NASA/TM-20230011420 AIAA 2023–4127

Thermal Management System Modeling in the Heat Transport System Simulation (HeaTSSPy) Package

Jeffryes W. Chapman, Hashmatullah Hasseeb, and Sydney L. Schnulo
Glenn Research Center, Cleveland, Ohio

August 2023

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI Program provides access to the NASA Technical Report Server—Registered (NTRS Reg) and NASA Technical Report Server—Public (NTRS) thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counter-part of peer-reviewed formal professional papers, but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., "quick-release" reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at http://www.sti.nasa.gov

- E-mail your question to help@sti.nasa.gov

- Fax your question to the NASA STI Information Desk at 757-864-6500

- Telephone the NASA STI Information Desk at 757-864-9658

- Write to:
  NASA STI Program
  Mail Stop 148
  NASA Langley Research Center
  Hampton, VA 23681-2199

NASA/TM-20230011420 AIAA 2023–4127

Thermal Management System Modeling in the Heat Transport System Simulation (HeaTSSPy) Package

Jeffryes W. Chapman, Hashmatullah Hasseeb, and Sydney L. Schnulo
Glenn Research Center, Cleveland, Ohio

Prepared for the
2023 AIAA/IEEE Electric Aircraft Technologies Symposium (EATS)
cosponsored by AIAA and IEEE
San Diego, California, June 14–16, 2023

National Aeronautics and
Space Administration

Glenn Research Center
Cleveland, Ohio 44135

August 2023

Acknowledgments

The authors would like to thank the NASA Revolutionary Vertical Lift Technology (RVLT), Convergent Aeronautics Solutions (CAS), and Transformational Tools and Technologies (TTT) projects for funding this work.

This work was sponsored by the Advanced Air Vehicle Program
at the NASA Glenn Research Center

Level of Review: This material has been technically reviewed by technical management.

This report is available in electronic form at https://www.sti.nasa.gov/ and https://ntrs.nasa.gov/

Thermal Management System Modeling in the Heat Transport System Simulation (HeaTSSPy) Package

Jeffryes W. Chapman, Hashmatullah Hasseeb, and Sydney L. Schnulo
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Abstract

This paper describes the development of a thermal management system (TMS) concept design and analysis software package called Heat Transport System Simulation (HeaTSSPy). Built within Python using the OpenMDAO framework, HeaTSSPy can be used to size and optimize an active (using liquid/air heat exchangers) or passive (using finned heat sinks) TMS. The package makes use of modular TMS elements that allow for the creation of different system architectures and includes components such as heat sinks, heat exchangers, liquid pumps, fans, ducts, air inlets, air nozzles, and liquid pipes. Modeling methods for these components include a combination of physics-based analytical and empirical equations that relate component sizing criteria to system performance. The HeatSSPy heat sink methods are fully detailed within this paper, while the heat exchanger methods are described and referenced from previous work. This paper also uses high-fidelity simulation to validate two different methods of calculating thermal resistance using CFD results. Once the methods are fully described, the code is exercised to compare an active TMS with that of a passive TMS. Design criteria for this study include rejected heat, system altitude, Mach number, and ambient temperature. These criteria are used to develop a TMS system with estimated performance metrics such as weight, drag, and operational power. Results of this paper show the crossover point when a passive system begins to weigh more than an active system for a given heat rejection.

Nomenclature

A	Area
AUqV	Area*overall thermal conductance/volume
A_R	Fin Aspect Ratio
b	Fin spacing (channel width)
C	Capacity rate
CFD	Computational Fluid Dynamics
D	Diameter
D_h	Hydraulic diameter
D	Channel diameter
EAP	Electrified Aircraft Propulsion
f	Friction factor
h	Heat transfer coefficient
HeaTSSPy	Heat Transport System Simulation
Ht	Height
k	Thermal conductivity
k_{air}	Thermal conductivity of air
K	Loss coefficient
L	Length
m	Thermal constant

N	Number
N_f	Number of Fins
Nu	Nusselt number
NTU	Number of Transfer Units
Pr	Prandtl number
R	Thermal resistance
Re	Reynolds number
Re_{D_H}	Reynolds number based on hydraulic diameter
Re_b	Reynolds number based on channel width
Re_b^*	Channel Reynolds number
RVLT	Revolutionary Vertical Lift Technology
t	Thickness
t_f	Fin Thickness
t_{base}	Base Thickness
T	Temperature
Q	Heat rejection
U	Overall thermal conductance
V	Volume
ν	Kinematic viscosity
W	Width
ε	Effectiveness
ε_f	Fin effectiveness
η	Efficiency
η_f	Fin efficiency
μ	Dynamic viscosity
ρ	Density

I. Introduction

The research of electrified aircraft propulsion (EAP) concepts has shown its potential in reducing emissions, increasing efficiency, and decreasing aircraft noise. [1] In any development of an EAP vehicle there will need to be electric machines, and these electric machines produce waste heat whose magnitude is a function of machine power and efficiency. While determining what to do with this heat is a topic of ongoing research, the conventional method is to reject this waste energy to the surrounding air. [2] Passive and active cooling systems are two methods that can be used to perform this heat rejection. For this paper, a passive or air-only cooling system refers to a heat sink that pulls heat from a load through a conductive interface, while an active or liquid/air cooling system refers to a system containing a coolant that gathers heat from a load and then transports it to a heat exchanger for rejection to air. All studies are completed using the Heat Transport System Simulation(HeaTSSPy) package, which is a python-based modular thermal management system (TMS) performance and weight estimation code. This paper details the development of HeaTSSPy's heat exchanger code, whereby the theory and/or sources used are discussed. In addition, a validation case for a passive heat sink is discussed alongside some build cases. Lastly, a study is conducted comparing the passive and active systems across a variety of rejection heat loads.

Developed by NASA, the HeaTSSPy software package is a modular framework created to size and simulate the performance of a TMS. While general, this software has been developed to facilitate the creation of EAP concept models and considers passive and active heat rejection systems. HeaTSSPy was written in Python and uses the OpenMDAO framework for TMS sizing optimizations.[3] The development of HeaTSSPy has taken place over several years and methods used within the package are demonstrated on a number of NASA EAP research vehicles including the Single-Aisle Turboelectric Aircraft with an Aft Boundary Layer propulsor (STARC-ABL), Parallel Electric-Gas Architecture with Synergistic Utilization Scheme (PEGASUS), a Revolutionary Vertical Lift Technology (RVLT) concept tiltwing, and an RVLT concept quadrotor. [4,5,6] Theory for the active system used within HeaTSSPy has been documented within several publications. [7,5] This paper lays out the general framework, describes the previous methods in work completed for the active systems, and details the passive system methods and modeling theory.

Validation of the HeaTSSPy passive finned heat sink module was performed by comparing the total thermal resistance predicted by HeaTSSPy, which involves correlations standard rectangular duct correlations and those from

Teertstra, with two heat sinks analyzed with computational fluid dynamics (CFD). Ideally comparisons would be performed with results from hardware testing. Here, CFD is used as an intermediate step to give some confidence to the accuracy of the methods. In this case, comparisons are made using a simple heat sink design and an earlier form of the X-57 heat sink; whereby, sink dimensions were transferred to HeaTSSPy and the resulting thermal resistances are compared.

Subsequently, a passive and active cooling system study is completed to demonstrate the capability of HeaTSSPy by designing and comparing two cooling systems. In this study the two cooling systems are designed to reject a range of heat loads at aircraft take-off conditions, which are 0 ft altitude and 0.25 MN, and optimized for minimum weight. It should be noted that this is a preliminary study, and that load types and changes due to cooling arrangement are not accounted for. Heat loads enter each cooling system directly at a constant temperature and the cooling systems are designed to reject this heat to the surrounding air.

Subsequent sections of this paper detail passive and active TMS methods, two passive TMS validation cases, and a TMS study. Specifically, a review of TMS simulation methods is shown in Section II. Details of the high-fidelity validation cases with low-fidelity comparisons are in Section III. System study results are then found in Section IV. Finally, summary and conclusions are given in Section V.

II. Software definition

The HeaTSSPy software is a modular TMS design package used to size and simulate the performance of custom TMS. Simulations are developed with sets of components or elements that come in two major types: fluid transport and thermal transport. With fluid transport, components are combined into fluid circuits where each component will input and output a standardized range of fluid properties such as mass flow, pressure, temperature, convection coefficient, density, dynamic viscosity, and kinematic viscosity. Fluid properties can be imported using input files or using the CoolProp software package. [8] During operation, fluid properties for a current state are defined based on pressure and temperature or enthalpy. Thermal transport is considered using thermal junctions that use appropriate thermodynamic transport equations. The HeaTSSPy package also considers two types of TMS: passive and active. Air transport components are common to the two systems, and include inlets, fans, ducts, and nozzles that are used to channel compressible gas into and out of a heat transfer interface. In some cases, a puller fan is also included to increase low speed mass flow. Validation for package elements was performed in several different methods. The active heat exchanger equations and results were compared against examples within the appendix of Ref. [9]. The passive heat sinks are validated with comparisons against CFD results, as shown below in this paper.

HeaTSSPy makes use of the OpenMDAO framework with custom functions created to facilitate the creation of a TMS. A FlowStart element is used to set the fluid properties and then special connect_flow functions are used to transfer these properties to the other elements. An example of the syntax is shown in Figure 1. Here a FlowStart element is created for a coolant (FS_coolant), in this case water, where W is mass flow, T is temperature, and P is pressure. FS_coolant is then fed into a load where the coolant picks up heat. Another flow start is created for air (FS_air) and then the load and FS_air outputs are fed into a plate-fin heat exchanger (HeatExchanger with type defined by hex_def) with side 1 being air and side 2 being coolant. The size of this heat exchanger is then set with the inputs; width, height1, and height2. These sizing parameters can then be optimized for a specific mission to guarantee the coolant exit flow drops to a reasonable temperature.

```
import openmdao.api as om
import heatsspy.api
from heatsspy.include.props_air import air_props
from heatsspy.include.props_water import water_props
from heatsspy.include.HexParams_PlateFin import hex_params_platefin

p = om.Problem()

p.model.add_subsystem('FS_coolant', FlowStart(thermo='file',fluid= water_props()),
                      promotes=[('W','Wcoolant'), ('T','Tcoolant'), ('P','Pcoolant')])

p.model.add_subsystem('load', HE_1side(thermo='file', fluid= water_props()), promotes=[ '*'])
connect_flow(self, 'FS_coolant.Fl_O', 'load.Fl_I')

p.model.add_subsystem('FS_air', FlowStart(thermo='file',fluid= air_props()),
                      promotes=[('W','Wair'), ('T','Tair'), ('P','Pair')])

p.model.add_subsystem('HeatExchanger', HE_2side(fluid1= air_props(), thermo1='file',
                      fluid2= fluid= water_props(), thermo2='file',
                      hex_def= hex_params_platefin()),
                      promotes_inputs=[width, height1, height2])
connect_flow(self, 'load.Fl_O', 'ACC.Fl_I2')
connect_flow(self, 'FS_air.Fl_O', 'ACC.Fl_I1')
```

Figure 1. Example HeaTSSPy syntax for a two-flow heat exchanger.

A. Active cooling system (heat exchanger) definition

In developing an active cooling system one or more coolants are used to transport heat to a liquid/air heat exchanger. An example of this type of TMS is shown in Figure 2. Here two loops of fluid are used, coolant and air. The coolant loop is driven by a pump first through a heat load, and then driven through a heat exchanger with two sides. Air is pulled through the inlet by a fan, then the heat exchanger's cold side, and finally exhausted through a nozzle. The theory for each element is detailed within Ref. 7. A brief overview of the heat exchanger component is given below, and an example of a crossflow heat exchanger is shown in Figure 3.

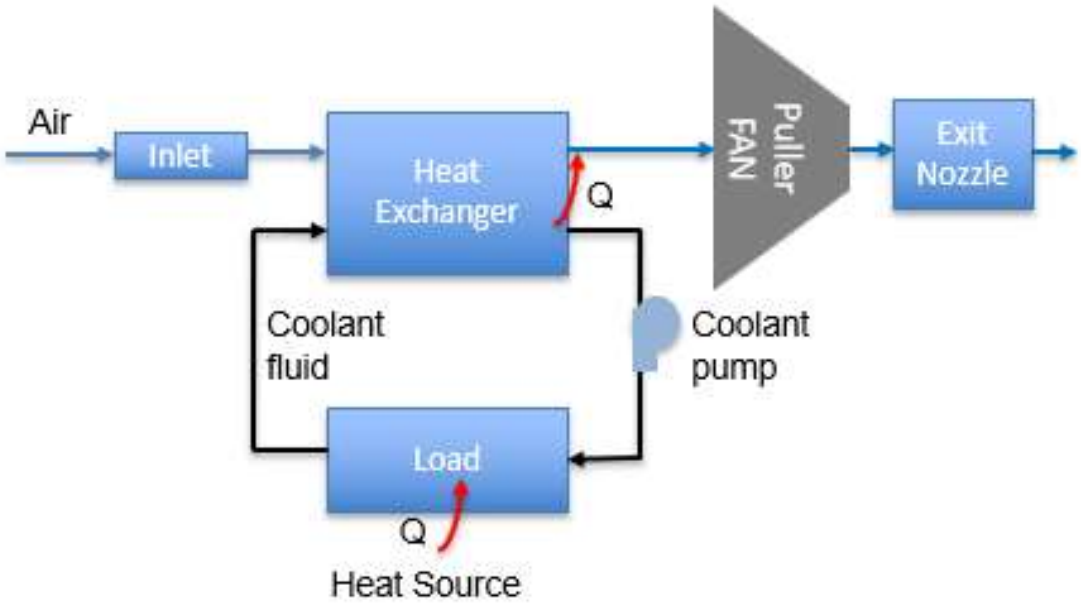

Figure 2. Active cooling system with a coolant loop moving load heat to a heat exchanger to be transferred to air.

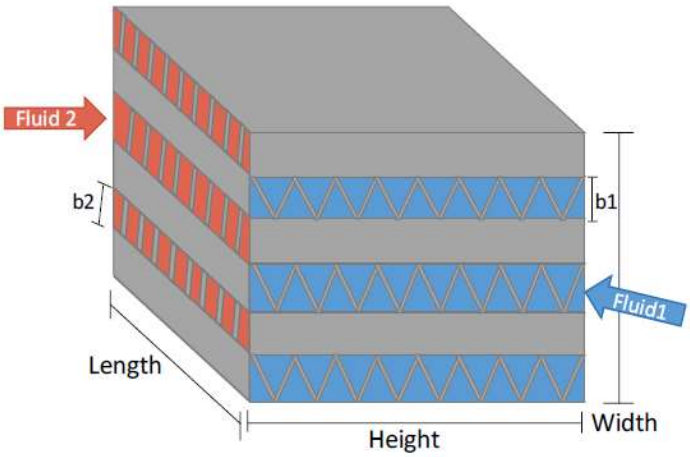

Figure 3. Plate-fin compact crossflow heat exchanger, with plate spacing denoted by "b".

The heat exchanger element updates flow properties using the state of the input fluids, mass flow, pressure, and temperature. [10] Heat exchanger performance within HeaTSSPY can be defined with a constant heat rejection (Q), constant effectiveness (ε), AUqV maps (defined in Ref. 5, with AUqV broken down as area (A), unit thermal conductance (U), and volume (V) defined as a function of mass flow and temperature), or characterization files with heat exchanger surface parameters defined in a combination of empirical and analytic methods. In both the AUqV and characterization file methods, heat exchanger effectiveness is estimated based on the heat exchanger flow arrangement as a function of NTU (Number of Transfer Units) and the capacity rate ratio. A simple expression for this is given in Eqns. 1-4 for a 2-sided heat exchanger, where Q is the heat transfer rate, C_{min} is minimum capacity rate, C_{max} is maximum capacity rate, $T_{h,i}$ is input hot side temperature, $T_{c,i}$ is input cold side temperature, ε is the effectiveness, and NTU is number of transfer units. [10, p.687], [5] Within Eqn. 2, effectiveness is defined as a function of NTU and the capacity rates and can be set by the flow arrangement type.

$$Q = \varepsilon * C_{min} * \left(T_{h,i} - T_{c,i}\right) \tag{1}$$

$$\varepsilon = function\left(NTU, \frac{C_{min}}{C_{max}}\right) \tag{2}$$

$$NTU = \frac{AU}{C_{min}} \tag{3}$$

$$\frac{1}{AU} = \frac{1}{AUqV_1 * V} + \frac{1}{AUqV_2 * V} \tag{4}$$

Characterization files make use of heat exchanger physical properties and empirical relations for thermal characterization. Each side of the heat exchanger is defined with several generalized surface criteria dictated by what type of heat exchanger it is. These criteria can be scaled up and down to change the size or number of plates of the heat exchanger. Currently, fin-plate or strip-fin heat exchangers are considered with parameters including fin pitch, plate spacing, total heat transfer area per volume, flow passage hydraulic radius, fin thickness, and fin area. These files also include specific empirical relationships related to the heat transfer coefficient. These methods are based on those laid out within the text Compact Heat Exchangers by Kays and London. [9] During design, the heat exchanger surface types (2 for a two-flow heat exchanger, i.e. air and water) are used to determine the generalized physical criteria (such as total heat transfer area per volume) then brought together with heat exchanger size (length, height), number of surface stacks (width), and fluid properties to calculate heat exchanger performance. For this calculation, the heat transfer coefficient is determined by a Colburn j factor, empirically determined from Reynolds number, or a Nusselt number. This heat transfer coefficient is combined with fin effectiveness and overall surface effectiveness to

calculate the AU term and heat rejected is calculated using Eqns. 1-3. One should note, this method bypasses the AUqV calculation shown in Eq. 4. Flow arrangement for the effectiveness calculation is also set explicitly as an equation within the characterization file. Heat exchanger pressure drop is determined using an extended Darcy-Weisbach equation with an empirically determined friction factor that accounts for inlet and exit pressure losses and changes in fluid density. [7], [9, p.36] An example of a heat exchanger characterization file is located within the Appendix.

Heat exchanger performance can also be estimated with an AUqV map. [5] This map is generated by running a wide variety of mass flows at different temperatures through a designed heat exchanger. The map for the designed heat exchanger is subsequently generated by back-calculating AU and then normalizing by its volume. This technique enables the size (number of surface stacks or width, height, and length) of the heat exchanger to be updated by changing the volume. These assumptions result in a constant heat exchanger aspect ratio (ratio of the height to width to length) with a weight that scales with the total volume. The total heat rejected by the heat exchanger then becomes a function of the coolant mass flows, temperatures, volume, and flow arrangement (by default crossflow), Eqns. 1-4. When sizing with the AUqV maps, the design rejected heat may change, but it is assumed that the application of the heat exchanger remains the same at the design point. This means any temperature limits, environmental conditions, and coolant types are the same as the developed map. Once the design is completed and map is scaled, the system may be run off design with representative heat exchanger effectiveness. The pressure drop across the heat exchanger is considered with another empirical look up table based on a reference mass flow that can be scaled to the design point. In practice, an AUqV map should be generated by first running an optimization with the characterization files or by obtaining heat exchanger test data. An example of an AUqV map is shown in Figure 4.

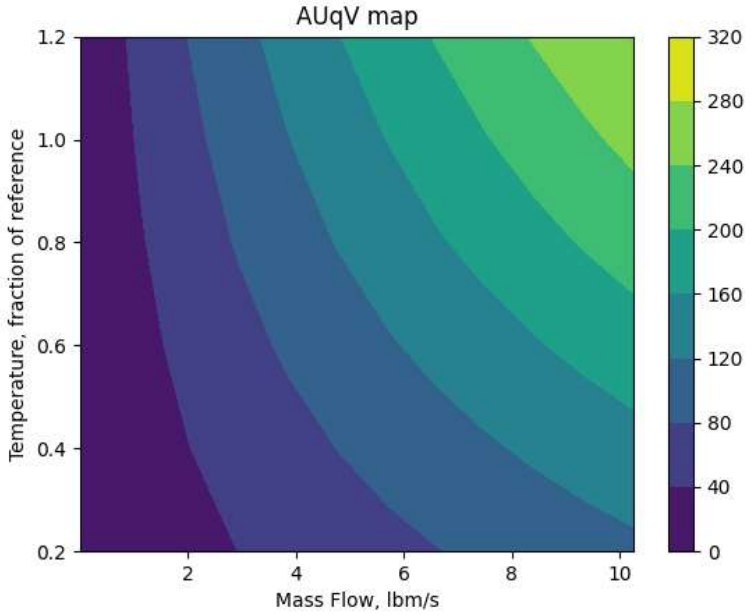

Figure 4: 2-side heat exchanger map, AUqV (Area*UnitThermalConductance/Volume).

B. Passive cooling system (heat sink) definition

The passive system is defined by a finned heat sink. The geometry of the heat sink, which includes straight rectangular fins, is shown in Figure 5.

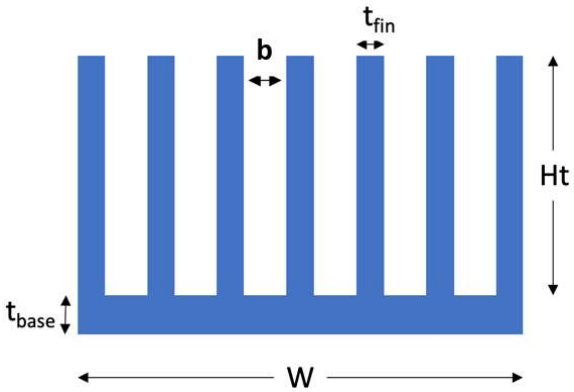

Figure 5: Heat sink geometry, where airflow and length (L) is into the page.

In the thermal management system, it is assumed that the heat sink base will have a conductive interface with a component that needs to be cooled (active junction). The performance of the heat sink is characterized by the thermal resistance of the geometry, the pressure drop of the fins, and the weight of the system. The thermal resistance of the heat sink is determined by equations from Reference 10. For a ducted finned heat sink with nonzero contact resistance, the resistance network is shown below in Figure 6:

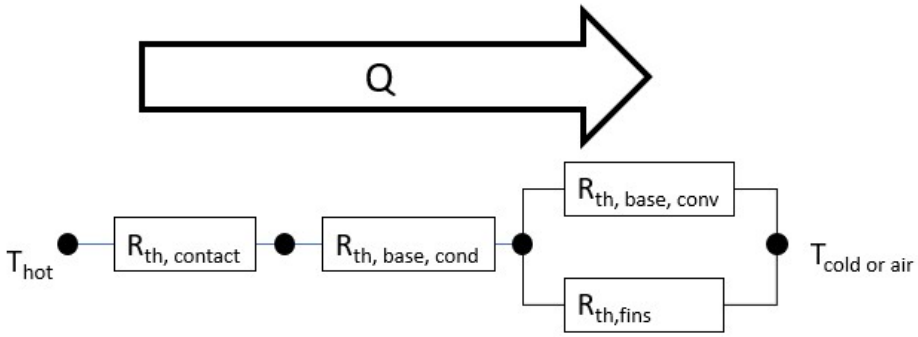

Figure 6: Thermal resistive network of the heat sink.

Where $R_{th,contact}$ is the contact resistance, $R_{th,base,cond}$ is the resistance of the sink base due to conduction, $R_{th,base,conv}$ is resistance due to convection at the base of the channels formed by the fins, and $R_{th,fins}$ is the fin resistance as a result of convection. In combining the parallel branch into an effective resistance term for convection of the fins and base, the simple resistance network reduces into a series circuit:

$$R_{th,total} = R_{th,contact} + R_{th,base,cond} + R_{th,fin\ system} \tag{5}$$

The resistance due to conduction in the base of the heat sink is determined:

$$R_{th,base,cond} = \frac{t_{base}}{k_{fin} LW} \tag{6}$$

Instead of applying individual equations for the base and fin convection terms, the total resistance from the fin system can instead be determined using an efficiency approach: [10]

$$R_{th,fin\ system} = \left(\frac{1}{R_{th,base,conv}} + \frac{1}{R_{th,fins}}\right)^{-1} = \frac{1}{\eta_o h A_t} \tag{7}$$

Where η_o is the overall surface efficiency of the finned system, h is the convection coefficient within the channels, and A_t is the total wetted surface area of the heat sink. This total wetted area is simply decomposed into two major contributions: [10, p.170]

$$A_t = (N_{fin} - 1)A_f + A_b \tag{8}$$

Where A_f is the total wetted surface area of a single fin, and A_b is the total area of the exposed base. These areas are easily determined using Figure 5:

$$A_f = 2H_t L \tag{9}$$

$$A_b = \left(W - N_{fins} t_{fin}\right)L \tag{10}$$

As seen in equation 7, when the total wetted surface area increases then the total resistance of the finned array decreases; however, this usually entails increasing the number of fins (N_{fin}) in the limit of creating smaller channels. With the area terms now defined, the overall efficiency of the finned system can now be determined in terms of the fin efficiency, η_f: [10, p. 171]

$$\eta_o = 1 - \frac{N_{fin} A_f}{A_t}(1 - \eta_f) \tag{11}$$

In this paper, we will assume the fins have an adiabatic tip condition; therefore, the fin efficiency can be estimated using the following equation: [10, p. 165]

$$\eta_f = \frac{\tanh(mH_t)}{mH_t} \tag{12}$$

where [10, p. 142]

$$m \equiv \left(\frac{hP_c}{k_{fin} A_c}\right)^{\frac{1}{2}} \tag{13}$$

Where P_c is the perimeter of the fin cross-section seen from above.

C. Characterization of the Heat Transfer Coefficient

Based on equation 7, one can see that the heat transfer coefficient is instrumental in determining thermal resistance of the heat sink; therefore, it is key to adequately model this parameter in HeatSSPY. For ducted flow, one must consider both laminar and turbulent flow in addition to developing flow. For laminar, hydrodynamically and thermally fully developed flow, the Nusselt number correlation becomes solely a function of the duct aspect ratio. For a constant heat flux case, the following correlation may be used to determine the local Nusselt number: [11]

$$Nu_{D_h} = 8.235(1 - 2.042A_R + 3.085A_R^2 - 2.477A_R^3 + 1.058A_R^4 - 0.186A_R^5) \tag{14}$$

Where Nu_{D_h} is the local Nusselt number based on hydraulic diameter, A_R is the duct aspect ratio, which is defined as $A_R \equiv b/Ht$ in this paper.

The hydraulic diameter for the fin channels can be determined by recognizing they're rectangular ducts:

$$D_h = \frac{2bH_t}{b + H_t} \tag{15}$$

For fully-developed turbulent flow, the local Nusselt number correlation becomes sensitive to surface roughness: [12, pp. 665-667]

$$Nu_{D_h} = \frac{\left(\frac{f_{fd}}{8}\left(Re_{D_h} - 1000\right)Pr\right)}{1 + 12.7\left(Pr^{\frac{2}{3}} - 1\right)\sqrt{\frac{f_{fd}}{8}}} \tag{16}$$
$$2300 \leq Re_{D_h} \leq 5x10^6, 0.5 < Pr < 2000$$

Where f_{fd} is the fully develop friction factor, and Pr is the Prandtl number.
The average Nusselt number in the channels can be estimated using the following approximation: [12, p. 668]

$$\overline{Nu}_{D_h} \approx Nu_{D_h,fd}\left[1 + \left(\frac{L}{D_h}\right)^{-0.7}\right] \tag{17}$$

It should be noted that the above correlations are specifically for ducted flow and not finned heat sinks. In [13], Teertstra et. al. developed a correlation for high aspect ratio finned sinks with no flow bypass. This correlation consists of a composite model spanning both the developing and fully-develop regimes:

$$\overline{Nu} = \eta_f Nu_i \tag{18}$$

$$Nu_i = \left[\left(\frac{Re_b^* P_r}{2}\right)^{-3} + \left(0.664\sqrt{Re_b^*}P_r^{\frac{1}{3}} \times \sqrt{1 + \frac{3.65}{\sqrt{Re_b^*}}}\right)^{-3}\right]^{-\frac{1}{3}} \tag{19}$$
$$0.26 \leq Re_b^* \leq 175$$

Where, $Nu_i$is the idealized Nusselt number, and Re_b^* is the channel Reynolds number. In this correlation, the channel Reynolds number is based upon channel width:

$$Re_b^* \equiv Re\frac{b}{L} = \frac{Ub^2}{\nu L} \tag{20}$$

With the average channel Nusselt number now known, the average heat transfer coefficient can be determined:

$$\bar{h} = \overline{Nu}\frac{k_{air}}{L_{characteristic}} \tag{21}$$

Where $L_{characteristic}$ is either the width of the channel or the hydraulic diameter depending on whether Teertstra's correlation is used or the ducted flow correlations, respectively.

Now that the average channel heat transfer coefficient is known, the total resistance due to convection in the heat sink is now defined by equations 5, 6 and 7.It now follows that the heat (Q) transferred through the heat sink to the flowing air can now be approximated using the following relation: [10, pp.98-99]

$$Q = \frac{T_{hot} - T_{cold}}{R_{th,tot}} \tag{22}$$

where T_{hot} and T_{cold} are average heat sink interface and incoming air temperatures, respectively.

D. Pressure Drop

Another performance metric that must be determined is the pressure drop of the system. The pressure drop (dP) of a heat sink with rectangular channels can be estimated using the Darcy-Weisbach formulation: [14]

$$\Delta P = \frac{1}{2}\rho V_{ch}^2 (f_{app}\frac{L}{D_h} + K_c + K_e) \tag{23}$$

where ρ is the fluid density, V_{ch} is channel velocity, f_{app} is the apparent friction factor, D_h is the channel hydraulic diameter, K_c is contraction loss coefficient, and K_e is expansion loss coefficient.

The loss coefficients can be estimated using correlations on sudden contraction and expansion: [14, p. 394]

$$K_c = 0.42\left(1 - \frac{D_{h,small}^2}{D_{h,big}^2}\right) for\ \frac{d}{D} \leq 0.76 \tag{24}$$

$$K_e = \left(1 - \frac{D_{h,small}^2}{D_{h,big}^2}\right) \tag{25}$$

Where $D_{h,small}$ refers to the hydraulic diameter of the smaller channel, and $D_{h,big}$ refers to the hydraulic diameter of the bigger duct (inlet/outlet). These minor losses are rather conservative so they may cause predictions in this paper to be higher than expected.

As was the case for the Nusselt number correlations, the flow friction factor depends on the flow regime. For laminar flow, the apparent friction factor (also known as "average") is estimated using the following equation: [12, p. 653]

$$f_{app} \approx \frac{4}{Re_{D_h}}\left[\frac{3.44}{\sqrt{L^+}} + \frac{\frac{1.25}{4L^+} + \frac{f_{fd}Re_{D_h}}{4} - \frac{3.44}{\sqrt{L^+}}}{1 + \frac{0.00021}{(L^+)^2}}\right] \tag{26}$$

Where the laminar, fully develop friction factor is given by the following equation:

$$f_{fd} = \frac{96}{Re_{D_h}}(1 - 1.3553A_R + 1.9467A_R^2 - 1.7012A_R^3 + 0.9564A_R^4 - 0.2537A_R^5) \tag{27}$$

The dimensionless length, L^+, is determined by normalizing the heat sink length by the hydraulic diameter and Reynolds number:

$$L^+ = \frac{L}{D_h Re_{D_h}} \tag{28}$$

This channel Reynolds number is defined in terms of the hydraulic diameter and the properties of the fluid:

$$Re_{D_h} = \frac{\rho V_{ch} D_h}{\mu} \tag{29}$$

For a known roughness,e, the turbulent fully developed friction factor can explicitly be determined using the following correlation: [12, p. 654]

$$f_{fd} = \left\{-2.0\log_{10}\left[\frac{2e}{7.54D_h} - \frac{5.02}{Re_{D_h}}\log_{10}\left(\frac{2e}{7.54D_h} + \frac{13}{Re_{D_h}}\right)\right]\right\}^{-2} \tag{30}$$

For turbulent flow, the apparent friction factor can be estimated using the following approximation: [12, p. 655]

$$f_{app} \approx f_{fd}\left(1 + \left(\frac{D_h}{L}\right)^{0.7}\right) \tag{31}$$

With the pressure drop now defined, a heat sink's performance can now be evaluated in the context of the resulting drag or pumping loss it might entail. Within the context of optimization codes, such as HeatSPPY, such a constraint should prevent the solver from trending towards an ever-increasing number of fins. With the pressure drop correlations now known, the objective function can be used to design a more pragmatic heat exchanger whose final performance metric of mass can now be assessed. This mass is easily determined after settling upon the number of fins in addition to the overall geometry of the heat exchanger by multiplying the solid volume by the density of the sink material.

III. High Fidelity Comparisons

A. Validation: A Simple Heat Exchanger

For the purposes of validation, a simple finned heat sink model was first generated in Solidworks and then analyzed in ANSYS Fluent for various inlet speeds. This finned heat sink, which has the same form as that shown in Figure 5, had a 10-Watt boundary condition applied to the bottom and a 20℃ inlet temperature condition. The parameters of this geometry are shown below in Table 1:

Table 1: Finned Heat Sink Parameters

Parameter	Value
L	25 mm
W	8 mm
Ht	20 mm
N_{fin}	6
t_{base}	3 mm
t_{fin}	0.5 mm
b	1.00 mm

The total thermal resistance of the heat sink and its pressure drop were estimated using the above formulations and then compared to CFD results for a range of far inlet velocities in order to capture both the laminar and turbulent regimes. The results of thermal resistance analysis is shown below for $1\ m/s \leq V \leq 20 m/s$:

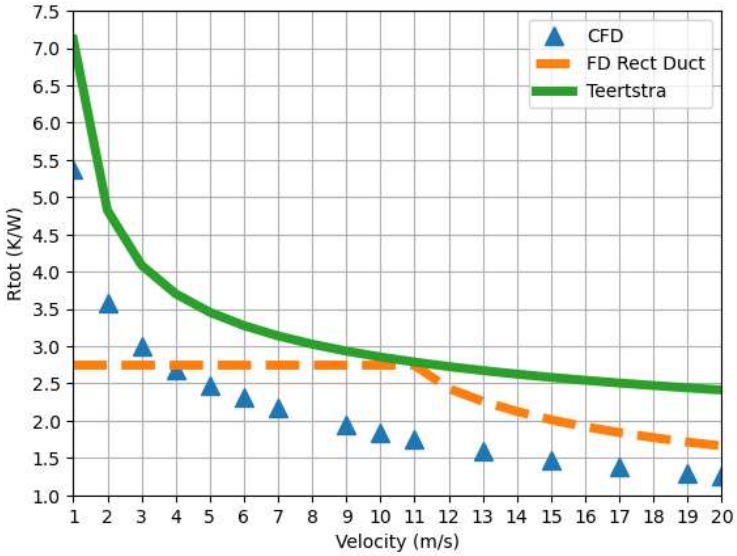

Figure 7: Comparison of total thermal resistance between CFD and Correlations.

The above Figure 7 plot shows that the CFD results are bounded above by the Teertstra correlation, which also follows the same shape as the CFD results. rectangular duct correlation, on the other hand, performs rather poorly until the flow velocity increases. The reason for this can be gleaned from the left part of the plot where the rectangular duct correlation provides a constant resistance for various inlet velocities. Before 11 m/s, the flow is still in the laminar regime – this means that the Nusselt number correlation is constant based upon the above formulations. The reason why Teertstra's correlation follows the same shape as the data is due to it covering both the developing and fully-develop flow regimes. Therefore, the fully-developed correlation used in this paper for a rectangular duct is insufficient and must be revised to include developing flow. The sheer magnitude of these differences can be seen in Table 4 in the Appendix.

Unlike the Nusselt number correlations, the pressure drop formulations discussed in this paper do cover the developing flow regime. This can be seen in the below figure where the code's estimates agree rather well with the CFD data.

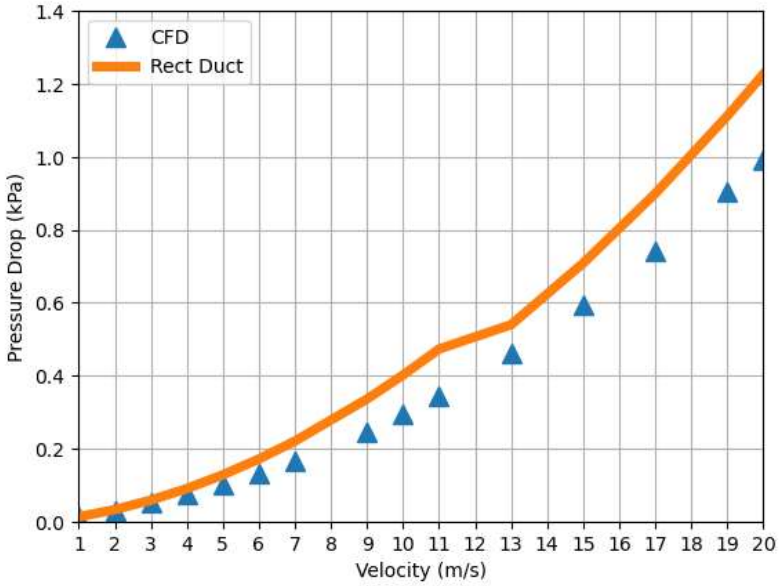

Figure 8: Pressure Drop Comparison between CFD and HeatSSPY.

To get a better idea of how well the pressure drop formulations fair, the data from figure 8 is tabulated below alongside percent differences. Here, one can see the data diverges the higher the inlet velocity becomes. This is most likely due to the conversative contraction and expansion correlations used in this paper. Since they are rather conservative, they'll become more prominent the higher the flow velocity becomes. Data from this study is located within Table 5 in the Appendix.

B. Validation: X-57 Heat Sink

An older design of the X-57 heat sink, which is shown below in Figure 9, was also used to validate the methods used in HeatSSPy. A high-fidelity model was similarly generated in ANSYS Fluent using the same techniques as the above section. In this case, an aluminum heat sink (shown below), is given a 1080 W boundary heat load with a 63℃ inlet condition flowing at 0.085 kg/s. The parameters for this heat sink are also shown below in Table 2.

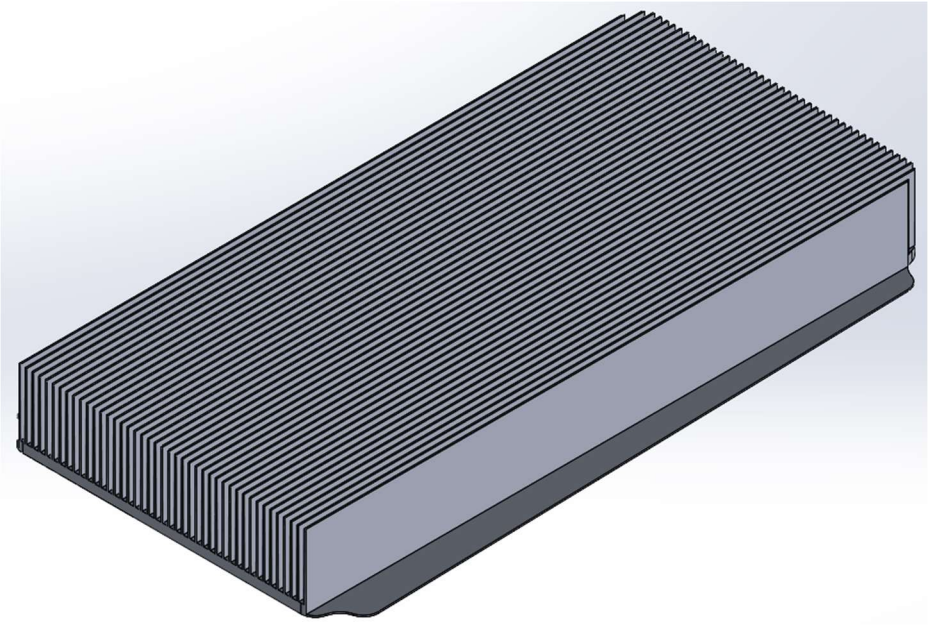

Figure 9: X-57 heat sink analyzed in this paper.

Table 2: X-57 parameters

Fin parameter	Value
L	388.60 mm
W	179.50 mm
Ht	41 mm
N_{fin}	43
t_{base}	6 mm
t_{fin}	1.11 mm (average)
b	3.14 mm (average)

To cut down on computational time, the heat sink was cut alongside its plane of symmetry and then evaluated with the fluid domain having its inlet and outlet extended for convergence purposes. The temperature profile borne from this analysis is shown below in Figure 10:

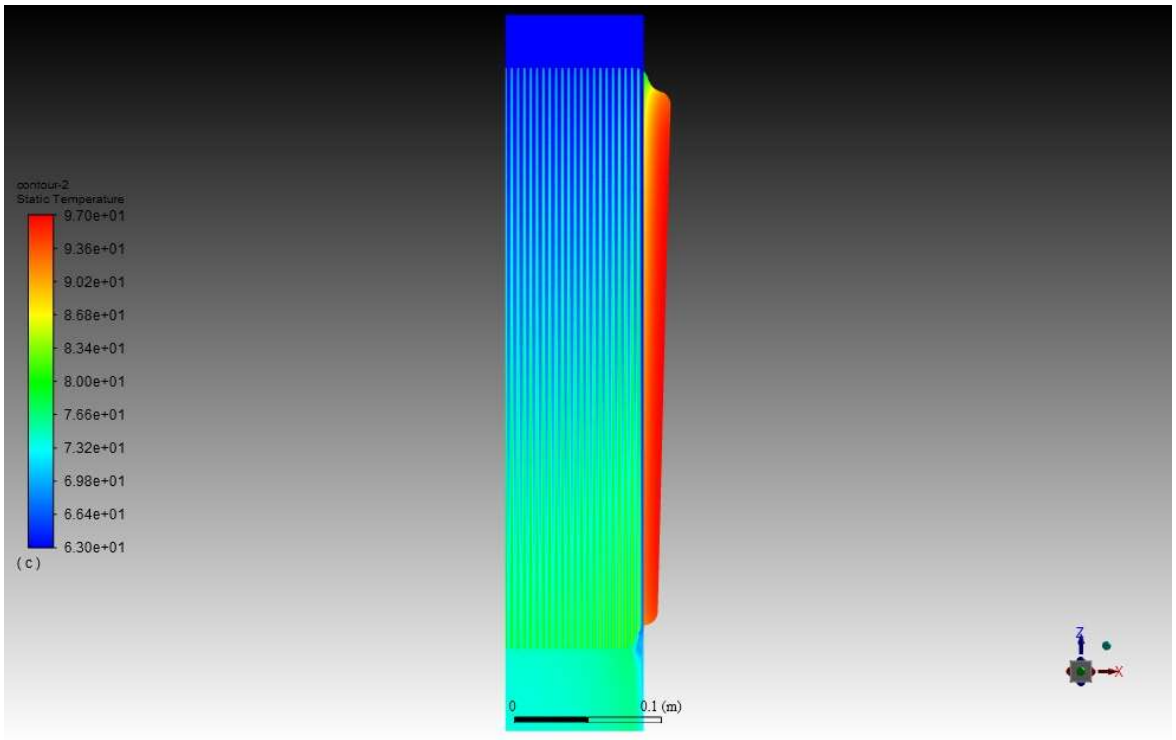

Figure 10: Heat sink temperature distribution.

The total resistance of this heat sink can now be determined after evaluated surface average temperature at the base ($T_{avg,base}$):

$$R_{th,total} = \frac{\Delta T}{\mathrm{Q}} = \frac{(84-63)K}{1080\,W} \approx 0.0194\frac{K}{W} \tag{32}$$

The estimates from the the rectangular duct and Teertstra correlations are tabulated below in Table 3 alongside percent differences. The values generated by these correlations, as shown in Figure 7, bound the CFD result but are still quite different.

Table 3: Total Thermal Resistance Comparison: X-57 Sink

Method	R_{th}, $\frac{K}{W}$	% Difference
CFD	0.0194	0
Ducted	0.0138	33.73
Teertstra	0.0335	-53.31

IV. Comparing passive and active cooling systems

A simple comparison between the TMS designed with the active (Figure 2) and passive cooling systems (Figure 5) was completed to provide a demonstration study for HeaTSSPy and to showcase the weight change of the two systems as designed rejected heat is increased. This study was not conducted on any specific heat source, but rather considers the system required to reject a set amount of heat through its respective medium and with an assumed temperature. The hot medium for the active system is coolant and for the passive system is an active junction that is attached to the heat sink (where hot medium is set to 150℃). For both systems the cold medium is air with a temperature consistent with a very hot conditions at 65℃. Design of the passive system uses finned heat sink height, width, length, number of fins, and air mass flow as design variables then constrains these by the pressure drop, number of fins related to heat sink size, and Reynolds number. Convection coefficient for the finned heat sink uses the Teertstra methods. The design of the active system makes use of the characterization files method and uses heat exchanger

height, width (number of stacks), and length along with the mass flows of the air and coolant to develop the design. Constraints consider Reynolds number, pressure drop, temperature ratio, and heat exchanger aspect ratio. In both design cases, Reynolds number is limited to 3000 (where the system assumptions are validated for both methods), a pressure drop is limited to 1%, and a minimum weight is used as an objective. Total weights for each system are summed up by adding the individual system components. For the air system this is the heat sink calculated by multiplying the final size by the material density, assumed to be aluminum. For the active system the weight is calculated by summing the weights of the heat exchanger, coolant within the heat exchanger (in this case water), coolant pump, and 2 m of coolant line. Additional structures like inlet ducting, outlet ducting, load cold plate, or air puller fans are not considered in this study, however a 30% increase in system size is assumed to account for additional structures not accounted for explicitly.

Results of the optimizations are shown in Figure 11. Here the active system's response is linear with increase in weight of about 10% over the 4 kW increase in rejected heat. The passive system has a parabolic form with an increase of about 9x over the same heat range. These trends lead to the passive system to be the lowest weight solution up until about 1.5 kW after which the active system weighs less. As noted above these results do not consider the source of the heat. For the active system this means the method of adding the heat to the coolant in the first place is not considered. This omission would add a considerable amount of weight to the active system pushing the blue line higher. It can be noted however, that if the active system's weight was doubled by adding in load heat rejection to the coolant, the cross over point would shift to about 2.8 kW.

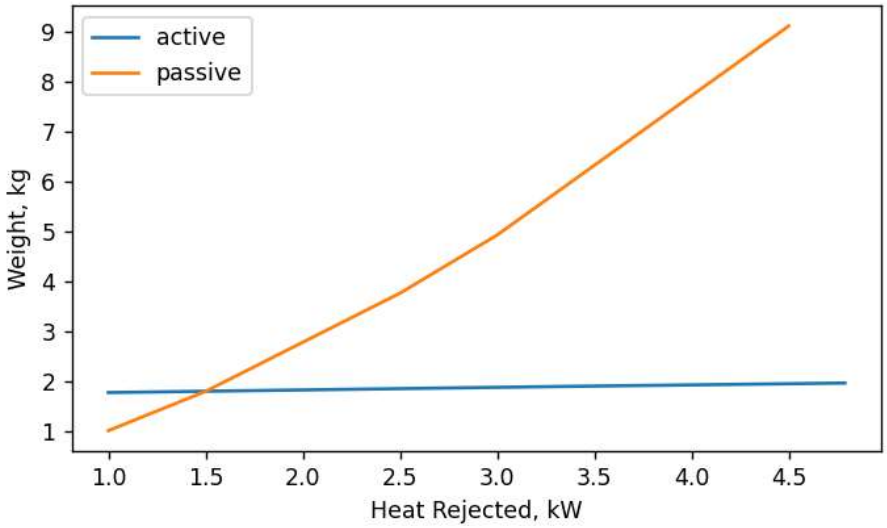

Figure 11: Optimization results for active and passive TMS.

V. Summary and Conclusions

This paper lays out the development of the HeaTSSPy software package and applies it to a system study comparing active and passive cooling methods. HeaTSSPy uses a modular approach to developing TMS packages with elements including heat exchangers (fluid-to-fluid, actively cooled), heat sinks (active junction-to-fluid, passive), ducts, pumps, inlets, fans, and nozzles. It is built in Python using the OpenMDAO framework and can use either the CoolProp software or property tables for thermal fluid definition. Heat exchanger performance is estimated using a constant energy, constant effectiveness, AUqV maps, or by defining the heat exchanger surface types using a formatted characterization file. Heat sink performance is calculated based on a rectangular fin geometry outlined in the methods in the paper. Validation of the heat sink model is performed on a simple heat exchanger and a complex heat exchanger based on a simplified X-57 design, showing comparisons for a high-fidelity simulation and two different methods of defining convection coefficient (rectangular duct and Teertstra). This validation shows that Teertstra correlation provides an upper bound on the CFD results and can be used as a conservative estimate in heat exchanger design. The rectangular duct formulations, as shown from the above results, will need to be updated to include the effects of developing flow in determining the heat transfer coefficient - this will be done in a future update of HeatSPPY. The pressure drop calculations, on the other hand, show a good trend with the test case; however, the diverging nature with increasing velocity should also be revisited. Finally, a simple demonstration study is presented that compares a passive

system with that of an active system showing system weight for each at varying rejected heat requirements. Results show that at low rejected heat the passive system weights less and at higher rejected heat the active system weighs less.

References

[1] Jansen, R., Bowman, C., Jankovsky, A., Dyson, R., Felder, J., "Overview of NASA Electrified Aircraft Propulsion Research for Large Subsonic Transports," AIAA 2017-4701, 2017 AIAA Power and Energy Forum, Atlanta, GA, 10-12 Jul., 2017.

[2] Moir, I., Seabridge, A., *Aircraft System: Mechanical, Electrical, and Avionics Subsystems Integration,* AIAA Education Series, John Wiley & Sons, Reston, VA, 2008.

[3] Gray, J., Moore, K.T., Naylor, B.A., "OpenMDAO: An Open Source Framework for Multidisciplinary Analysis and Optimization." 13th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA Aviation Forum, AIAA-2010-9101, Fort Worth, TX, Sept. 13–15, 2010.

[4] Chapman, J., Hasseeb, H., Schnulo, S., "Thermal Management System Design for Electrified Aircraft Propulsion Concepts," AIAA 2020-3571, AIAA Propulsion and Energy Forum, Virtual Event, Aug. 24-28, 2020.

[5] Chapman, J., Thomas, G., "Development and Integration of a Thermal Management Simulation for a Quadrotor Parallel Hybrid Propulsion System," AIAA 2021-3336, AIAA Propulsion and Energy Forum, Virtual Event, Aug. 9-11, 2021.

[6] Hendricks, E., Aretskin-Hariton, E., Chapman, J., Gray, J., Falk, R., "Propulsion System Optimization for a Turboelectric Tiltwing Urban Air Mobility Aircraft," ISABE-2019-24365, International Society for Air Breathing Engines, Canberra, ACT, Sept. 22, 2019.

[7] Chapman, J., Schnulo, S., Nitzche, M.," Development of a Thermal Management System for Electrified Aircraft," AIAA-2020-0545, AIAA Science and Technology Forum, Orlando, Fl, Jan 6-10, 2020.

[8] Bell, I., Wronski, J., Quoilin, S., Lemort, V., "Pure and Pseudo-pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp," Journal of Industrial Engineering Chemistry Research, vol. 53, number 6, pages 2498-2508, 2014.

[9] Kays, W.M., London, A.L., Compact Heat Exchangers, 3rd Ed., Krieger Publishing Co., Malabar, Florida, 1984.

[10] Incropera, DeWitt, Bergman, Lavine, *Fundamentals of Heat and Mass Transfer*, 7th Ed., John Wiley and Sons, 2007.

[11] Shah, R.K., London, A.L., *Laminar Flow Forced Convection in Ducts: A Source Book for Compact Heat Exchanger Analytical Data*, Academic Press, New York, 1978.

[12] Nellis, G., and Klein, S., *Heat Transfer*, Cambridge University Press, Cambridge, 2008.

[13] Teertstra, P., Yovanovich, M., Culham, J., Lemczyk, T., "Analytical Forced Convection Modeling of Plate Fin Heat Sinks," Fifteenth Annual IEEE Semiconductor Thermal Measurement and Management Symposium, San Diego, CA, 1999.

[14] White, F. M., *Fluid Mechanics*, 7th Edition, McGraw-Hill, New York, NY, 2008.

APPENDIX

Table 4: Total Thermal Resistance Comparison: CFD vs Correlations

$V_{inlet}, \frac{m}{s}$	R_{tot}(CFD), $\frac{K}{W}$	$R_{tot}^{Ducted}, \frac{K}{W}$	$R_{tot}^{Teerstra}, \frac{K}{W}$
1	5.380	2.744	7.115
2	3.586	2.744	4.823
3	2.996	2.744	4.087
4	2.684	2.744	3.703
5	2.477	2.744	3.455
6	2.315	2.744	3.276
7	2.180	2.744	3.138
9	1.947	2.744	2.934
10	1.844	2.744	2.855
11	1.748	2.744	2.786
13	1.588	2.260	2.676
15	1.470	2.012	2.581
17	1.377	1.840	2.505
19	1.300	1.713	2.441
20	1.266	1.661	2.412

Table 5: Pressure Drop Comparison: CFD vs Correlations

$V_{inlet}, \frac{m}{s}$	ΔP(CFD), Pa	ΔP(HeatSSPY), Pa	$\% \, Difference$
1	13.160	12.681	3.706
2	30.060	32.274	7.103
3	50.200	58.141	14.660
4	73.670	89.989	19.943
5	100.640	127.684	23.689
6	131.360	171.146	26.304
7	165.810	220.307	28.228
9	246.740	335.481	30.484
10	293.80	401.376	30.949
11	345.460	472.740	31.112
13	462.330	539.433	15.393
15	594.930	708.201	17.384
17	742.280	899.020	19.099
19	905.060	1111.743	20.496
20	992.170	1226.278	21.106

```
Example heat exchanger characterization file for a plate fin type heat exchanger.
#-------------------------------------------------------------------------------
# Heat Exchanger Characterization file (plate fin)
#-------------------------------------------------------------------------------
import numpy as np

class hex_params_platefin():
    #Define Surface strip fin plate fin surface (SFPFS) parameters fit from Kays and London figure 10-58 (air side)
    b1 = 5.08e-3 # plate spacing in meters
    beta1 = 2360 # Total heat transfer area/volume between plates in m**2/m***3
    r_h1 = 3.75e-4 # flow passage hydraulic radius in meters
    delta1 = 0.102e-3 # fin thickness in meters
```

```
AfqA1 = 0.850 # Fin area/total area

#Define Surface strip fin surface (SFS) parameters fit from Kays and London figure 10-61 (liquid side)
b2 = 1.91e-3 # plate spacing in meters
beta2 = 2490 # Total heat transfer area/volume between plates in m**2/m***3
r_h2 = 3.51e-4 # flow passage hydraulic radius in meters
delta2 = 0.102e-3 # fin thickness in meters
AfqA2 = 0.611 # Fin area/total area

a = 0.3e-3 #plate thickness in meters

alpha1 = b1*beta1/(b1+b2+2*a)
sigma1 = b1*beta1*r_h1/(b1+b2+2*a)
alpha2 = b2*beta2/(b2+b1+2*a)
sigma2 = b2*beta2*r_h2/(b2+b1+2*a)

# Heat exchanger material properties
k_material = 237. # thermal conductivity of Al, W/m2/K
rho_material = 2700. # density of Al, kg/m3

# Pressure variables
K_c1 = 0.40 # entrance coefficient SFPFS
K_e1 = 0.08 # exit coefficient SFPFS
K_c2 = 0.55 # entrance coefficient SFS
K_e2 = 0.65 # exit coefficient SFS

#-----------------------------------------------
# surface Colburn j factor values from curve fits
# for calculating convection coefficient (h) derived from Kays and London
#-----------------------------------------------
def get_j(self, Re, side_number):
   if side_number == 1:
      # surface 1 (SFPFS)
      self.j_1 = 0.3153*Re**-0.441
   else: # side_number == 2
      # surface 2 (SFS)
      self.j_2 = 0.0165*Re**-0.091

def get_j_partials(self, Re, side_number):
   if side_number == 1:
      # surface 1 (SFPFS)
      self.dj_dRe1 = - 0.3153*0.441*Re**-(1+0.441)
   else: # side_number == 2
      # surface 2 (SFS)
      self.dj_dRe2 = - 0.0165*0.091*Re**-(1+0.091)

#-----------------------------------------------
# surface friction factor (f) values from curve fits
# for calculating pressure drop (dP)
#-----------------------------------------------
def get_f(self, Re, side_number):
   if side_number == 1:
      # surface 1 (SFPFS)
      self.f1 = 3.0146*Re**-0.55
   else: # side_number == 2
      # surface 2 (SFS)
      self.f2 = 0.0264*Re**-0.119

def get_f_partials(self, Re, side_number):
   if side_number == 1:
      # surface 1 (SFPFS)
      self.df_dRe1 = - 3.0146*0.55*Re**-(1+0.55)
   else: # side_number == 2
      # surface 2 (SFS)
      self.df_dRe2 = - 0.0264*0.119*Re**-(1+0.119)
```

```
#-----------------------------------------------------------------
# Cross flow heat exchanger : Incropera, Table 11.3, eqn. 11.32
#-----------------------------------------------------------------
# Set effectiveness to NTU and capacity rate ratio relationship
#
def get_eff(self,NTU, CR):
    # Populate effectiveness
    self.effect = 1 - np.exp((1/CR)*NTU**0.22*(np.exp(-CR*NTU**0.78)-1))  #unmixed cross flow
#
# define partial derivatives
#
def get_eff_partials(self,NTU,CR):
    # Populate effectiveness partials wrt CR and NTU
    N = NTU
    E = np.exp(-CR*N**0.78)
    Em1 = np.exp(-CR*N**0.78)-1
    N22 = N**0.22
    self.deffect_dCR  =  np.exp(N22*Em1/CR)*(N22*Em1/CR**2 + N*E/CR)
    self.deffect_dNTU = -np.exp(N22*Em1/CR)*(0.22*Em1/CR/N**0.78 - 0.78*E*N**2.77566e-17)
```