Topology Optimization of Heating Chamber of Vaporizing Liquid Microthrusters

Silva, Marsil A. C.; Cordeiro Guerrieri, Daduí; Cervone, Angelo; Gill, Eberhard

Publication date
2018

Document Version
Accepted author manuscript

Published in
ESA Space Propulsion 2018 Conference, Seville, Spain

Citation (APA)
Silva, M. A. C., Cordeiro Guerrieri, D., Cervone, A., & Gill, E. (2018). Topology Optimization of Heating Chamber of Vaporizing Liquid Microthrusters. In *ESA Space Propulsion 2018 Conference, Seville, Spain* Article SP2018_00108

TOPOLOGY OPTIMIZATION OF HEATING CHAMBER OF VAPORIZING LIQUID MICROTHRUSTERS

M. A. C. Silva [(1)], D. C. Guerrieri, A. Cervone, E. Gill

Faculty of Aerospace Engineering, Delft University of Technology; Delft, The Netherlands.

[(1)]*Email: m.deathaydecostaesilva@tudelft.nl*

ABSTRACT: Vaporizing Liquid Microthrusters (VLM) have recently received attention as promising propulsion technology for highly miniaturized spacecraft due to its high thrust levels and low power consumption. This paper presents the results of numerical optimization of the parameters for the design of the heating chamber of VLMs that use water as the propellant. The optimization is aimed to increase the heat transfer coefficient of the heating chamber in order to maximize the heat convection while minimizing the heat and pressure losses from the inlet to the nozzle as well as the size of the device. The simulations are carried out in a combined environment using Computational Fluid Dynamics (CFD) and an optimization tool to run the algorithms. The results of the optimization are compared to the results of a comprehensive experimental campaign and are intended to be used in the next design of the VLMs produced by TU Delft that will fly on-board of a PocketQube.

KEYWORDS: Vaporing Liquid Microthruster, PocketQube, Optimization

1. INTRODUCTION

Micropropulsion has been recognized as one of the key development areas for the next generation of highly miniaturized spacecraft such as CubeSats and PocketQubes. It will extend the range of applications for this class of satellites to include missions where, e.g., formation flying, station keeping, or space debris maneuvers are required.

Many concepts of micropropulsion have been proposed during the last decades in order to provide these satellites with mentioned capabilities. Most of the systems are manufactured using MEMS (Micro Electro-Mechanical Systems) [1] fabrication technologies and generate thrust by ejecting gases or plasma at high velocities, for example liquid propellant microthrusters [2], solid propellant microthrusters [3], [4], and cold-gas microthrusters [5], [6]. Other concepts generate thrust by other means, for example accelerating a spray of particles [7] or using the solar radiation pressure [8].

An interesting option for CubeSats and PocketQubes are resistojets which use a resistive heater to heat up the propellant and eject it through a nozzle. In the context of MEMS micropropulsion systems there are two variants commonly found in the literature: the Low Pressure Microresistojet (LPM) [9–11] and the Vaporizing Liquid Microthruster (VLM) [12]. The VLM has received attention due to its ability to provide high thrust levels with relatively low power consumption. The thruster uses the gases generated in the vaporization to produce thrust using a nozzle. The vaporization is usually done by applying power to resistive heaters that could be integrated into the device or externally attached to it. The nozzle is usually a convergent-divergent nozzle that can accelerate the propellant to supersonic velocities.

This paper presents the numerical optimization of the parameters used to design the vaporization chamber of a VLM system. Water is used as propellant as it has been proved to be an interesting option for this kind of propulsion system [13]. The vaporization chamber contains pillars that help to increase the heat transfer to the fluid improving the efficiency of the vaporization. The goal of the optimization is to find the shape of the pillars inside the chamber that maximizes the heat transfer and minimizes the pressure drop. The system is implemented in an environment combining MATLAB and COMSOL. The optimization algorithms run in MATLAB defining the geometry used in COMSOL to solve the fluid dynamics and heat transfer problems. Four different optimization algorithms are tested and compared in terms of the best solutions found.

2. VAPORIZING LIQUID MICROTHRUSTER

The Vaporizing Liquid Microthruster considered in this paper is manufactured using MEMS and silicon technology. The details of the complete design and characterization of the devices produced at TU Delft are presented in [14]. The VLM system is composed of a tank to store the liquid propellant in this case water, a valve to control the fluid flow, and a thruster to vaporize the propellant and accelerate it to space.

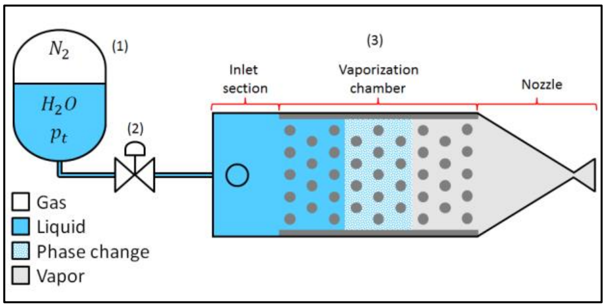

Figure 1 – Diagram of a VLM system showing a tank (1), a valve (2) and a thruster (3).

The thruster is composed by an inlet section where the fluid flows in, a vaporization chamber used to increase the enthalpy of the fluid to the boiling point, and a nozzle to accelerate the vapor. The chamber contains resistive heaters attached to the back side of the thruster chip. These heaters are made out of molybdenum and are deposited on the surface of the chip during the manufacturing.

3. OPTIMIZATION

3.1 Problem formulation

The goal of the optimization is related to two important aspects of the VLM system which are the energy efficiency and the propulsion efficiency. The former relates to the efficiency in the heat transfer process occurring inside the vaporization chamber. The latter relates to the friction losses in the fluid flow inside the vaporization chamber that affects the thrust and specific impulse. Friction losses are known to impact the nozzle performance as well but here we focus only on the chamber aspects.

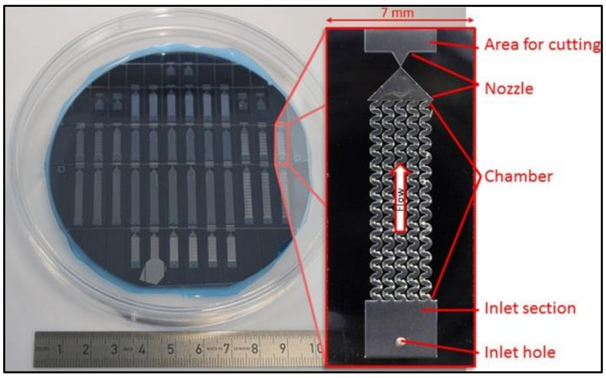

Figure 2 – Wafer and thruster made at TU Delft [14]. The form factor of the chip is 7 x 17 mm.

The optimization variables are the coordinates of the points defining the shape of the pillars inside the chamber. These pillars can have any shape so long as there are no intersecting or coincident lines.

In this paper, the total heat transfer coefficient h is used to measure the efficiency in the heat transfer and it is desired to be as high as possible, i.e. it has to be maximized. The propulsion efficiency is measured by the pressure drop caused by the pillars.

The problem is reduced to the area surrounding one of the pillars of the vaporization chamber. Figure 3 shows the diagram of the section used in the simulations. The average heat transfer coefficient is calculated for the entire section whereas the pressure drop is calculated from the left to the right boundaries.

The objective function, given by (1.1), is the sum of the heat transfer coefficient h and the pressure drop Δp normalized by the values given by a circular pillar (index 0) as the one in Figure 3.

$$F(h, \Delta p) = \frac{h_0}{h} + \frac{\Delta p}{\Delta p_0} \qquad (0.1)$$

The heat transfer coefficient has to be maximized whereas the pressure drop has to be minimized. Both objectives are given the same weight because they are normalized by a reference value which makes them comparable to each other.

The vector of optimization variables contains the coordinates x and y of the n points composing the edges of the pillar:

$$X = \{x_1, \dots, x_n, 0, y_1, \dots, y_{n-1}, 0\} \quad (0.2)$$

The first and last y values are fixed in order to keep the consistency of the model.

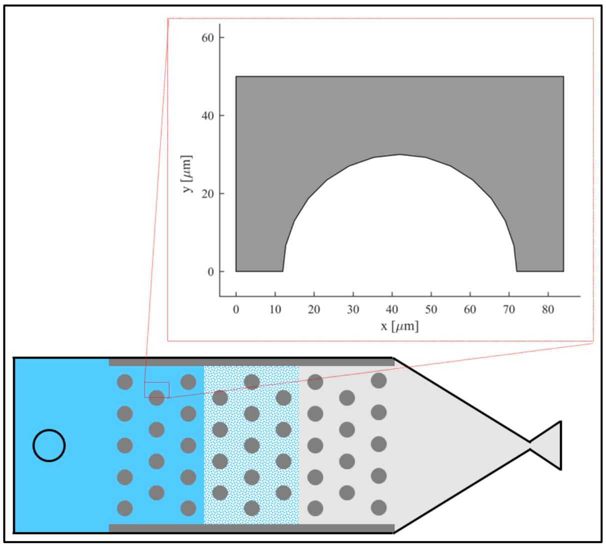

Figure 3 – Detail of the section used in the simulations. The liquid flows from left to right with a given flow rate. The top and bottom sides are symmetry boundaries.

3.2 Algorithms

Four evolutionary algorithms have been selected in order to evaluate their performance in solving the problem of topology optimization of the vaporization chamber of VLM systems.

The algorithms are Genetic Algorithm (GA) [15], Particle Swarm Optimization (PSO) [16], Evolutionary Strategy (ES) [17], Biogeography Based Optimization (BBO) [18] and Differential Evolution (DE) [19]. The details of each method are given in the following.

Genetic Algorithm: each candidate solution has a set of optimization variables that are treated as chromosomes and each variable is treated as a gene. Based on the fitness value of each solution (that is related to the objective function) the algorithm applies genetic operators, such as mutation and crossover, to create new evolved solutions and generate the best one after a stop criterion is met.

Genetic Algorithm – pseudocode

1: Initialize population
2: Compute fitness of all solutions
3: **while** *stop criteria not met* **do**
    4: Select candidates based on fitness
    5: Generate new solutions
    6: Compute fitness of new solutions
    7: Update the current population
8: Present the best solution

Particle Swarm Optimization: in this method the solutions are considered particles moving in the search space. Each particle moves with certain velocity according to its own best position, the best position of neighboring particles and the best position of the entire population of solutions. The solutions are classified by the values of the objective function.

Particle Swarm Optimization – pseudocode

1: Initialize population (position and velocity)
2: Compute fitness of all solutions
3: **while** *stop criteria not met* **do**
    4: Calculate new velocities for each particle
    5: Update the positions
    6: Compute fitness of all solutions
    7: Update the current population
8: Present the best solution

Evolutionary Strategy: this algorithm computes the solutions based on evolutionary operators such as mutation and selection. The solutions are randomly mutated and combined to generate new solutions and the best ones in terms of the objective function are kept for the next generations whereas the others are discarded.

Evolutionary Strategy – pseudocode

1: Initialize population
2: Compute fitness of all solutions
3: **while** *stop criteria not met* **do**
    4: Select random candidates to cross
    5: Generate new solutions
    6: Apply mutation and selection operators
    7: Compute fitness of new solutions
    8: Update the current population
9: Present the best solution

Biogeography Based Optimization: this method uses the concept of migration of species between habitats. Each solution is treated as a habitat that shares information with other habitats due to migration. The emigration and immigration rates are used to generate new solutions based on the existing ones. A mutation operator can also be applied.

```
Biogeography Based Optimization – pseudocode
1: Initialize population
2: Compute fitness of all solutions
3: while stop criteria not met do
    4: Compute migration rates
    5: Generate new solutions
    6: Apply mutation operator
    7: Compute fitness of new solutions
    8: Update the current population
9: Present the best solution
```

Differential Evolution: unlike the previous methods, this one is not directly related to a natural process such as migration but it linearly combines three existing solutions into a new solution that is either added to the current population of solutions or simply discarded.

```
Differential Evolution – pseudocode
1: Initialize population
2: Compute fitness of all solutions
3: while stop criteria not met do
    4: Select three random candidates
    5: Generate new solution
    6: Compute fitness of the new solution
    7: Update the current population
8: Present the best solution
```

3.3 Implementation

The algorithms were implemented in MATLAB using the codes presented in [18] and [20]. The optimization algorithms generate the points defining the geometry in Figure 3. These values are then updated to the model structure used by COMSOL which runs the CFD code that outputs the heat transfer coefficient and the pressure drop for the given geometry. The algorithm used in all methods is presented in Figure 4.
The objective function is then calculated using (1.1). The stop criterion is set as the number of iterations of the algorithm. This allows the comparison between the different methods also in terms of computational effort.

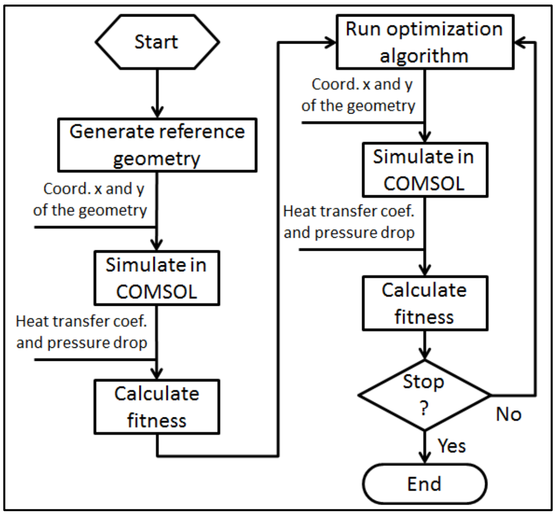

Figure 4 – Algorithm used in the simulations

4. RESULTS

All the algorithms were set to run 100 iterations and to start with the same randomly generated initial solutions. The population of solutions was set to 20 candidates. A list of all parameters is presented in Table 1.

Figure 5 shows the convergence plot of all the algorithms after 100 iterations. As it is clear, the GA ends with the best solution. The BBO and PSO end with solutions that are close in terms of the objective function but the geometries generated are very different as shown in Figure 6.

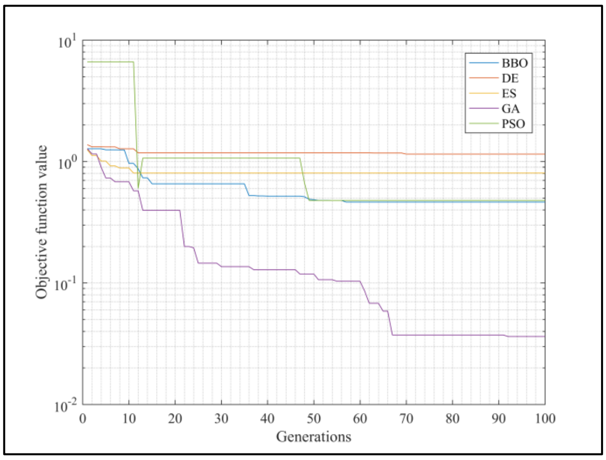

Figure 5 – Convergence plot of all algorithms.

Figure 7 shows the evolution of the best solution during the execution of the GA. We can see that some of the features of the geometry are actually evolving along time and not only being randomly generated.

Table 1 – Parameters used in the algorithms.

Algorithm	Parameter	Value
	Mutation rate	0.01
BBO	Max. migration rate	1
	Elite size	2
DE	Weighting factor (F)	0.5
	Crossover constant (CR)	0.5
ES	Offspring size	10
	Elite size	2
	Crossover type	single point
GA	Mutation rate	0.01
	Elite size	2
PSO	Inertial constant	0.25
	Elite size	2

Figures 8-10 show the plots of the velocity, pressure and temperature fields of the best solution.

Table 2 shows the values of the objective function together with the heat transfer coefficient and pressure drop for the best solutions of each algorithm. As we can see the GA presents the best values for both criteria and a much lower objective function than the other algorithms.

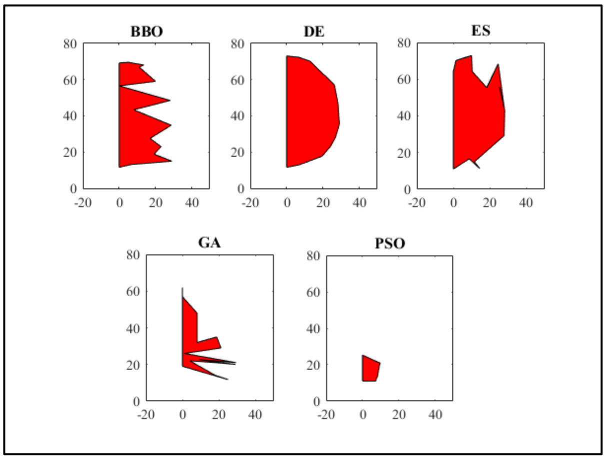

Figure 6 – Best solutions found by all algorithms.

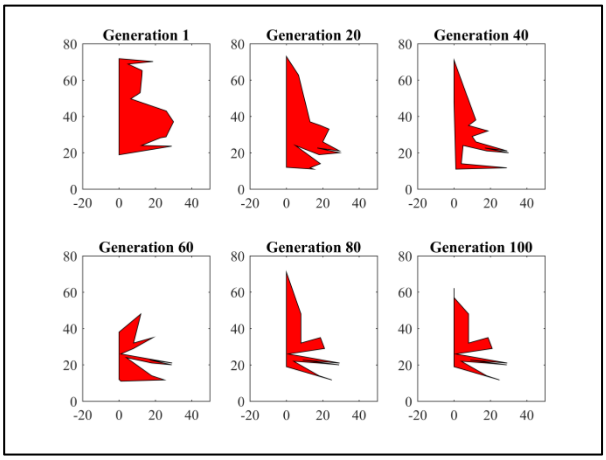

Figure 7 – Generations of the best solution found by the GA.

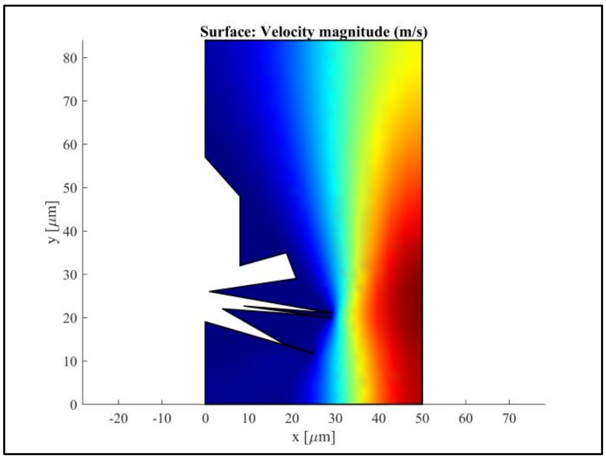

Figure 8 – Velocity field of the best solution found by the GA.

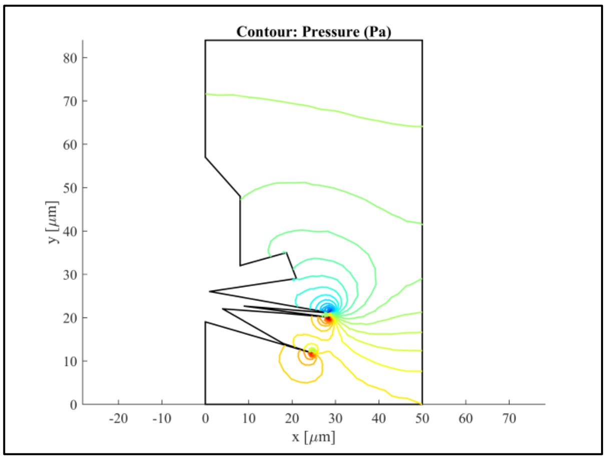

Figure 9 – Pressure field of the best solution found by the GA.

Table 2 – Optimization results after 100 generations.

Algorithm	h [W/(m^2 K)]	Δp [Pa]	Objective function
BBO	4.17E+13	2.968678	0.463811
DE	1.13E+13	6.602596	1.150814
ES	3.27E+13	6.055864	0.802772
GA	2.92E+14	0.191197	0.036234
PSO	7.56E+12	0.22926	0.478261

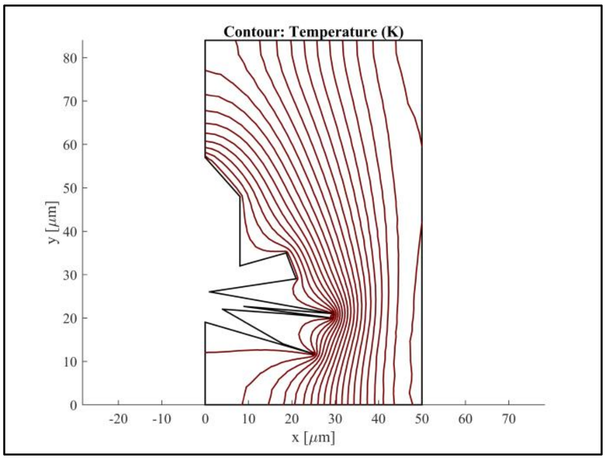

Figure 10 – Temperature field of the best solution found by the GA.

5. CONCLUSIONS

This paper presented the approach and results of optimization of VLM systems. The goal of the optimization is define an optimal shape for the pillars of the vaporization chamber of VLM system. A combined simulation environment was established to allow the information exchange between the optimization software (MATLAB) and the CFD software (COMSOL).

Several different algorithms were tested in this combined simulation environment in order to evaluate them and select the best one to be applied in future developments.

The genetic algorithm has performed better than the other methods and was able to generate a solution that satisfies all the constraints of the problem and provides very low value for the objective function which is composed by the heat transfer coefficient and the pressure drop of one of the pillars of the vaporization chamber. This improved performance could be attributed to the combination between the crossover operator that enhances the information sharing and the nature of the problem which does not have equality constraints which limits large individual changes to the optimization variables.

Future work will be done to extend the number of generations the algorithm runs in order to achieve better results that can be considered for the next generation of the Vaporizing Liquid Microthrusters developed by TU Delft. Also, more constraints have to be added to the optimization in order to produce solutions that are more technically feasible, for example, the minimum feature size that can be manufactured with MEMS and structural constraints such that not too narrow or thin walls are generated that might be easily broken during operation.

ACKNOWLEDGEMENTS:

The present work was supported by CNPq, Conselho Nacional de Desenvolvimento Científico e Tecnológico – Brasil.

REFERENCES

[1] M. A. C. Silva, D. C. Guerrieri, A. Cervone, and E. Gill, “A review of MEMS micropropulsion technologies for CubeSats and PocketQubes,” *Acta Astronautica*, vol. 143, pp. 234–243, 2018.

[2] A. London, A. Epstein, and J. Kerrebrock, “High-pressure bipropellant microrocket engine,” *Journal of Propulsion and Power*, vol. 17, no. 4, pp. 780–787, 2001.

[3] D. H. Lewis., S. W. Janson, R. B. Cohen, and E. K. Antonsson, “Digital micropropulsion,” *Sensors and Actuators A: Physical*, vol. 80, no. 2, pp. 143–154, 2000.

[4] C. Rossi, T. D. Conto, D. Estève, and B. Larangot, “Design, fabrication and modelling of MEMS-based microthrusters for space application,” *Smart Materials and Structures*, vol. 10, no. 6, p. 1156, 2001.

[5] J. Kohler, J. Bejhed, H. Kratz, F. Bruhn, U. Lindberg, K. Hjort, and L. Stenmark, “A hybrid cold gas microthruster system for spacecraft,” *Sensors and Actuators A: Physical*, vol. 97–98, no. 0, pp. 587–598, 2002.
[6] L. Stenmark and A. Eriksson, “Cold Gas Micro Thrusters,” in *NanoTech 2002 - "At the Edge of Revolution*, American Institute of Aeronautics and Astronautics, 2002.
[7] R. Krpoun and H. Shea, “Integrated out-of-plane nanoelectrospray thruster arrays for spacecraft propulsion,” *Journal of Micromechanics and Microengineering*, vol. 19, no. 4, p. 045019(10pp), 2009.
[8] A. Pukniel, V. Coverstone, R. Burton, and D. Carroll, “The dynamics and control of the CubeSail mission: A solar sailing demonstration,” *Advances in Space Research*, vol. 48, no. 11, pp. 1902–1910, 2011.
[9] D. C. Guerrieri, A. Cervone, and E. Gill, “Analysis of Nonisothermal Rarefied Gas Flow in Diverging Microchannels for Low-Pressure Microresistojets,” *Journal of Heat Transfer*, vol. 138 (11), no. 11, pp. 112403–112403–11, 2016.
[10] A. D. Ketsdever, D. C. Wadsworth, S. Vargo, and E. P. Muntz, “The Free Molecule Micro-Resistojet: An Interesting Alternative to Nozzle Expansion,” in *34th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*, 1998.
[11] K. Palmer, H. Nguyen, and G. Thornell, “Fabrication and evaluation of a free molecule micro-resistojet with thick silicon dioxide insulation and suspension,” *Journal of Micromechanics and Microengineering*, vol. 23, no. 6, p. 065006, 2013.
[12] E. . Mukerjee, A. . Wallace, K. . Yan, D. . Howard, R. . Smith, and S. . Collins, “Vaporizing liquid microthruster,” *Sensors and Actuators A: Physical*, vol. 83, no. 1–3, pp. 231–236, 2000.
[13] D. C. Guerrieri, M. A. C. Silva, A. Cervone, and E. Gill, “Selection and Characterization of Green Propellants for Micro-Resistojets,” *ASME Journal of Heat Transfer*, vol. 139, no. 10, p. 9, 2017.
[14] M. A. C. Silva, D. C. Guerrieri, H. van Zeijl, A. Cervone, and E. Gill, “Vaporizing Liquid Microthrusters with integrated heaters and temperature measurement,” *Sensors and Actuators A: Physical*, vol. 265, pp. 261–274, 2017.
[15] D. Whitley, “A genetic algorithm tutorial,” *Statistics and Computing*, vol. 4, no. 2, pp. 65–85, Jun. 1994.
[16] J. Kennedy, “Particle Swarm Optimization,” in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds. Boston, MA: Springer US, 2010, pp. 760–766.
[17] N. Hansen, “The CMA Evolution Strategy: A Comparing Review,” in *Towards a New Evolutionary Computation: Advances in the Estimation of Distribution Algorithms*, J. A. Lozano, P. Larranaga, I. Inza, and E. Bengoetxea, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 75–102.
[18] D. Simon, “Biogeography-Based Optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 6, pp. 702–713, Dec. 2008.
[19] R. Storn and K. Price, “Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces,” *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, Dec. 1997.
[20] D. Simon, “Biogeography-based optimization.” 2008.