MLACCD USER MANUAL

ALPHA RELEASE

PREFACE

In this manual, a quick guide is provided on how to properly use the MLACCD software for the user. A brief introduction of the software, the system requirement, the software installation, and the solutions to potential installation problems are discussed in Section 1. In Section 2, the guide of the software different graphical user interfaces (GUIs) of the software is provided. In Section 3, exceptions of the software such as the potential problems that the user might encounter, and the possible pretreatment of the cooling surfaces are discussed.

DESCRIPTION

The MLACCD software is designed based on the python tkinter GUI library, and powered by the TensorFlow. It contains models of MLACCD implementation, and an expandable design data base. This software provides a user-friendly interface that guide the researchers and industrial users to create their own MLACCD channels with desired conformal cooling topologies to minimize the resulting part surface temperature and increase the part quality. The functions of the software include:

- Provide the mesh design of the temperature variance minimized (TVM) cooling channels with zigzag, spiral, and conformal porous structures (CPS) based on the user defined cooling surface in .stl format.
- Provide the data of the control lines for MLACCD cooling channels for simulation and research purposes.
- A flexible and expandable machine learning design database that can improve the speed of TVM simulations.

DEVELOPERS

Zhenyang Gao – development of the software and the website

Yaoyao Fiona Zhao – project supervisor and the head of the research lab

TABLE OF CONTENTS

1. Overview

1.1. Introduction

The effectiveness of the cooling system in the injection molding processes significantly affects production efficiency and part quality. The current cooling system design is strictly limited by conventional manufacturing processes such as the casting and drilling process, which typically create straight or simple cooling channels. However, the maturing additive manufacturing (AM) technology allows the design and fabrication of complex conformal cooling channels. Typical advantages of conformal cooling are reduced cooling cycle

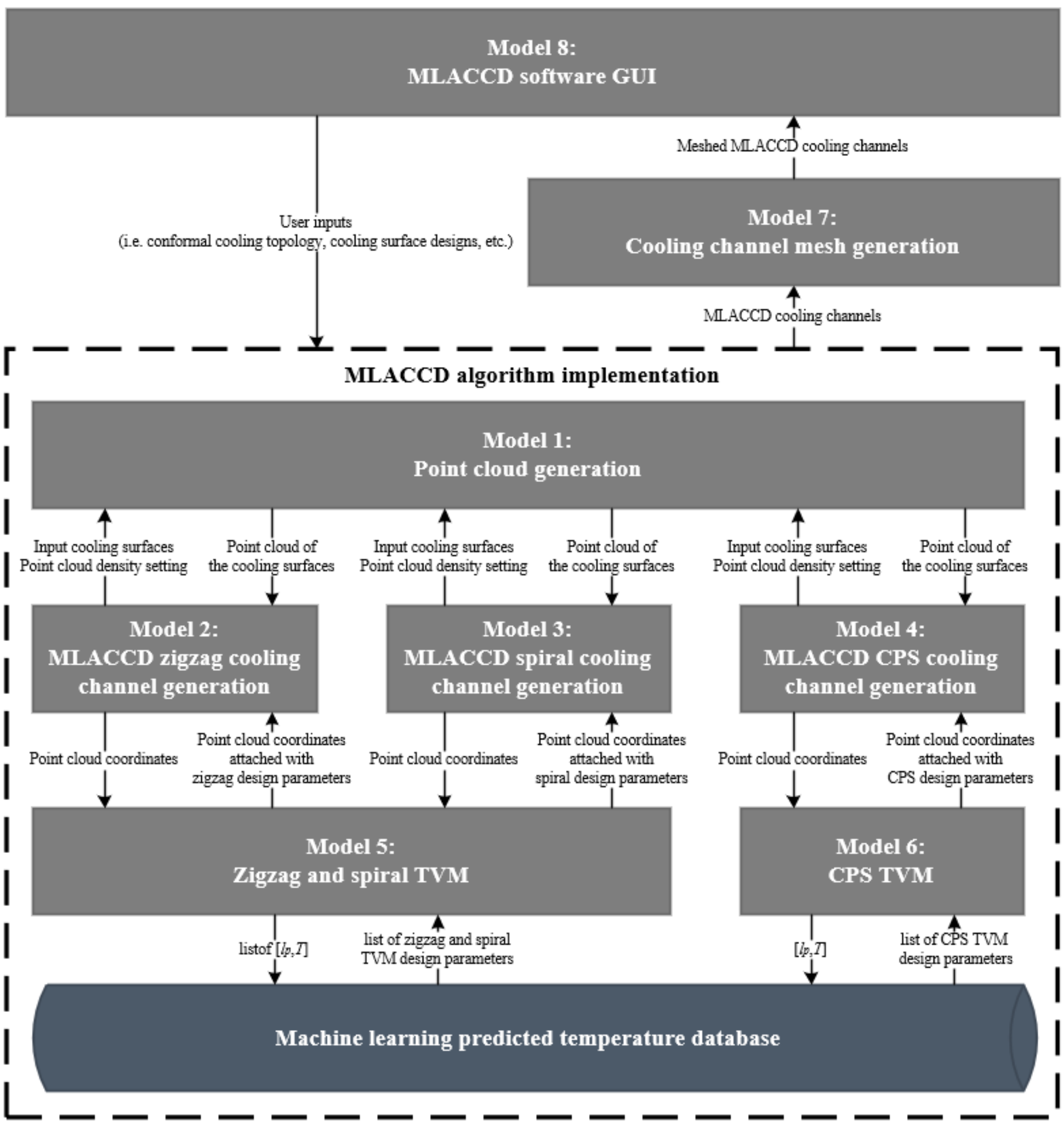


Figure 1. MLACCD software model relations

time, smaller temperature variance, and better cooling quality. However, the existing conformal cooling designs do not support parts with non-uniform thickness values, which leads to high temperature variance. To improve the conformal cooling design in terms of the temperature variance, the machine learning aided conformal cooling design (MLACCD) software is provided, where a machine learning aided design method is proposed to create cooling system which conforms not only to the part surface but also to the part thickness values. The current version of the software is able to support the topologies including zigzag, spiral, and conformal porous structures (CPS). If ones are interested in designing new topologies, please contact the ADML McGill research team for an updated version of the software.

The MLACCD software is designed based on the python tkinter GUI library, the pre-designed models that has the MLACCD implementation, and a data base composed of machine learning predicted temperature. The relations between the models are shown in Figure 1. Basically, functions within the Model 1 to Model 7 are called by the Model 8 (GUI model) so that the interactions between the user and the MLACCD programs can be achieved. Specifically, these MLACCD algorithms are integrated by the GUI model with a proper calling sequence, so that the corresponding graphical interfaces will be displayed once the user interacts with the software. To improve the maintainability of the software, the software models are designed with high cohesion (the functions within the same model are built to achieve a common goal), and low coupling (unnecessary relations between models are removed).

1.2. System requirement

Operating System: Windows 7 or later (64-bit recommended)

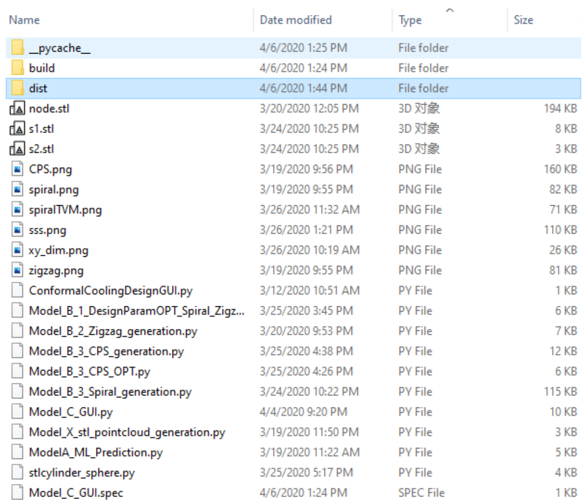
RAM: 8GB or more (recommended)

Python: python 3.5 – 3.7, 64-bit (available at https://www.python.org/downloads/, recommend version is 3.6.8: https://www.python.org/downloads/release/python-368/)

TensorFlow: tensorflow 1.5.0 to tensorflow 1.12.0 (available through pip, eg. “pip install tensorflow==1.12.0”)

1.3. Installation

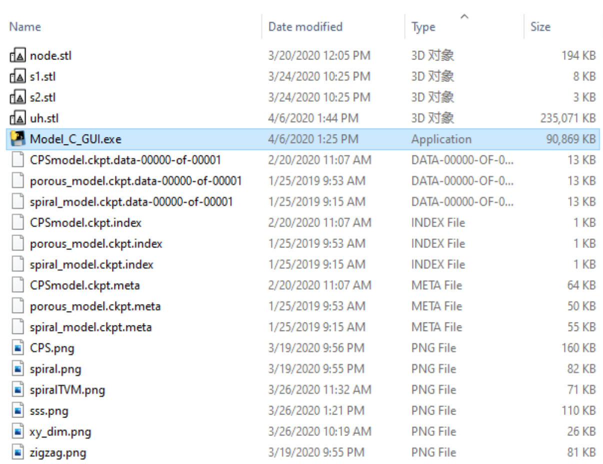
Download the MLACCD.zip at the same website, unzip it under the same directory where the TensorFlow is installed. To request the unzip passcode, please contact the ADML McGill research team. With in the MLACCD software folder, the .exe file is under the folder called “dist” as shown in Figure 2. Double-click the “Model_C_GUI.exe” to execute



Name	Date modified	Type	Size
__pycache__	4/6/2020 1:25 PM	File folder	
build	4/6/2020 1:24 PM	File folder	
dist	4/6/2020 1:44 PM	File folder	
node.stl	3/20/2020 12:05 PM	3D 对象	194 KB
s1.stl	3/24/2020 10:25 PM	3D 对象	8 KB
s2.stl	3/24/2020 10:25 PM	3D 对象	3 KB
CPS.png	3/19/2020 9:56 PM	PNG File	160 KB
spiral.png	3/19/2020 9:55 PM	PNG File	82 KB
spiralTVM.png	3/26/2020 11:32 AM	PNG File	71 KB
sss.png	3/26/2020 1:21 PM	PNG File	110 KB
xy_dim.png	3/26/2020 10:19 AM	PNG File	26 KB
zigzag.png	3/19/2020 9:55 PM	PNG File	81 KB
ConformalCoolingDesignGUI.py	3/12/2020 10:51 AM	PY File	1 KB
Model_B_1_DesignParamOPT_Spiral_Zigz...	3/25/2020 3:45 PM	PY File	6 KB
Model_B_2_Zigzag_generation.py	3/20/2020 9:53 PM	PY File	7 KB
Model_B_3_CPS_generation.py	3/25/2020 4:38 PM	PY File	12 KB
Model_B_3_CPS_OPT.py	3/25/2020 4:26 PM	PY File	6 KB
Model_B_3_Spiral_generation.py	3/24/2020 10:22 PM	PY File	115 KB
Model_C_GUI.py	4/4/2020 9:20 PM	PY File	10 KB
Model_X_stl_pointcloud_generation.py	3/19/2020 11:50 PM	PY File	3 KB
ModelA_ML_Prediction.py	3/19/2020 11:22 AM	PY File	5 KB
stlcylinder_sphere.py	3/25/2020 5:17 PM	PY File	4 KB
Model_C_GUI.spec	4/6/2020 1:24 PM	SPEC File	1 KB

Figure 2. The folder of the executable file

as shown in Figure 3. After the software is executed, a command window will pop up to allow additional communication message to be displayed as shown in Figure 4. The software will examine whether the required modules are installed, if not (it is rare to happen, but if happen please try to unzip it in the correct folder or re-install the missing module), a message indicating the missing module will be displayed in the command window and the



Name	Date modified	Type	Size
node.stl	3/20/2020 12:05 PM	3D 对象	194 KB
s1.stl	3/24/2020 10:25 PM	3D 对象	8 KB
s2.stl	3/24/2020 10:25 PM	3D 对象	3 KB
uh.stl	4/6/2020 1:44 PM	3D 对象	235,071 KB
Model_C_GUI.exe	4/6/2020 1:25 PM	Application	90,869 KB
CPSmodel.ckpt.data-00000-of-00001	2/20/2020 11:07 AM	DATA-00000-OF-0...	13 KB
porous_model.ckpt.data-00000-of-00001	1/25/2019 9:53 AM	DATA-00000-OF-0...	13 KB
spiral_model.ckpt.data-00000-of-00001	1/25/2019 9:15 AM	DATA-00000-OF-0...	13 KB
CPSmodel.ckpt.index	2/20/2020 11:07 AM	INDEX File	1 KB
porous_model.ckpt.index	1/25/2019 9:53 AM	INDEX File	1 KB
spiral_model.ckpt.index	1/25/2019 9:15 AM	INDEX File	1 KB
CPSmodel.ckpt.meta	2/20/2020 11:07 AM	META File	64 KB
porous_model.ckpt.meta	1/25/2019 9:53 AM	META File	50 KB
spiral_model.ckpt.meta	1/25/2019 9:15 AM	META File	55 KB
CPS.png	3/19/2020 9:56 PM	PNG File	160 KB
spiral.png	3/19/2020 9:55 PM	PNG File	82 KB
spiralTVM.png	3/26/2020 11:32 AM	PNG File	71 KB
sss.png	3/26/2020 1:21 PM	PNG File	110 KB
xy_dim.png	3/26/2020 10:19 AM	PNG File	26 KB
zigzag.png	3/19/2020 9:55 PM	PNG File	81 KB

Figure 3. The executable file

software will be closed. Wait for a few seconds and the main user interface of the software will pop up as shown in Figure 5.

Figure 4. The MLACCD command window

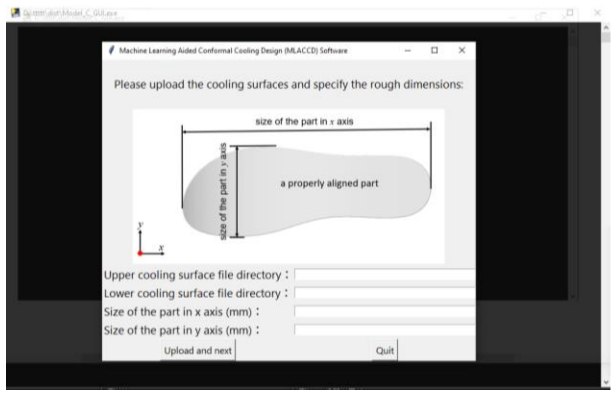


Figure 5. The MLACCD main interface

1.4. Solutions to potential installation problems

(1) Encountered “Could not find a version that satisfies the requirement tensorflow (from version: none)” when install tensorflow

Solution: this is probably due to the 32-bit version of python and pip, try to uninstall the python and re-install it with 64-bit.

(2) Encountered “No Module Named ‘_pywarp_tensorflow_internal’” and “Failed to load the native TensorFlow runtime”, which results in a crash of the MLACCD command window.

Solution: there are different reasons of causing this problem. This may due to the version of the tensorflow is not correct (recommend tensorflow 1.12.0), and may due to the requirement of the CUDA of your computer. Here are some solutions on-line:

Failed to load the native TensorFlow runtime:

https://github.com/tensorflow/tensorflow/issues/18503

No Module Named '_pywarp_tensorflow_internal':
https://github.com/tensorflow/tensorflow/issues/11571

2. Software interface guide

The flowchart of the software is shown in Figure 6, where are four steps for the software to generate the meshed MLACCD channels for the user. The main menu GUI is shown in Figure 7, where the user is asked to type in the file directories for the upper and lower cooling surface of the part, and the estimations of the part size in *x* and *y* directions. To explain, the file directories are used to upload the cooling surfaces from specified locations for analysis, and the sizes of the part are used in the calculation of the point cloud density, which will be discussed later.

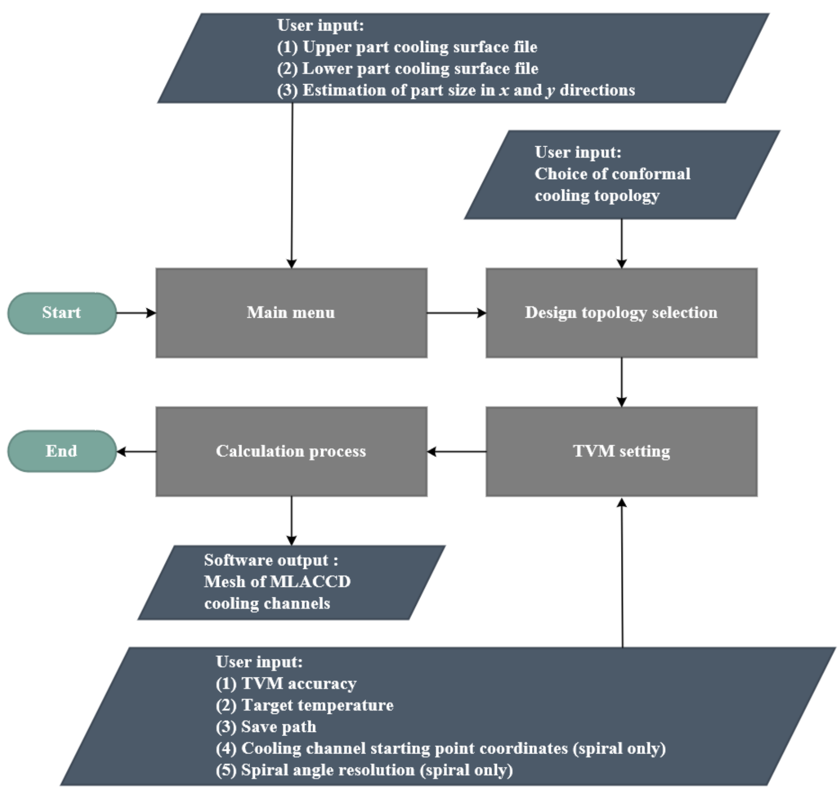


Figure 6. MLACCD software flowchart

By clicking the "Upload and next" button, the software navigates to the design topology selection menu as shown in Figure 8. In this menu, three conformal cooling topologies are provided, where the topology that user prefers can be selected for TVM analysis. The porous cooling channels are not implemented in this interface, since CPS is able to provide a better cooling performance and a smaller pressure drop as a special type among different porous cooling systems [13].

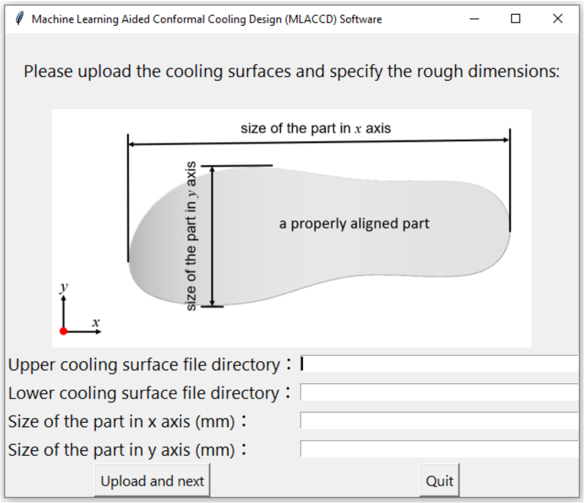


Figure 7. MLACCD software main menu

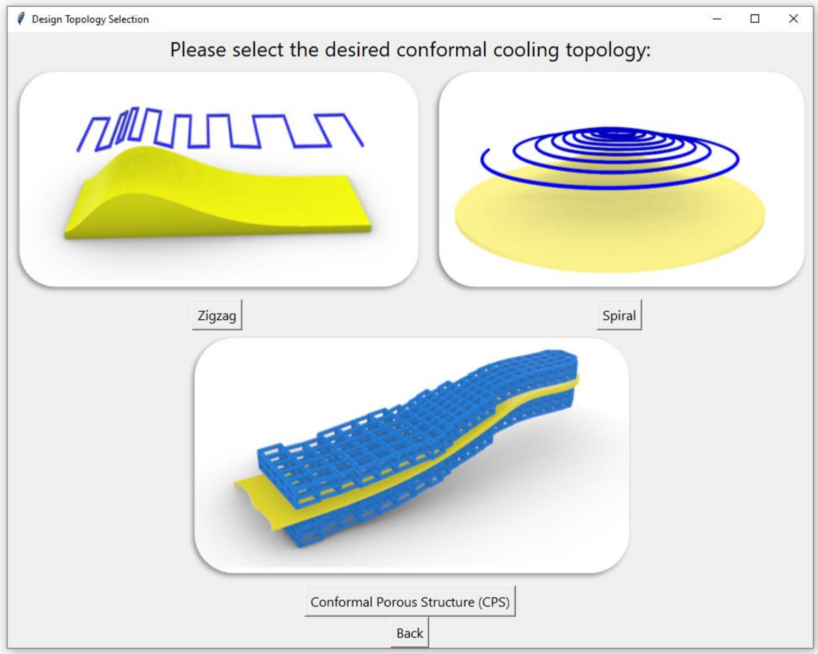
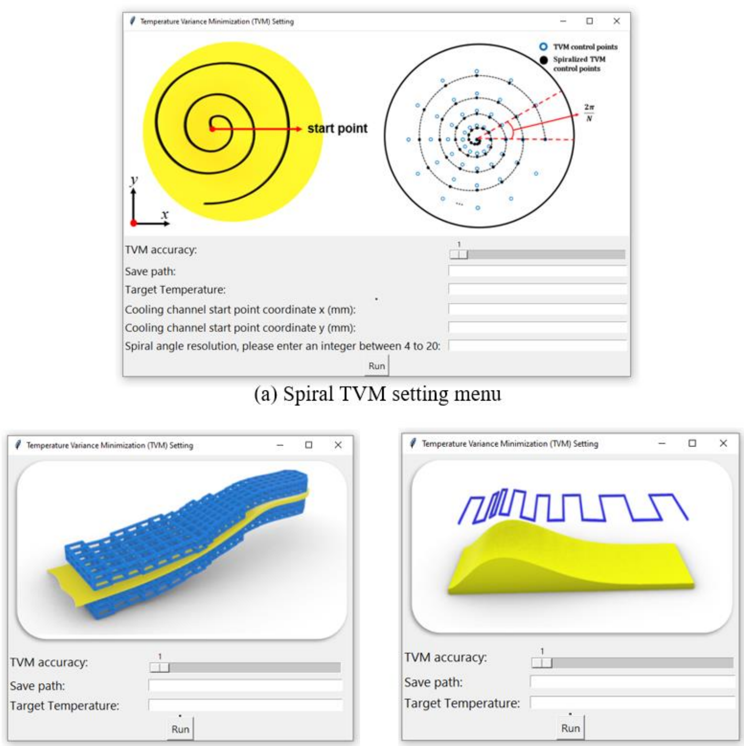


Figure 8. Design topology selection menu

After the cooling topology is selected, the TVM setting menu is shown is acquire the inputs including the TVM accuracy, save path, and the target temperature from the user as shown in Figure 9. For spiral cooling channels TVM setting as shown in Figure 9a, additional user inputs such as the spiral angle resolution *N*, and the cooling channel start point coordinates are required to fully constrain the control lines of the cooling channels. The TVM accuracy together with the size of the part reflects the density of the point cloud during the analysis as shown in equation (6-1):

$$\rho_{point\ cloud} = \frac{\text{TVM accuracy} \times \text{global accuracy}}{s_x s_y} \quad (6\text{-}1)$$

where s_x, s_y are the estimate size of the part in *x* and *y* directions, $\rho_{point\ cloud}$ is the point cloud density. The global accuracy is implemented as 500 inside the current version of the software to achieve a simulation time ranging from roughly 5 minutes to 10 hours according to different TVM accuracy defined by the user. Figure 10 shows the window that appears once the TVM setting is completed and the calculation process starts. The software displays a program dialog box and a process bar to provide the information about the current objective and progress of the calculation and potential warning messages for the user. Once the calculation is finished, the meshed MLACCD channels will be saved



(a) Spiral TVM setting menu

(b) CPS TVM setting menu

(c) Zigzag TVM setting menu

Figure 9. Temperature variance minimization (TVM) setting menu

to the save path, where the supported input and output format of the current version software is STL.

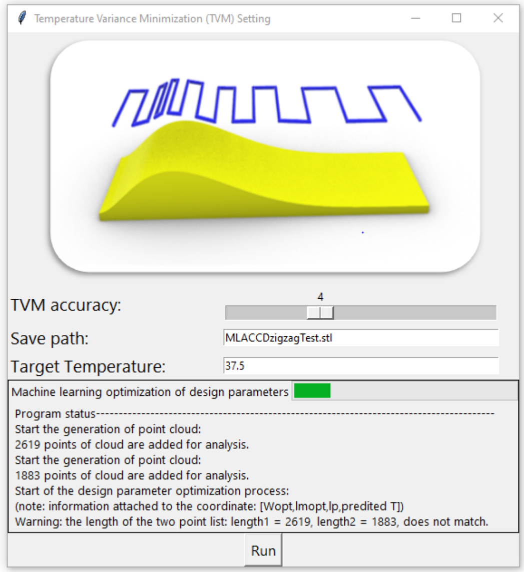


Figure 10. MLACCD program dialog window

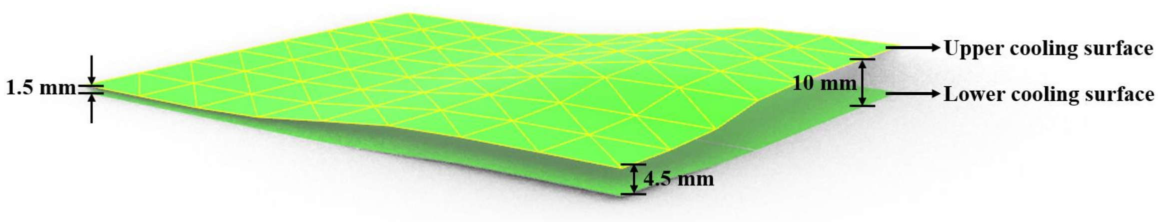


Figure 11. Example cooling surfaces

To illustrate the expected output of the software, example cooling surfaces as shown in Figure 11 are provided, where this test part has a highly non-uniform thickness values distributed long its surface. Following the instructions of the software as shown in Figure 7 to Figure 10, the mesh of example MLACCD channels can be generated, where the CPS is selected for these cooling surfaces for illustration purposes. The MLACCD CPS cooling channels generated by the software are shown in Figure 12, which has optimized cell size and pitch to part surface distance distributed along the cooling surface of the part. Based on the mesh of the MLACCD channels, the user can either make the MLACCD wireframe re-built for simulation purposes or generate the final mold design using the mesh Boolean subtraction between the mold object and the mesh of the MLACCD channels.

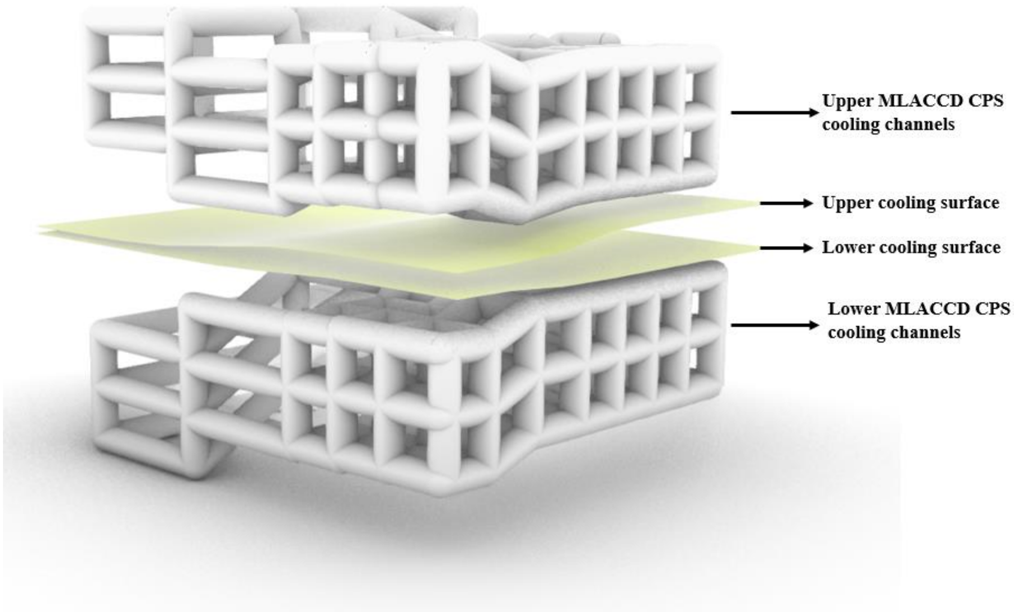


Figure 12. MLACCD CPS cooling channels for example cooling surfaces

3. Exceptions

Exceptional operations are required to generate the MLACCD cooling channels for the parts with special properties such as the one with large geometrical angle variations as shown in Figure 13, or the one containing very small geometrical features as shown in Figure 14. For the parts with large geometrical angle variations as shown in Figure 13a,

(a) Example part with large geometrical angle variations
(b) Separated cooling regions
(c) MLACCD cooling channels for the part with large geometrical angle variations

Figure 13. Generation of cooling channels for the part with large geometrical angle variations

the total cooling surface can be divided into different cooling regions. Specifically, these cooling regions are separated by the cooling surface division lines, which are the edges of two neighboring surfaces with the angles that are close to 90 degrees. After the total cooling surface is divided into different cooling regions as shown in Figure 13b, the MLACCD cooling channels can be formed based on the cooling channels that are separately constructed for each cooling region as shown in Figure 13c.

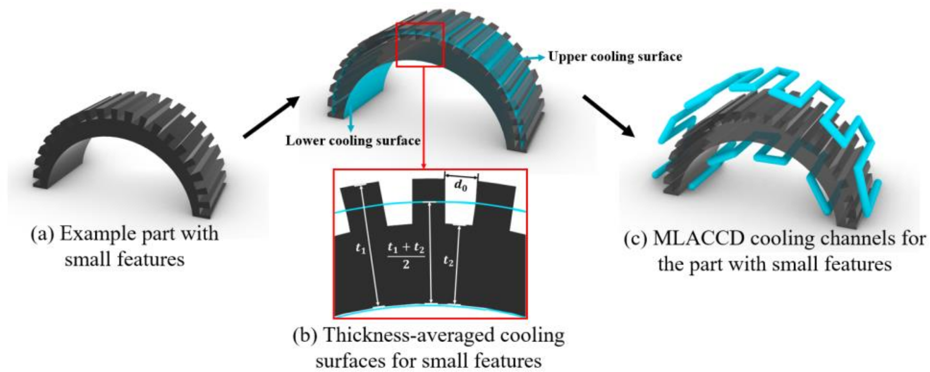


Figure 14. Generation of cooling channels for the part with small features

For the parts with very small geometrical features which have the feature size d_0 smaller than the minimum pitch width of the cooling channels as shown in Figure 14, the thickness-averaged cooling surfaces are required to generate the MLACCD cooling channels. The reason of calculating the thickness-averaged cooling surfaces is that the dimensions of those detailed features are too small for the cooling channels being conformal to them. For an example part with gear features as shown in Figure 14a, the distance between the two cooling surfaces would be the average thickness of the gear features calculated in Figure 14b. Based on the thickness-averaged cooling surface, the MLACCD cooling channels for the part with small features can be generated as shown in Figure 14c.

References

[1] Z. Gao, G. Dong, Y. Tang, Y. Zhao. Machine learning aided design of conformal cooling channels for injection molding. ASME. Journal of Mechanical Design. (In Press)
[2] Z. Gao, Y. Zhao. Machine learning aided design and optimization of conformal porous structure. IDETC-CIE 2020.
[3] Z. Gao, Y. Zhao. Machine learning aided optimization of conformal cooling channels for injection molding. HI-AM Conference 2019.

[4] Z. Gao, Y. Zhao. Machine learning aided design of conformal cooling channels for injection molding, Master thesis, McGill University, 2020.