the system, because it focuses on the exhaust problem of all the heat load coming from plasma, which can be divided in radiative and conductive contributions. The former is roughly isotropic and it correspond to the bigger part, instead the latter is strongly anisotropic and, even if lower, it is concentrated on a very small surface, the divertor. So far only solid divertors are used in practice, but for future purposes like the EU DEMO reactor, bigger heat loads can melt the surface in some cases (like disruptions or ELM events) and better solutions must be investigated. In fact, many ideas were proposed and, among these, LMD (Liquid Metal Divertor) configuration is one of the most promising ones. It is not by chance that this thesis work focuses on this promising solution.

So many international efforts have been spent since 50s to succeed in such appealing technology. Past and future research reactors were and will be created – ITER, JET or DTT are just a few examples – with one goal in mind: create a FPP able to supply electricity to the grid, economically feasible and with positive gain (Output Energy produced – Input Energy consumed > 0). This project is intended to be national or intranational and in our case the joint venture EU reactor will be DEMO, hopefully the final milestone before accessing the long wished era of commercial fusion reactors. This reactor will be far way bigger than any other ones already constructed and also its output power will be much larger. For this reason, we should improve the technology available so far and this thesis aims to go on with the research activities about the divertor, the most critical component of the reactor in terms of power exhaust

1.2 Divertor

Now, in order to have a better outlook of the divertor problem, let us go back to Equation 1.5. Nowadays this one is the most exploited reaction taking in consideration the economical (abundance of fuel on earth, expensive machines) and technological (nominal temperatures needed) feasibility. In order to sustain fusion reactions (like Equation 1.5 indeed) we have to reach such big temperatures as discussed in previous paragraphs and for this aim a magnetic confinement approach is considered here, which roughly speaking accelerates gas particles until plasma state is reached. Despite SCs issues – like EM and mechanical stresses or quenching mitigation and prevention –, for now this is all we need to start a plasma. Knowing this, however, we have to take in consideration that this very high temperature gas at a certain time must impinge somewhere. When a target (limiter or divertor)

is correctly placed, the plasma is automatically divided in two regions: core and SOL[5] plasma. The former is the most important one because most of nuclear reactions take place and for this reason the temperatures are and must be the highest possible (about 1-20 keV). Moreover, this region is characterized by closed magnetic field lines, around which ions and electrons are coerced to whirl at a certain velocity and radius. Without this part, no self-sustained plasma can exist. The latter instead have lower temperatures (from 1 to 100 eV), lower pressures (around 0.01 bar against the near atmospheric pressure of the core) and open magnetic field lines which end right on the target. The separatrix – called also LCMS (Last Closed Magnetic Surface) – is the threshold between the two. In the end, as concern the width of the two, for the former case we talk about meters (approximately equal to the minor radius of the machine), meanwhile for the latter only of centimetres or millimetres. This phenomenon is due to the fact that transport of particles in the direction parallel to the field lines is preferred rather than diffusion in radial direction. In fact, when particles go out from the core, crosses separatrix and enter SOL, their velocity along field lines is close to the sound speed in the medium, meanwhile on the radial direction is far way lower.
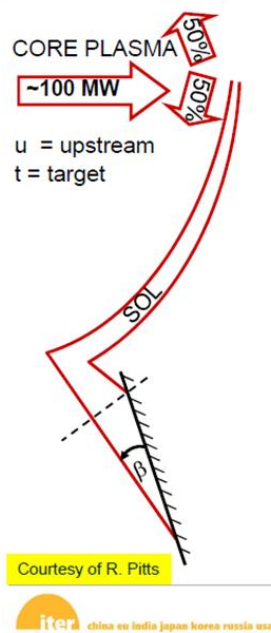
Figure 1.6 – SOL plasma zoom in tilted divertor case [Nuclear Fusion Engineering Masters, Torino, 11 January 2011, page 18].

In synthesis, taking into consideration these phenomena and other contributions, SOL width can be defined starting from the formula of λ_q (i.e. "power width"), proposed by Goldston [2], which helps us to see its inverse proportionality with respect to I_p (plasma current).

---

[5] Acronym for Scrape-Off Layer

$$\lambda_q = 5671 \cdot P_{sol}^{1/8} \frac{(1+k^2)^{5/8} a^{17/8} B^{1/4}}{I_p^{9/8} R} \left(\frac{2\bar{A}}{\bar{Z}^2(1+\bar{Z})} \right)^{7/16} \left(\frac{Z_{eff}+4}{5} \right)^{1/8}$$

Equation 1.10 – λ_q definition.

This seems quite unusual because the bigger the power output, the bigger the plasma current, the lower λ_q and so the impact surface. This feature, indeed, brings us to the main point inside the power exhaust discussion: the exhaust of heat by conduction, advection and irradiation. Even if the latter is the biggest one, it is distributed on a surface much larger and so the limit of a few MW/m^2 that can be handled by a solid surface is not reached and things are much easier. Instead, when SOL plasma impacts on a surface, that PFU experiences a heat load, by particle conduction and advection, much bigger and this is the main challenge. Few tens of MW/m^2 are predicted to be reached in DEMO reactor operation and normal ITER-like solid targets may be obsolete. The explanation stands in the λ_q inverse proportional behaviour with respect to I_p and the major radius R slight increase. In fact, if the power output from separatrix is 100 MW for ITER (Figure 1.6) and 150 MW for DEMO, the wetted area does not increase that much because of its direct proportionality with λ_q and R, which summed contribution does not lead to a big increase.

Now, changing topic, let us focus on the target technology. So far 2 types have been studied: limiter and divertor. The former is a physical object which is placed inside the toroidal chamber. Thanks to this entity it is forced the mechanical creation of the separatrix at a certain radial distance from the toroidal axis. The latter, on the other hand, exploits the positioning of a coil under the machine, along the toroidal direction, which changes the conformation of the magnetic field and gives birth to an X-point, where magnetic field is 0[6], and a private plasma with characteristics far way less exotic than core plasma.

---

[6] Here we are discussing about the typical single null configuration with only one X-point (see Figure 1.7 (left)). Many other ideas have been studied, like the double null one, but they are not of our interest for this work
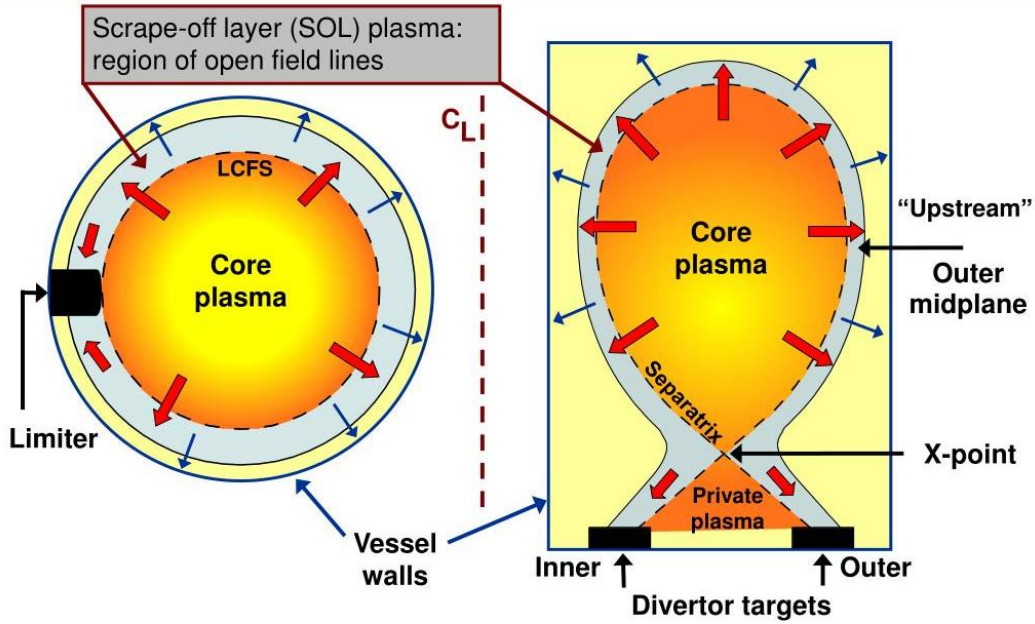
Figure 1.7 - Core and SOL plasma formation in limiter (at the left) and divertor (at the right) configuration [Nuclear Fusion Reactor Physics, prof. Fabio Subba, sl_12].

When SOL intersects DPs, plasma particles interact with the target in many ways: neutrals can penetrate the structure of the PFU, PFU atoms can be extracted (or more technically, sputtered) and so on.
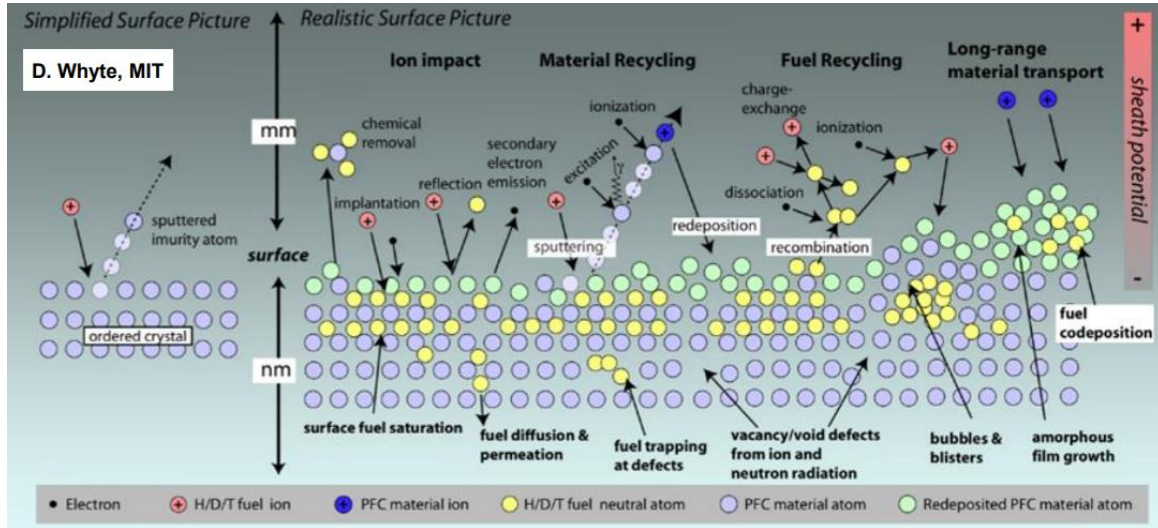
Figure 1.8 - plasma-wall interaction phenomen.

The latter in particular will cause a net erosion of the target and at the same time a contamination of the plasma which can be beneficial up to a certain point. The particular reason for this circumstance consists in the fact that external particles interact with plasma ones through many ways and the result will be a deceleration of the latter by isotropic emission, line radiation and ionization. At the very beginning it is quite good because the heat load at DPs would be highly decreased and the peaks

11

smoothed. Furthermore, on the wave of this phenomenon, it was invented the "impurity seeding" which aims to inject inside SOL plasma – in specific points like nearby the X-point – a gas, better if pure and inert (indeed the most used is Ar[7]). Nonetheless, if the process goes out of control we will have a considerable amount of non-fuel atoms inside the plasma, which can arrive to pollute the core itself and cause the shut down of the reactor (the isotropic emission would be too much [6] [7]).

Summarizing, the good divertor is an optimum between different features:

- Capability of handling heat loads of few tens of MW/m^2 in steady state operation and approximately few GW/m^2 in case of ELM or disruption events
- Minimizing the impurity in the plasma guaranteeing a good "effective ion charge parameter" Z_{eff} (<1.5)
- Low affinity with Tritium avoiding a big permeation and retention of the same, helped also by a purification system[8]

1.3 State of the art

1.3.1 ITER solid divertor

First of all, let us remember that ITER is still under construction and it is estimated to have the first plasma near the end of 2025. Even if this date has been postponed few times across the years[9], the project has been already decided and above all what is of our interest: the divertor. For this aim a solid-like configuration was chosen, considering that heat loads will be 3-4 times lower than DEMO ones. Ranging to values between 5 to 10 MW/m^2 in nominal conditions, a solid target design is easier and more feasible to be implemented here. In Figure 1.9 below we can see the mock-up of the PFU in the ITER divertor cassette and it is clear how simple is the technology: a tube covered by a monoblock of solid tungsten with water flowing in through in forced convection with swirl tapes which enhance the heat transfer. Even if other designs are still under investigation, the structure is always similar: the tungsten armour, the copper alloy (or his composite CuCrZr also called EUROFER) tube and the water inlet temperature (about 130 °C) are a constant.

---

[7] Previously N_2 was used but it ended up creating ammonium-like molecules with D and T and having other reactive contaminants in the vacuum chamber is not absolutely what we want, instead noble gases are far way better. For the same reason, quite recently divertor material was changed from Graphite to Tungsten because, even if in the second case we have more sputtering, in the first one C creates methane-like molecule and is highly reactive with the environment

[8] The extraction of Tritium is not only beneficial for target structural materials, preserving it from decay irradiation, but also for the fuel cycle, because in this way Tritium lost can be recollected and injected again inside the core

[9] According to first previsions we could have seen by far the first reactor operation
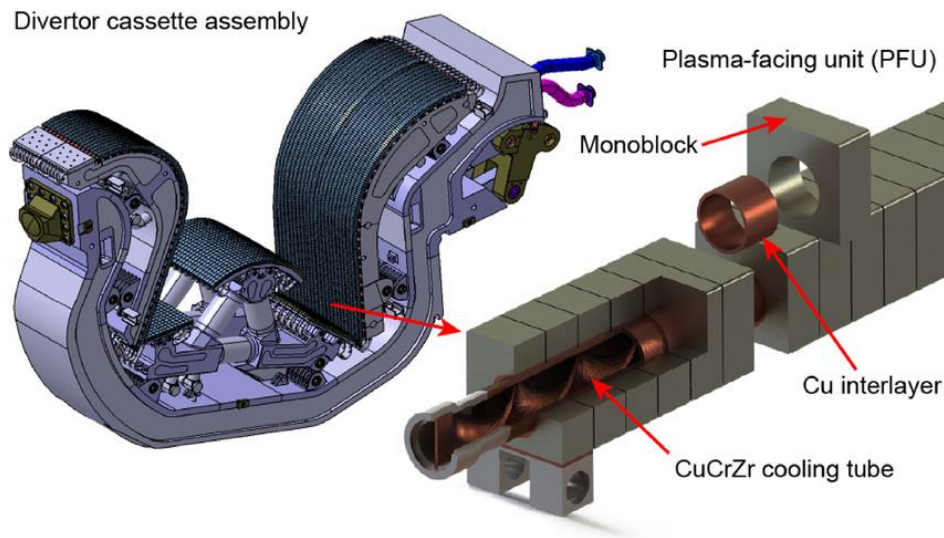
Figure 1.9 - ITER divertor cassette and outer vertical target PFU particular [9].

Due to different thermal expansions of the 2 materials, there can be an important thermal stress contribution which limits the lifetime of the component, expected to be around 2 full-power years. In addition to this, also the embrittlement by neutron irradiation and the previously introduced erosion (by sputtering process) are other important causes of degradation of the PFU, especially in case of thermal shocks, like disruption or similar events.

1.3.2 DEMO divertors

As we know, in DEMO reactor the power output is expected to be 3-4 times larger than ITER. Nevertheless, this is not the same for the wetted area which increases but not that much. So we need to improve the technology available, because the one discussed so far – solid type divertor – is no more sufficient. For this purpose, DTT is one of the fundamental test projects, in the roadmap for fusion energy, for the testing of different cutting edge divertors. Few designs were proposed so far but the most promising – and most important for this work – are the LMDs ones, which inherit the advantages of solid divertors and solve the typical disadvantages like:

- Tritium retention and neutron irradiation damages
  Tritium and neutrons actually penetrates the liquid structure of the LM, but it is then purified by an external system. Differently from ITER case, here the working material is liquid and, as we can assume, it is far way easier to purify it instead of a solid structure. Moreover, this discussion is applied to the reservoirs of LM; whereas the impurities that impact on the CPS structure are removed simply along with the extraction by evaporation of the LM.

13

- Resilience to shock transients

  If we have disruption or ELM events, the heat load – even if for very small periods of time – increases so much that LM experiences local huge vaporization. Nonetheless, from the reservoirs the same quantity of liquid is taken and no damages are expected

- Net erosion

  In the same way as the previous point, also for this case if any amount of LM is extracted by sputtering or other events (described in Figure 1.8), the same quantity is taken from the reservoirs

So, from the point of view of the divertor, the "self-healing" characteristic is a stunning advantage with respect to previous technologies, but an important drawback has been hidden so far. In fact, all the LM which is eroded by the PFU goes inside the plasma chamber and contaminates the plasma and, as already said, until a certain point it is quite good because it decreases the heat load dissipated by conduction and advection on the target surface, but if it goes up too much the reactor risks to shut down. Moreover, when SOL plasma impacts on the LM surface, MHD (magnetohydrodynamic) effects may rise precarious balance and instabilities on the LM surface, which can cause therefore a splashing effect with the result of a droplet ejection inside the vacuum chamber. In order to face the different disadvantages, few proposals have been discussed about, but for the aim of this work we will focus on two in particular: the vapour box and the CPS.

Vapour box

This design has been conceived by Nagayama [3], composed of 2 different chambers and a pool or CPS-like target where plasma arrives. As we can see from Figure 1.10, we have an Evaporation Chamber (EC) where LM is free to vaporize and a Differential Chamber (DC) which provides differential pumping between main chamber (MC, i.e. the plasma chamber) and EC. Here the LM tend to recondense along the surfaces, which is recollected and reinjected in the pool or the porous structure.

The entrances must be kept very small, but at the same time we have to consider that fluctuations of the magnetic field may change the strike point position and so the direction of the separatrix (the yellow line in Figure 1.10). In conclusion, we must avoid 2 different phenomena: the intersection between separatrix and apertures of the chambers – which can cause their local melting – and the excessive evaporation – resulting in metal vapour entering into MC. If the former contribution can be solved by enlarging the gaps, the latter can be handled exactly in the opposite way, forcing us to find the optimum size.
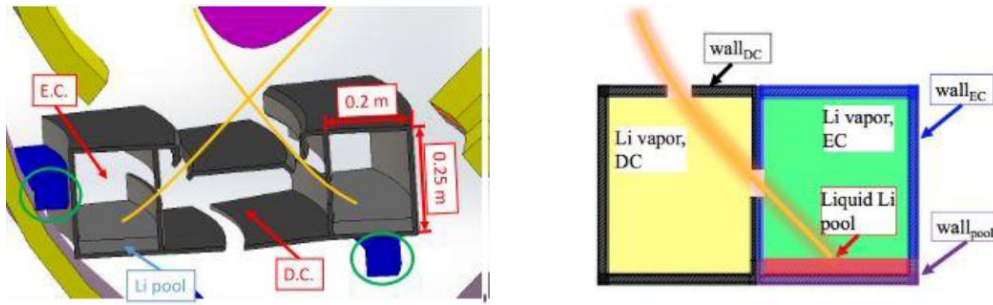
Figure 1.10 - vapor box divertor configuration [4].

In conclusion, two are the main advantages of this proposal: the "vapor shield" effect and the cooling by phase changing. The former stands for a cloud of the liquid metal just vaporized which interacts with the SOL plasma. The relative cool neutral gas interacts with high energetic plasma particles causing a deceleration of the second ones thanks to energy dissipation by line radiation, bremsstrahlung and ionization. Instead, regarding the latter phenomenon, for this particular design, due to the big evaporated mass flow rates, the cooling contribution from vaporization – $\Gamma_{eva} \cdot \Delta H_{eva}$ @~0.6 bar – is quite relevant. For the next paragraph, where no EC and DC is included in the model, this will be not the case – so lower values of evaporated mass flow rates possible – and at the end of the thesis it will be supported by calculations.

CPS

The Capillary Porous Structure is the fundamental divertor technology studied in this work and here we will just briefly introduce three different proposals (Figure 1.11), trying to compare the most relevant features (see Table 1-1). The main idea is to place along the target a porous material, wetted by the LM, which is able to create a capillary force that balances all the other ones, like friction losses, gravity and JxB force

$$P_C >= \Delta p_L + \Delta p_G + \Delta p_F + \Delta p_{MHD}$$

Equation 1.11 – CPS pressure balance inequality.

Δp_L: hydrayulic pressure drop

Δp_G: hydrostatic pressure drop (linked to gravitational force)

Δp_L: pressure drop at evaporation surface due to liquid-vapour phase change

Δp_{MHD}: pressure drop due to MHD effect

P_C: capillary pressure

$$P_C = 2 \cdot \sigma \cdot \cos(\theta) \, / \, r_{eff}$$

Equation 1.12 – P_C definition.

where σ is the surface tension of the LM, θ the wetting angle and r_{eff} the effective radius (linked to the width of the wire mesh)
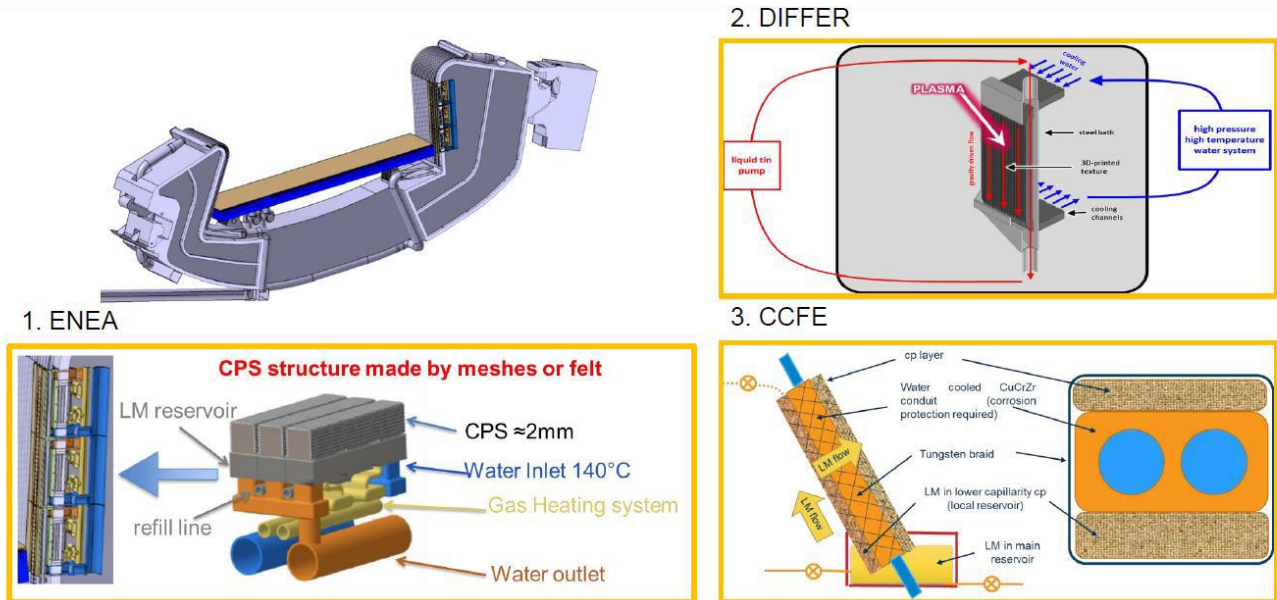
Figure 1.11 - CPS different proposals [5].

1) ENEA concept

   This one is the main argument of the thesis and for a detailed description it is strongly encouraged the further reading of this work, especially chapter 2. Here the cooling of the structure and the heating of the LM (in order to keep it liquid) are decoupled and worked out respectively by a water and a gas circuit. Moreover, the LM has a crossflow with respect to water direction. The former goes in the toroidal direction, instead the latter along the poloidal one, from up to bottom.

2) DIFFER concept

   As for the previous case, the water goes from the upper part to the bottom one[10], but here also the LM flows in the same way, obviously between plasma and solid structure. We can conclude as well that the LM goes in the same direction as gravity, so the capillary force needed may be slightly lower rather than the other cases

3) CCFE concept

   The particularities of this mock-up are 2. First the LM flows from below – where we have the reservoir of a LM pool – to the top, so a strong capillary force is needed to overcome the

---

[10] We have to remember that these configurations are all Vertical Target concepts

gravitational force (which is about on the opposite direction). Secondly the water here acts both as cooler and heater for the LM and for this reason its inlet must be higher than the melting temperature of the LM (e.g. $T > \approx 232$ °C for Sn)

As previously said the CPS target can be coupled with the vapour box concept, exploiting the big advantages of cooling by phase changing and "vapor shield" effect, which on the other hand in the pure CPS case are not that relevant.

Design choice	ENEA	DIFFER	CCFE	
Liquid metal	Sn	Sn	Sn	
CPS type (2 mm thick)	Mesh/felt	3D printed	Braid	Different technology options
Sn resupply	Capillary reservoir	Capillary/flow	Capillary reservoir	
CPS max pore size (μm)	50	50	20	
Water temperature*	120 °C	180 °C	250 °C	
Water pressure (bar)*	50	150	150	
Water flow speed (m s^{-1})*	12	14	16	Different cooling assumptions
Sn liquid?	>5 MW m^{-2}	>5 MW m^{-2}	Always	
ICHF (MW m^{-2})*	32 (W/Cu)	56	25	
MCHF @ 20 MW m^{-2}	1.6	2.8	1.25	
Sn heating	Gas in PFC	Gas in pipeline	Water	

*c.f. DEMO ITER-like PFC: 120 °C, 50 bar 16 m s^{-1} ICHF=33 MW m^{-2} MCHF=1.65

Table 1-1 - comparison through different technologies of LMDs in the CPS design [5].

1.4 Aim of the thesis

This thesis is the result of the joint venture between Politecnico di Milano and Politecnico di Torino. In particular, the NemoLab group – belonging to the energy department "Galileo Ferraris" of Politecnico di Torino – entrusted me with the task of modelling the LMD CPS mock-up proposed by ENEA using the opensource software openModelica and Dymola. This is the development of a laboratory session inside the Nuclear Fusion Reactor Engineering course which took my curiosity and inspired me to study more deeply this technology and contribute to the research activities.

From the practical point of view, it was implemented a code with different levels of approximation, which is able to help us understand the thermal behaviour of the components – in steady state and transient conditions –, let us see if there are local melting or vaporization points respectively in the structure materials or in the LM and controlling the water flow and its characteristics.

2.   LMD ENEA Design

As introduced in chapter 1, few CPS mock-ups have been studied so far for DEMO divertor application, but here we will focus on one in particular, the one proposed by ENEA. This concept is characterized by the particularity of having both in-vessel and ex-vessel components. For an organized description of the model, we will go from the in-vessel components, starting from the cassette and going more deeply inside up to the analysis of the cross section of the single unit. Then we will see which are and why we need ex-vessel parts.

2.1 Geometry

First of all, we have to know that, along the toroidal direction, 54 cassettes are placed inside the VV, where SOL plasma is supposed to impinge[11]. For each cassette we have 24 modules, disposed in 8 packs of 3, each one in parallel to the next one as shown in Figure 2.2.b.
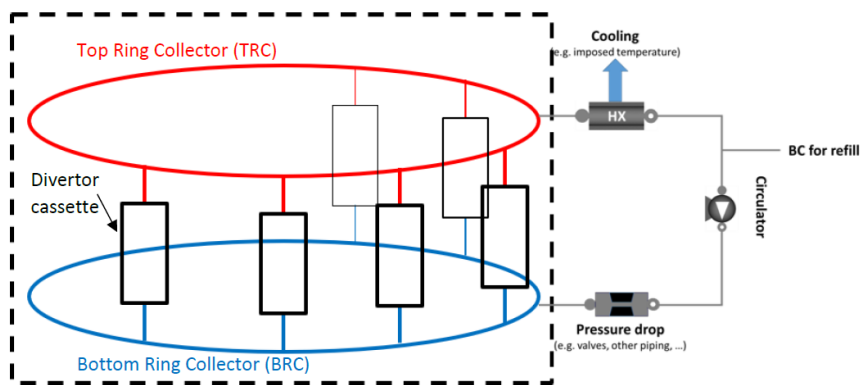
Figure 2.1 - Simplified diagram of the divertor cassette configuration with the TRC and BRC and the coupled cycle which aims to cool, refill, purify the LM and also to overcome its pressure drops. Beware that here only 6 of the total 54 cassettes are represented inside the dashed rectangle at the left .

---

[11] For now on, we consider the single null configuration with the additional coil below the machine, used to change the magnetic field lines and to create the X-point and the private plasma
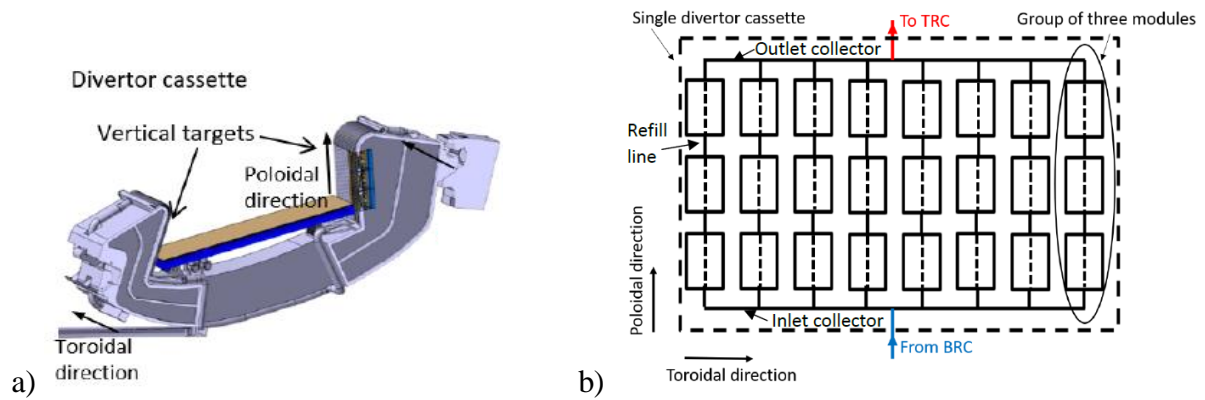
Figure 2.2 – At the left (a), overview of a real divertor cassette. At the right (b), scheme of one VT with the 8 x 3 modules typical arrangement.

In essence, we have a group of three modules, which is repeated 8 times along the toroidal direction inside the VT. Looking Figure 2.2.b and Figure 2.4.a, we can see how a group of three modules is modelled. The LM comes from the lower part, where the BRC is placed. Then, before going straight to the TRC, a part of its mass flow rate is splitted through each unit. Afterwards, the remaining LM is cooled with a heat exchanger, purified with a particular system, pumped because there can be localized and delocalized pressure losses through all the circuit and in the end there is also a source of LM which aims to refill the circuit (Figure 2.3, grey circuit, and Figure 2.1). In fact, the LM in the porous structure – so inside the module – tend to be eroded and to evaporate, so, in order to close the loop, we have to recollect the LM inside the VV which is in gas form and then recondensed along the surfaces of the FW.
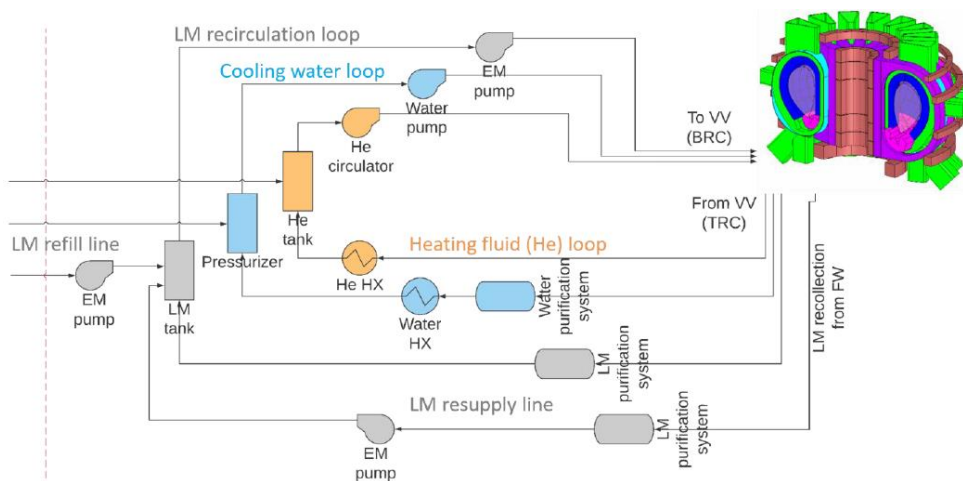
Figure 2.3 – Simplified scheme of the ex-vessel LMD circuits in DEMO reactor design (CAD taken from [19]). The grey loop corresponds to the LM one, the blue one to the water and in the end the ochre one to He gas

Now, let us have a better look on the module itself. Starting from the group of three modules of Figure 2.2.b, in the next picture we have a more precise scheme which explains us better how it is done one module and each unit. In Figure 2.4.a we can see that from the BRC the LM flows till the TRC but a

part of the mass flow rate is extracted and given to the units – the small rectangles. Where does the LM go after being "sucked" by the CPS structure inside the unit? It is sputtered or evaporates, so it exits the CPS and enters the VC (follow m_{net} flow in Figure 2.4.b).
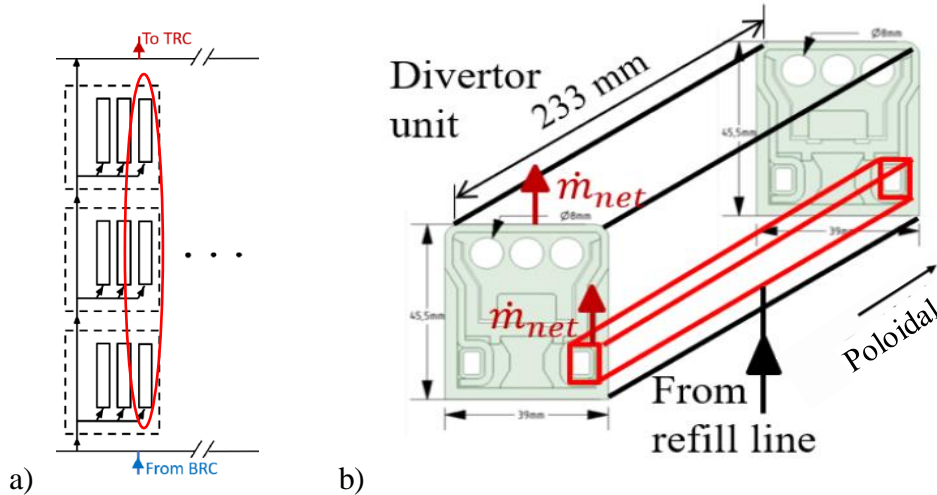
Figure 2.4 – At the right (b) the design of one single unit, represented at the left (a) schematically with a rectangle. One module is encircled by the red oval in the left diagram, consisting in 3 elementary units in series

The dashed rectangle in Figure 2.4.a represents the actual physical most elementary units: a series of 3 of the latter define a group of three modules and so on up to the whole divertor. We can see how it looks like in reality in the bottom centre part of Figure 2.5. We will see in the next paragraphs that the LM is cooled by the water "in-situ", right where the plasma arrives, but after the water is heated, it is cooled again outside the divertor unit where power is extracted to generate electric power (Figure 2.3, blue circuit). The gas heating system, instead, have the function to heat up the plasma maintaining it liquid and guaranteeing its flow. As the water circuit, the gas enters from the beginning of the first unit of the module and exits from the ending of the last unit and from here is recollected, pumped up in order to face the pressure losses, brought to its nominal temperature of 350 °C and reinjected inside the modules (Figure 2.3, ochre circuit).
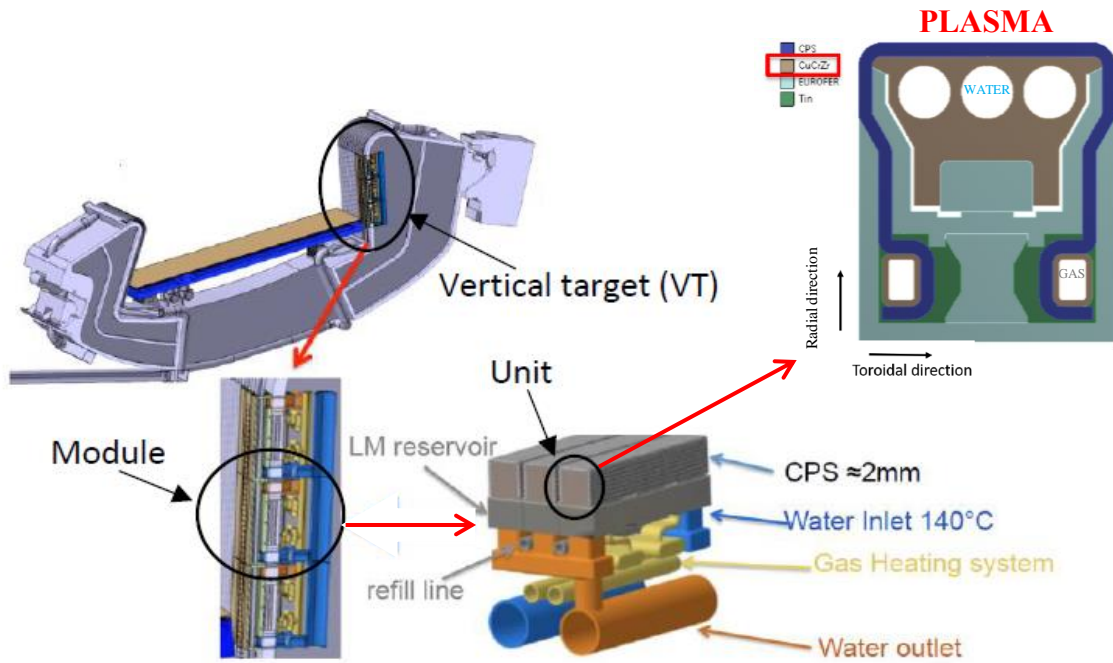
Figure 2.5 - Overview of how the divertor really looks, from the cassette on the top left, to the group of 3 modules on the bottom left, to the group of 3 units on the right, till the cross section of the CPS at the top right [10].

Now, let us focus on the cross section of the single unit (Figure 2.6), which thermal behaviour of each layer is divided through Figure 2.7, Figure 2.8 and Figure 2.9, instead the overall one is represented in Figure 2.10.
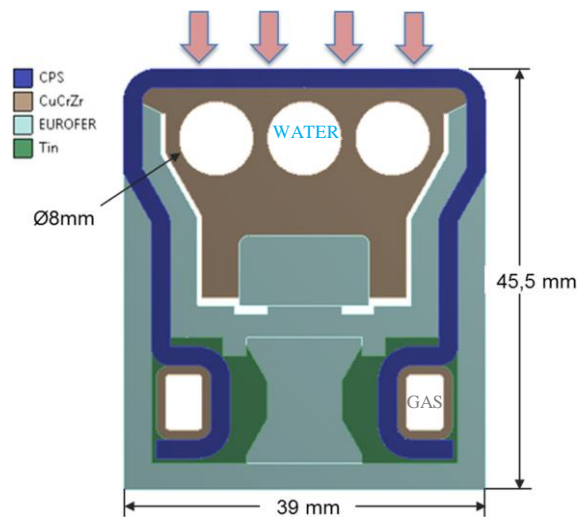
Figure 2.6 - Zoom of the cross section of the CPS divertor proposed by ENEA [10]. In Figure 2.10 we can see the thermal behaviour of the whole cross section in function of different heat loads from plasma

2.1.1 EUROFER

The structural body is made of EUROFER, in light blue in the picture above and in Figure 2.7. This is a special material because it has very good thermal and irradiation properties: very low activation

probability, a considerable resilience to thermal shocks and a good temperature window. Its operative temperature ranges should be between 300°C and 550 °C to avoid further degradation besides neutron irradiation damages. Contact resistance with CuCrZr is neglected.

This steel structure is machined from a single body and it is attached to the heat sink on one side and to the reservoirs on the other one. It is mounted as last and has the function of containing the LM and also anchoring the CPS. The reservoirs are fixed to the EUROFER body with screw joints but, for safety reason, this cannot be employed for a real FPP application. For the purpose of the research activities of Roccella ([10]) these were considered, whereas for DEMO application they can be easily replaced by weldings, without big modifications on the overall thermal behaviour.
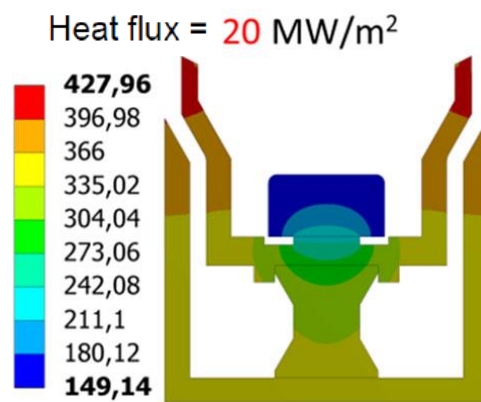
Figure 2.7 - Analysis of the thermal behaviour of the EUROFER structure considering a plasma heat flux input of 20 MW/m² (simulations performed with Ansys software) [10].

2.1.2   CuCrZr

The heat sink structure material is CuCrZr because it is usually the reference one for divertor application. By the way, many research activities have been conducted about a composite of W/Cu (e.g. with 70% wt. of W) but it is quite critical to handle with huge contents of Cu because of its high activation and material degradation [11,12].

The heat sink structure is fixed to the steel one with pins inserted in horizontal slot. In this way, knowing that they have different thermal expansion coefficients, they can slide between the two parts, reducing the thermo-mechanical stresses. For our analysis this material is the fundamental one because it is located just between the porous structure, which faces the plasma, and the cooling channels. For this reason, it is important to take in consideration its properties starting from thermal

conductivity – crucial to define the global thermal resistance/conductance of the solid – up to density and specific heat capacity[12].

The heat sink is fixed at first to the structure, then the CPS around it and only in the end the steel box is placed. In Figure 2.8 we can see the CuCrZr zoom with respect to the total cross section of Figure 2.6.
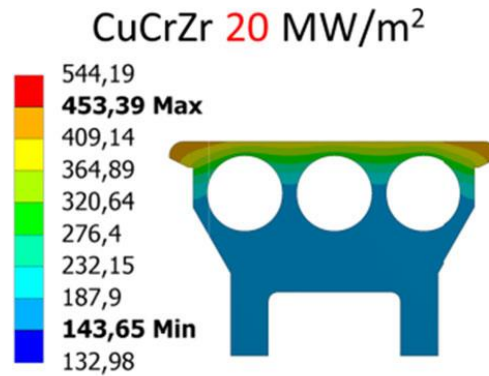
Figure 2.8 - Analysis of the thermal behaviour of the CuCrZr structure considering a plasma heat flux input of 20 MW/m² (simulations performed with Ansys software) [10]

2.1.3 CPS

The Capillary Porous Structure is composed by an armour – made by a tungsten matrix with the arrangement shown in Figure 2.9 – wetted by the LM which flows through it. Even if it can be calculated the percentage of LM in time and calculate the overall thermal properties of the CPS, for the sake of simplicity, the mixture is considered 50% wt. of W and 50% wt. of LM [15][13].

Unlike solid targets, the CPS thickness can be really small due to the fact that the refilling times are small enough to ensure the replenishment and so the self-healing feature of this design. Indeed, the W mesh sucks the LM from reservoirs passively, just thanks to the capillary forces and without an external pumping action. As we can see from Equation 1.11, P_C must be at least equal to all the losses and other forces acting on the LM. In this way, after erosion and evaporation events, the mass flow rate lost can be replaced just extracting it from the reservoirs. The flow follows the m_{net} arrows from Figure 2.4.b.

The pore diameter is far smaller than the CPS thickness (tens of micrometres against 2 mm) and its size can change. According to different values of porosity[14] we can modify CPS performance, without

---

[12] We can find semi-empirical correlations for the temperature window of our interest in the ITER Material Properties Handbook [13]

[13] For the semi-empirical correlations for W, Sn and Li conductivity look up to respectively [14], [17] and [16]

[14] Maximum pore diameter: 50 micrometres

affecting that much the thermal behaviour. What is important for the latter is the fact of modelling the CPS as a homogeneous structure with same W and Sn volumes and CPS total volume. Guaranteing these conditions, an affordable thermal analysis can be done with good approximation.
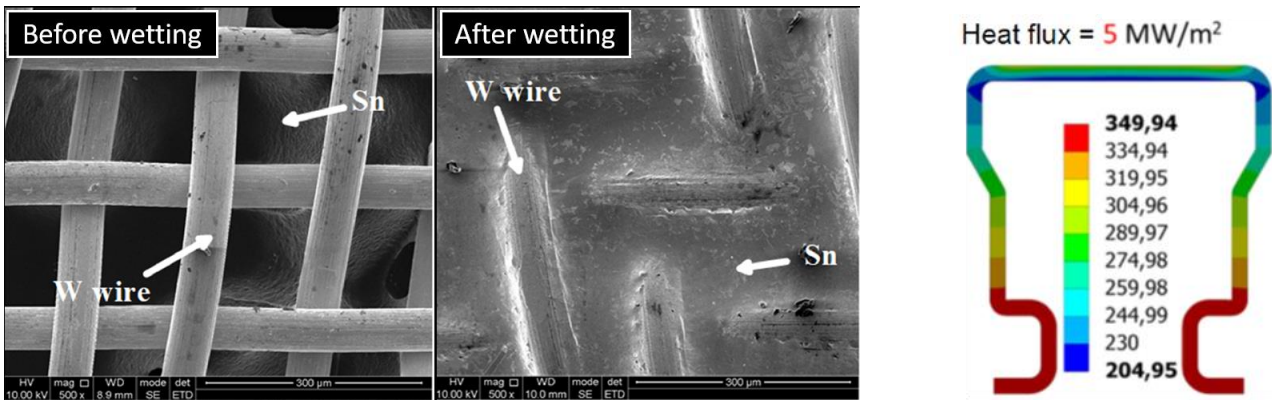
Figure 2.9 - At the left, a comparison between the tungsten matrix with and without LM [18]. At the right, analysis of the thermal behaviour of the CPS structure (wetted by Sn) considering a plasma heat flux input of 5 MW/m² (simulations performed with Ansys software) [10]. In this case the input is no more 20 MW/m² and so the maximum temperature reached by the LM is the one of the gas circuit (350 °C), in the proximity of it. Meanwhile, at the top, water power extraction is so high that it cools down the LM till the freezing point (232 °C for Sn).
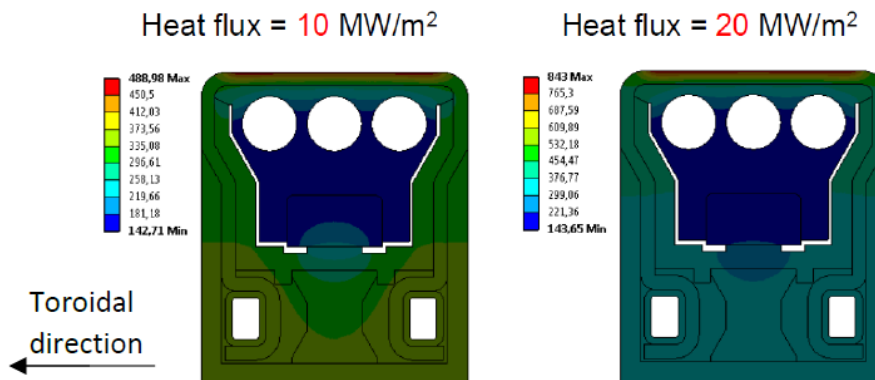
Figure 2.10 - Analysis of the thermal behaviour of the whole cross section (LM is Sn here) considering different plasma heat flux inputs (simulations performed with Ansys software) [10].

2.2 Water and gas circuits

The water and gas loops have both in-vessel and ex-vessel components. From the below representation of the group of three modules we can have an idea of how the former are designed. For example, let us look to the water system. A series of blue pipelines have the aim to distribute fresh water from the upper part of the module[15]. Afterwards, when water reaches the end of the module, it goes to the orange sector. Then all water stream coming from cassettes is recollected in an ex-vessel circuit where it is cooled back to its inlet temperature value and pumped up to overcome all

---

[15] Remember that we are talking about a VT, so the Figure 2.11 is not as it looks, but it is rotated of 90°

the loop pressure losses. In the end, flow is divided again for each cassette – to the blue water inlet pipes previously described – and the loop is closed.

On the other hand, very similar consideration can be done for the gas heating circuit – even if in Figure 2.11 we do not have a clear diversification between inlet and outlet. The only big differences are two: a compressor rather than a pump and the absence of a purification system. The former is quite obvious, instead the latter can be justified by the noble gas characteristic of He, which is very weakly reactive and subject to small activation due to neutron irradiation.
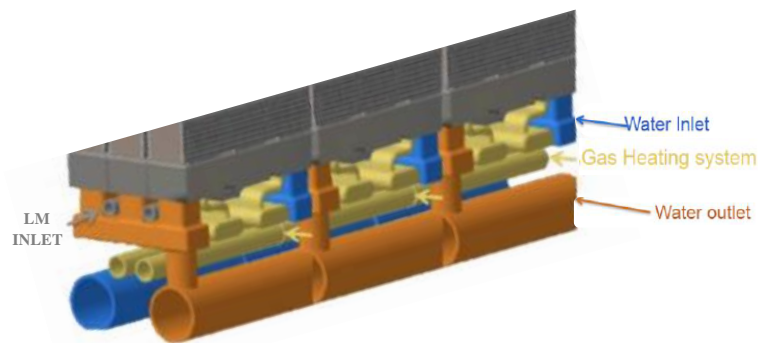
Figure 2.11 – Overview of the in-vessel components of water, gas and LM loops. The grey channels stand for the LM refill line, whereas gas arrives to the unit thanks to the pipes coloured in yellow.

In Figure 2.12 we see highlighted the water section of the single divertor unit. Three channels are designed in order to cool down the whole structure and it is not by chance that they are right close to the side where plasma impinges. In this way cooling of the most critical point of LM and the CuCrZr structure is far way more effective and the probabilities of CPS dryout and solid melting decrease. Moreover, as briefly introduced before, water enters from the upper part, flows through the first divertor unit, it goes inside the next unit and so on till the end of the module. We do not have three different thermally and hydraulically decoupled circuits for each unit, but only one. Obviously we would prefer the former choice, but in absence of detailed information about other components in the neighbourhoods of the divertor, this layout was chosen, keeping in mind that a limit amount of penetration is imposed. It is not excluded that with future research achievements there will be the possibility to go back to the other more efficient option.
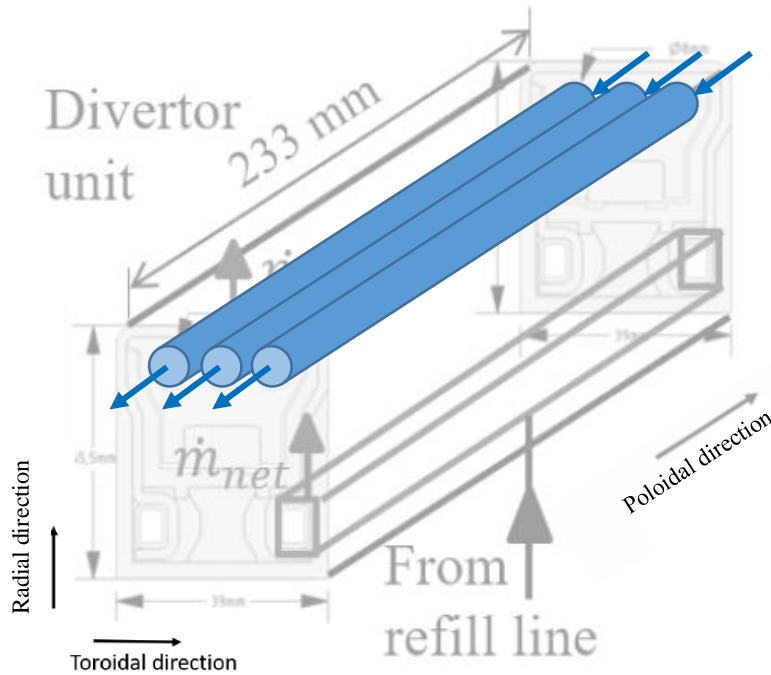
Figure 2.12 – Zoom of the channels where water flows in a single divertor unit, highlighting the flow direction with the blue arrows.

In Table 2-1 we can see the nominal values according to the design choices

Parameters	Design value
Water pressure (bar)	50
Channel diameter (mm)	8
Channel cross section (mm^2)	~50.3
Water velocity (m/s)	~12
Inlet temperature (°C)	130
Pressure drop (bar/m)	3.5
Critical Heat Flux (MW/m^2)	42.3
Peaking factor	1.75
CHF Margin (ICHF/20 MW/m^2)	1.21

Table 2-1 – Water circuit. Beware that these are the design values according to Roccella's work [10]. In Chapter 3 we will precise what will be the design choices and why the modeler chose them

The gas circuit, on the other hand, when it enters the modules it flows in the bottom part as indicated in Figure 2.6, where there are the reservoirs. In this way the LM in the reservoirs is kept liquid and it can be refilled in the CPS structure. Besides, looking the 3D profile of the unit in Figure 2.13, we can see that the flow is consistent with the water one, going from up to bottom inside the VT.
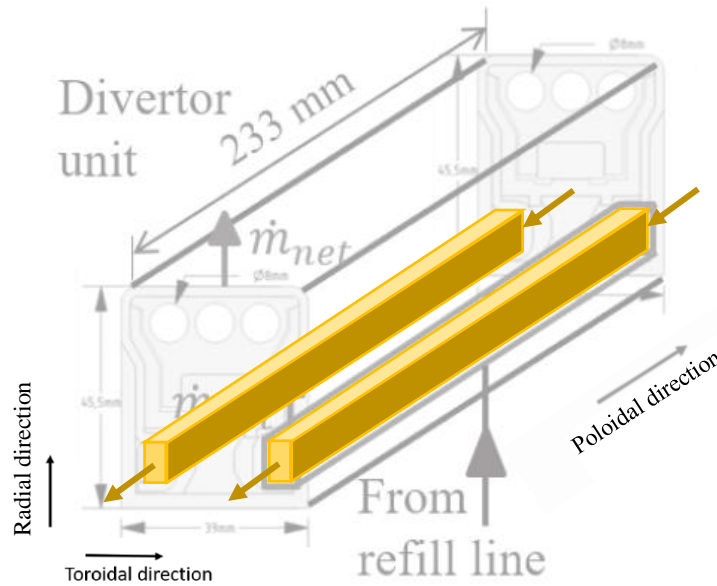
Figure 2.13 – Zoom of a single divertor unit, where they are highlighted in yellow the channels where gas flows, while the arrows stand for the direction of He (from upper module part, to the lower side).

Below, the values chosen for the fundamental parameters of the circuit

Parameters	Design value
Gas pressure (bar)	50
Channel cross section (mm^2)	~12
Gas velocity (m/s)	~ 0.13
Inlet temperature (°C)	350

Table 2-2 – Gas circuit. Consider that in [10] there are no specification about these values apart from the last one. For this reason, they were chosen the other ones as an attempt and put in input inside our model.

2.3 LM aim

After having discussed accurately about the divertor unit cross section with all the characteristics of the structural materials, the water and the gas channels, now let us focus on the LM. Its aim is not to extract heat even if it is a good conductor, but to protect the solid structure. In fact, differently from the solid ITER-like divertor, here we have a layer of LM which, in case of disruption or ELM, it evaporates and is promptly refilled (with a certain delay). In this way we avoid huge thermal shocks and the risk of melting of CuCrZr and EUROFER. Remember also that the 2 solid structures have different thermal properties and so different thermal dilatation coefficients, with the effect of non-zero stress gradients at the interface in case of excessive heating up of the whole body. Unfortunately, this phenomenon may result in a degradation of the unit if not kept inside certain safety values, with

the need of substituting the component[16]. Furthermore, keep in mind that LM evaporates during normal operation and especially during high-thermal-load transients. Consequently the latent heat of vaporization is extracted and used to cool down the metal still in liquid form. We will see from calculations if this contribution is negligible or not, both for Li and Sn.

2.4 LM choice

The choice of the LM is still under investigation because no perfect metal exists on planet earth. For this reason, few with good trade-offs have been studied and the most relevant ones are two: Li and Sn. But before jumping on the discussion about these two options let us start from the requirements needed for CPS design.

- Low atomic number
- Strong surface tension, to restrain splash damages (Equation 1.11)
- Low vapor pressure, to limit excessive evaporation outflow which leads to plasma contamination and dilution
- Material compatibility, linked to corrosion, wettability and chemical stability with the other materials it comes in touch with
- Low activation by neutron flux
- High latent heat, in order to extract a part of heat with the phase changing process
- Low T retention

Quantity	Unit	Meaning	Li	Sn
Z		Charge	3	50
A	amu	Mass number	6.94	118.7
ρ	kg/m^3	Density at melting temperature	512	6990
T_m	°C	Melting temperature	180.5	231.9
T_b	°C	Boiling temperature	1347	2270
ΔH_{vap}	·10^{6}kJ/kg	Latent heat of evaporation	1.02	35.15
k	W/(m K)	Thermal conductivity at melting temperature	45	30
E_{iz}	eV	First ionization potential	5.39	7.34

Table 2-3 – Li and Sn properties[17].

---

[16] Beware that for maintenance operation we must shut down the reactor for long periods of time, so quite relevant economic costs. For this reason we choose to create cutting-edge components rather than cheaper but more fragile ones

[17] Beware that the latent heat of evaporation values refer to ambient pressures, even if inside VC we can have lower values of pressure. Nevertheless, it can be demonstrated that for such low pressure values the latent heat remains with good approximation constant, so it can be assumed correctly that these data are good also for our application

2.4.1 Li

Lithium was the first LM to be kept into consideration. It has a very low atomic number $(Z = 5)$[18], so even if its evaporation is quite relevant for temperatures bigger than 600 K, Li vapour can enter even core plasma but without considerable effects like for tungsten or tin case. Moreover, Li has good wettability which helps capillary forces to be predominant, but also a quite reactive compatibility with water. For this reason between water channels and flowing Li it is needed a protective interlayer (of copper alloy like the previously introduced CuCrZr). Then we can classify two different operative conditions which can be implemented for Li: fast and slow flow concepts. The former is related to Li flowing fast through the target, keeping temperatures quite low and reaching the low-recycling regime. In this condition it has been seen that Li can be beneficial for plasma which is able to reach and maintain H-mode (High performance mode) operation: it broadens the electron temperature profile, improving consequently the energy confinement time; it acts as a passive pump for deuterium gas; it reduces the quantity of residual gases (typical of vacuum systems). In fact, if we look at Figure 1.8 at the "Fuel Recycling" process we can have an idea of how it works. In a nutshell, when D/T ions impinge on the target, they recombine and they accumulate on the surface till a certain saturation point. As soon as it is reached, neutrals start to go back to the plasma and so fuel particles are recycled. This process is quite important if we have a solid or a quasi-static liquid wall (high-recycling regime), whereas negligible if we have a fast flow (low-recycling regime). In particular, for what concerns the former condition, the slow flow concept must be cited, which is associated to our CPS concept where LM stream is quasi-static. In this case, unfortunately, we lose the benefits of the low-recycling regime due to higher temperatures involved[19]. In the end, let us remember that T retention can be considered negligible for temperatures higher than 650 K, due to LiD (and LiT) dissociation. Knowing that the main advantage of Li was the low-recycling feature, if we are interested in a CPS application, other metals should be investigated.

2.4.2 Sn

Tin has a quite big atomic number $(Z = 50)$, but a very low evaporation rate (up to approximately 1350 °C (Figure 2.14) so it fits well the CPS application without the help of an external box (no relevant vapor are predicted to go in the vacuum chamber). Moreover, due to many ionization levels, Sn vapor helps to radiate the power coming from SOL plasma (Figure 2.15), flattening the conductive

---

[18] In this way, even if Li arrives to core plasma, its contribution to effective charge should be low, so not a big problem

[19] The threshold can be considered around 450 °C

and advective heat flux contribution which impinges on the divertor [21]. Finally, Sn has a very high latent heat and it is much less chemically aggressive than Li, so the corrosion compatibility is far way better for Sn [22].
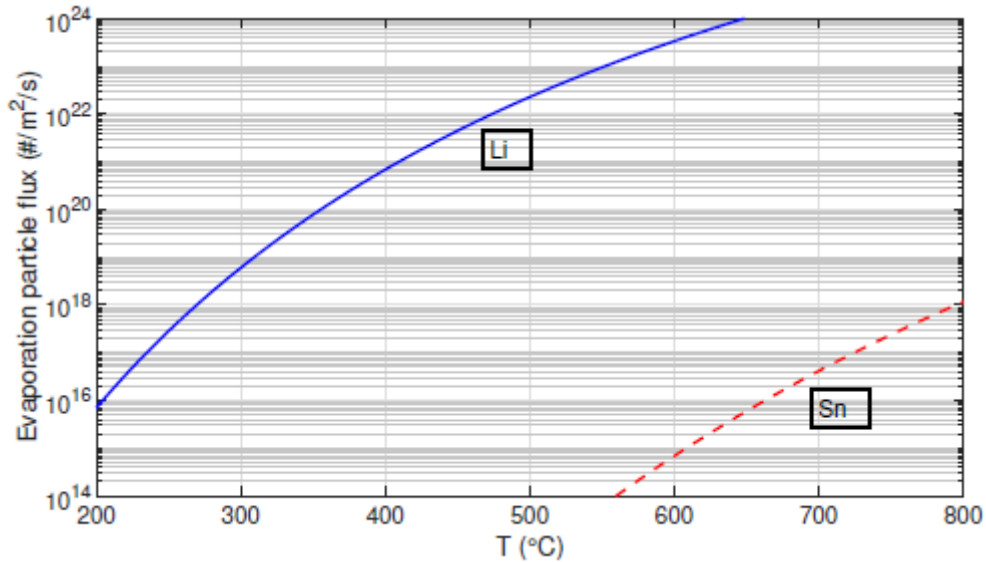
Figure 2.14 – Curve about the particles evaporated per unit area and time, in function of the temperature for Li and Sn cases. It is undeniable the advantage of Sn, even if atomic weight is bigger than Li.
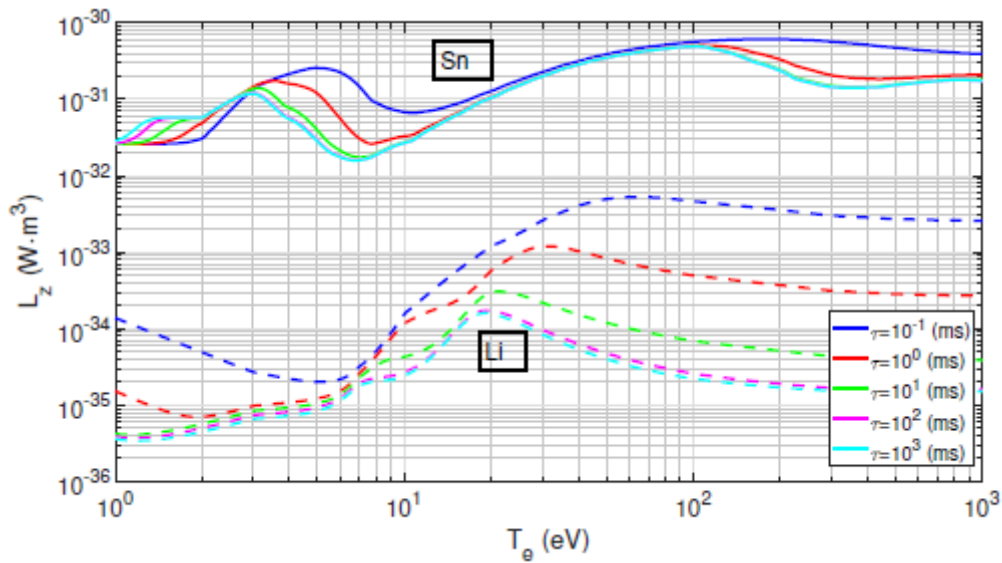
Figure 2.15 – Behaviour of the power density L_z emitted by radiation in function of the electron temperature T_e in case of plasma interacting with Li [23] and Sn vapor [24]. We can see different curves for both metals because different values of τ (residence time) have been considered. We can conclude that $L_{z,Li}$ is strongly dependent on τ, whereas for Sn it is less sensitive and, on the other hand, it is always bigger.

3 SOLPS Data

3.1 Background research activities

Before going right on the model description, let us start from a detailed discussion about the input data, courtesy of G.F. Nallo and his research activities in the field [25].

First of all, let us begin from the two prototypes used for the experiments. One's aim was the analysis of the configuration of CPS wetted by Li, the other one for the Sn case. The former is represented in Figure 3.1. Very briefly in this layout a W very thin (2 mm) plate faces the plasma and protects a CPS layer, which is cooled by an impinging water gas atomized spray. Water temperature is designed to be at 20°C, which allows to have heat transfer coefficients of about $70 - 100$ kW/(m^2·K). This configuration aims to maintain the PFS at quite low temperature values and, as a consequence, low evaporation particle fluxes are implied. The maximum tolerable heat flux is claimed to be ≈ 20 MW/m^2. The latter instead, uses the CPS mock-up by ENEA already discussed in chapter 2. For this configuration water flows at 120 °C inside the channels and the resultant heat transfer coefficients can be seen in Roccella's work [10]. The handling capabilities of the plasma heat flux of this design go up to about 40 MW/m^2, limited by the CHF of the coolant. Beware that CuCrZr upper temperature limit can be reached even for lower plasma heat fluxes. Consequently, new heat sink materials (like W/Cu composite) will be under investigation in the next future, even though for our treatment we will stick to this one.
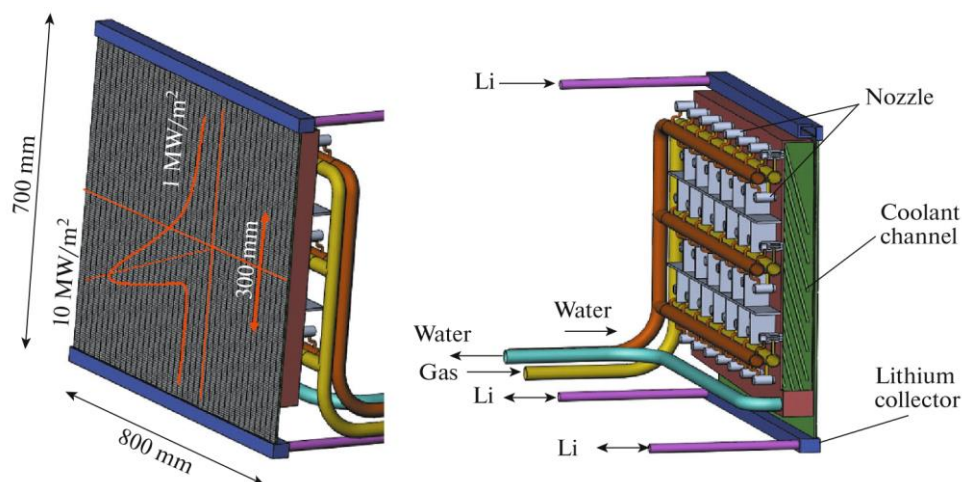
Figure 3.1 – Overview of the model proposed by Vertkov [26] for the experiment activities about the Li-wetted CPS design [26].

3.1.1 Simulation setup

Calculations have been performed using different assumptions. First of all no transients have been analysed, but only steady state conditions. We will see later on that this will affect partially the simulations, especially during transient phases. The domain considered for both plasma and neutral species is the one in Figure 3.2, keeping in mind that toroidal symmetry is assumed here. In this model SOL plasma and pedestal[20] can be described as multi-fluid (using Braginskii-like equations), whereas neutrals behave for assumption like a single fluid. Target erosion is computed self-consistently with the plasma and neutrals distribution.
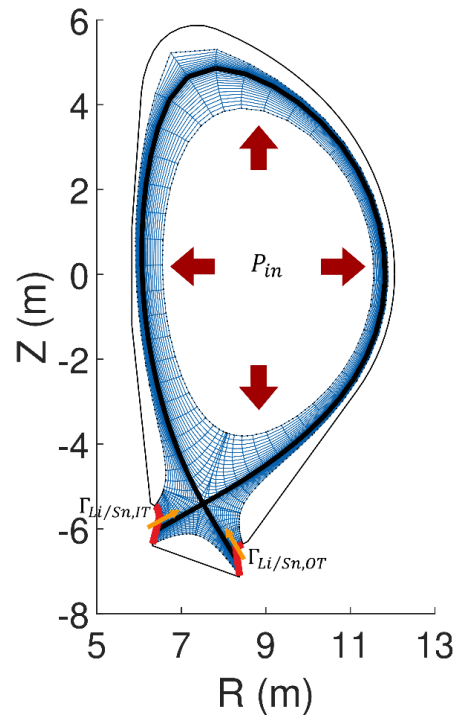
Figure 3.2 – Sketch of the EU DEMO poloidal cross section. Here we can see the power input coming from the core indicated by the red arrows, the separatrix in black and the SOL (outside separatrix) and pedestal (inside separatrix) regions inside the blue mesh where fluid equations have been solved. In the end we have also the divertor targets in red and the $\Gamma_{Li/Sn,IT}$ and $\Gamma_{Li/Sn,OT}$ erosion mass flow rates described by the orange arrows [25].

In order to obtain the output values, a self-consistent approach have been achieved by coupling two models:

1) A 2D SOL plasma model with SOLPS ITER software, which aims to calculate everything that concerns about the blue mesh region (Figure 3.2): from SOL plasma temperature and density profiles to advection and conduction heat load on divertor targets; from the impurity

---

[20] Pedestal is the interface region between SOL and core plasma, inside the separatrix. Indeed, in Figure 3.2 we can see three zones inside the blue mesh: private plasma at the bottom, SOL outside the separatrix and then pedestal the remaining one, between the separatrix and the internal white oval-shaped region. In this part we have closed magnetic field lines like in the core, although all the fundamental properties like p, T or n_e drop down, from core plasma values to the very low ones of SOL.

flux to core plasma to the radiated power in SOL – direct outcome of the plasma-vapor interactions

2) A thermal model, which computes the temperatures on the PFS and in addition the LM evaporation and sputtering rates

3.1.2 Target erosion model

The gross erosion mass flow rate for Li is:

$$
\Gamma_{Li,gross} = \Gamma_{D+} f(\beta) \left[\frac{Y_{phys,D+\to Li}(E_{D+})}{3} + \frac{Y_{ad}}{1 + A\exp\left(\frac{E_{eff}}{kT}\right)} \right]
$$

$$
+ \Gamma_{Li+} f(\beta) \left[\frac{Y_{phys,Li+\to Li}(E_{Li+})}{3} + \frac{Y_{ad}}{1 + A\exp\left(\frac{E_{eff}}{kT}\right)} \right] + \dots
$$

$$
+ \Gamma_{Ar+} f(\beta) \left[\frac{Y_{phys,Ar+\to Li}(E_{Ar+})}{3} + \frac{Y_{ad}}{1 + A\exp\left(\frac{E_{eff}}{kT}\right)} \right] + \dots
$$

$$
+ \eta \cdot \frac{p_v(T,\beta) - p_a}{\sqrt{2\pi m_{Li} kT}}
$$

Equation 3.1 – Gross erosion mass flow rate for Li [25].

- Γ : particle fluxes of impinging ions
- E : energy of impinging ions
- "…" : it stands for the ionized species (Li^{2+}, Ar^{2+}, …) not explicitly reported but still considered for the calculus
- B : D concentration at the surface. Due to big surface temperatures, D retention in Li can be assumed negligible as well as β contribution => f(β) = 1
- Y_{phys} : physical sputtering yield
- Second term in square parenthesis : thermal sputtering
- Last term : evaporation contribution calculated with Hertz-Knudsen equation [27]
- k : Boltzmann constant
- T : target surface temperature
- $p_v(T)$: vapor pressure
- eta : "sticking coefficient" for evaporation which in first approximation is assumed to be 1.66 [28]
- m_{Li} : mass of a single Lithium atom

Instead, the Sn gross erosion mass flow rate is:

$$
\begin{aligned}
\Gamma_{Sn,gross} = &\ \Gamma_{D+} f\left(\beta\right) \left[Y_{phys,D+\to Sn}\left(E_{D+}\right)\right] \\
&+ \Gamma_{Sn+} f\left(\beta\right) \left[Y_{phys,Sn+\to Sn}\left(E_{Sn+}\right)\right] + \ldots \\
&+ \Gamma_{Ar+} f\left(\beta\right) \left[Y_{phys,Ar+\to Sn}\left(E_{Ar+}\right)\right] + \ldots \\
&+ \eta \cdot \frac{p_v\left(T,\beta\right) - p_a}{\sqrt{2\pi m_{Sn}kT}}
\end{aligned}
$$

Equation 3.2 – Gross erosion mass flow rate for Sn.

Where we have similar considerations as above, taking into account that different empirical formulas were used to calculate some quantities, like vapor pressure (in HK relation) and Y_{phys}. At last we have to say that for both Li and Sn p_a is assumed here ≈ 0, even if other authors have proposed other values (e.g., $\approx p_{plasma}$ in case of vapor shielding regime [30]).

3.1.3 Prompt redeposition coefficient

Prompt redeposition refers to those particles that are ionized within the magnetic sheath, i.e. approximately within one gyro radius from the PFS. In that region, due to the strong electric fields, they are immediately redeposited in the target, without contributing to well-known phenomena like line radiation. For the point of view of vapor shielding, it is as if such particles were not emitted at all.

For the calculus of the net eroded mass flow rate going in SOL plasma – as shown in Figure 3.2 by the orange arrows – we must add also this contribution:

$$
\Gamma_{Li/Sn,\ net} = \Gamma_{Li/Sn,\ gross} \cdot \left(1 - f_{redep}\right)
$$

Equation 3.3 – Net erosion mass flow rate for Li/Sn.

where f_{redep} was taken by [29]. The formulation of f_{redep} depends on the plasma parameters in the neighbourhood of the target. Besides, in the calculus of f_{redep}, multiple ionizations events in the sheath edge and effects of different ion and electron temperatures are neglected. Lastly, beware that the same prompt redeposition coefficient is applied to both the evaporated and the sputtered mass flow rate contributions.

3.1.4 Target surface temperature distribution

Now we divide the following treatment in two parts, according to the two different prototypes used for each LM testing ground.

The Li target has a flat geometry (Figure 3.1) and so simulations were performed dividing it into a certain number of 1D finite difference thermal models, each corresponding to a specific poloidal location. The W and the CPS substrates were examined as attached layers, without evaluating any thermal resistance between the two. As previously introduced, with the assumption of LM slow motion and no occurrence of splashing phenomena or droplet ejection, CPS can be simplified with a solid structure characterized by average thermal properties (i.e. thermal conductivity and so on). Moreover, Li can be considered as pure, neglecting any impurity contamination on the surface or bulk (which may change the overall thermal properties). Cooling due to evaporation processes, temperature-dependent properties and enthalpy of refilling metal were included. As a final remark, we must mention that a Robin-type BC (i.e. uniform heat distribution condition) was imposed on the side where atomized water is supposed to impinge. The coolant temperature is 20 °C, meanwhile the constant heat transfer coefficient is supposed to be $8 \cdot 10^4$ W / (m$^2 \cdot$K).

For the Sn case instead, as already said, the CPS geometry proposed by ENEA was used for simulations (see ch. 2). Unlike previous paragraph, here the layout is a bit more complex. For this reason, for each poloidal discretized coordinate a 2D model based on FEM method was implemented to calculate the thermal performances of the unit and only thereafter coupled with SOLPS-ITER code. The coupling is self-consistent, considering that few modifications were made in order to take into account the specific geometry, the materials proposed by ENEA design and the cooling by evaporation. In the end, we specify that each "slice" have been examined individually because of the very little temperature difference of the coolant between inlet and outlet. In other words the heat transfer coefficient on the radial direction is so big compared to the one on the poloidal direction that the latter is neglected. This assumption is a very good time saver and quite conservative, because in the real application, for example near the peak load, all the heat will not be handled by only one finite volume, but also shared with the neighbouring ones, smoothing the spike. For the coolant inlet temperature and heat transfer coefficients we refer to [10].
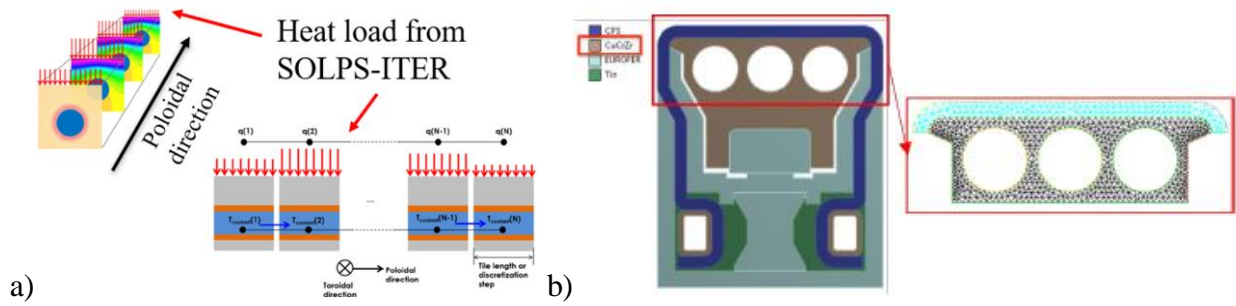
Figure 3.3 – Schematic of the thermal behaviour of ITER W divertor according to FEM code approach (a). At the right the usual CAD of the CPS cross section divertor unit [10] with the zoom on the mesh created to model each grid cell of the Sn divertor (b).

For simplification, for each barycentre of the SOLPS-ITER discretized slice a poloidal section was chosen. Therefore, a 36-elements array of temperature values is given to SOLPS-ITER at each time step, which use this information to evaluate evaporation and thermal sputtering. A corresponding array of 36 plasma heat fluxes is created at the end of the time step and it is sent back to the thermal model and the coupling between the two codes is made (Figure 3.4).
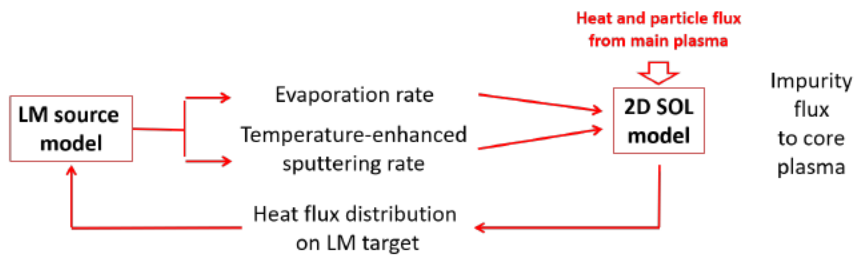
Figure 3.4 – Resume about the coupling strategy between the thermal model and SOLPS-ITER method.

3.1.5 Results

At the end of the study, G.F. Nallo in [25] shows the results of his work in two different scenarios: firstly without Ar seeding and then with it. For the following treatment we will refer only to the latter, since his aim was to justify the choice of the actual usefulness of Ar injection, whereas ours consists only in understanding if our CPS prototype can sustain certain thermal loads from plasma. Knowing this, let us have a look of the final data, which it was used as input for the thesis work.

3.2 Dataset

3.2.1 ξ

A vector of size 1x36 contains the poloidal coordinates (in m) at which the heat flux has been computed. Its elements go from negative to positive values because the origin 0 point is set in the strike point[21], where we have the maximum possible heat flux.

From Figure 3.5, representing the ξ vector, we can see some dashed coloured vertical lines which indicate the position of each unit. In fact, the distance between each subsequent vertical line (except for the black one) is always 233 mm, i.e. the divertor unit length along the poloidal direction (Figure 2.4.b).
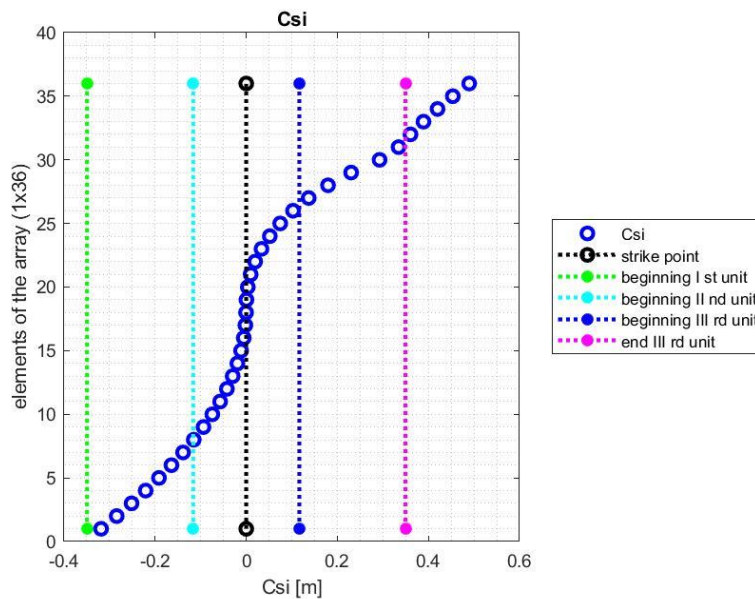
Figure 3.5 – ξ values represented in function of the position inside the 1x36 array. The first divertor units – everyone in parallel along the toroidal direction – goes from the green to the cyan dashed vertical lines, the central one – in series to the former – from the cyan to the blue ones, the last ones from the blue to the magenta ones. We can immediately note that the more we get close to the strike point – inside the mid unit –, the denser the discretization mesh becomes.

From Figure 3.5 we can note that there are values outside the limit of the third unit. It may seem unreasonable, but we have to consider that these data were the result of the experiments on plasma-wall interactions with certain surface dimensions which are not consistent to our case. For this reason from now on they will not be considered the last 4 values, taking into account only the [1,32] elements of the (1x36) arrays.

3.2.2 Area

Also here a vector of size 1x36 contains the areas (in m^2) of the discretized PFS referring each to the corresponding ξ poloidal coordinate (Figure 3.6). Beware of two considerations. First, the area is

---

[21] The strike point is defined as the position where separatrix crosses the target

computed for the whole toroidal extension of the tokamak, so if we want the area of a single unit for a specified j position on the poloidal direction, we should simply calculate AreaSingUnit(j) = Area(j) / number_of_units. In the same way we can arrive to the areas for each cassette or group of three modules. Secondly, these areas are centred in ξ. So, on the toroidal dimension the base is constant and set equal to PFS_{base} = 39 mm (for each single unit), instead on the poloidal one we have the heights of the rectangles PolLength(j) = AreaSingUnit (j) / PFS_{base}. These segments are then centred in ξ as shown in Figure 3.7 (left). Furthermore, these data were obtained from experiments on a general PFS, so not consistent with our geometry. As a consequence, we can have overlappings (Figure 3.7 right) and knowing that these areas are multiplied to the heat fluxes, a global bigger heat flow is given to the divertor. From calculations we can derive the relative error in this way:

$$Err_{overl} = Q_{overl} / Q_{tot}$$

Equation 3.4 – Err_{overl} definition.

Q_{overl} is the surplus heat flow, equal to $A_{overl} \cdot HF$, where A_{overl} is the vector collecting the overlapping areas and HF the corresponding heat fluxes. Instead Q_{overl} is the total heat flow array given as input to the units in the model, obtained by simply multiplying two vectors: Area and HF (Heat Flux). Beware that these values were found only considering one single unit. Nonetheless, if it would be scaled up to the total divertor size nothing should change.

For PFS_{base} = 39 mm we have a mean error $Err_{overl} \approx 4$ % (maximum value almost 28 %), so this approach is quite conservative: to our model it is given a heat flow input bigger than the real (expected) one. Although, if we convert to a PFS_{base} of 40 mm, we would have a mean error of almost 2 % and a peak (always at the same position as before, i.e. between 30 and 31 in the graph in Figure 3.7) of about 23 %.
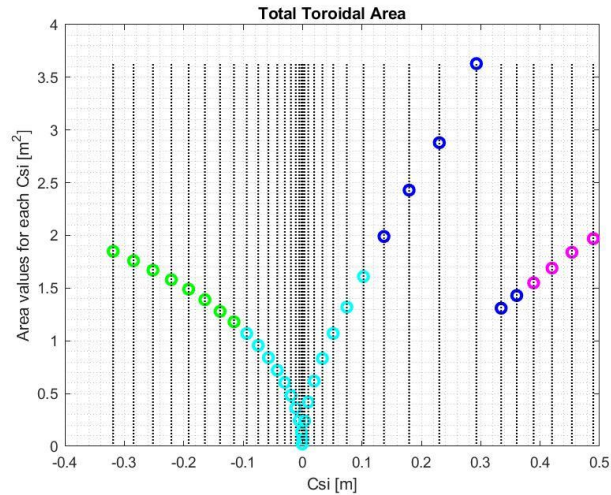
Figure 3.6 – Representation of the area values (coloured circles) for each position on the poloidal direction ξ, indicated by the dotted vertical lines. In green we have the elements in the first unit, in cyan the ones in the central unit, in blue the ones of the last unit, instead in magenta the ones discarded because outside our domain of interest. To be noted, as already previously said, that the closer we get to the strike point, the more packed it becomes the discretization (black vertical dotted lines) and also the smaller the discretized areas are. We need lower values of areas for those positions close to the strike point, in order to grasp better the thermal behaviour of the slices where gradients change rapidly and the target is highly stressed.
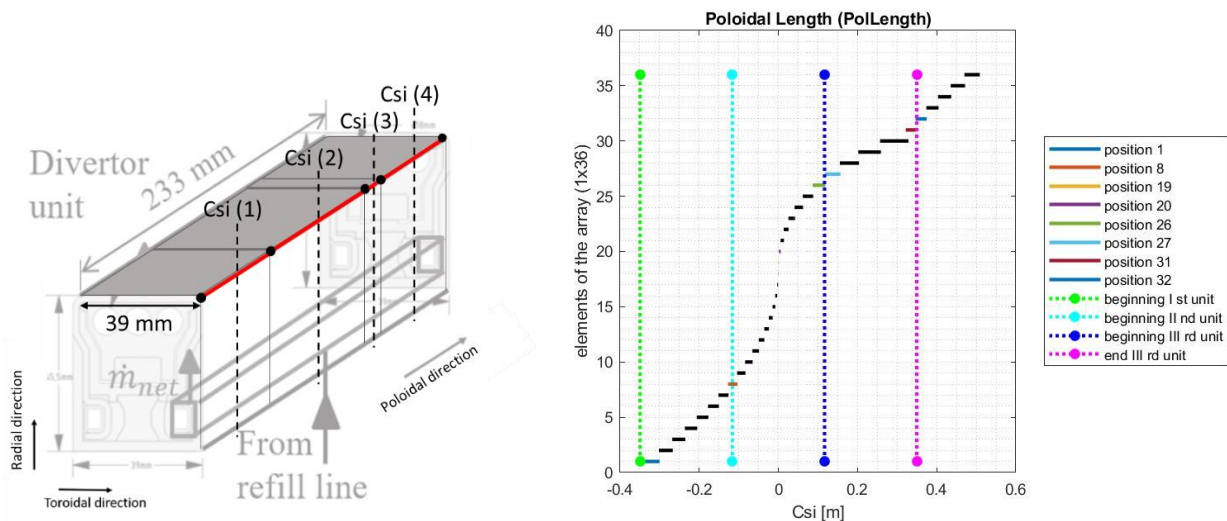
Figure 3.7 – At the left, a sketch of a single divertor unit with a random discretization of the PFS, keeping in mind that all the small rectangles are centred along the poloidal direction in the ξ positions. However we have to consider that the experiments needed to obtain "Area" are not consistent with our geometry, but with a general PFS. For this reason there could be overlappings and showing them is the aim of the graph at the right, where on the coordinate we have ξ, on the ordinate the position on the array and the horizontal segments stand for the red lines in the left picture. Between position 30 and 31 (in brown) we can see the most evident overlapping, which as well refers to the biggest error Err_{overl} mentioned above.

3.2.3   Heat Flux

A vector of size 1x36 contains the heat fluxes (in W/m^2) corresponding to each ξ poloidal coordinate (Figure 3.8 and Figure 3.9). Remember that these values are the result of the steady-state nominal condition scenario with Ar seeding, (3.1.5). Moreover, as already stressed out in the previous subsection ξ, we have 4 final values ([33:36]) which go outside our 3 units design. By the way we

can see that its contribution is quite low with respect to the other units (lower graph), albeit not that much if we consider $Q_{tot} = Area \cdot HF$. If we calculate the sum of the Q_{tot} contributions for each real and unreal unit, we can find the overall heat load divided per each unit:

Tin

Unit	Total heat load W
I	$1.78 \cdot 10^5$
II	$7.32 \cdot 10^7$
III	$3.62 \cdot 10^6$
IV (not real)	$7.99 \cdot 10^3$

Table 3-1 – Total heat load exhausted by each unit in the case of Tin LM. The second one is clearly the most critical one, especially where separatrix crosses the target, then we have the third one which faces the private plasma and in the end the first one which sees the ending part of SOL. The fourth one relates to the rejected values, not needed for our research activity.

Lithium

Unit	Total heat load W
I	$1.06 \cdot 10^5$
II	$9.11 \cdot 10^7$
III	$6.69 \cdot 10^6$
IV (not real)	$5.22 \cdot 10^4$

Table 3-2 – Same as Table 3-1 but in case of Lithium as LM.

We can conclude that the total heat load discarded – outside our design module – is not so small, though it is assumed that it is handled by the structural materials and no further investigations are done in this thesis work.
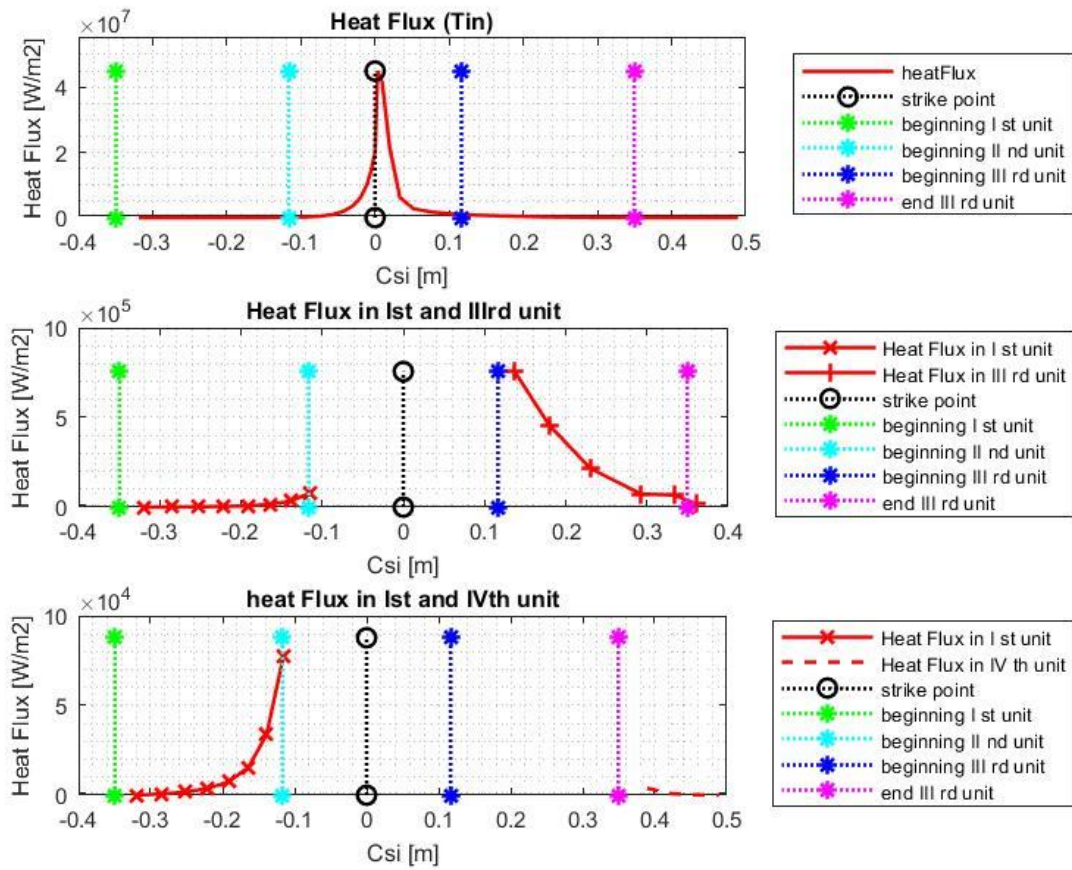
Figure 3.8 – In the upper graph we have the raw data of the Heat Flux in function of ξ, regarding the medium Tin. In the central and in the bottom one, instead, we have only the values related to the first (bottom) unit, the last (upper) unit and an imaginary fourth unit, possibly placed over the other ones in the VT.
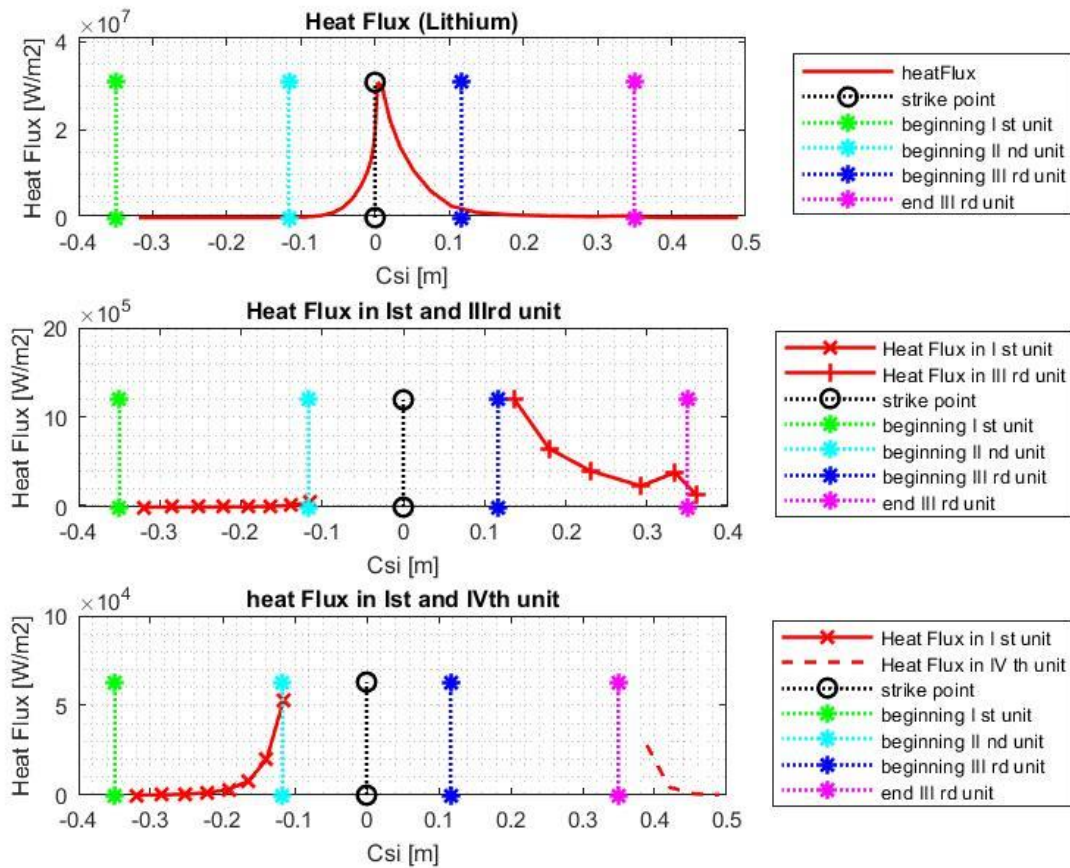
Figure 3.9 – This is the same as Figure 3.8, but related to a configuration where Lithium is wetting the CPS. The curves are quite different because – as told in 2.4 – Li should have a bigger evaporation rate, so more important vapor shielding effect and in the end a different behaviour of the plasma heat flux with the PFS.

3.2.4 Sputtered particle flux

A vector of size 1x36 reports the gross sputtered particle flux (in number_of_particles / (m^2 · s)) corresponding to each ξ poloidal coordinate (Figure 3.10). For the net sputtered mass flow rate we should multiply these values with the areas of our interest, the $m_{Li/Sn}$ (mass of one single atom of Li/Sn) and add the contribution of the prompt redeposition coefficient.
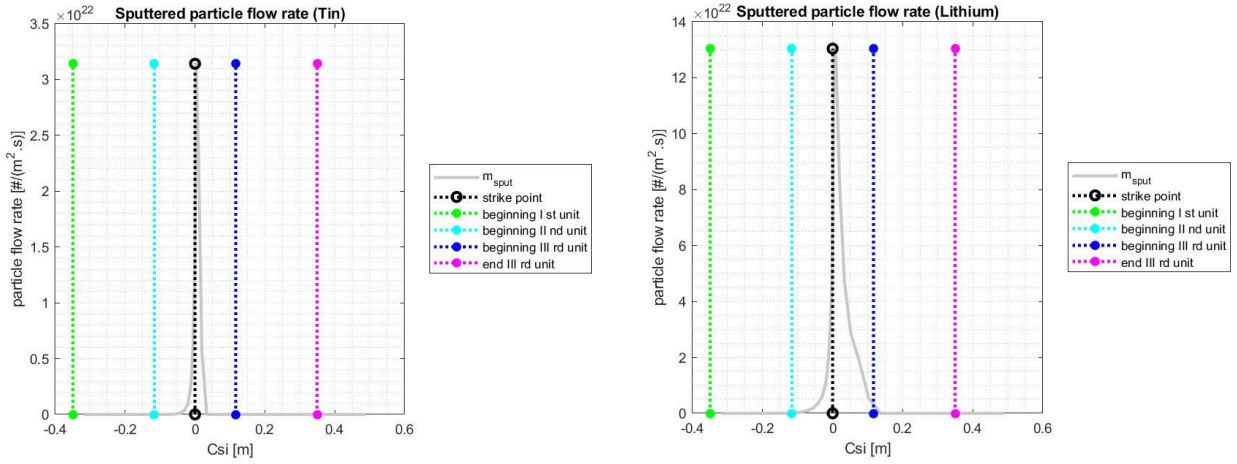
Figure 3.10 – Comparison between the sputtered particle fluxes in case of Tin (left) and Lithium (right) configurations.

3.2.5 Prompt redeposition coefficient

A vector of size 1x36 includes the prompt redeposition coefficients characteristic of each ξ poloidal coordinate (Figure 3.11). They are all adimensional numbers.
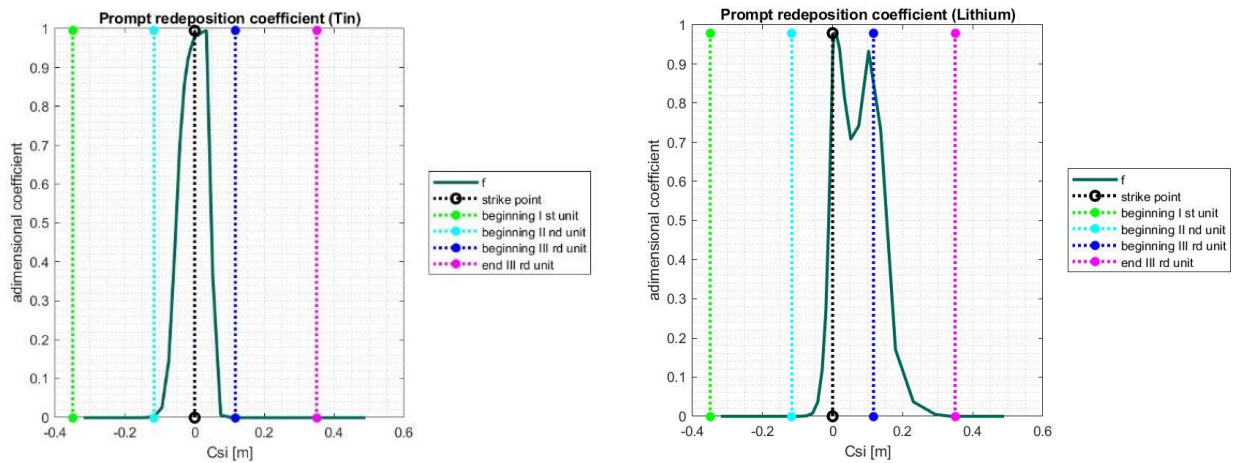
Figure 3.11 – Prompt redeposition coefficients in case of Tin (left) and Lithium (right) configurations.

3.2.6 Surface temperature from heat flux

A table of size 81x2 contains in the first column a set of arbitrary[22] heat fluxes (in MW/m^2) and in the second one the corresponding average[23] temperatures of the PFS (in °C) computed according to the 2D heat conduction calculations on the unit cross section already mentioned in 3.1 (Figure 3.12).

---

[22] In MATLAB language it can be expressed as [0:0.5:40] or linspace(0,40,81)

[23] With respect to a certain X heat flux impinging on a surface with a Y area, in a 0D approximation the latter is considered as one single point and only the average overall surface temperature is given as output
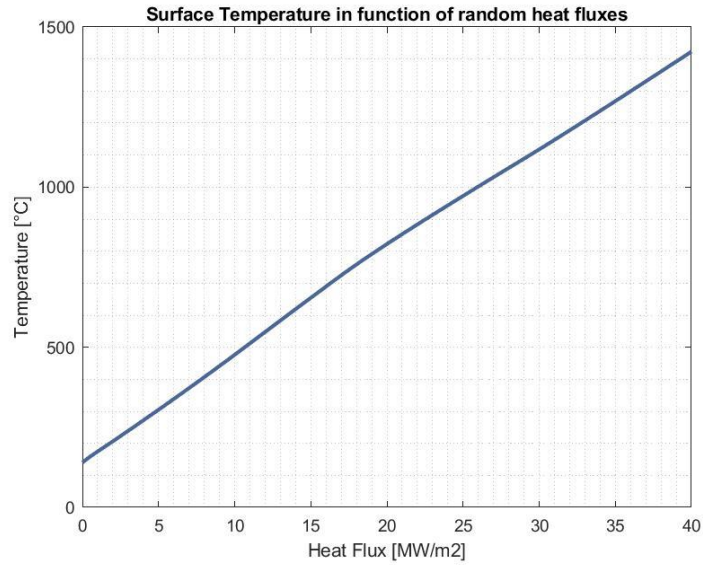
Figure 3.12 – Surface LM temperature in function of casual values of heat fluxes. The behaviour is quite linear and it is the same in the cases of Li and Sn as LMs.

4 Modelica Modelization

4.1 Model Description

For the model implementation it was used previously the OpenSource software OpenModelica, then Dymola version, which is quite similar to the former, except for better graphics layout, a better explanation of the errors and other features. Not by chance the latter is available for a certain fee, but thanks to the NEMO group there was the permission to use it freely from the DENERG's servers and go on more quickly with the work. The models were developed both writing dedicated objects and exploiting models available in the open source ThermoPower Modelica library, developed by Politecnico di Milano [31].

In the following subsections it will be briefly presented the model, explaining all the relevant components and the important equations considered. In appendix A, instead, it will be reported all the written code illustrating what is necessary to run the total final model.

4.1.1 CPS model

In Figure 4.1 we have the diagram view of the model which is consistent with the slice represented at the right. The central LatoPL[] "Flow1DFV2w" stands for the green parallelepipeds at the right. This component represents an array of NN elements – constituting each unit – with a parallel of two sections which, in fact, are the two green volumes. This layout is due to the fact that all the slices are considered adiabatic and no heat flows along the poloidal direction (i.e. from one slice to the adjacent ones) for assumptions. Thus, for each ξ we have again two independent sections because each one is related to one reservoir, from which it takes the LM and it lets it out from above, when the end of the parallelepiped is reached. Even if it should be more correct creating a model which keeps into account the distributed local loss along the toroidal direction of the LM, for now the modelling choice was to maintain this level of complexity. Therefore, the LM goes so slowly inside the component that a quasi-static flow model was used and for the heat transfer model the convection cooling contribution was neglected. Relations like the Dittus-Boelter one would fail in such conditions and so it was chosen the constant heat transfer coefficient, varying it ad hoc for our scope. In fact the CPS is composed by the W mesh, as already mentioned, and by the LM, with weight composition rate of 50 – 50 % for hypothesis (see 2.1.3). Trying to be as realistic as possible, we used the semiempirical correlations, which express the thermal conductivities of these materials in function of the LM temperature, which

is changing in time. For all the code particularities have a look at "VariabHeatTransferCoefficient" in Appendix A.

Then, to conclude the LM discussion, a couple of words about the operating pressure. Even though the capillary pressure for the CPS is expected to be ≈ 100 kPa [30], in order to avoid the risk of overfilling the CPS an attempt pressure was chosen among the interval $50 - 70$ kPa. In fact, the PFS sees on one side the high vacuum conditions of the VV and on the other one the reservoirs overpressure. The capillary force has to overcome this and other contributions (like the JxB and gravitational forces), so 60 kPa was decided as a reasonable try for this purpose. In simulations this value will remain constant, as desired.
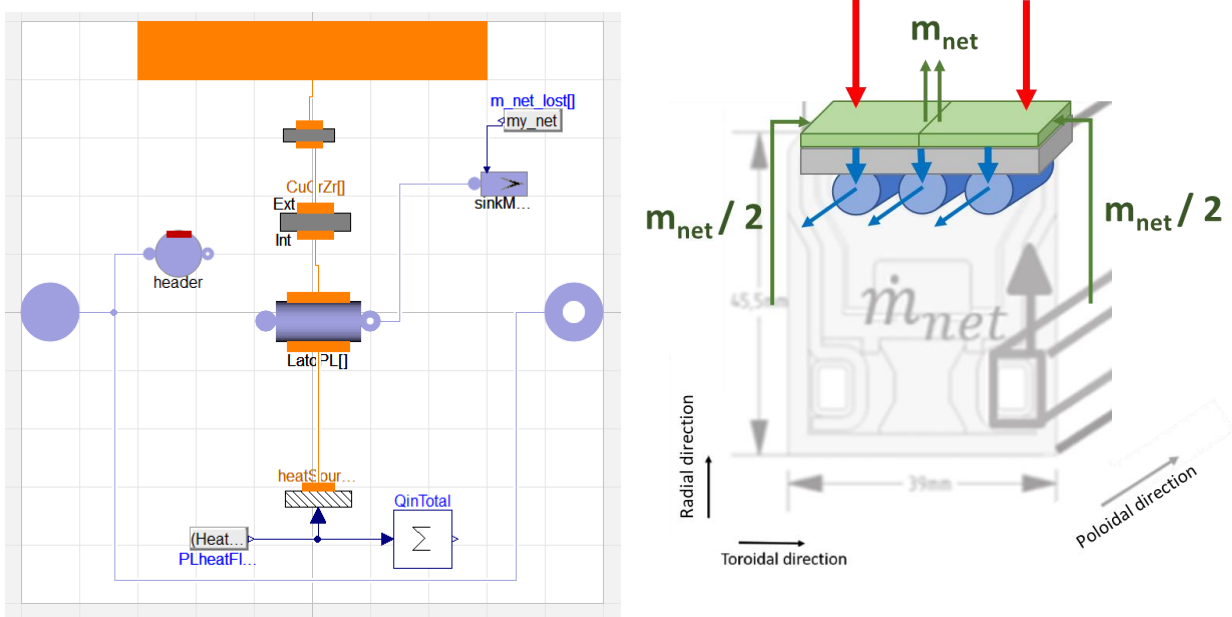
Figure 4.1 – At the left, diagram view of the model ("cps" in appendix A) representing the single divertor unit inside the VT. At the right, overview of a single slide describing all the characters in play.

Afterwards, underneath we have the "HeatSource1DFV" component which gives to LatoPL[] the input power from the plasma after deduction of the evaporation cooling contribution. Above, instead, we have the CuCrZr layer which is again an array of the same NN elements, but without the separation in two different volumes as in the case of LatoPL[]. Indeed, it is simply a structural material and its function is to exchange heat, from the CPS to the water channels. By the way, one big assumption was made here: a simple slab of thickness L_half_wall · 2 = 1.8 · 2 mm = 3.6 mm was selected, without considering the complex geometry of the CuCrZr in contact with the water channels, which may change the heat transfer characteristics as well. Even if a more detailed mesh of this layer would be better for the final simulation result, for now, this level approximation was assumed acceptable.

Above there is the Converter component which aim is to manage the interface between the NN array configuration of the components listed so far and the single-component one with NN elements. Because of the cross-flow configuration of LM going in the toroidal direction and water in the poloidal one, it was necessary to write this ad hoc component, otherwise neither OpenModelica, nor Dymola would have been capable to handle this problem automatically. Thus, at the top we have the wall which will be responsible of exchanging heat from CPS to water tubes.

The "SinkMassFlow" extracts the total eroded mass flow rates, which values are given as input to this component. It is again an array of NN elements, consistent with LatoPL[], instead the "Header" is single-component only and it is connected with the flow1DFV2w thanks to manual connection relations inside the for loop of the equation section. Roughly speaking, it takes the LM from "FlangeA" at the left and it splits the flow in NN divisions, consistent with LatoPL[]. In the end, the LM not needed by this unit, goes from "FlangeA" to "FlangeB" (direct line between the two components) without interferences, straight to the next unit or to the refill line loop.

4.1.2 Water circuit

In the water loop (Figure 4.2) we have a "SourceMassFlow" and a "SinkPressure" because for the aim of this research activity it was judged enough. Closing the loop would be obviously more correct but in the simulations they would not appear such big differences. Moreover, the "ramp" block is needed to go from low values of water flow rates to nominal ones. It is not at the same time as the power and sputtering ramps, because it was observed that in this eventuality bad resonance effects on water temperatures would occur. To solve this situation, we let the water ramp start and conclude before the start-up phase. The three "flow1DFV"'s represent the group of three parallel water channels inside each divertor unit showed in Figure 4.1. The "DHTVolumes" at the top refer to the different heat transfer walls for each divertor unit. unitA stands to the bottom one, unitB to the central one and unitC to the top one. Not by chance we can see here below that the flow of water comes from unitC to unitA (see 2.2 for further details).
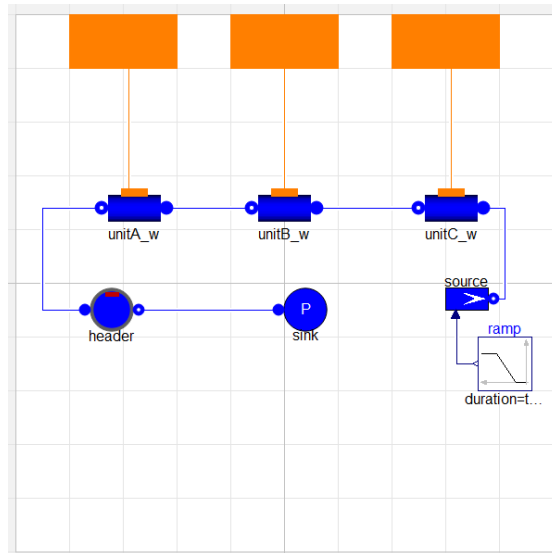
Figure 4.2 – Diagram view of the water circuit ("Water_circuit_ud" in appendix A).

4.1.3   He circuit

Also in this case (Figure 4.3) we have two components like the "SourcePressure" and the "SinkMassFlow", because for the aim of this work it is not needed to go too much in details implementing the whole circuit. The "LM" flow1DFV component is the one which exchanges heat with the refill line, keeping it at 350 °C. "header" is present in order to act as a buffer: it helps the hydraulic physics of the problem but also smoothing the simulations – a good aid for the numerical calculations. For the same reason it was used also for the CPS model and the water circuit. In the end the "PressDrop" component takes into consideration the pressure drops of the circuit.
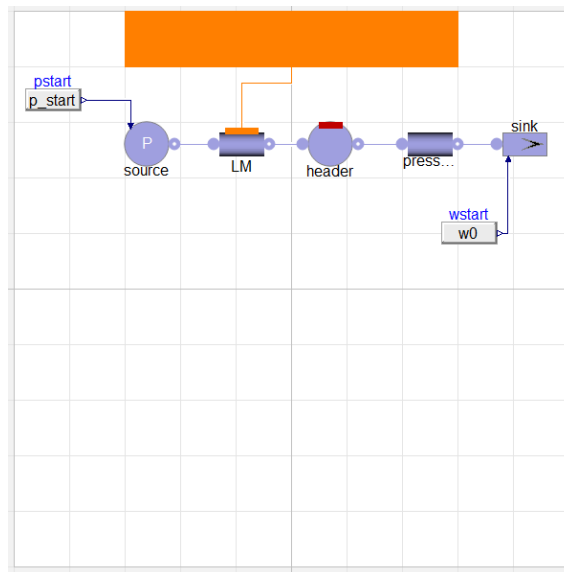
Figure 4.3 – Diagram view of the gas circuit ("Helium_circuit" in appendix A).

4.1.4 Coupling model

Last but not least, it is reported the coupling model (Figure 4.4) which takes all the components listed so far, make them interact and run the simulations, which results we will see in next chapter. Again, here we do not have a closed loop but an open circuit, for the same reason as before. We have the usual "SinkMassFlow", "SourcePressure" and "PressDrop" already discussed about. Thus, HeHX is the "Flow1DFV" component which carries the LM from the reservoirs and makes it exchange heat with the helium circuit (He). unitA_LM, unitB_LM and unitC_LM are all "cps" components, referring each to the corresponding real unit. They are then associated to the water circuit with the orange connections ending to the unnamed component, which stands for the water circuit previously mentioned. Therefore, at the top we have the two models "myPowerTransFcn" corresponding to the two power and sputtering inputs. The former, besides, is connected to the "CombiTable1D" block, set-up according to the data described in 3.2.6. In a practical point of view, for a certain power input, this table interpolates the output Surface Temperature, exploiting the values obtained by SOLPS simulations. Ultimately there is at the bottom the "System" component which sets the system wide properties and default.
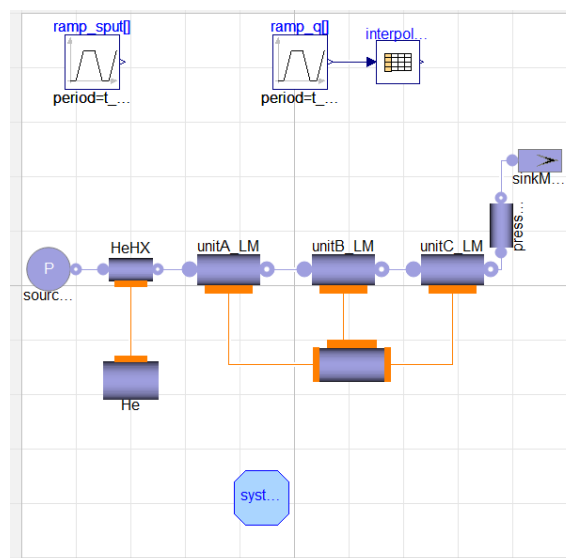
Figure 4.4 – Diagram view of the coupling model ("final_model" in appendix A).

5 Results

Now, with all the assumptions expressed, the code compiled and the input data in our possess, we can move forward, straight to the simulation analysis. We start with the Li case, then similar comments will be done for the Sn one.

5.1 Li

5.1.1 Nominal Power

The first thing we have done was giving a ramp of power and a simultaneous one of sputtered mass flow rate from 0 power – and of course also 0 sputtering – condition, till the nominal one, to mimic the load during the start-up phase. We have already stressed out that simply giving a ramp – also for the sputtering contribution – is not correct because it does not take in consideration the start-up phenomena (see 3.1.1). Nevertheless, it is assumed that this factor would not affect that much the final result. What is surprising is the freezing of the LM after larger and larger heat fluxes coming from the plasma. Thus, in order to understand this phenomenon let us start from the beginning, i.e. the surface temperature computed by the model.

According to the interpolating table of 3.2.6, a linear increase of the heat flux leads to the same behaviour of the Surface Temperature and this is what we register in Figure 5.1.
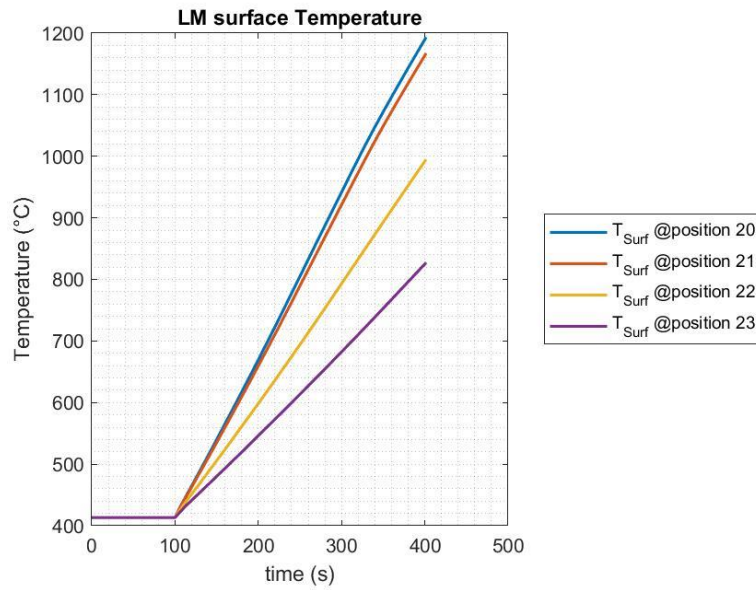
Figure 5.1 – Target surface temperature distribution in function of time.

The computed surface temperature profile is then used to compute the profile of the evaporated mass flux according to the HK equation. In this way we obtain the evaporated mass flow rates, with which we can arrive to the total eroded mass flow rate. In Figure 5.2 we can see what happens in the central most stressed unit.
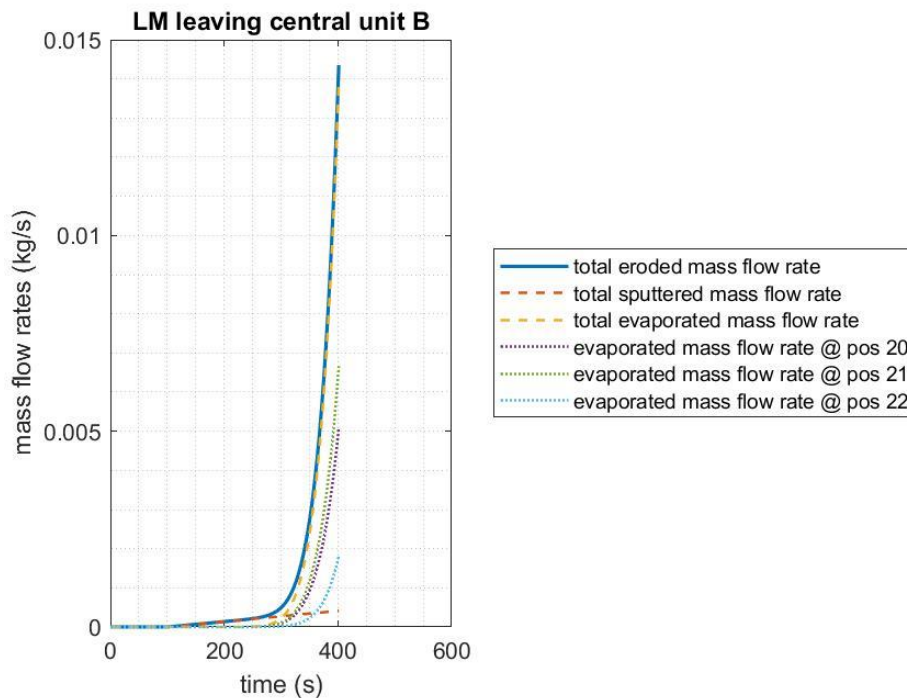
Figure 5.2 – Sputtered, evaporated and total eroded mass flow rates regarding the central unit – the one where the strike point is located. The first one is linearly increasing, consistently with the assumptions described above. The second one instead has that particular curvature because of the exponential dependence of the vapor pressure on the surface temperature, which is contained in the HK relation, and after 300 s starts to become important. As we can see, the contributions of the slices near the strike point (i.e. 20, 21 and 22) dominate the overall behaviour.

A total eroded LM mass flow rate exiting the target bigger than ~10 g/s should ring more than one alarm. The consequent evaporation cooling – the reduction of the power to be exhausted by the cooling channels due to the phase change – would indeed affect the remaining LM still in liquid form. In fact, right for the most critical positions along the target, such as the 20th, 21st and 22nd, the heat flows through the layers of the target are quite counterintuitive, showing a non-expected descending behaviour (Figure 5.3 and Figure 5.4).
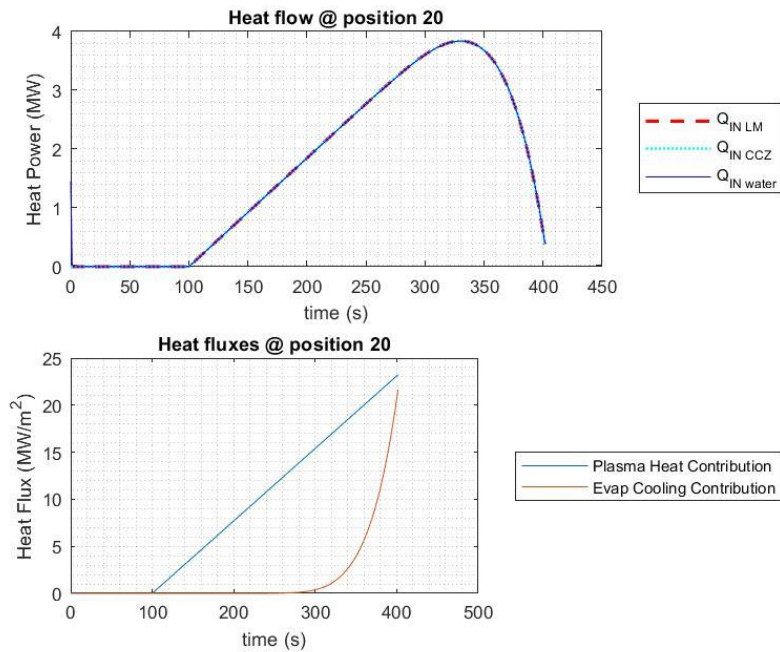
Figure 5.3 – Above we can see the evolution of heat flows along the target – power balance verified – only regarding the "slice" at position 20, which as we have already said is considered adiabatic with respect to the other ones. In red dashed line we have the power incoming from plasma also considering the cooling due to the latent heat of evaporation. In cyan dotted line there is the power exiting from the LM and going to the CuCrZr slab (CCZ) and in the end in blue line the power going from the CuCrZr to the water channels. Beneath, we have the plasma input power contribution, in blue, increasing linearly as imposed, and in red the evaporative cooling one which starts to become important after t = 300 s. Similar considerations can be done also for the graphs below.
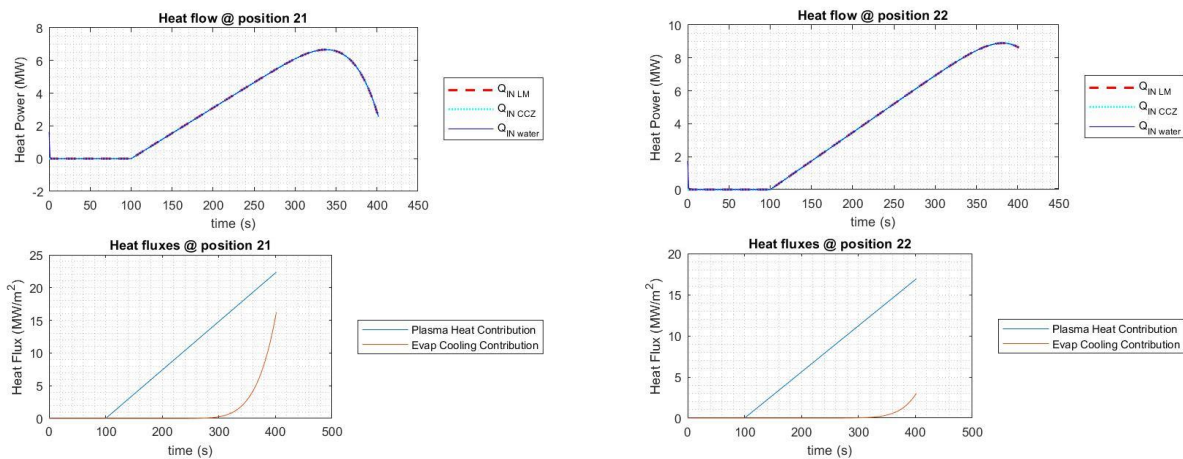
Figure 5.4 – Heat flow and heat flux evolution for position 21 and 22.

This trend can be justified by the evaporative heat flux contribution which starts to get relevant at a certain time and for specified ξ positions. This is the moment and place where LM temperatures begin to decrease, even reaching the freezing point (Figure 55.5 and Figure 5.6).
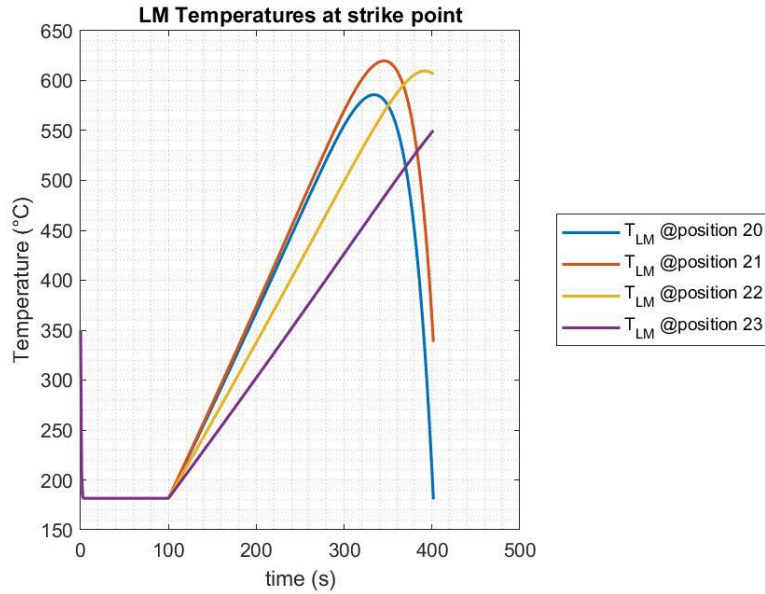
Figure 55.5 – LM bulk temperatures inside the discretized elements near the strike point. Position 20 is the slice where LM becomes solid, leading to the stop of the simulation. It should be noticed that the initial sudden cooling down from the starting condition (reservoirs BC) of 350 °C till about 181 °C (water BC) is related to the initial phase (with reactor still shut down) when a certain amount of water was put inside the water channels, forcing the LM to go at that temperature, and should therefore be disregarded.
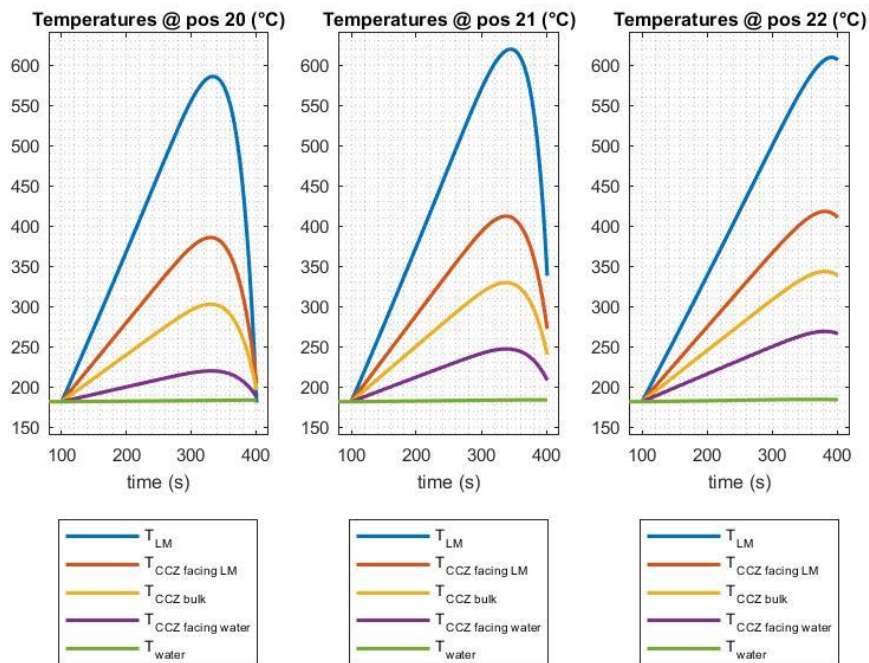
Figure 5.6 – Temperature evolutions along the target with respect to 20th, 21st and 22nd "slices". In blue we have the LM temperature, in orange, yellow and purple the temperatures of the CuCrZr layer and in green the water bulk temperature. Around t = 400 s the simulation stops because the LM in position 20 freezes and our model is incapable of simulating this physical situation.

At last, let's see the water circuit. The pressure is the design one (50 bar) and it remains so along the whole simulation, considering that along the tube there are some pressure losses which, at present, are not fully characterized due to the preliminary stage of the design[24] (Figure 5.7). The temperature, instead, along the pipeline, tend to increase, as expected, but with a slight decrease at the end in the central unit, probably due to the effect of excessive evaporative cooling already discussed about (Figure 5.7).
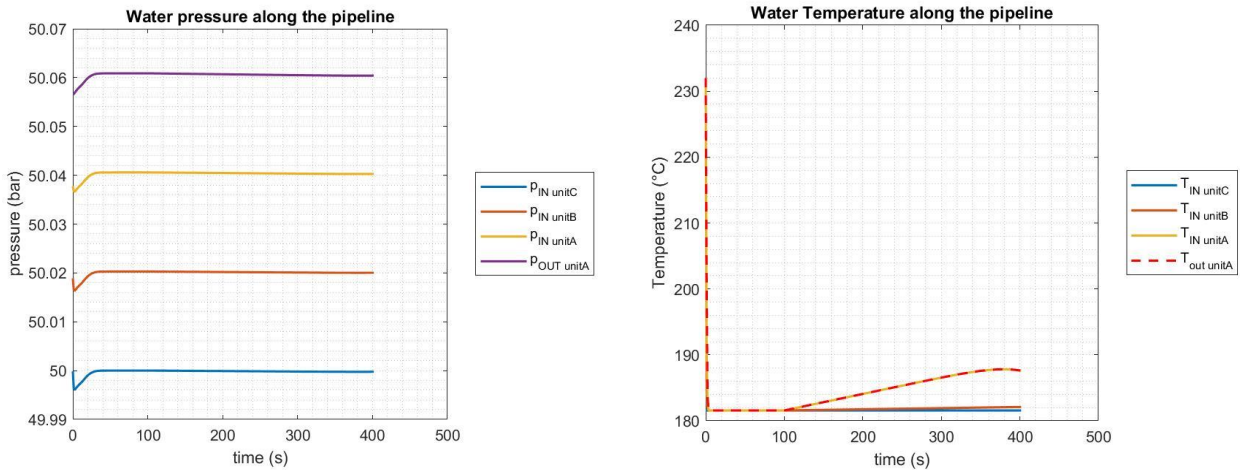
Figure 5.7 – At the left, the water pressure trends inside the target are shown. It is helpful to recall that the water flows from the top (inlet of upper unit C), to the bottom (outlet of lower unit A) of the vertical target. At the right, the evolutions of the temperature of the water along the channels are reported. We start from about 180 °C of first unit C (in blue), to the inlet of central unit B with a slight increase (in orange), then we arrive to the outlet of unit B – or inlet of unit A – (in yellow) and in the end to the red dashed line. As a side remark, it is useful to remind that, as said in the caption of Figure 55.5, at the initial instants of time there is an abrupt change of the temperature (and also of pressure). This is again due to the moment (t = 0 s) when it is put in contact the water at 180 °C and the rest of the target in equilibrium at 350 °C.

5.1.2 Half Full Power

Since what we have said so far, another simulation has been run with the same model, but only with half full power as final state and in the same way half full value of final total sputtered mass flow rate. In this case the results are easier to understand, since in this situation the role of evaporation is less significant. Moreover, it was analysed a possible transient disruption event – after reached the nominal power – but setting its peak power of only 20 % the nominal reactor value. We know that, in real applications, these abrupt events may reach even 2 or 3 order of magnitude bigger than the full reactor power, for a very short period of time, but my model, so far, is not capable to withstand such fast power excursions.

---

[24] It was supposed a certain model and a certain value of pressure loss for each water channel, but it is not the real expected one. We leave the correct substitution of these values in further developments of this work

First of all, we consider the surface temperature which is consistent with the power input, as before. We can see that at steady state the value reached is lower than the ones in the previous setup conditions, but during the trapezoid transient the peak surface temperatures are almost equal to the ones shown before, when the described evaporating phenomenon starts to become relevant (Figure 5.8). Indeed, in this short period of time, the evaporation gets important and makes the heat flows drop down, but only in the most critical slices (Figure 5.9 and Figure 5.10).
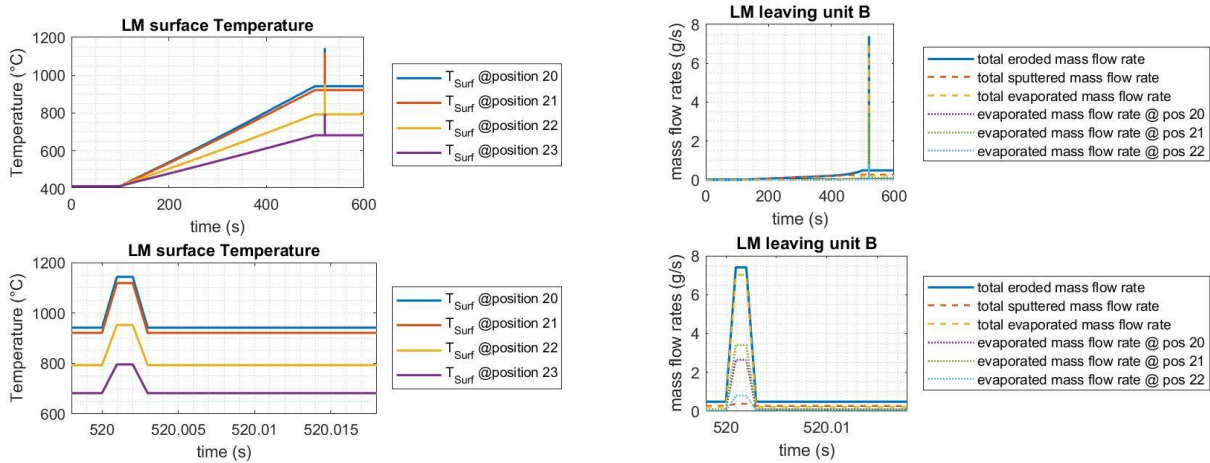
Figure 5.8 – At the left, overview of the LM surface temperature evolution in the most stressed points of the target, with the zoom in the bottom of the transient. At the right, focus on the LM exiting the target and entering the VC. On the bottom the usual focus on the "disruption" event, showing the big contribution of the evaporation mass flow rates.
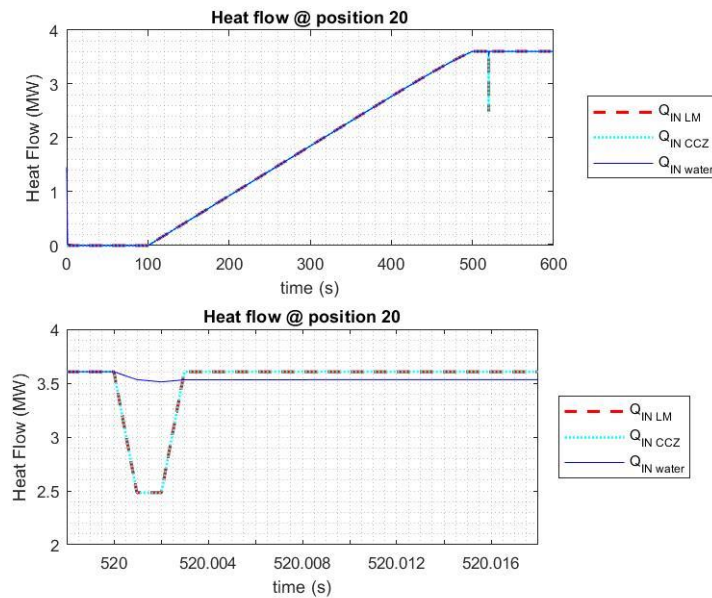
Figure 5.9 – Heat power flowing through the target structure. From the bottom graph we can note the descending behaviour of the power incoming from the plasma and going to the CuCrZr layer. Similar comments can be done for the position 21 represented below, whereas for the 22nd – as well for the other ones – we have an increasing heat flow as we would predict, considering the low evaporation shown in the bottom right graph of Figure 5.8. Thus, a final comment can be made about the heat flow extracted by the water channels, which follows the heat load coming from the LM – it increases if the latter does the same and vice versa – but with a certain delay. Then it goes back to its steady state conditions after a certain long period with respect to the characteristic time of the trapezoid (i.e. about 3 ms).
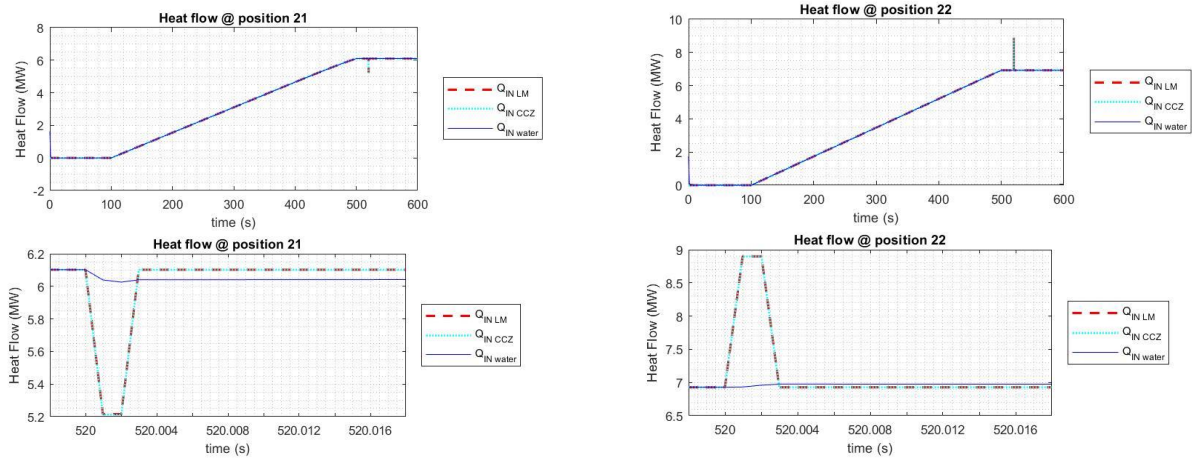
Figure 5.10 – Heat flow evolution in time for position 21 and 22.

After making all this preamble, the crucial graphs of this subsection are now considered: the temperature evolution of LM, CuCrZr layer and water (Figure 5.11 and Figure 5.12). Except for positions 20 and 21, in 22 and all the other points in the divertor, inside the trapezoid transient, the expected direct relation is retrieved: for an increasing input power, the same trend is observed for the target temperature.
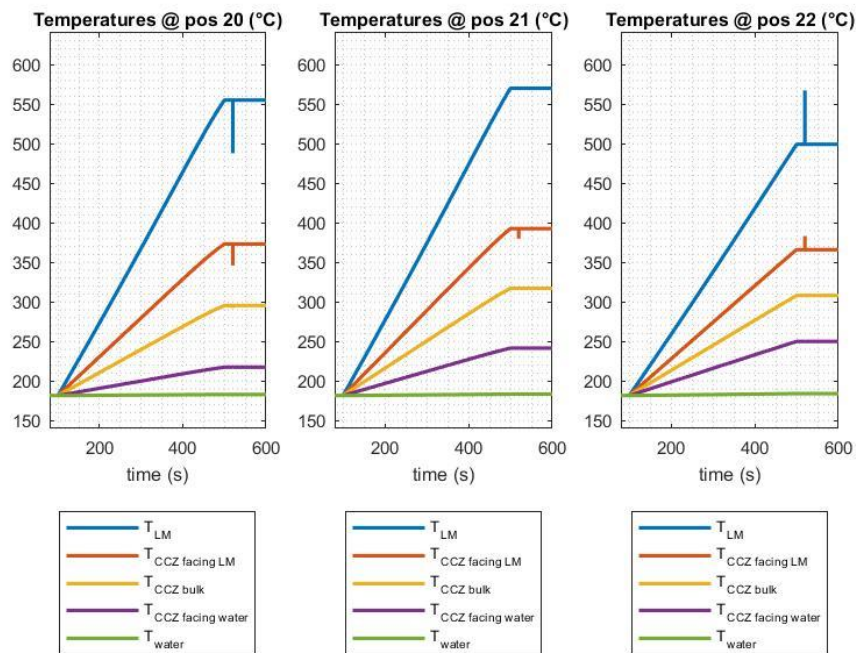
Figure 5.11 – Temperature trends in the most critical positions in the target.
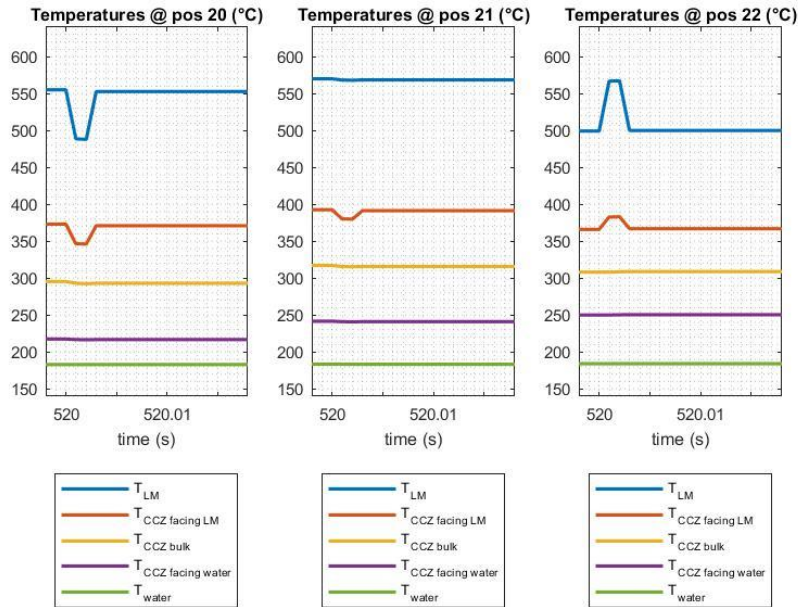
Figure 5.12 – Focus of the above graphs in proximity of the "disruption" transient.

Finally, in the water circuit we do not have appreciably differences in the pressure trends, whereas we do see a big difference in the temperatures (Figure 5.13), which, as expected, have a lower delta ($T_{OUT} - T_{IN}$) along the pipeline compared to the nominal full power case. Therefore, as told in the caption of Figure 5.9, the heat flow extracted by the water changes quite smoothly and for this reason the temperatures involved do not significantly change during the "disruption".
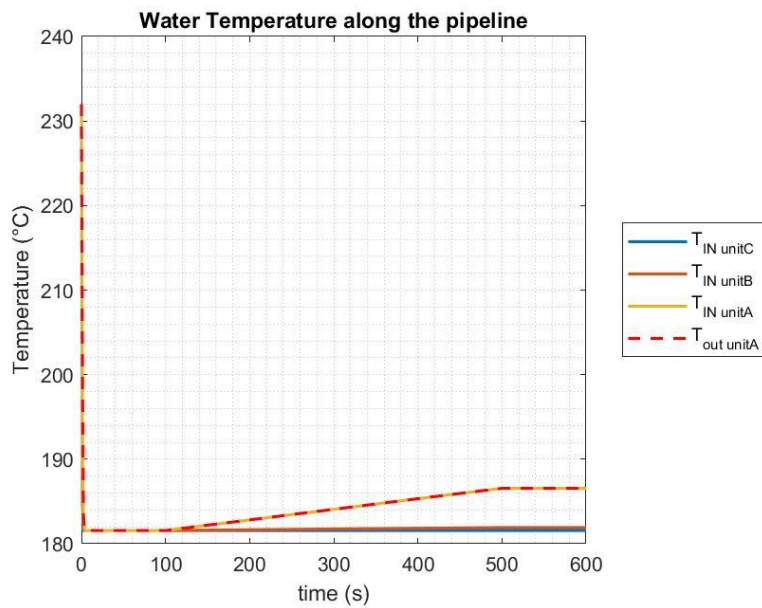
Figure 5.13 – Water temperature evolution along the channels inside the target CPS structure.

5.2 Sn

5.2.1 Nominal Power

For the Sn case, conversely, the evaporation is lower and we are able to reach higher values of surface and bulk temperatures (Figure 5.14). This is exactly the advantage of choosing this LM which doesn't have big fluxes of particles towards the VC like in the Li case.
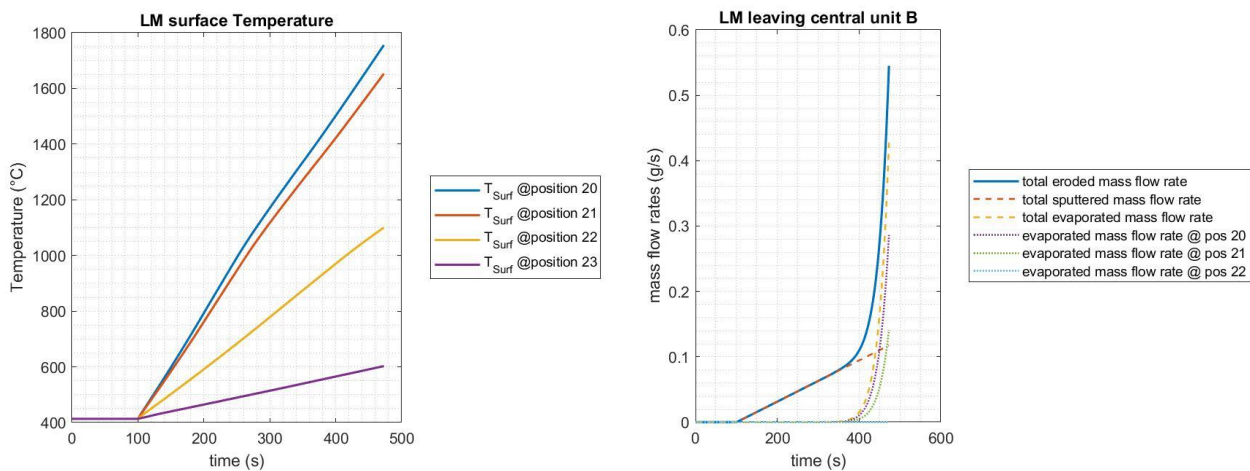
Figure 5.14 – At the left, surface temperature evolution in the most critical points. At the right, the total eroded, sputtered and evaporated mass flow rates with also the contribution of the latter only for positions near the strike point (dotted lines). With respect to Figure 5.2 it is clear the difference of almost 2 orders of magnitude in the values of LM mass flow rates leaving the target.

The heat flows through the target for the most stressed slices again show a descendent behaviour (Figure 5.15 and Figure 5.16 (left)) and, even if here we sustain bigger heat fluxes in input from the plasma, at a certain point we arrive again to the freezing point for Sn (Figure 5.16 (right) and Figure 5.17).
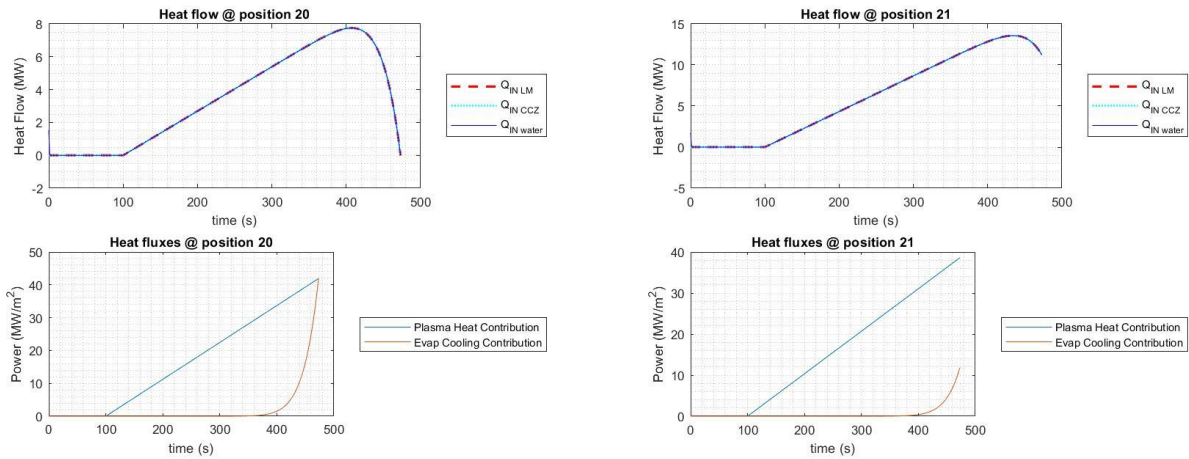
Figure 5.15 – Above, heat flowing from LM to water along the target. Below, comparison between the heating flux from the plasma and the cooling one from the evaporation process. Evident is the overwhelming contribution of the evaporation cooling in the graphs at the left, referring to the same position (20) where the LM solidify (see Figure 5.16, right graph).
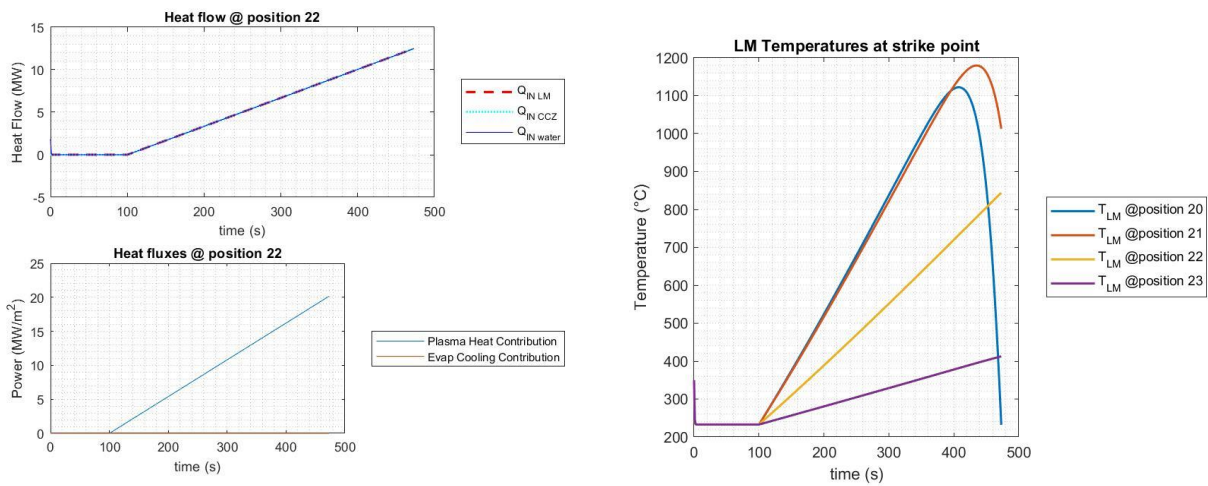
Figure 5.16 – At the left, the same graphs of the Figure 5.15. Here, such as in the other points of the divertor (except for positions 20 and 21), the evaporation is so low that it does not affect the heat flow and so the temperatures inside the target. Indeed, so we observe in the right picture, where the temperatures regarding the slices 22 and 23 go up linearly, as the input plasma power. Instead, in the strike point (blue line) the LM reaches the solid state and makes the simulation stop.
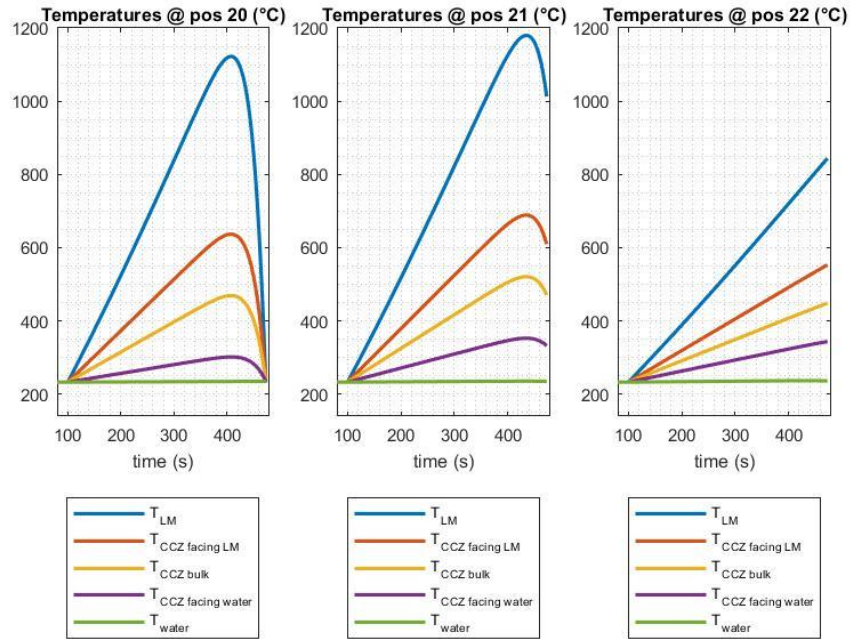
Figure 5.17 – Temperature evolution inside the target components along the most stressed slices. Compared to Figure 5.6 here we reach higher values, probably due to the lower cooling by evaporation and, to a lesser extent, to the smaller thermal conductance of the CPS.

Finally, in the water circuit, the pressures along the pipeline remain quite constant and close to the Li values, instead the temperature difference between outlet and inlet is much lower (Figure 5.18).
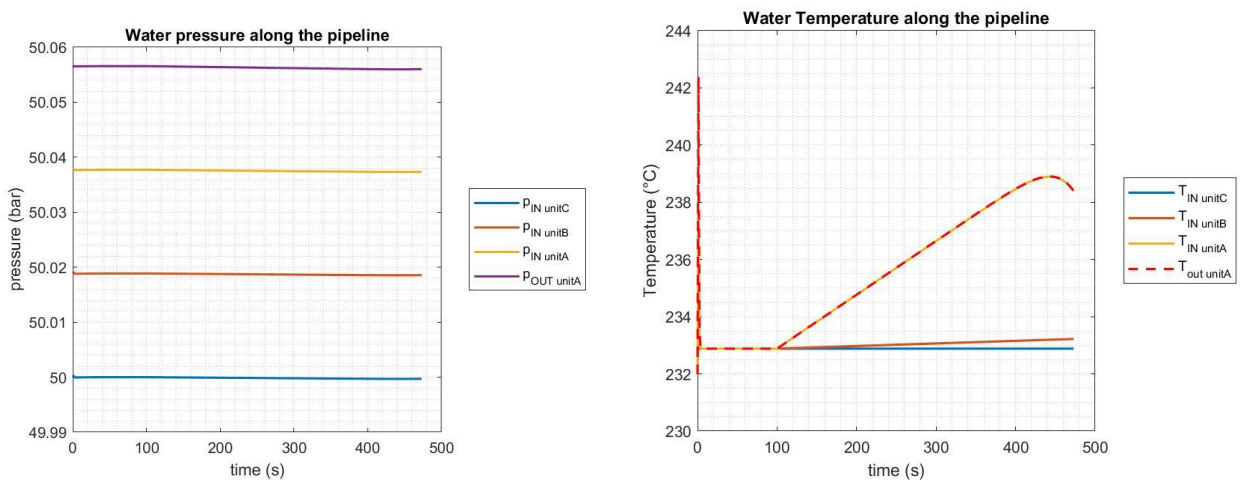
Figure 5.18 – At the left, the water pressure trends from the inlet of the upper unit C, till the outlet of the lower one A. Beware that the friction model is the same as the above Li case. At the right, the water temperature evolution along the channels inside the VT.

5.2.2 80 % Full Power

Here, differently from before, we will give an initial start-up ramp from 0-power to 80 % of the nominal condition and then a "disruption" event at 10 % of the full reactor power. Again, the same percentage variations as for the power are assumed for the sputtering evolution. This choice was justified by the interest of investigating the maximum value of heat load input tolerable by my model before stopping the simulations.

In fact, the surface temperatures reached are quite big, but not enough to make evaporation a significant contribution (Figure 5.19). Nonetheless, we can see that it is not totally negligible, indeed in Figure 5.20 and Figure 5.21, showing the heat flows for position 20 and 21, the power is rapidly decreasing in the trapezoid transients, as we would expect in case of relevant cooling by evaporation. Similar considerations can be done for the temperature trends in Figure 5.22 and Figure 5.23.
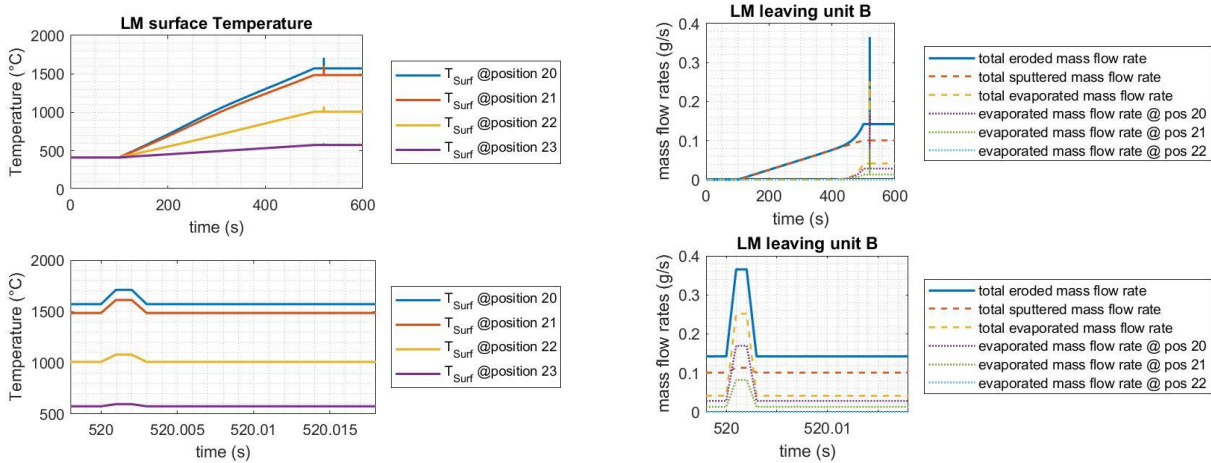
Figure 5.19 – At the left, the surface temperature, in the most stressed slices, as a function of time. At the right, the graph representing the sputtering, evaporating and total eroded contributions of the LM, demonstrating that the evaporation is quite negligible, except for the trapezoid transient period of time.
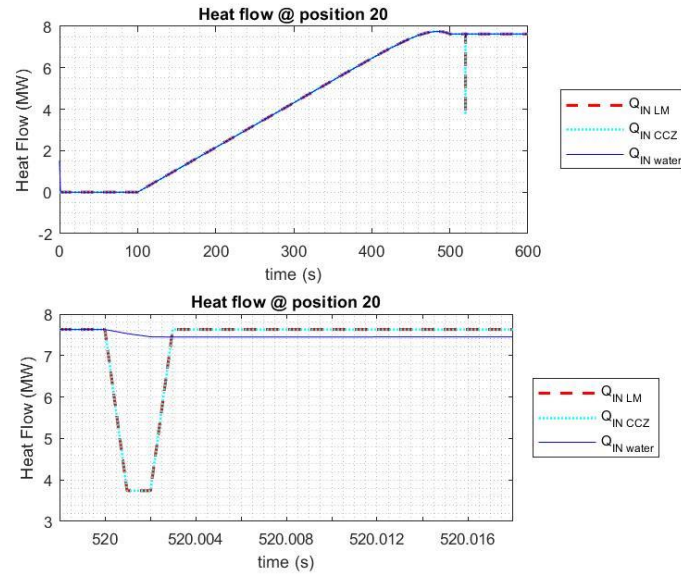
Figure 5.20 – Above, the heat flowing through the target, noting at first sight, at the end of the ramp, the triggering of the descending behaviour, though late enough not to cause freezing of the LM. Observe the effect on the temperature of the LM at the same position in Figure 5.22. Below, the particular of the trapezoid transient. Here, as expected, the power decreases because of the increasing contribution of evaporation.
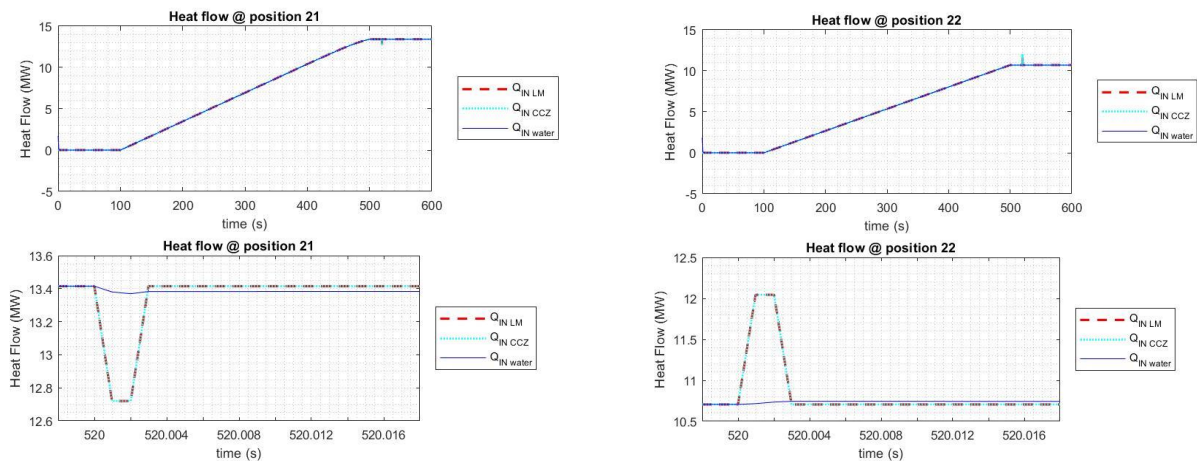
Figure 5.21 – Same considerations as in the Figure 5.20 caption, except for the fact that, at the end of the ramp, here we do not have any descending trend and, in the trapezoid transient at position 22 – such as in the other points of the divertor –, the power increases consistently with the plasma power input.
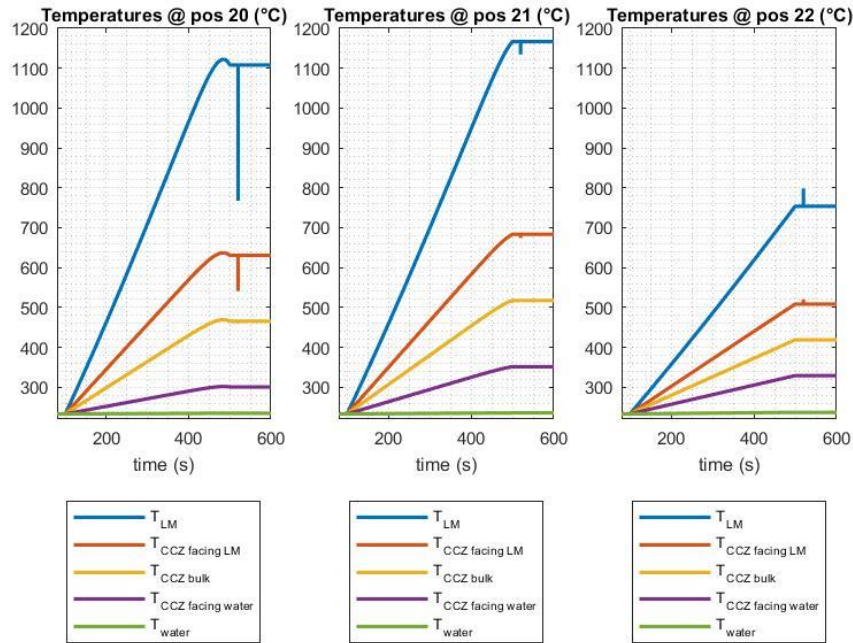
Figure 5.22 – Temperature evolutions inside the target slices in the neighbourhood of the strike point. Observe the slight decrease at the end of the start-up ramp of the LM and CuCrZr temperature in the left graph, as previously foretold.
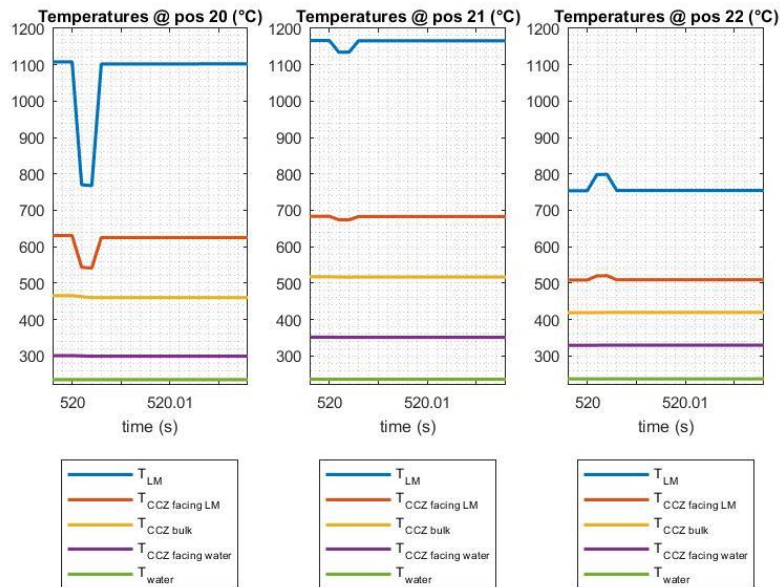
Figure 5.23 – Zoom of Figure 5.22 in the vicinity of the "disruption" transient.

In the end, a few words about the water circuit. The pressure, again, doesn't appreciably change with respect to the Sn full power case and also the temperature is quite similar, apart from the final descending phase (Figure 5.24). Moreover, as for the Li half power condition, the "disruption" event

doesn't significantly affect the water circuit, neither considering the temperatures involved, nor the pressure, the mass flow rate and so on.
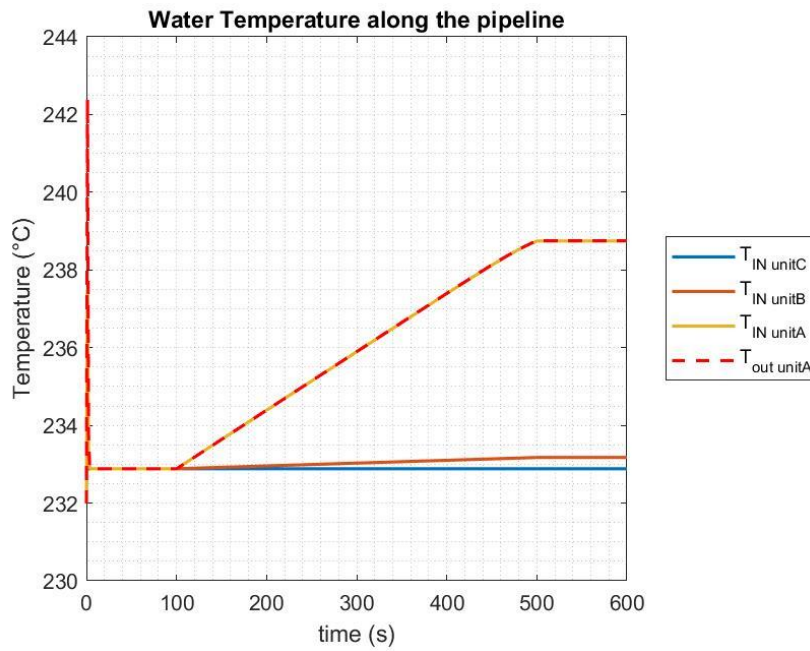
Figure 5.24 – Temperature of the water along the channels inside the VT.

Conclusion and Future development

Phenomenology investigations and model improvement

Finally, at the end of this work, they may be suggested some guidelines for further developments. As we have seen the model is not finished yet and there are also still some unexplained behaviours. More research activities are then needed to better understand and simulate more realistically what's happening inside the divertor.

First of all, it must be investigated what occurs inside the CPS when larger and larger heat fluxes are coming in from the plasma. In the previous chapter we've seen how, for both Li and Sn, the nominal reactor power was not tolerable because, the bigger the input heat load, the bigger the evaporation of the LM, the bigger the power extracted – due to the latent heat of evaporation – from the remaining metal still in liquid form and the colder the latter it gets, reaching even the freezing point. Maybe there could be a threshold input heat flux for each different LM, from which the evaporation starts to become quite significant, leading to a bigger mass flow rate "sucked" from the reservoirs and perhaps to the exiting from the initial assumption of quasi-static flow inside the CPS. Nonetheless, what is fundamental for these simulations is going on with the complexity of the model. They are then recommended several big changes:

1) implementing a freezing model, discretized for each ξ position as well each "flow1DFV" inside the CPS OpenModelica model. It is suggested writing a simple code which controls the temperature of the LM. Just before it would get solid and the simulation would stop, it has to say to the model that particular finite volume has just become solid and no further mass flow rate must be provided from the reservoirs

2) a better heat transfer model must be implemented for the CPS layer. So far it was approximated as a solid slab, because relations like the Dittus-Boelter one would have failed for such low mass flow rates like the starting one (reactor shut down). Nevertheless, for higher input heat fluxes, higher amount of LM should evaporate and as rapidly as possible the same quantity must be replaced, extracting it from the refill line. So the flow would change in this condition and maybe the solid hypothesis would no longer be valid. Moreover, when the LM would freeze – as told in the previous point –, we should return to the solid approximation model, considering on the other hand other semi-empirical relations. For these reasons, a better attention must be paid on the LM flow and its heat transfer properties

3) up to now the CPS was analysed with a 0-dimensional approach, calculating for each point of the mesh only the average temperatures and extracting the total eroded mass flow rate at the end of the CPS volume. We should implement on the other hand a "flow1DFV" model which is not only discretized on the poloidal length (array of independent slices), but also on the toroidal direction, improving the information about the LM inside the CPS. Therefore, this solution can be useful also to consider a distributed sink of LM along each different toroidal volume. Indeed, in reality the LM doesn't obviously wait the symmetry axis (i.e. the end of each 0D volume) to exit the CPS after sputtering and evaporation events, but it leaves the W felt locally. With this upgrade the model should be more representative of the real behaviour of this technology

4) the CuCrZr layer was approximated to a slab but as we know from subsection 4.1.1, it has a diverse geometry and especially heat transfer areas with the water channel different from the simple base of the parallelepiped. It should be improved the mesh of this structure as well, refining the heat transfer process with the water

5) as a footnote, it may be changed the discretization of the "flow1DFV" of the water channels inside the units. So far, the built-in OpenModelica model was used but it is not that correct, because the whole length of the single channels is divided in NN equal sections, corresponding to each element of the CPS "flow1DFV" array (see 4.1.1). We know very well that the latter are not equal, instead are consistent with the input areas of Figure 3.6, which, multiplied by the constant CPS thickness (2 mm), define different mesh volumes. For this reason, the length of each discretized element of the "flow1DFV" model of the water channels must be equal to the poloidal lengthes of each CPS grid described by LatoPL[]

At the end of all these enhancements, being careful that the new model should be ok for both the LMs proposed so far – or maybe also for newer ones which are going to be investigated in the next future –, the model should be tested in different conditions. Obviously it should operate well in steady-state nominal conditions, but it should also be able to reproduce the system behavior during real disruption events because they are right the most critical point of the validation of this divertor technology. If this characteristic would result satisfactory, perhaps a real application of this design for the EU DEMO fusion reactor could be not only possible but even promising.

Appendix

Appendix A - Model Code

If the reader wants to run the simulations with the code implemented here, he should simply get a paid version of Dymola, open it, load the ThermoPower and MediaLMD libraries (given to me by kind concession of Antonio Froio and Giuseppe Francesco Nallo during the course of Nuclear Fusion Reactor Engineering, held by Professor Roberto Zanino), create a blank model or package (recommended because it keeps everything in a single file), load all the code below and simulate the "final_model". Thus, beware that the version reported here of the code refers to Sn as LM, half full reactor power as final steady state value and 20 % of nominal condition for the "disruption" trapezoid-like transient. If you would want to change the ramp or transient setting, you should simply change the parameters "PowFrac" and "TransFrac", whereas, for the LM it is more complicated:

1) Go to CPS model, comment `replaceable package Medium = MediaLMD.Media.Tin;` and comment out `replaceable package Medium = MediaLMD.Media.Lithium;` (at the beginning of the text) for the passage from Tin to Lithium. Vice versa for the opposite case

2) Stay in CPS model, comment `parameter Modelica.SIunits.SpecificEnergy LatHeat = 35.15e9 " J / kg *Sn*";` and comment out the one of Li or vice versa

3) Go on final_model and do the same as point 1)

4) Remain on final_model, comment the section "new Sn parameters" and remove from comment "Li parameters" or vice versa, being careful to insert the correct file paths

5) Stay on the final_model, comment out `final parameter Modelica.SIunits.Mass m_Li = MM / Na "kg/num.particelle";` and comment `final parameter Modelica.SIunits.Mass m_Sn = MM / Na "kg/num.particelle";` (or vice versa) and the same for the respective HK relations in the `equation` section

6) Go to the model VariabHeatTransferCoefficient and use for k_{LM} the semi-empirical correlations for Sn and Li if you want to change from constant to variable LM properties behaviour

CPS model

```
model cps
//    replaceable package Medium = MediaLMD.Media.Lithium;
    replaceable package Medium = MediaLMD.Media.Tin;
//    parameter Modelica.SIunits.SpecificEnergy LatHeat = 1.02e9 " J / kg *Li*";
    parameter Modelica.SIunits.SpecificEnergy LatHeat = 35.15e9 " J / kg *Sn*";
  parameter Integer nelems = 36;
  parameter Integer ii = 1 "this is the first element I consider from the array of SOLPS
data values";
  parameter Integer NN = 12 "this is the number of the elements of the array I consider f
or this unit";
  parameter Modelica.SIunits.Length Len = 39e-3 "+ 32e-
3 *2 equal to the distance the LM crosses along the cps. Len is only the effective length
 where more important heat transfer gradients intervene";
  parameter Modelica.SIunits.Length L_cps = 233e-
3 "length of cps along the poloidal direction";
  parameter Modelica.SIunits.MassFlowRate m_dot = 100 "trial mass flow rate value";
  parameter Modelica.SIunits.Length T_cps = 2e-3 "thickness of the CPS structure";
  parameter Modelica.SIunits.Area A_tot = T_cps * L_cps "in this way I consider the real
configuration of the first NN elements. To understand 2.275 see notes (GEOMETRY)";
  parameter Modelica.SIunits.Length omega_hyp[NN] = my_A / Len;
  parameter Modelica.SIunits.Area A_cross[NN] = T_cps * omega_hyp;
  parameter Modelica.SIunits.Diameter Diam[NN] = 2 * omega_hyp * T_cps ./ (omega_hyp + T_
cps * fill(1,NN));
  parameter Modelica.SIunits.Density rho0;
  parameter Modelica.SIunits.ThermalConductivity k0;
  parameter Modelica.SIunits.ThermalConductivity k_cps = 95;
  parameter Modelica.SIunits.ThermalConductivity k_CCZ = 347 "near 340";
  parameter Modelica.SIunits.SpecificHeatCapacity cm_CCZ = 420 "near 410";
  parameter Modelica.SIunits.Density rho_CCZ=8750   "near 8800";
  parameter Modelica.SIunits.Mass M_CCZ[NN] = rho_CCZ .* my_A * L_half_wall "Mass";
  parameter Modelica.SIunits.Length L_half_wall = 2e-3;
  parameter Modelica.SIunits.Length L_half_cps = T_cps / 2;
  parameter Modelica.SIunits.Pressure p_start;
  parameter Modelica.SIunits.Temperature T_start;
  parameter Modelica.SIunits.Temperature T_min(displayUnit = "K") = 506;
  constant Real pi = Modelica.Constants.pi;
  // let's take only the values we need
  parameter Modelica.SIunits.Length csi[nelems];
  parameter Modelica.SIunits.Area A_csi[nelems];
  parameter Modelica.SIunits.Area my_A[NN] = A_csi[ii:ii + NN - 1];
  parameter Modelica.SIunits.Length my_csi[NN] = csi[ii:ii + NN - 1];
  type MassFlowRatePerUnitArea = Real(unit = "kg/(m2.s)");
  ThermoPower.Gas.FlangeA infl(redeclare package Medium = Medium) annotation (
    Placement(visible = true, transformation(origin = {-90, 0}, extent = {{-10, -
10}, {10, 10}}, rotation = 0), iconTransformation(origin = {-100, 0}, extent = {{-20, -
20}, {20, 20}}, rotation = 0)));
  ThermoPower.Gas.FlangeB outfl(redeclare package Medium = Medium) annotation (
    Placement(visible = true, transformation(origin = {90, 0}, extent = {{-10, -
10}, {10, 10}}, rotation = 0), iconTransformation(origin = {100, 0}, extent = {{-20, -
20}, {20, 20}}, rotation = 0)));
  ThermoPower.Gas.Flow1DFV2w LatoPL[NN](redeclare each package Medium = Medium, each Nt =
 2,
    A=A_cross,each L = Len / 2, Dhyd = Diam, each QuasiStatic = true, each Tstartbar = T_
start, omega = omega_hyp, each pstart = p_start, each rhonom = rho0, each wnom = m_dot /
NN, redeclare
      model HeatTransfer =
        VariabHeatTransferCoefficient (each halfThick = L_half_cps, my_A = my_A), redecla
re
      model HeatTransfer2 =
        VariabHeatTransferCoefficient (each halfThick = L_half_cps, my_A = my_A)) annotat
ion (
    Placement(visible = true, transformation(origin = {2, -3}, extent = {{-18, -
17}, {18, 17}}, rotation = 0)));
```

```
  Modelica.Blocks.Sources.RealExpression PLheatFlow[NN](y = (HeatContrib - CoolEvContrib)
 .* my_A) annotation (
    Placement(visible = true, transformation(origin = {-32, -78}, extent = {{-
10, 10}, {10, -10}}, rotation = 0)));
  ThermoPower.Gas.SinkMassFlow sinkMassFlow[NN](redeclare each package Medium = Medium, e
ach T = T_start, each p0 = p_start, each use_in_w0 = true) annotation (
    Placement(visible = true, transformation(origin = {66, 44}, extent = {{-10, -
10}, {10, 10}}, rotation = 0)));
  Modelica.Blocks.Sources.RealExpression m_net_lost[NN](y = my_net) annotation (
    Placement(visible = true, transformation(origin = {76, 66}, extent = {{10, -10}, {-
10, 10}}, rotation = 0)));
  outer Modelica.SIunits.HeatFlux heatFlux[nelems];
  outer Modelica.SIunits.MassFlowRate m_net[nelems];
  outer Modelica.SIunits.MassFlowRate m_net_eva[nelems];
  outer Modelica.SIunits.MassFlowRate m_net_sput[nelems];
  Modelica.SIunits.HeatFlux HF[NN] = heatFlux[ii:ii + NN - 1];
  Modelica.SIunits.MassFlowRate my_net[NN] = m_net[ii:ii + NN - 1];
  Modelica.SIunits.MassFlowRate my_net_eva[NN] = m_net_eva[ii:ii + NN - 1];
  Modelica.SIunits.MassFlowRate my_net_sput[NN] = m_net_sput[ii:ii + NN - 1];
  Modelica.SIunits.MassFlowRate LMEvap = sum(my_net_eva);
  Modelica.SIunits.MassFlowRate LMSput = sum(my_net_sput);
  Modelica.SIunits.MassFlowRate LMErodTot = sum(my_net);
  Modelica.SIunits.HeatFlux HeatContrib[NN] = HF * 1e6;
  Modelica.SIunits.HeatFlux CoolEvContrib[NN] = my_net_eva * LatHeat ./ my_A;
  Modelica.SIunits.HeatFlux HeatCoolDiff[NN] = HeatContrib - CoolEvContrib;
  Real DerHeatCoolDiff[NN] = der(HeatCoolDiff);
  Modelica.SIunits.DimensionlessRatio EvapToTotHeatContrib = if min(HeatContrib) > 0 then
 sum(CoolEvContrib ./ HeatContrib) else 0;
  Modelica.SIunits.DimensionlessRatio EvapToTotHeatContribSingle[NN] = if min(HeatContrib
) > 0 then CoolEvContrib ./ HeatContrib else fill(0,NN);

  ThermoPower.Thermal.DHTVolumes wall(N = NN) annotation (
    Placement(visible = true, transformation(extent = {{-
60, 80}, {60, 100}}, rotation = 0), iconTransformation(extent = {{-
60, 40}, {60, 60}}, rotation = 0)));
  ThermoPower.Thermal.MetalWallFV CuCrZr[NN](
    M=M_CCZ,
    cm=cm_CCZ,
    each WallRes=true,
    UA_ext=k_CCZ*my_A/L_half_wall,
    each Tstartbar=T_start)
    annotation (Placement(transformation(extent={{-14,46},{16,16}})));

  ThermoPower.Gas.Header header(redeclare package Medium = Medium, Tmstart = T_start, V =
 A_tot * Len * 100, S = A_tot * 10, pstart = p_start, Tstart = T_start, noInitialPressure
 = true, noInitialTemperature = false) annotation (
    Placement(transformation(extent = {{-56, 10}, {-36, 30}})));
  Converter converter(Nw = NN) annotation (
    Placement(transformation(extent={{-12,72},{10,50}})));
  Modelica.Blocks.Math.Sum QinTotal(nin = NN) annotation (
    Placement(transformation(extent = {{28, -88}, {48, -68}})));
  ThermoPower.Thermal.HeatSource1DFV heatSource1DFV[NN] annotation (
    Placement(transformation(extent = {{-12, -50}, {16, -78}})));
equation
  connect(infl, outfl) annotation (
    Line(points = {{-90, 0}, {-68, 0}, {-68, -92}, {70, -
92}, {70, 0}, {90, 0}}, color = {159, 159, 223}));
  for i in 1:NN loop
    connect(LatoPL[i].infl, header.outlet);
  end for;
  connect(PLheatFlow.y, QinTotal.u) annotation (
    Line(points = {{-21, -78}, {26, -78}}, color = {0, 0, 127}));
  connect(PLheatFlow.y, heatSource1DFV.power) annotation (
    Line(points = {{-21, -78}, {2, -78}, {2, -69.6}}, color = {0, 0, 127}));
  connect(sinkMassFlow.in_w0, m_net_lost.y) annotation (
    Line(points = {{60, 49}, {60, 64}, {65, 64}, {65, 66}}, color = {0, 0, 127}));
  connect(LatoPL.outfl, sinkMassFlow.flange) annotation (
    Line(points = {{20, -3}, {30, -3}, {30, 44}, {56, 44}}, color = {159, 159, 223}));
```

```
  connect(header.inlet, infl) annotation (
    Line(points = {{-56, 20}, {-68, 20}, {-68, 0}, {-90, 0}}, color = {159, 159, 223}));
  connect(heatSource1DFV.wall, LatoPL.wall2) annotation (
    Line(points = {{2, -59.8}, {2, -11.5}}, color = {255, 127, 0}));
  connect(CuCrZr.int, LatoPL.wall) annotation (Line(points={{1,26.5},{1,16.25},{
          2,16.25},{2,5.5}}, color={255,127,0}));
  connect(CuCrZr.ext, converter.Arr) annotation (Line(points={{1,35.65},{1,57.7},
          {-1,57.7}}, color={255,127,0}));
  connect(converter.nonArr, wall) annotation (Line(points={{-1,64.41},{-1,74.205},
          {0,74.205},{0,90}}, color={255,127,0}));
  annotation (
    Line(points = {{-4, -3.8}, {0, -3.8}, {0, 53.8}}, color = {255, 127, 0}),
    Icon(graphics={  Rectangle(fillColor = {159, 159, 223}, fillPattern = FillPattern.Hor
izontalCylinder, extent = {{-80, 40}, {80, -40}}), Text(extent = {{-100, -40}, {100, -
80}}, textString = "%name")}));
end cps;
```

Converter

```
model Converter "Generic metal wall model with Nw finite volumes"
  extends ThermoPower.Icons.MetalWall;
  parameter Integer Nw = 1 "Number of volumes on the wall ports";
  ThermoPower.Thermal.DHTVolumes Arr[Nw](each N = 1) "Internal surface" annotation (
    Placement(transformation(extent = {{-40, 20}, {40, 40}}, rotation = 0)));
  ThermoPower.Thermal.DHTVolumes nonArr(final N = Nw) "External surface" annotation (
    Placement(transformation(extent = {{-40, -42}, {40, -20}}, rotation = 0)));
equation
  for i in 1:Nw loop
    0 = Arr[i].Q[1] + nonArr.Q[i] "Energy balance";
    Arr[i].T[1] = nonArr.T[i];
  end for;
end Converter;
```

Variable Heat Transfer Coefficient (Heat Transfer model)

```
model VariabHeatTransferCoefficient "Variable heat transfer coefficient"
  extends ThermoPower.Thermal.BaseClasses.DistributedHeatTransferFV;
  Modelica.SIunits.ThermalConductance UA_CPS[Nw] "conductance of the cps (heat transfer c
oefficient)";
  Medium.ThermalConductivity k[Nw] "Thermal conductivity";
  Medium.ThermalConductivity k_LM[Nw] "Thermal conductivity of Tin";
  Medium.ThermalConductivity k_w[Nw] "Thermal conductivity of W";
  parameter Boolean adaptiveAverageTemperature = true "Adapt the average temperature at l
ow flow rates";
  parameter Modelica.SIunits.PerUnit sigma = 0.1 "Fraction of nominal flow rate below whi
ch the heat transfer is computed on outlet volume temperatures";
  Modelica.SIunits.PerUnit w_nom "Ratio between actual and nominal flow rate";
  Medium.Temperature Tvol[Nw] "Fluid temperature in the volumes";
  parameter Modelica.SIunits.Length halfThick = 2.275e-3 / 2;
  parameter Modelica.SIunits.Area my_A;
  parameter Real f = 0.5;
equation
  assert(Nw == Nf - 1, "The number of volumes Nw on wall side should be equal to number o
f volumes fluid side Nf - 1");
  w_nom = abs(w[1]) / wnom;
  for j in 1:Nw loop
    k_LM[j] = Medium.thermalConductivity(fluidState[j]);
//    k_LM[j] = 13.90 + 0.02868*Tvol[j] "*Sn* semi-empirical correlations";
//    k_LM[j] = 40.1328 + 0.01908*Tvol[j] "*Li* semi-empirical correlations";
    k_w[j] = 174.9274 - 0.1067 * Tvol[j] + 5.0067 * 1e-5 * Tvol[j] ^ 2 - 7.8349 * 1e-
9 * Tvol[j] ^ 3;
    k[j] = f * k_w[j] + (1 - f) * k_LM[j];
    UA_CPS[j] = k[j] / halfThick * my_A;
    Tvol[j] = if not useAverageTemperature then T[j + 1] else if not adaptiveAverageTempe
rature then (T[j] + T[j + 1]) / 2 else (T[j] + T[j + 1]) / 2 + (T[j + 1] - T[j]) / 2 * ex
p(-w_nom / sigma);
    Qw[j] = (Tw[j] - Tvol[j]) * UA_CPS[j];
  end for;
end VariabHeatTransferCoefficient;
```

He circuit

```
model Helium_circuit
  replaceable package Medium = Modelica.Media.IdealGases.SingleGases.He;
  constant Real pi = Modelica.Constants.pi;
  parameter Modelica.SIunits.Diameter D = sqrt(4*A_cross/pi);
  parameter Modelica.SIunits.Diameter D_HX = D * 1e1;
  final parameter Modelica.SIunits.Area A_cross = 12e-6;
  final parameter Modelica.SIunits.Area A_cross_HX = pi * D_HX ^ 2 / 4;
  final parameter Modelica.SIunits.Area A_lat = 2 * pi * D * L;
  final parameter Modelica.SIunits.Volume V_HX = A_cross_HX * L;
  parameter Modelica.SIunits.Length L = 233e-3;
  parameter Modelica.SIunits.Length Omega = pi * D;
  parameter Modelica.SIunits.Length Omega_HX = pi * D_HX;
  parameter Modelica.SIunits.MassFlowRate w0 = 10;
  parameter Modelica.SIunits.Density rho0 = 4;
  parameter Modelica.SIunits.Pressure p_start = 5000000;
  parameter Modelica.SIunits.Temp_K T_start = 350 + 273.15;
  ThermoPower.Gas.Flow1DFV LM(A = A_cross, Dhyd = D, FFtype = ThermoPower.Choices.Flow1D.
FFtypes.NoFriction, L = L, redeclare
      package Medium = Medium, initOpt = ThermoPower.Choices.Init.Options.noInit, noIniti
alPressure = true, redeclare
      model HeatTransfer =
        ThermoPower.Thermal.HeatTransferFV.DittusBoelter (Re_min = 1000), Tstartbar = T_s
tart, dpnom(displayUnit = "Pa"), omega = Omega, pstart = p_start, rhonom = rho0, wnom = w
0) annotation (
    Placement(visible = true, transformation(origin={-18,52},  extent = {{-10, -
10}, {10, 10}}, rotation = 0)));
  ThermoPower.Thermal.DHTVolumes wall(N = 1) annotation (
    Placement(visible = true, transformation(extent = {{-
60, 80}, {60, 100}}, rotation = 0), iconTransformation(extent = {{-
60, 80}, {60, 100}}, rotation = 0)));
  ThermoPower.Gas.SinkMassFlow sink(redeclare package Medium = Medium, T = T_start, p0 =
p_start, use_in_w0 = true) annotation (
    Placement(visible = true, transformation(origin={84,52},    extent = {{-
10, 10}, {10, -10}}, rotation = 0)));
  ThermoPower.Gas.SourcePressure source(redeclare package Medium = Medium, T = T_start, p
0(displayUnit = "Pa"), use_in_p0 = true) annotation (
    Placement(visible = true, transformation(origin = {-52, 52}, extent = {{-10, -
10}, {10, 10}}, rotation = 0)));
  Modelica.Blocks.Sources.RealExpression pstart(y = p_start) annotation (
    Placement(visible = true, transformation(origin = {-86, 68}, extent = {{-10, -
10}, {10, 10}}, rotation = 0)));
  Modelica.Blocks.Sources.RealExpression wstart(y = w0) annotation (
    Placement(visible = true, transformation(origin={64,20},    extent = {{-10, -
10}, {10, 10}}, rotation = 0)));
  ThermoPower.Gas.Header header(
    redeclare package Medium = Medium,
    Tmstart=T_start,
    V=V_HX,
    S=A_cross_HX,
    pstart=p_start,
    Tstart=T_start,
    noInitialPressure=true)
    annotation (Placement(transformation(extent={{4,42},{24,62}})));
  ThermoPower.Gas.PressDrop pressDrop(
    redeclare package Medium = Medium,
    wnom=w0,
    dpnom(displayUnit="Pa") = 10,
    FFtype=ThermoPower.Choices.PressDrop.FFtypes.Kf,
    K=1e-3,
    A=A_cross,
    Kfc=1e-2,
    pstart=p_start,
    Tstart=T_start)
            annotation (Placement(transformation(extent={{40,42},{60,62}})));
```

```
equation
  connect(LM.wall, wall) annotation (
    Line(points={{-18,57},{-18,74},{0,74},{0,90}},
                                    color = {255, 127, 0}));
  connect(wstart.y, sink.in_w0) annotation (
    Line(points={{75,20},{78,20},{78,47}},        color = {0, 0, 127}));
  connect(pstart.y, source.in_p0) annotation (
    Line(points = {{-75, 68}, {-58, 68}, {-58, 58.4}}, color = {0, 0, 127}));
  connect(source.flange, LM.infl) annotation (
    Line(points={{-42,52},{-28,52}},        color = {159, 159, 223}));
  connect(LM.outfl, header.inlet)
    annotation (Line(points={{-8,52},{4,52}}, color={159,159,223}));
  connect(header.outlet, pressDrop.inlet)
    annotation (Line(points={{24,52},{40,52}}, color={159,159,223}));
  connect(sink.flange, pressDrop.outlet)
    annotation (Line(points={{74,52},{60,52}}, color={159,159,223}));
  annotation (
    Icon(graphics={  Text(extent = {{-100, -60}, {100, -
100}}, textString = "%name"), Rectangle(origin = {1.42109e-
14, 10}, fillColor = {159, 159, 223},
            fillPattern =
                                                            FillPattern
.HorizontalCylinder, extent = {{-100, 70}, {100, -70}})}));
end Helium_circuit;
```

Water circuit

```
model Water_circuit_ud
  replaceable package WaterMedium = Modelica.Media.Water.StandardWater;
  constant Real pi = Modelica.Constants.pi;
  parameter Integer nmods = 3 * 8 * 54;
  parameter Integer Nt;
  parameter Real Cf = 1e-1;
  parameter Modelica.SIunits.Time t_start = 5;
  parameter Modelica.SIunits.Time t_ramp = 60;
  parameter Modelica.SIunits.Diameter D = 8e-3;
  final parameter Modelica.SIunits.Area A_cross = pi * D ^ 2 / 4;
  parameter Modelica.SIunits.Length L = 233e-3;
  final parameter Modelica.SIunits.Volume Vol = A_cross * L;
  parameter Modelica.SIunits.Length Omega = pi * D;
  parameter Modelica.SIunits.MassFlowRate w0 = m_start_w;
  parameter Modelica.SIunits.MassFlowRate m_start_w = 55.8e1;
  parameter Modelica.SIunits.MassFlowRate m_fin_w = 55.8e1;
  parameter Modelica.SIunits.Density rho0w=1000;
  parameter Modelica.SIunits.Pressure p_start_w=5000000;
  parameter Modelica.SIunits.SpecificEnthalpy h_start = 1e6;
  parameter Modelica.SIunits.Temp_K T_start_w=506    "equal to the min Temp before the cha
nging of phase solid liquid of Sn";
  parameter Modelica.SIunits.Temp_K T_min=506;
  parameter Integer NNA = 2 "number of nodes";
  parameter Integer NwA = NNA - 1;
  parameter Integer NNB = 2 "number of nodes";
  parameter Integer NwB = NNB - 1;
  parameter Integer NNC = 2 "number of nodes";
  parameter Integer NwC = NNC - 1;

  ThermoPower.Thermal.DHTVolumes wallA(N = NwA) annotation (
    Placement(visible = true, transformation(origin = {-60, 90}, extent = {{-20, -
10}, {20, 10}}, rotation = 0), iconTransformation(origin = {-86.6667, 40}, extent = {{-
40, -6.66667}, {40, 6.66667}}, rotation = 90)));
  ThermoPower.Thermal.DHTVolumes wallB(N = NwB) annotation (
    Placement(visible = true, transformation(origin = {0, 90}, extent = {{-20, -
10}, {20, 10}}, rotation = 0), iconTransformation(origin = {-
0.333306, 90.8333}, extent = {{-59, -9.83334}, {59, 9.83334}}, rotation = 180)));
  ThermoPower.Thermal.DHTVolumes wallC(N = NwC) annotation (
    Placement(visible = true, transformation(origin = {60, 90}, extent = {{-20, -
10}, {20, 10}}, rotation = 0), iconTransformation(origin = {86.6667, 40}, extent = {{-
40, -6.66667}, {40, 6.66667}}, rotation = 90)));
  ThermoPower.Water.SourceMassFlow source(redeclare package Medium =
        WaterMedium,                                                      T = T_s
tart_w, h = h_start, p0 = p_start_w, use_T = true, use_in_w0 = true) annotation (
    Placement(visible = true, transformation(origin = {68, -6}, extent = {{-
10, 10}, {10, -10}}, rotation = 0)));
  ThermoPower.Water.SinkPressure sink(redeclare package Medium = WaterMedium, use_T = tru
e, T = T_start_w, h = h_start, p0 = p_start_w) annotation (
    Placement(visible = true, transformation(origin={8,-10},     extent = {{-10, -
10}, {10, 10}}, rotation = 0)));
  ThermoPower.Water.Flow1DFV unitB_w(A = A_cross, Dhyd = D,
                      H = -L,
    Cfnom=Cf,
    HydraulicCapacitance=ThermoPower.Choices.Flow1D.HCtypes.Downstream,

            L = L, redeclare
      package Medium =
        WaterMedium,

                                    N = NNB, Nt = Nt, Nw = NwB,
    dpnom(displayUnit="Pa"),

                                        hstartin = h_start, hstartout = h_start
```

```modelica
, noInitialPressure = true, omega = Omega, pstart = p_start_w, rhonom = rho0w, wnom = w0,
 fixedMassFlowSimplified = true, redeclare
      model HeatTransfer =
        ThermoPower.Thermal.HeatTransferFV.DittusBoelter (

                                                                    redeclare

          package Medium =
            WaterMedium,

                                              Re_min = 1000)) annotation (
    Placement(visible = true, transformation(origin = {0, 28}, extent = {{12, -12}, {-
12, 12}}, rotation = 0)));
  ThermoPower.Water.Flow1DFV unitC_w(A = A_cross, Dhyd = D,
                          H = -L,
    Cfnom=Cf,
    HydraulicCapacitance=ThermoPower.Choices.Flow1D.HCtypes.Downstream,

            L = L, redeclare
     package Medium =
       WaterMedium,

                                      N = NNC, Nt = Nt, Nw = NwC,
    dpnom(displayUnit="Pa"),

                                                hstartin = h_start, hstartout = h_start
, noInitialPressure = true, omega = Omega, pstart = p_start_w, rhonom = rho0w, wnom = w0,
 fixedMassFlowSimplified = true, redeclare
      model HeatTransfer =
        ThermoPower.Thermal.HeatTransferFV.DittusBoelter (

                                                                    redeclare

          package Medium =
            WaterMedium,

                                              Re_min = 1000)) annotation (
    Placement(visible = true, transformation(origin = {62, 28}, extent = {{12, -12}, {-
12, 12}}, rotation = 0)));
  ThermoPower.Water.Flow1DFV unitA_w(A = A_cross, Dhyd = D,
                          H = -L,
    Cfnom=Cf,
    HydraulicCapacitance=ThermoPower.Choices.Flow1D.HCtypes.Downstream,

            L = L, redeclare
     package Medium =
       WaterMedium,

                                      N = NNA, Nt = Nt, Nw = NwA,
    dpnom(displayUnit="Pa"),

                                                hstartin = h_start, hstartout = h_start
, noInitialPressure = true, omega = Omega, pstart = p_start_w, rhonom = rho0w, wnom = w0,
 fixedMassFlowSimplified = true, redeclare
      model HeatTransfer =
        ThermoPower.Thermal.HeatTransferFV.DittusBoelter (

                                                                    redeclare

          package Medium =
            WaterMedium,

                                              Re_min = 1000)) annotation (
    Placement(visible = true, transformation(origin = {-56, 28}, extent = {{12, -12}, {-
12, 12}}, rotation = 0)));
  Modelica.Blocks.Sources.Ramp ramp(height = m_fin_w, duration = t_ramp, offset = m_start
_w, startTime = t_start) annotation (
    Placement(transformation(extent = {{92, -40}, {72, -20}})));
```

```
  ThermoPower.Water.Header header(
    redeclare package Medium = WaterMedium,
    V=A_cross*L*3e2,
    S=A_cross*1e2,
    H=3*L,
    pstart=p_start_w,
    hstart=h_start,
    Tmstart=T_start_w,
    noInitialPressure=true)
    annotation (Placement(transformation(extent={{-74,-20},{-54,0}})));

equation

  connect(unitA_w.wall, wallA) annotation (
    Line(points = {{-56, 34}, {-58, 34}, {-58, 90}, {-60, 90}}, color = {255, 127, 0}));
  connect(unitB_w.wall, wallB) annotation (
    Line(points = {{0, 34}, {0, 90}}, color = {255, 127, 0}));
  connect(unitC_w.wall, wallC) annotation (
    Line(points = {{62, 34}, {62, 84}, {60, 84}, {60, 90}}, color = {255, 127, 0}));
  connect(ramp.y, source.in_w0) annotation (
    Line(points = {{71, -30}, {64, -30}, {64, -11.6}}, color = {0, 0, 127}));
  connect(unitC_w.infl, source.flange) annotation (
    Line(points = {{74, 28}, {82, 28}, {82, -6}, {78, -6}}, color = {0, 0, 255}));
  connect(unitB_w.infl, unitC_w.outfl) annotation (
    Line(points = {{12, 28}, {50, 28}}, color = {0, 0, 255}));
  connect(unitB_w.outfl, unitA_w.infl) annotation (
    Line(points = {{-12, 28}, {-20, 28}, {-20, 28}, {-44, 28}}, color = {0, 0, 255}));
  connect(unitA_w.outfl, header.inlet) annotation (Line(points={{-68,28},{-90,28},
          {-90,-10},{-74.1,-10}}, color={0,0,255}));
  connect(header.outlet, sink.flange)
    annotation (Line(points={{-54,-10},{-2,-10}}, color={0,0,255}));
  annotation (
    Icon(graphics={  Rectangle(extent = {{-80, 80}, {80, 0}}, lineColor = {0, 0, 0},
            fillPattern =
          FillPattern.HorizontalCylinder, fillColor = {159, 159, 223})}));
end Water_circuit_ud;
```

Coupling model

```
model final_model
      replaceable package Medium = MediaLMD.Media.Tin;
//      replaceable package Medium = MediaLMD.Media.Lithium;
   final parameter Modelica.SIunits.Mass m_Sn = MM / Na "kg/num.particelle";
//   final parameter Modelica.SIunits.Mass m_Li = MM / Na "kg/num.particelle";

    // carichiamo tutti gli input che ci servono
    //   Sn parameters
    parameter Modelica.SIunits.Area A_tot[nelems] = vector(Modelica.Utilities.Streams.r
eadRealMatrix("Z:/desktop Remoto/LMdata_20210509/Sn/areaTarget.mat", "areaTarget", 1, nel
ems)) /nmods "";
    // parameter Modelica.SIunits.Area A_csi[nelems] = A_tot * 35 / sum(A_tot) "questa
è la correzione sull'area che serviva prima quando i dati erano totalmente sballati";
    parameter Modelica.SIunits.Length csi[nelems] = vector(Modelica.Utilities.Streams.r
eadRealMatrix("Z:/desktop Remoto/LMdata_20210509/Sn/csiTarget.mat", "csiTarget", 1, nelem
s));
    parameter Modelica.SIunits.DimensionlessRatio f_red[nelems] = vector(Modelica.Utili
ties.Streams.readRealMatrix("Z:/desktop Remoto/LMdata_20210509/Sn/promptRedeposition.mat"
, "promptRedeposition", 1, nelems));
    parameter MassFlowRatePerUnitArea m_sput[nelems] = m_Sn * vector(Modelica.Utilities
.Streams.readRealMatrix("Z:/desktop Remoto/LMdata_20210509/Sn/sputterPhysical.mat", "sput
terPhysical", 1, nelems)) "m_Sn * ... because here sputterPhysical is in particles/(m2.s)
";
    parameter Modelica.SIunits.HeatFlux q_s[nelems] = vector(Modelica.Utilities.Streams
.readRealMatrix("Z:/desktop Remoto/LMdata_20210509/Sn/heatFluxTargetOutb.mat", "heatFluxT
argetOutb", 1, nelems)) "(* 1400 / 1010: previous correction) here I load the values of h
eat flux from data to our variable q_s";
    Modelica.Blocks.Tables.CombiTable1D interpolTemp(columns = fill(2, nelems), fileNam
e = "Z:/desktop Remoto/LMdata_20210509/Sn/surfTempFromHeatFlux.mat", tableName = "surfTem
pFromHeatFlux", tableOnFile = true) annotation (
      Placement(visible = true, transformation(origin={38,82},    extent = {{-8, -
8}, {8, 8}}, rotation = 0)));
/*
  // Li param
  parameter Modelica.SIunits.Area A_tot[nelems] = vector(Modelica.Utilities.Streams.readR
ealMatrix("Z:/desktop Remoto/LMdata_20210509/Li/areaTarget.mat", "areaTarget", 1, nelems)
) / nmods "";
  parameter Modelica.SIunits.Length csi[nelems] = vector(Modelica.Utilities.Streams.readR
ealMatrix("Z:/desktop Remoto/LMdata_20210509/Li/csiTarget.mat", "csiTarget", 1, nelems));
  parameter Modelica.SIunits.DimensionlessRatio f_red[nelems] = vector(Modelica.Utilities
.Streams.readRealMatrix("Z:/desktop Remoto/LMdata_20210509/Li/promptRedeposition.mat", "p
romptRedeposition", 1, nelems));
  parameter MassFlowRatePerUnitArea m_sput[nelems] = m_Li * vector(Modelica.Utilities.Str
eams.readRealMatrix("Z:/desktop Remoto/LMdata_20210509/Li/sputterPhysical.mat", "sputterP
hysical", 1, nelems)) "m_Li * ... because here sputterPhysical is in particles/(m2.s)";
  parameter Modelica.SIunits.HeatFlux q_s[nelems] = vector(Modelica.Utilities.Streams.rea
dRealMatrix("Z:/desktop Remoto/LMdata_20210509/Li/heatFluxTargetOutb.mat", "heatFluxTarge
tOutb", 1, nelems)) "here I load the values of heat flux from data to our variable q_s";
  Modelica.Blocks.Tables.CombiTable1D interpolTemp(columns = fill(2, nelems), fileName =
"Z:/desktop Remoto/LMdata_20210509/Li/surfTempFromHeatFlux.mat", tableName = "surfTempFro
mHeatFlux", tableOnFile = true) annotation (
    Placement(visible = true, transformation(origin = {38, 82}, extent = {{-8, -
8}, {8, 8}}, rotation = 0)));
  */
  constant Real pi = Modelica.Constants.pi;
  parameter Integer Nt = 3;
  parameter Modelica.SIunits.Length L = 233e-3;
  parameter Modelica.SIunits.MassFlowRate w0 = 0.495 * Nt;
  parameter Modelica.SIunits.Density rho0w(displayUnit = "kg/m3") = 1000;
  parameter Modelica.SIunits.Pressure p_start_w=5000000;
  parameter Modelica.SIunits.SpecificEnthalpy h_start = 1e6;
  parameter Modelica.SIunits.Temp_K T_start_w = Medium.T_min + 1 "equal to the min Temp b
efore the changing of phase from liquid to solid of LM";
  parameter Integer nelems = 36;
```

```
  parameter Integer ii = 1;
  parameter Integer NN = 36;
  parameter Integer iiA = 1;
  parameter Integer NNA = 8 "number of nodes";
  parameter Integer NwA = NNA - 1;
  parameter Integer iiB = 9;
  parameter Integer NNB = 18 "number of nodes";
  parameter Integer NwB = NNB - 1;
  parameter Integer iiC = 27;
  parameter Integer NNC = 6 "number of nodes";
  parameter Integer NwC = NNC - 1;
  parameter Integer nmods = 3 * 8 * 54;
  final parameter Modelica.SIunits.MolarMass MM = Medium.MM_const;
  constant Real Na = Modelica.Constants.N_A;
  constant Modelica.SIunits.DimensionlessRatio eta = 1.66;
  constant Real k_b = Modelica.Constants.k;
  parameter Modelica.SIunits.Density rho0 = Medium.tableDensity[1, 2];
  parameter Modelica.SIunits.ThermalConductivity k0 = Medium.tableConductivity[2, 2];
  parameter Modelica.SIunits.Length Len = 102.7e-3;
  parameter Modelica.SIunits.Length L_half_wall=1.8e-3;
  parameter Modelica.SIunits.MassFlowRate m_dot = 22;
  final parameter Modelica.SIunits.Volume Vol = Area * Len;
   parameter Modelica.SIunits.Length omega_hyp = pi * Diam;
   parameter Modelica.SIunits.Diameter Diam = sqrt(4 * Area / pi);
  parameter Modelica.SIunits.Area Area = 530.1e-6;
  parameter Modelica.SIunits.Pressure p_start=60000;
  parameter Modelica.SIunits.Temperature T_start=350 + 273.15;
  parameter Modelica.SIunits.Time t_start = 1e2;
  parameter Modelica.SIunits.Time t_ramp = 4e2;
  parameter Modelica.SIunits.Time t_disr = 1e-3;
  parameter Modelica.SIunits.Time t_startrans = t_start + t_ramp + 20;
  parameter Modelica.SIunits.DimensionlessRatio PowFrac=0.5;
  parameter Modelica.SIunits.DimensionlessRatio TransFrac=0.2;
   type MassFlowRatePerUnitArea = Real(unit = "kg/(m2.s)");
   inner ThermoPower.System system(initOpt = ThermoPower.Choices.Init.Options.fixedState,
T_amb(displayUnit = "K"), T_wb(displayUnit = "K"), p_amb(displayUnit = "Pa")) annotation
(
    Placement(visible = true, transformation(origin = {-12, -78}, extent = {{-10, -
10}, {10, 10}}, rotation = 0)));

      ThermoPower.Gas.Flow1DFV HeHX(redeclare package Medium = Medium, A = Area, Dhyd =
Diam,
       L=L,
       HydraulicCapacitance=ThermoPower.Choices.Flow1D.HCtypes.Upstream,
            Tstartbar = T_start, omega = omega_hyp, pstart = p_start, rhonom = rho0, w
nom = m_dot, QuasiStatic = true,
       redeclare model HeatTransfer =
          ThermoPower.Thermal.HeatTransferFV.DittusBoelter (Re_min=1000))

                             annotation (
        Placement(visible = true, transformation(origin={-58,6},   extent = {{-
10, 10}, {10, -10}}, rotation = 0)));
      Helium_circuit He annotation (
        Placement(visible = true, transformation(origin={-60,-36},   extent = {{-10, -
10}, {10, 10}}, rotation = 0)));

  cps unitA_LM(m_dot = 1e-3, k0 = k0,
    L_half_wall=L_half_wall,          A_csi = A_tot, NN = NwA, T_start = T_start, csi = c
si, ii = iiA, p_start = p_start, rho0 = rho0) annotation (
    Placement(visible = true, transformation(origin={-24,6},     extent = {{-
14, 14}, {14, -14}}, rotation = 0)));
  cps unitB_LM(m_dot = 1e-3, k0 = k0,
    L_half_wall=L_half_wall,          A_csi = A_tot, NN = NwB, T_start = T_start, csi = c
si, ii = iiB, p_start = p_start, rho0 = rho0) annotation (
    Placement(visible = true, transformation(origin={18,6},     extent = {{-
14, 14}, {14, -14}}, rotation = 0)));
  cps unitC_LM(m_dot = 1e-3, k0 = k0,
```

```
    L_half_wall=L_half_wall,              A_csi = A_tot, NN = NwC, T_start = T_start, csi = c
si, ii = iiC, p_start = p_start, rho0 = rho0) annotation (
    Placement(visible = true, transformation(origin={58,6},      extent = {{-
14, 14}, {14, -14}}, rotation = 0)));
  Modelica.SIunits.Temperature T_s[nelems] "(start = fill(Medium.T_min + 1, nelems)) afte
r having interpolated the values in the block interpolTemp I assign the output to T_s whi
ch will be the surface temperature I need for the calculations";
  Modelica.SIunits.Pressure p_sat[nelems];
  Modelica.SIunits.MassFlowRate m_eva[nelems];
  parameter Modelica.SIunits.DimensionlessRatio k[NN] = fill(1, NN) - f_red;
  inner Modelica.SIunits.HeatFlux heatFlux[nelems] = ramp_q.y;
  inner Modelica.SIunits.MassFlowRate m_net[nelems];
  inner Modelica.SIunits.MassFlowRate m_net_eva[nelems];
  inner Modelica.SIunits.MassFlowRate m_net_sput[nelems];

  Water_circuit_ud Water(Nt = Nt, t_start = t_start * 1e-
1, t_ramp = t_start * 0.5, m_start_w = w0 * 1e-
1, m_fin_w = w0 * 1e0, NNA = NNA, NNB = NNB, NNC = NNC, NwA = NwA, NwB = NwB, NwC = NwC,
T_start_w = T_start_w, h_start = h_start, p_start_w = p_start_w, rho0w = rho0w, w0 = w0)
annotation (
    Placement(visible = true, transformation(origin={21,-35},     extent = {{-15, -
15}, {15, 15}}, rotation = 0)));
    ThermoPower.Gas.PressDrop pressDrop( A = Area, FFtype = ThermoPower.Choices.PressDrop
.FFtypes.Kinetic,
    K=0.01,
                        redeclare
      package Medium =
                                            Medium, Tstart = T_start, dpnom( dis
playUnit = "Pa") = 100, pstart = p_start, wnom = m_dot) annotation (
      Placement(visible = true, transformation(origin={76,22},     extent = {{-10, -
10}, {10, 10}}, rotation = 90)));
  ThermoPower.Gas.SourcePressure sourcePressure(redeclare package Medium = Medium, T = T_
start, p0 = p_start) annotation (
    Placement(transformation(extent = {{-100, -4}, {-80, 16}})));
  ThermoPower.Gas.SinkMassFlow sinkMassFlow(redeclare package Medium = Medium, T = T_star
t, p0 = p_start, w0 = m_dot) annotation (
    Placement(transformation(extent = {{80, 36}, {100, 56}})));

  myPowerTransFcn ramp_sput[nelems](
    height=m_sput .* A_tot*PowFrac,                              duration = t_ramp,
    amplitude=m_sput .* A_tot*TransFrac,
                            startTrans = t_startrans, rising = t_disr, width = t_dis
r, falling = t_disr, period = t_startrans, nperiod = 1, startTime = t_start) annotation (
    Placement(transformation(extent = {{-84, 72}, {-64, 92}})));
  myPowerTransFcn ramp_q[nelems](
    height=q_s*1e-6*PowFrac,                            duration = t_ramp,
    amplitude=q_s*1e-
6*TransFrac,                                                                          s
tartTrans = t_startrans, rising = t_disr, width = t_disr, falling = t_disr, period = t_st
artrans, nperiod = 1, startTime = t_start) annotation (
    Placement(transformation(extent = {{-8, 72}, {12, 92}})));
equation
  T_s = Modelica.SIunits.Conversions.from_degC(interpolTemp.y);
  m_net = m_net_eva + m_net_sput;
  m_net_eva = m_eva .* k;
  m_net_sput = ramp_sput.y .* k;
  m_eva = A_tot * eta * m_Sn .* p_sat ./ sqrt(2 * pi * m_Sn * k_b .* T_s) "HK relation";
//   m_eva = A_tot * eta * m_Li .* p_sat ./ sqrt(2 * pi * m_Li * k_b .* T_s) "HK
relation";
  p_sat = Medium.saturationPressure(Medium.setState_pT(fill(p_start, nelems), T_s));
  connect(unitC_LM.infl, unitB_LM.outfl) annotation (
    Line(points={{44,6},{32,6}},        color = {159, 159, 223}));
  connect(unitB_LM.infl, unitA_LM.outfl) annotation (
    Line(points={{4,6},{-10,6}},        color = {159, 159, 223}));
  connect(unitA_LM.wall, Water.wallA) annotation (
    Line(points={{-24,-1},{-24,-29},{8,-29}},     color = {255, 127, 0}));
  connect(unitB_LM.wall, Water.wallB) annotation (
    Line(points={{18,-1},{18,-21.375},{20.95,-21.375}},        color = {255, 127, 0}));
```

```
  connect(unitC_LM.wall, Water.wallC) annotation (
    Line(points={{58,-1},{58,-29},{34,-29}},          color = {255, 127, 0}));
  connect(ramp_q.y, interpolTemp.u) annotation (
    Line(points = {{13, 82}, {28.4, 82}}, color = {0, 0, 127}));
  connect(sourcePressure.flange, HeHX.infl) annotation (Line(points={{-80,6},{-76,
          6},{-76,4},{-68,4},{-68,6}}, color={159,159,223}));
  connect(He.wall, HeHX.wall) annotation (Line(points={{-60,-27},{-60,-4},{-58,-4},
          {-58,1}}, color={255,127,0}));
  connect(HeHX.outfl, unitA_LM.infl)
    annotation (Line(points={{-48,6},{-38,6}}, color={159,159,223}));
  connect(sinkMassFlow.flange, pressDrop.outlet)
    annotation (Line(points={{80,46},{76,46},{76,32}}, color={159,159,223}));
  connect(unitC_LM.outfl, pressDrop.inlet)
    annotation (Line(points={{72,6},{76,6},{76,12}}, color={159,159,223}));
  annotation (
    Icon(coordinateSystem(extent = {{-100, -
100}, {120, 100}}), graphics={  Text(extent = {{-100, -60}, {100, -
100}}, textString = "%name"), Rectangle(origin = {0, 40}, fillColor = {159, 159, 223},
          fillPattern =

                                    FillPattern.HorizontalCylinder, extent = {{-
80, 40}, {80, -40}})}),
    experiment(
      StopTime=600,
      __Dymola_fixedstepsize=0.0001,
      __Dymola_Algorithm="Dassl"),
    Diagram(coordinateSystem(extent = {{-100, -100}, {100, 100}})));
end final_model;
```

myPowerTransFcn

```
block myPowerTransFcn "Generate ramp and trapezoidal signal of type Real"
  parameter Real height = 1 "Height of ramps" annotation (
    Dialog(groupImage = "modelica://Modelica/Resources/Images/Blocks/Sources/Ramp.png"));
  parameter Modelica.SIunits.Time duration(min = 0.0, start = 2) "Duration of ramp (= 0.0
 gives a Step)";
  parameter Real amplitude = 1 "Amplitude of trapezoid" annotation (
    Dialog(groupImage = "modelica://Modelica/Resources/Images/Blocks/Sources/Trapezoid.pn
g"));
  parameter Modelica.SIunits.Time startTrans "Rising time of trapezoid";
  parameter Modelica.SIunits.Time rising(final min = 0) = 0 "Rising duration of trapezoid
";
  parameter Modelica.SIunits.Time width(final min = 0) = 0.5 "Width duration of trapezoid
";
  parameter Modelica.SIunits.Time falling(final min = 0) = 0 "Falling duration of trapezo
id";
  parameter Modelica.SIunits.Time period(final min = Modelica.Constants.small, start = 1)
 "Time for one period";
  parameter Integer nperiod = 0 "Number of periods (< 0 means infinite number of periods)
";
  extends Modelica.Blocks.Interfaces.SignalSource;
protected
  parameter Modelica.SIunits.Time T_rising = rising "End time of rising phase within one
period";
  parameter Modelica.SIunits.Time T_width = T_rising + width "End time of width phase wit
hin one period";
  parameter Modelica.SIunits.Time T_falling = T_width + falling "End time of falling phas
e within one period";
  Modelica.SIunits.Time T_start "Start time of current period";
  Integer count "Period count";
initial algorithm
  count := integer((time - startTrans) / period);
  T_start := startTrans + count * period;
equation
  when integer((time - startTrans) / period) > pre(count) then
    count = pre(count) + 1;
    T_start = time;
  end when;
  y = offset + (if time < startTime then 0 else if time < startTime + duration then (time
 - startTime) * height / duration else height) + (if time < startTrans or nperiod == 0 or
 nperiod > 0 and count >= nperiod then 0 else if time < T_start + T_rising then amplitude
 * (time - T_start) / rising else if time < T_start + T_width then amplitude else if time
 < T_start + T_falling then amplitude * (T_start + T_falling - time) / falling else 0);
  annotation (
    Icon(coordinateSystem(preserveAspectRatio = true, extent = {{-100, -
100}, {100, 100}}), graphics={  Line(points = {{-80, 68}, {-80, -
80}}, color = {192, 192, 192}), Polygon(points = {{-80, 90}, {-88, 68}, {-72, 68}, {-
80, 90}}, lineColor = {192, 192, 192}, fillColor = {192, 192, 192},
          fillPattern =

                                            FillPattern.Solid), Line(points = {{-90, -
70}, {82, -70}}, color = {192, 192, 192}), Polygon(points = {{90, -70}, {68, -62}, {68, -
78}, {90, -70}}, lineColor = {192, 192, 192}, fillColor = {192, 192, 192},
          fillPattern =

                                            FillPattern.Solid), Text(extent = {{-
147, -152}, {153, -112}}, textString = "period=%period"), Line(points = {{-81, -70}, {-
60, -70}, {-30, 40}, {9, 40}, {39, -70}, {61, -70}, {90, 40}})}),
    Diagram(coordinateSystem(preserveAspectRatio = true, extent = {{-100, -
100}, {100, 100}}), graphics={  Polygon(points = {{-81, 90}, {-87, 68}, {-75, 68}, {-
81, 90}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                            FillPattern.Solid), Line(points = {{-
81, 68}, {-81, -80}}, color = {95, 95, 95}), Line(points = {{-91, -70}, {81, -
```

```
70}}, color = {95, 95, 95}), Polygon(points = {{89, -70}, {67, -65}, {67, -76}, {89, -
70}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                              FillPattern.Solid), Polygon(points = {{-
46, -30}, {-48, -41}, {-44, -41}, {-46, -
30}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                              FillPattern.Solid), Line(points = {{-46, -
30}, {-46, -70}}, color = {95, 95, 95}), Polygon(points = {{-46, -70}, {-48, -60}, {-
44, -60}, {-46, -70}, {-46, -70}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                              FillPattern.Solid), Text(extent = {{-80, -
46}, {-42, -55}}, textString = "offset"), Text(extent = {{-49, -71}, {-6, -
81}}, textString = "startTime"), Text(extent = {{-80, 95}, {-
47, 80}}, textString = "y"), Text(extent = {{66, -78}, {89, -
89}}, textString = "time"), Line(points = {{-31, 82}, {-31, -
70}}, color = {95, 95, 95}, pattern = LinePattern.Dash), Line(points = {{-11, 59}, {-
11, 40}}, color = {95, 95, 95}, pattern = LinePattern.Dash), Line(points = {{19, 59}, {19
, 40}}, color = {95, 95, 95}, pattern = LinePattern.Dash), Line(points = {{39, 59}, {39,
-30}}, color = {95, 95, 95}, pattern = LinePattern.Dash), Line(points = {{-
31, 76}, {59, 76}}, color = {95, 95, 95}), Line(points = {{-
31, 56}, {39, 56}}, color = {95, 95, 95}), Text(extent = {{-
3, 86}, {24, 77}}, textString = "period"), Text(extent = {{-
11, 68}, {18, 59}}, textString = "width"), Line(points = {{-43, 40}, {-
11, 40}}, color = {95, 95, 95}, pattern = LinePattern.Dash), Line(points = {{-40, 40}, {-
40, -30}}, color = {95, 95, 95}), Text(extent = {{-77, 11}, {-
44, 1}}, textString = "amplitude"), Polygon(points = {{-31, 56}, {-24, 58}, {-24, 54}, {-
31, 56}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                              FillPattern.Solid), Polygon(points = {{-
11, 56}, {-18, 58}, {-18, 54}, {-
11, 56}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                              FillPattern.Solid), Polygon(points = {{-
31, 76}, {-22, 78}, {-22, 74}, {-
31, 76}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                              FillPattern.Solid), Polygon(points = {{59,
 76}, {51, 78}, {51, 74}, {59, 76}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                              FillPattern.Solid), Line(points = {{-81, -
30}, {-31, -30}, {-11, 40}, {19, 40}, {39, -30}, {59, -
30}, {79, 40}, {99, 40}}, color = {0, 0, 255}, thickness = 0.5), Polygon(points = {{-
40, 40}, {-42, 29}, {-38, 29}, {-
40, 40}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                              FillPattern.Solid), Polygon(points = {{-
40, -30}, {-42, -20}, {-38, -20}, {-40, -30}, {-40, -
30}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                              FillPattern.Solid), Line(points = {{59, 84
}, {59, -
30}}, color = {95, 95, 95}, pattern = LinePattern.Dash), Polygon(points = {{39, 56}, {32,
 58}, {32, 54}, {39, 56}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                              FillPattern.Solid), Polygon(points = {{19,
 56}, {26, 58}, {26, 54}, {19, 56}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
```

```
          fillPattern =

                                          FillPattern.Solid), Polygon(points = {{19,
 56}, {12, 58}, {12, 54}, {19, 56}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                          FillPattern.Solid), Polygon(points = {{-
11, 56}, {-4, 58}, {-4, 54}, {-
11, 56}}, lineColor = {95, 95, 95}, fillColor = {95, 95, 95},
          fillPattern =

                                          FillPattern.Solid), Text(extent = {{-
35, 68}, {-
6, 60}}, textString = "rising"), Text(extent = {{16, 68}, {44, 60}}, textString = "fallin
g")}),
    Documentation(info = "<html>
<p>
The Real output y is a trapezoid signal:
</p>

<p>
<img src=\"modelica://Modelica/Resources/Images/Blocks/Sources/Trapezoid.png\"
    alt=\"Trapezoid\">
</p>
</html>"));
end myPowerTransFcn;
```

References

[1] EIA. International Energy Outlook 2019

[2] R. J. Goldston, "Heuristic drift-based model for the power scrape-off width in Hmode tokamaks," *38th EPS Conf. Plasma Phys. 2011, EPS 2011 - Europhys. Conf. Abstr (*, vol. 35 1, pp. 385–388, 2011.)

[3] Y. Nagayama, "Liquid lithium divertor system for fusion reactor," *Fusion Eng. Des.*, vol. 84, no. 7–11, pp. 1380–1383, 2009.

[4] G. F. Nallo *et al.*, "Modeling the lithium loop in a liquid metal pool-type divertor," 61 *Fusion Eng. Des.*, vol. 125, pp. 206–215, 2017.

[5] T. W. Morgan et al., "Experiments and modelling in development of a liquid metal conceptual divertor design for the European DEMO", presented at ISLA conference, University of Illinois at Urbana-Champaign, Oct. 2019

[6] D. Relter et al., Nucl. Fus. 30 (1990)2141: T. Putterich, EPFW Split, Dec 2014 p112

[7] J. P. Freidberg, Plasma Physics and Fusion Energy (Cambridge University Press, 2007)

[8] Yican Wu, Sümer Şahin, 3.13 Fusion Energy Production, Editor(s): Ibrahim Dincer, Comprehensive Energy Systems, Elsevier, 2018, Pages 538-589, ISBN 9780128149256

[9] Pitts et al., Physics conclusions in support of ITER W divertor monoblock shaping. Nuclear Materials and Energy. (2017).

[10] S. Roccella et al. "CPS Based Liquid Metal Divertor Target for EU-DEMO". In: *Journal of Fusion Energy* (2020), (7pp). issn: 1572-9591.

[11] E. Tejado et al., The thermo-mechanical behaviour of W-Cu metal matrix composites for fusion heat sink applications: The influence of the Cu content. J. Nucl. Mater. 498, 468–475 (2018).

[12] A. von Muller et al., Melt infiltrated W-Cu composites as advanced heat sink materials for plasma facing components of future nuclear fusion devices. Fusion Eng. Des. 124, 455–459 (2017).

[13] ITER MATERIAL PROPERTIES HANDBOOK, Document No. G 74 MA 16, File Code: ITER-AK02-3304 / 3108 / 3112, Publication Package No. 9

[14] ITER MATERIAL PROPERTIES HANDBOOK, Document No. S 74 MA 2, File Code: ITER-AM01-3112, Publication Package No. 2

[15] T.W. Morgan a , ∗, A. Vertkov b et al., Power handling of a liquid-metal based CPS structure under high steady-state heat and particle fluxes, Nuclear Materials and Energy 12 (2017) 210–215

[16] ITER MATERIAL PROPERTIES HANDBOOK, Document No. S 74 MA 2, File Code: ITER-AR01-3112, Publication Package No. 1

[17] I. V. Savchenko et al., Measurement of Liquid Tin Heat Transfer Coefficients within the Temperature Range of 506–1170 K, In. *Teplofizika Vysokikh Temperatur* (2011), Vol. 49, No. 4, pp. 524–528.

[18] Zongbiao Ye et al. "The investigation of plasma-induced wettability of liquid tin-capillary porous system". In: Nuclear Materials and Energy 20 (Aug. 2019), p. 100694. issn: 23521791.

[19] Boštjan Končar et al. "Initial optimization of DEMO fusion reactor thermal shields by thermal analysis of its integrated systems". In: *Fusion Engineering and Design* 125.July (2017), pp. 38–49. issn: 09203796.

[20] I. E. Lyublinski et al., "Status of design and experimental activity on module of lithium divertor for KTM tokamak," Fusion Eng. Des., vol. 88, no. 9–10, pp. 1862– 1865, 2013.

[21] T. W. Morgan, D. C. M. Van Den Bekerom, and G. De Temmerman, "Interaction of a tin-based capillary porous structure with ITER/DEMO relevant plasma conditions," *J. Nucl. Mater.*, vol. 463, pp. 1256–1259, 2015.

[22] R.E Nygren et al. «Liquid surfaces for fusion plasma facing components - A critical review. Part I: physics and PSI». In: *Nuclear Materials and Energy* (2016).

[23] H P Summers. *The ADAS User Manual, version 2.6*. Tech. rep. 2004.

[24] T. Putterich et al. "Impurity limits in a reactor grade fusion device". In: *42nd European Physical Society Conference on Plasma Physics, EPS 2015, paper n. 4.111*. 2015.

[25] G.F. Nallo, "Modelling liquid metals for nuclear fusion and fission reactors", Dissertation Doctoral Program in Energetics (33.rd cycle), Chapter 5.2, pp. 62-89

[26] A Vertkov et al. "The concept of lithium based plasma facing elements for steady state fusion tokamak-reactor and its experimental validation". In: *2018 IAEA Fusion Energy Conference - EX/P1-12*. Gandhinagar, 2018.

[27] Jafar Safarian and Thorvald A. Engh, "Vacuum Evaporation of Pure Metals", In: METALLURGICAL AND MATERIALS TRANSACTIONS A (February 2013).

[28] J. Knaster, T. Kanemura, and K. Kondo, "An assessment of the evaporation and condensation phenomena of lithium during the operation of a Li(d,xn) fusion relevant neutron source", Heliyon, vol. 2, no. 12, pp. 1–23, 2016.

[29] A. V. Chankin, D. P. Coster, and R. Dux. "Monte Carlo simulations of tungsten redeposition at the divertor target". In: *Plasma Physics and Controlled Fusion* 56.2 (2014), 025003 (11pp). issn: 07413335.

[30] M. Iafrati, "Phenomenology of plasma-wall interaction using liquid metals in tokamak devices," Ph.D. thesis, Università degli studi di Padova, 2018.

[31] F. Casella and A. Leva, "Modelica open library for power plant simulation: design and experimental validation," Proceedings of 3rd International Modelica Conference, pp. 41-50, Linköping, Sweden, 3-4 November 2003.