SUSHEAT

Smart Integration of Waste and Renewable Energy for Sustainable Heat Upgrade in the Industry

D4.4 AI driven TES

Document Summary Information

Grant Agreement No.:	*101103552*	**Acronym:**	SUSHEAT
WP:	*WP4 – Thermal Energy Storage: Design and Optimization*		
Task:	*Task T4.4. Artificial intelligence driven new TES designs*		
Deliverable ID:	*D4.4 – AI driven TES*		
Submission date:	*13-06-2025*	**Due date:**	*30-04-2025*
Leader:	*Dr. Carles Mateu Piñol (UDL)* *Prof. Luisa F. Cabeza (UDL)* *Dr. Emiliano Borri (UDL)* *Nadiya Mehraj (UDL)*		
Participants:			
Dissemination level:	*Public*		
Reviewers:	*Rubén Barbero*		
Approver:	*Name WP-leader (Partner)*		

☒ Report ☐ Demonstrator ☐ Open Research Data Pilot ☐ Other

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or CINEA. Neither the European Union nor the granting authority can be held responsible for them. Grant Agreement No 101103552.

Executive Summary

This deliverable is the main outcome of the work carried out in “Task T4.4. Artificial Intelligence driven new TES designs”. This document details the application of the rules and constraints that the biologically inspired design must adhere to, as decided in Task 4.3. The task focused on producing designs that, following and adhering to the aforementioned rules, produced as many optimal designs for the inner piping of the tanks as possible, given the constraints and goals provided.

Revision history (quality control)

Version	Issue Date	% Complete	Changes	Contributor(s)
V1.0	2025-06-02	90 %	First draft sent out for revison	Dr. Carles Mateu Piñol (UDL) Prof. Luisa F. Cabeza (UDL) Dr. Emiliano Borri (UDL) Nadiya Mehraj (UDL)
V1.0	2025-06-02	100 %	Final version	Dr. Carles Mateu Piñol (UDL) Prof. Luisa F. Cabeza (UDL) Dr. Emiliano Borri (UDL) Nadiya Mehraj (UDL)

Table of contents

List of Tables

List of Figures

Glossary of terms and abbreviations

Acronyms
AI – Artificial Intelligence
CAD – Computer Aided Design
CFD – Computational Fluid Dynamics
GA – Genetic algorithm
STL – Stereolithography (file format)
TES – Thermal Energy Storage

Symbols

- w_e, e as the effectiveness (and it's weight)
- w_E, E as the energy stored in the tank (and it's weight)
- w_P, P as the power transfer (and it's weight)
- w_C, C as the charging time (and it's weight)
- w_D, D as the discharging time (and it's weight)
- w_p, p as the pressure drop (and it's weight)
- w_U, U as the uniformity (and it's weight)
- w_R, R as the Reynolds score (and it's weight)

Greek letters

Subscripts

1. Introduction

One of the goals of the SUSHEAT project was to produce and test designs for Thermal Energy Storage (TES) tanks that, applying biologically inspired design principles, were as optimal as possible on their task: rapidly storing thermal energy. The final goal was to test real implementations of those designs (by means of building such prototypes using metal 3D printing).

1.1. Mapping SUSHEAT Outputs

Table 1. Adherence to SUSHEAT's GA Deliverable & Tasks Descriptions.

TASKS	
Task T4.4. Artificial intelligence driven new TES designs	**Sections 2 to 5,**
The goal of this task is to apply AI techniques for the optimal design of biomimetic-based PCM storage tanks. The aim is to model, design, and evaluate (CFD) the thermal performance of the tanks. This task will be performed in line with T4.3 by providing inputs and optimized designs for the laboratory tests and obtaining results from the laboratory tests to calibrate the models, and the TES design. *Different materials will be evaluated for the design of the tank as well as different AI techniques for the elaboration and optimization of the design, genetic algorithms, machine learning, deep reinforcement learning among others.*	This deliverable reports the final TES design obtained with AI techniques, based on the rules described in D4.3 - Bio-inspired TES Rules. Regarding the CFD evaluation, ADSC have analyzed the charging and discharing of the TES via CFD simulations that were compared obtained experimental results by UDL which will be reported in D4.5 "Lab-scale TES design".
DELIVERABLE	
D4.4 – AI driven TES	**Sections 2 to 5**
AI driven TES designs for SUSHEAT. The report will analyse and show the performance of the tanks and the AI methods that have been considered and used. (T4.4)	In this report, the biomimetic rules and a description of the genetic algorithm used are shown in Section 2. The designs proposed are shown in Section 3 with the final TES design obtained with AI shown in Section 4.

1.2. Deliverable Overview and Structure

This deliverable is structured as follows: Section 2 reports the biomimetic rules and a description of the genetic algorithm. The designs proposed are shown in Section 3. The results of the optimization are shown in Section 4. The conclusions of this report are in Section 5.

2. Biomimetic rules and genetic algorithm

This chapter is a brief summary of deliverable D4.3; it introduces, briefly, the required concepts to be able to follow this document, for more details, the reader can refer to D4.3.

2.1. Genetic algorithms

The chosen optimization algorithm for this project is genetic algorithms; a genetic algorithm, briefly explained, is an optimization method that mimics the process of natural selection by iteratively evolving a population of potential solutions towards an optimal (or quasi-optimal) solution.

The inner workings of genetic algorithms are quite simple, the solution(s) to the problem, are encoded as a genome. That is, all the defining characteristics of the TES tank pipes are encoded as a set of “numbers” (the genome). For each possible genome, i.e. each different combination of defining characteristics (usually called parameters) we have a different TES tank piping design (an individual).

Also, we require a function that can work as an evaluation function, one that given a design, outputs a measure of the “goodness” (called fitness in genetic algorithms) of such design. This function must allow comparing different individuals, choosing the best of them.

Once we have these basic building parts, we can use natural selection-inspired algorithms to find the optimal solution. To that end, we start by creating several (hundreds) of random individuals (potential solutions) to the problem. We use the fitness function to rank them according to their goodness. We then remove the worst ones, keeping the best solutions, and we proceed to build new individuals, but this time we build them based on the existing ones, using similar tools to that found in nature: mutation, crossover, etc. Once we have this new generation of solutions, we start over again by evaluating them, culling the worst part of the population, and so on.

2.2. Genome parameters

As defined in D4.3, where a more detailed definition is provided, these are the defining parameters of the TES tank designs.

The parameters can be divided into three separate parts: shell, parent pipe, and pipe parameters.

2.2.1. Shell parameters

Shell parameters are those defining the enclosing tank, we have considered two parameters: shell diameter and shell height. Both determine the total volume of the tank and are constrained together (there is a minimum and a maximum determined by manufacturing capabilities).

2.2.2. Parent pipe parameters

The parent pipe is the pipe entering and exiting the TES tank. Due to this pipe comes from the outside, with its own set of constraints (manufacturing and external requirements), which makes this pipe has different treatment from the other pipes.

The affecting parameters are pipe length, pipe diameter, and pipe thickness.

2.2.3. Pipe parameters

Pipe parameters determine how the pipes inside the TES tank are shaped. These are the core of the design. The main parameters are:

- Branching factor: number of branches.
- Branching point: where the parent pipe branches out.

- Branching angle: the connecting angle of the branch.
- Pipe length: the length of pipes at this level.
- Pipe diameter: the diameter of the branch.
- Pipe thickness: the wall thickness of the pipe in this branch.

2.3. Fitness function

One key element in solving design problems using genetic algorithms is the existence of a fitness function. This function has to "grade" the goodness of any solution. As the driver behind GAs is the survival of the fittest, we need to be able to select the fittest of the solutions from the solution pool we have. The usual operation of the fitness function is to measure the fitness function of each individual as a numerical value associated with it. This value, or values, doesn't necessarily have to have a straightforward physical interpretation, that is, we don't need a precise physical model. We will use the value to compare individuals between them, rank them, and remove the worst. So, the fitness function must provide us with a comparable measure to choose between individuals.

In our case, the fitness function considers several aspects of the design, related to the quality of the design from a TES tank perspective: the amount of energy we can store, and charging and discharging speed are the most obvious ones. As the fitness function is what drives the selection of individuals for removing or procreating, the GA functions as an optimization algorithm maximizing said fitness function.

The fitness function used for the design problem was:

$$Printability \times \left(w_e \times e + w_E \times E + w_P \times P + w_C \times C + w_D \times D + w_p \times p + w_U \times U + w_R \times R\right)$$

It is a weighted sum of the factors we are considering when evaluating designs. With:

- w_e, e as the effectiveness (maximizing)
- w_E, E as the energy stored in the tank (maximizing)
- w_P, P as the power transfer (maximizing)
- w_C, C as the charging time (minimizing)
- w_D, D as the discharging time (minimizing)
- w_p, p as the pressure drop (minimizing)
- w_U, U as the uniformity (maximizing)
- w_R, R as the Reynolds score

All factors are weighted, and the actual weights have been computed by a hypergrid parameter search. With the weights as:

- $w_e = 0.2$
- $w_E = 0.15$
- $w_P = 0.1$
- $w_C = 0.15$
- $w_D = 0.15$
- $w_p = 0.10$
- $w_U = 0.10$
- $w_R = 0.05$

Then a final factor, printability, acts as a penalty weight to all the fitness scores from the thermal behavior of the tank.

If a tank is printable, Printability equals 1, so the fitness score is unchanged. With tanks that have too thin walls or hanging surfaces (hard or impossible to print), Printability equals 0.1 if any of those characteristics occurs, or 0.01 if both characteristics occur simultaneously. This penalizes heavily the fitness score, by up to a factor of 1/100th, but doesn't remove those individuals from the population. This allows us to keep non-printable designs on the gene pool that may have interesting features but skews selection towards the most printable versions of those designs. This will lead to introducing those interesting features into the genes of printable designs via cross-breeding.

3. Design proposals

With all the preceding rules, fitness functions, and genome design. The project proceeded to build a genetic algorithms-based system to produce, first, new design specifications that maximize the fitness function, i.e., that optimize the design to fulfil the requirements, and then to build a system to translate those design specifications into a CAD-like language so designs can be turned into fabrication.

3.1. Proposed models

The system starts with the following constraints and parameters:

- Branching: Figure 1 shows results with 5 branches. For this deliverable, in order to easily compare our results with and without AI optimization, we used 5 branches; there are 4 around the central axis, and one centered on the axis.
- Sizes. We constrained the size to be a printable tank, 300 mm x 300 mm x 300 mm, the same size as the manually designed tanks we used to test for previous deliverables. This size, 300 mm x 300 mm x 300 mm, is the printable volume limit for our existing suppliers.

Once established these constraints, we run the GA system for 500 generations, with the following evolution of the fitness function:

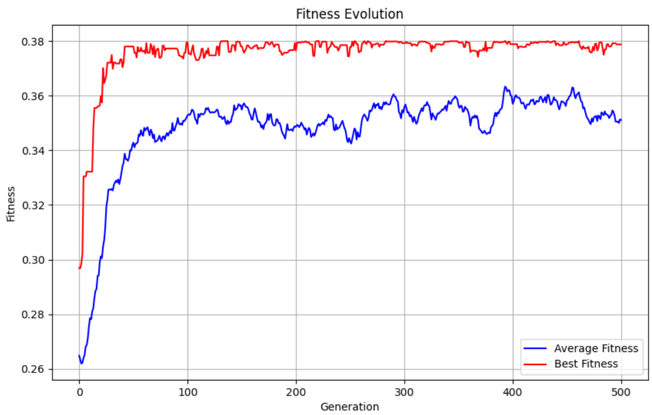

Fig. 1. Fitness function evolution per generation during the optimization process shows the fitness function value of the best individual in each generation and the average for all individuals in each generation.

As it can be seen clearly, the genetic algorithm is able to rapidly increase the fitness, i.e., the value, of all the designs as generations progress. In fact, although we experiment for 500 generations, at around 100 generations fitness function values are already almost at the best value we are going to get.

For the experiment we are showing the fitness function above, the resulting winning design is shown in Figure 2, where we can see the 5 branches on 3 levels that were obtained.

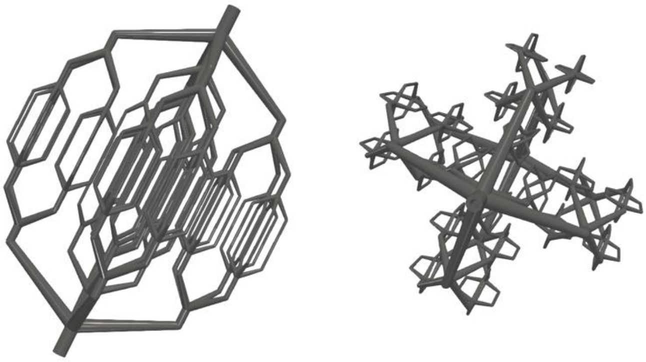

Fig. 2. Fully AI-generated design.

With the following characteristics:

- Minimum feature size: 0.6 mm. The minimum dimension (diameter, width, etc.) of any feature of the design is a constraint related to manufacturing the tank that ensures the design can be 3D printed. Designs that violate this constraint receive a high penalty, removing them from the gene pool.
- Minimum wall thickness: 0.5 mm. Another fabrication constraint. This thickness will allow for metal 3D printing without issues. Other manufacturers, other materials or other 3D printing processes can have different requirements.
- Maximum branching depth: 3. We restricted the design to have 3 different branching levels (and 3 rejoining levels). We imposed this restriction for ease of comparison with pre-existing manually designed tanks, and, given the maximum fabrication size, higher branching factors made designs harder to fabricate.

3.2. CAD designs

As the resulting designs are only vectors of numbers (i.e., genomes), to easily visualize such designs, and, more interestingly, to be able to use these designs for manufacturing, we have implemented a tool that takes in genetic descriptions of the designs and creates CAD drawings of the resulting TES tank designs.

This system uses an open-source readily available CAD software/language/solution called OpenSCAD. OpenSCAD is a parametric constructive geometry-based CAD system, where designs are specified constructively via a geometric specification language. Our system translates the results of the optimization phase (a set of numbers) into a 3D design, built using a set of geometry primitives we have developed.

We choose OpenSCAD because, first, is readily available, it's open-source and has been implemented into multiple architectures, that allow us to use it in both desktop computers or laptops or on high-performance computing solutions (clusters, HPC, etc.). Secondly, its building language is a mature, well-known and tested solution, with high-quality documentation. And, finally, OpenSCAD is an established solution in the 3D printing world, where it is commonly used for programmatically generated designs. It has plenty of options and features that help when designing and preparing designs for 3D manufacturing, including, but not limited to, the ability to export designs to file formats common for 3D printing software, for example to STL format. Even, in some cases, OpenSCAD, with the appropriate companion libraries, can generate the GCode files required for the most common printers on the market, without relying on external software, as slicers.

4. Results

As previously shown, the goal of this work package was to get an AI-optimized design for the heat transfer fluid piping inside the TES tank that was as optimal as possible. We have obtained such a design, but the results from this work package go beyond a single AI-optimized design. The result is a method and framework that allows to build designs that are optimal given a set of constraints. In fact, once this initial design is lab tested after it is fabricated with metal 3D printing, we plan to keep iterating from lab test to design to lab test. Improving the constraints and the fitness function from the lab results.

The optimized design, already shown in previous sections is the one shown in Fig. 3, with a measure axis, for scale.

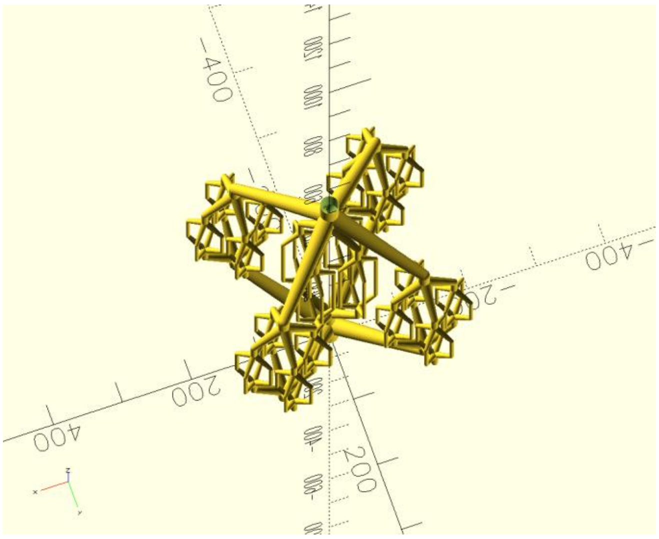

Fig. 3. Fully AI-generated design CAD view with rules for measure.

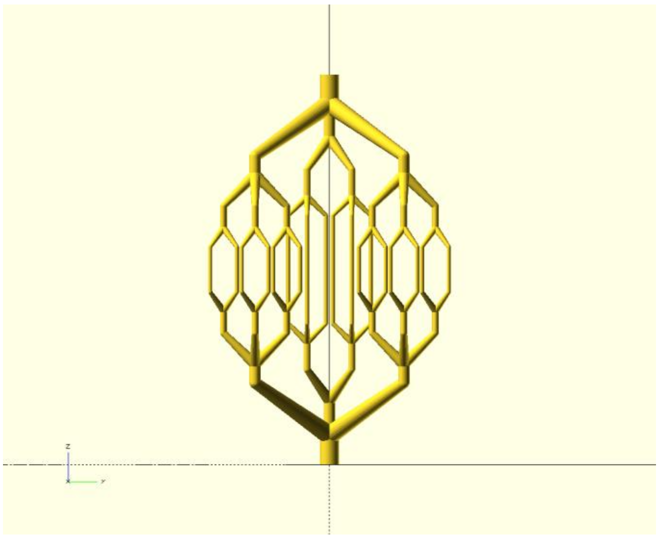

Fig. 4. Side view of AI-generated, showing the forced Y (vertical) symmetry.

The design measures 300 mm x 300 mm. And has some additional constraints, regarding symmetry for two reasons, one to ease fabrication and testing in the laboratory, and the other for easy comparison with previous designs already tested in Cabeza et al., 2024, that can be easily seen in Fig. 4. This forced vertical symmetry can be removed or modified from the generation algorithm.

Additionally, we also forced the branching factor to be the same as we used on previous designs to keep comparisons fair. But the branching factor can be changed or freed.

Branching angles are constrained to a set of values, the idea behind that is that those angles can be fabricated. As we gain more experience fabricating such models, we can extend the constraints so we can have the widest range of possible branch angles while still being able to 3D print the tanks. These parameters are the ones that will probably change more after laboratory testing of the tank. Moreover, fabrication-related parameters will surely change once we move from small lab-scale tanks to bigger-scale tanks.

5. Conclusions

We have been able to design an optimized biomimetic TES tank using AI, the initial goal of the work package. Our output is, not only, the optimized design but we have built also the necessary tools to easily modify such design to satisfy changing demands or changing specs, which will be useful after the laboratory testing phase and to consider the fabrication experience. The next step is to, physically, build the designed tank via a 3D printing manufacturer and test it in laboratory conditions. After that, and after considering the information gathered during fabrication and testing, the next round of optimization and re-design will take place to move to lab testing bigger and more “real scale” tanks.

This line of research is just in its infancy, as AI techniques progress at an astounding pace, and computational resources still grow at Murphy’s Law rhythm, the tools used for this kind of optimization can easily grow to consider more and more aspects of the design, even contemplating doing CFD like analysis for each iteration of the evolution of the tank designs. As for the implications on the SUSHEAT results, as the expected output of SUSHEAT will include a TES tank on the setup for heat recycling, the availability, in the near future, of better and more optimal TES tanks will surely improve the heat reuse capability of the setup.

References

Cabeza, L.F., Mani Kala, S., Zsembinszki, G., Vérez, D., Risco Amigó, S., Borri, E., 2024. Development of a Bio-Inspired TES Tank for Heat Transfer Enhancement in Latent Heat Thermal Energy Storage Systems. Applied Sciences 14, 2940. https://doi.org/10.3390/app14072940