Research Article

Performance of Gradient-Based Solutions versus Genetic Algorithms in the Correlation of Thermal Mathematical Models of Spacecrafts

Eva Anglada,[1] Laura Martinez-Jimenez,[2] and Iñaki Garmendia[2]

[1]*Industry and Transport Division, TECNALIA, Mikeletegi Pasealekua 2, 20009 San Sebastián (Donostia), Spain*
[2]*Mechanical Engineering Department, Engineering School of Gipuzkoa, University of the Basque Country (UPV/EHU),*
Plaza de Europa 1, 20018 San Sebastián (Donostia), Spain

Correspondence should be addressed to Eva Anglada; eva.anglada@tecnalia.com

The correlation of the thermal mathematical models (TMMs) of spacecrafts with the results of the thermal test is a demanding task in terms of time and effort. Theoretically, it can be automatized by means of optimization techniques, although this is a challenging task. Previous studies have shown the ability of genetic algorithms to perform this task in several cases, although some limitations have been detected. In addition, gradient-based methods, although also presenting some limitations, have provided good solutions in other technical fields. For this reason, the performance of genetic algorithms and gradient-based methods in the correlation of TMMs is discussed in this paper to compare the pros and cons of them. The case of study used in the comparison is a real space instrument flown aboard the International Space Station.

1. Introduction

Thermal control of spacecrafts and experiments is one of the key technologies needed to ensure the success of any space mission. This technology tries to make sure that the temperature of any component of the spacecraft or experiment is always inside the range foreseen during the design. Very high or very low temperatures can damage the different components of the spacecrafts, so these extreme temperatures must be avoided. Also, heat transfer between different components must be maintained under control. The design of the thermal control system is usually done following the hot and cold cases technique. This approach (consult the book of Karam [1] for information) bases the design on the coldest and the hottest situations expected during the mission. It is performed by means of two thermal mathematical models (TMMs) that represent the hot and the cold cases. The books of Gilmore and Redor [2, 3] can be also consulted for more information about the spacecraft thermal control.

Prior to any space mission, the thermal mathematical models of the spacecraft are created and different sets of temperatures are numerically calculated for particular mission phases or events. The use of the Thermal Lumped Parameters (TLP) method is very frequent in the space industry. This method divides the instrument in isothermal nodes with associated thermal inertias (MCs). These nodes are connected by means of conductive conductances (GLs) and radiative conductances (GRs). Reference [4] explains in detail this methodology.

These computational models must be verified by comparison with the results of the thermal balance test. The spacecraft or the experiment is placed inside a vacuum chamber where different thermal conditions are applied and it is tested under different operation modes, reproducing the situations expected in orbit. These boundary conditions, the operation modes, and the different thermal properties of the materials produce a range of thermal gradients inside the spacecraft. The temperatures of the different parts of the spacecraft are

measured with thermocouples that are located in strategic positions, mainly in places where the thermal engineers need to know the temperatures with some degree of accuracy. As a result, various distributions of temperatures are obtained when the thermal test is done. For more detailed information about the thermal control testing, the book of Meseguer et al. [5] can be consulted.

In parallel, the TMMs of the spacecraft have predicted sets of temperatures for the situations tested in the thermal balance test. In an ideal world, both sets of temperatures would be the same, if the TMMs were constructed with care and the geometry and material properties of the components were known exactly. However, as it could be expected, this is not the case. There are always differences between the measured temperatures and the predicted ones, even for TMMs done with extreme care. Although several well-established methods exist to calculate the TMM parameters, these methods have some limitations, so this type of modeling is always approximated. For example, Garmendia et al. explain in [6] the difficulties associated with the calculation of the conductive conductances. Moreover, there are parameters, for example, the contact conductances, whose exact calculation is almost impossible.

The main objective of the correlation task is to minimize the differences between the calculated and measured temperatures. Therefore, the correlation problem can be formulated as an optimization problem. This optimization is performed modifying the parameters of the TMM (mainly the GLs, GRs, and MCs) in such a way that the temperatures calculated by the thermal solver are as near as possible to the temperatures measured in the test. If this target is achieved, the thermal engineers will have a reliable TMM that can be used to explore thermal scenarios that could hardly (or not at all) be studied by thermal tests.

This type of optimization problem has the added difficulty that there is not a unique solution; that is, different sets of parameters of the TMM can produce almost the same temperature results. As an extreme situation, it could be possible to have a set of thermal parameters that produce a set of temperatures which fits exactly the measured temperatures but has no physical meaning. As a consequence, the changes in the TMM parameters should be done taking into account the physics behind the calculation of the parameters (geometry of the different parts of the spacecraft, material properties, etc.). This is possible when the correlation is tackled manually but it is difficult when automatic correlation is performed by means of optimization methods.

Different mathematical approaches can be used to handle the optimization problem already described, and a very complete description of them can be found somewhere else [7]. In our previous works [8, 9], our attention was focused on genetic algorithms (GAs), which are good candidates to solve the global optimization problem present in the TMM correlation. At the same time, we felt that classical optimization methods (mainly, gradient-based methods) deserved a deeper attention, as they provide good and operative solutions in other technical fields [10]. Although this type of methods tends to find the local optimum, we think that thermal parameters (GLs, GRs, and MCs) of TMMs produced

carefully by thermal engineers are, in many cases, close enough to the real values and, as a consequence, the solution provided by the thermal solvers is a good starting point for using classical optimization methods. In a previous work [11], the use of several algorithms of Broyden class in TMM correlation was compared with the results obtained by means of a GA. Although some limitations were detected for the use of Broyden class algorithms, the results obtained were promising. For this reason, it has been considered of interest to evaluate more in deep the application of classical methods to the TMM correlations. In particular, the extensive work done by Powell, presented in [12–16], and the availability of FORTRAN subroutines that implemented this work were crucial to the decision to make a comparison between classical optimization methods and genetic algorithms.

All in all, we decided to evaluate the performance of both types of algorithms, basing the comparison on the reduced thermal mathematical model of an experiment that was flown aboard the International Space Station (ISS). This experiment, called TriboLab, executed several tribology experiments during its mission. More information about it can be found in [17]. The thermal control system of this instrument was formed by the radiator, the heaters, the thermostats, and the multilayer insulation (MLI).

2. Optimization Methods in TMM Correlations

The detailed mathematical description of the different optimization methods employed in the correlation of thermal tests is outside the scope of this paper. However, we feel that a succinct description of the employed algorithms and the software that implements them is a necessary approach to the study we are doing.

The two types of algorithms studied require the use of one thermal analysis software to calculate the fitness function value. An in-house developed software able to solve steady-state and transient thermal problems including the use of thermostats and time-dependent boundary conditions has been used. It solves the set of N nonlinear transient algebraic equations (see (1)) that are obtained using the Thermal Lumped Parameters method, which is briefly explained in Section 3. The set of equations is solved by means of the Newton-Raphson linearization technique, using the iterative methods included in the public domain subroutines of ITPACK 2C package developed by Kincaid et al. [18].

$$\sum_{j=1}^{n} \mathrm{GL}\,(i,j)\,\left(T_i - T_j\right) + \sum_{j=1}^{n} \sigma \cdot \mathrm{GR}\,(i,j)\,\left(T_i^4 - T_j^4\right)$$
$$+ \mathrm{MC}_i \frac{dT_i}{dt} = q_i. \tag{1}$$

2.1. Classical Optimization Methods (Powell Approach). We have used four different available FORTRAN subroutines to generate four programs to run correlations. These subroutines are public domain and can be used subjected to some conditions (see [19]). They are based on an early work of Professor Powell in 1969, the subroutine VA05, available in [12]. This subroutine minimizes the sum of squares of m

given functions, f_j, $j = 1, 2, \ldots, m$, each of n variables, $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $m \geq n$, without the use of any partial derivatives, that is, to find $\mathbf{x}$ to minimize. In our case, m is the number of nodal temperature differences ($T_{\text{measured}} - T_{\text{calculated}}$) multiplied by the number of time instants where temperatures are calculated (in steady state, 1 time instant). Also, n is the number of thermal parameters (GLs, GRs, and MCs) whose real values are looked for. The method that VA05 uses is a compromise between three different algorithms for minimizing a sum of squares, namely, Newton-Raphson, Steepest Descent, and Marquardt. The subroutine automatically obtains an approximation to the first derivative matrix following the ideas of Broyden:

(i) *Subroutine TOLMIN*: this subroutine was developed later, in 1989. The subroutine minimizes a general differentiable objective function subject to linear constraints. The same definition of the objective function as that in the subroutine VA05 applies here but now linear constraints can be applied. This is a very interesting point, as the values of the thermal parameters must always be positive to have physical sense. This subroutine needs the user to supply the partial derivatives of the function with respect to the thermal parameters (gradient-based method)

(ii) *Subroutine NEWUOA*: the year of publishing of this software is 2004. The software seeks the least value of a function F (in our case, the sum of the temperature differences again) but this time without constraints and without derivatives. By this definition, it is clear that we are using a subroutine similar to the VA05 software but with a different approach. In fact, a quadratic approximation Q of the F function is used to obtain a fast convergence ratio and successive values of the Q approximation are used as the calculations progress

(iii) *Subroutine BOBYQA*: this piece of software was presented in 2009. The subroutine finds a minimum of a function F subject to bounds in the variables. The user does not supply derivatives. In our case, this particular approach seems to be very interesting, as extreme values for the thermal parameters can be imposed. These extreme values are, in fact, related to physical values of the problem; for instance, values of one of the thermal inertia MC(1) must be between 2000 J/˚C and 3000 J/˚C. As can be seen by this definition, BOBYQA is related with TOLMIN software

(iv) *Subroutine LINCOA*: this is the final work of Professor Powel, made public in 2014, before his passing away in April 2015. LINCOA stands for linearly constrained optimization. The software can be used even if the derivatives of the F function are not available. Linear constraints are possible with this software and emphasis is put on the successive calculations of the Q quadratic function that approximates F function

2.2. Genetic Algorithms. Genetic algorithms (GAs) are optimization algorithms of general purpose inspired in Darwin's theory of evolution. The main advantage of these types of algorithms is to be able to find the global optimum independently of the characteristics of the function. Although they are approximated algorithms, that is, do not guarantee the finding of the exact optimum, they are able to provide a good approximation in a reasonable time.

The basic idea consists in generating a random population of individuals, where each of them represents one possible solution to the problem. The population progresses by means of crossover and mutation operators that generate new individuals, the children, from two previous individuals, the parents. The parents are randomly selected from the population but better individuals (those that represent better solutions to the problem) have more probabilities to be selected. The children from better parents are expected to be better individuals, so new better individuals are generated with the successive generations.

Genetic algorithms are a well-established method of optimization. Although there are many possible ways of implementation, one of them must be selected depending on the problem. For this reason, only a brief description of the configuration of the GA used in this case is included here. References [20–22] can be consulted for more information about GAs.

The genetic algorithm used in this case is an in-house development similar to the GA used in [8] where the detailed description can be found. The fitness function used is defined in (2) and (3), where N corresponds to the number of temperature points or nodes, K corresponds to the number of instants, and J corresponds to the number of cases. The parents are selected by proportional selection based on a weighted fitness, following (4), by means of stochastic sampling with replacement. The crossover operator used corresponds to the simple arithmetic crossover. The range of the mutation operator is modified when the fitness has not been improved in last 20 iterations. The new population is created using an elitist strategy, so it is formed by the children and by the best individual of the previous population. The values of the definition parameters used are collected in Table 1.

$$F(X) = \sum_{j=1}^{J} \left[\sum_{k=1}^{K} \left[\sum_{n=1}^{N} \left(\text{node error}_{j,k,n} \right) \right] \right] \tag{2}$$

$$\forall \left(\text{node error}_{j,k,n} \right) > \text{target},$$

$$\text{node error}_{j,k,n} = \left| \left(T_{\text{calculated}} \right)_{j,k,n} - \left(T_{\text{measured}} \right)_{j,k,n} \right|, \tag{3}$$

$$f_i' = f_i - \beta \quad \text{where } \beta = \min_{i \in \text{population}} \left(f_i \right). \tag{4}$$

The GA is an approximated algorithm where the final results are influenced by some random values. For this reason, for each case of study subjected to correlation, the GA has been run 5 times obtaining 5 correlated models. The reason of this procedure is twofold. On one side, it makes it possible to evaluate the percentage of success of the algorithm, and on the other side it permits evaluating the variability between the different results obtained.

TABLE 1: Values of the GA definition parameters.

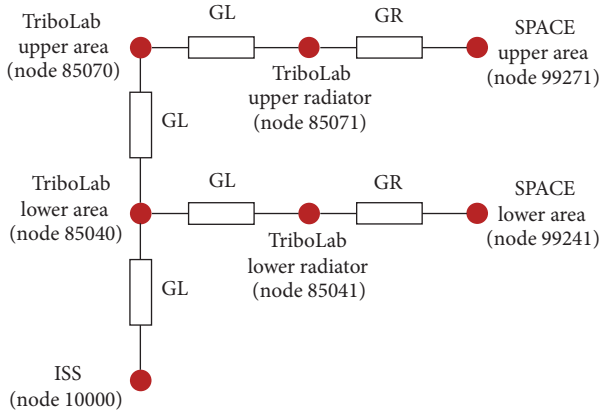| Parameters	Target (°C)	Iteration max.	Population in steady-state cases	Population in transient cases	Crossover probability (P_c)	Mutation probability (P_m)	Variation of P_m
Value	0.5	10,000	10	20	0.2	0.8	1.1

FIGURE 1: Thermal network of the TriboLab reduced model (reprinted from Acta Astronautica, 108 (2015), Eva Anglada and Iñaki Garmendia, Correlation of thermal mathematical models for thermal control of space vehicles by means of genetic algorithms, 1–17, Copyright (2014), with permission from Elsevier).

3. Thermal Mathematical Model Description

The thermal mathematical model used to evaluate the performance of the studied algorithms corresponds to the reduced model of the TriboLab experiment previously introduced. The design of the thermal control system of this experiment was performed following the hot and cold cases methodology and the thermal mathematical model is based on the Thermal Lumped Parameters method. As a result, two different models were obtained. A detailed model formed by 47 nodes and the reduced model used in this work. The latter is shown in Figure 1. This reduced model is formed by 7 nodes, where 4 of them are diffusive nodes representing different instrument areas (85040, 85041, 85070, and 85071) and the other 3 are boundary nodes that represent the ISS and the outer space (10000, 99271, and 99241).

In order to evaluate the performance of the compared algorithms, it is interesting to have a model where reference temperatures for all nodes of the model and the theoretical values of the variables are known. In this way, the results of the correlation can be better evaluated not only in terms of error level but also checking that temperature values are good for all nodes and that reasonable values have been obtained for the correlated variables. For this reason, instead of using as reference temperatures the values measured during thermal tests, the following procedure has been employed.

The TriboLab reduced model has been assumed as the reference model. That is, the values of its parameters (GLs, GRs, and MCs) are considered the real ones and the temperatures obtained from this model are taken as reference temperatures substituting the measured temperatures.

This model has been transformed modifying the value of its parameters (GLs, GRs, and MCs) randomly. As a result, a new model, called base model hereafter, has been obtained. This base model has been subjected to correlation to evaluate the performance of the compared algorithms. The four GLs, the two GRs, and the four MCs that form the model have been modified in a percentage up to 75%. This implies big differences between the theoretical values of the parameters of the reference model and the parameters used in the base model. These differences are higher than the expected ones in real cases, but this way the capacity of the algorithms to succeed in the correlation is tested with a challenging case. This model is the same as that previously used in the work presented in [8], where the detailed information about its values can be found.

The number of variables included in the correlation is 6 in the steady-state cases (the 4 GLs and the 2 GRs) and 10 in the transient cases (the 4 GLs, the 2 GRs, and the 4 MCs).

4. Results and Discussion

Results have been evaluated considering the temperature values, the values assigned to the correlated variables, and the CPU times. The accuracy of the temperature results has been quantified by means of the error expression shown in the following equation:

$$\text{error} = \frac{\sum_{i=1}^{N} \sqrt{\left(T_{\text{reference}} - T_{\text{calculated}}\right)^2}}{N}. \tag{5}$$

The error levels existing in the base model subjected to correlation, that is, the initial situation, are collected in Table 2.

4.1. Steady-State Cases. A first approximation to evaluate the capacity of the algorithms to correlate TMMs has been done with the steady-state cases. Table 3 shows the error values achieved in the correlations with Powell's algorithms. The temperature results obtained for the hot case can be considered to be exact. For the cold case, the results only can be considered to be exact with the NEWUOA algorithm; the rest are not so accurate but they reduce the error from 9.3°C to less than 0.45°C, which is a good result. These values are similar to those obtained with the GA (see Table 4).

The analysis of the temperature values obtained in all cases in each node (see Figures 2 and 3) shows that the values are good in all of them. No nodes with anomalous temperatures exist in the correlated models.

As it has been introduced previously, one of the difficulties of the TMM correlation is the fact that the solution is not unique. Therefore, different combinations of the involved

TABLE 2: Temperature errors of the base models (°C).

Case	Steady-state hot case	Steady-state cold case	Transient hot case	Transient cold case
Error (°C)	8.074	9.308	7.928	5.778

TABLE 3: Error values achieved with Powell's algorithms for steady-state cases (°C).

	TOLMIN	NEWUOA	BOBYQA	LINCOA
Hot case (°C)	0.000	0.000	0.000	0.000
Cold case (°C)	0.351	0.000	0.442	0.351

TABLE 4: Error values achieved with the GA for steady-state cases (°C).

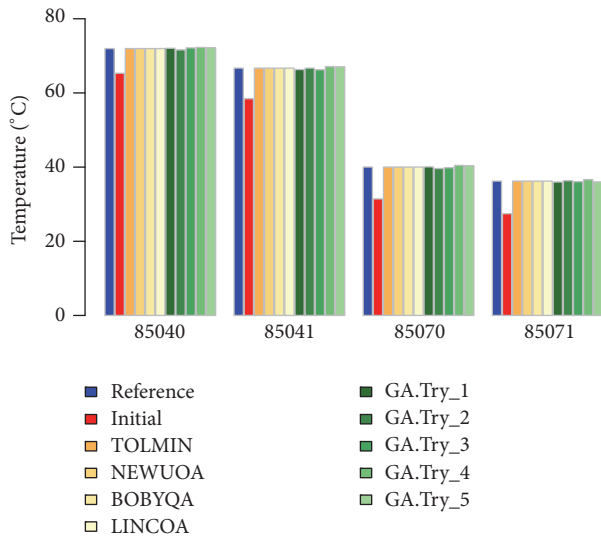|	GA Try 01	GA Try 02	GA Try 03	GA Try 04	GA Try 05	GA Average
Hot case (°C)	0.190	0.216	0.237	0.392	0.301	0.267
Cold case (°C)	0.407	0.310	0.439	0.318	0.293	0.353

FIGURE 2: Steady-state hot case. Temperatures obtained from correlations.
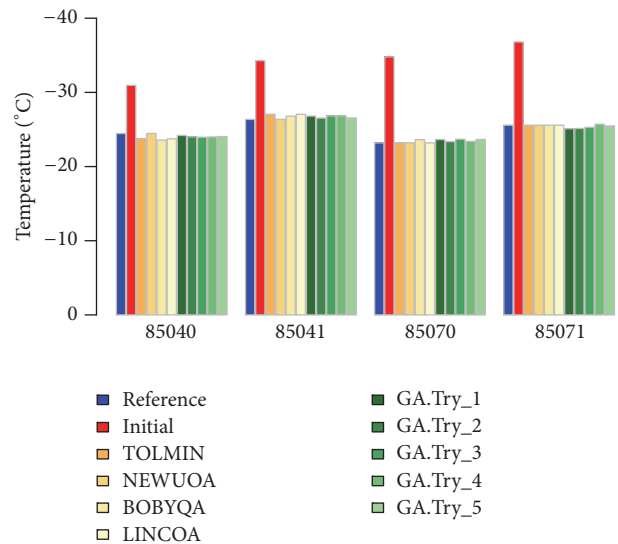

FIGURE 3: Steady-state cold case. Temperatures obtained from correlations.

parameters (GLs, GRs, and MCs) may give solutions which fulfil the error criteria used to evaluate the correlation but assign values with no physical sense for some variables. For this reason, it is interesting to evaluate the values obtained for the different variables in order to check whether some algorithms provide better values than others.

Figure 4 shows the values obtained for the variables in the correlation of the hot and cold steady-state cases with Powell's algorithms. The values obtained for each variable in the hot case correlation by the different algorithms do not present big differences, with the exception of the value of GL(10000, 85040) obtained with TOLMIN. Clearly, the values obtained are close to the initial values and not to the reference values; that is, the algorithm tends to find a local minimum close to the initial values.

The values obtained in the correlation of the cold case present higher variability compared to the hot case. In some

cases, for example, for the GRs, the values are closer to the reference values than to the initial ones. But, in other cases as GL(85040, 85070), two of the values assigned are equal to zero which is really far from the reference value.

Figure 5 shows the same values but obtained with the genetic algorithm. In general, the results can be considered similar to those obtained with the classical methods. No big differences are found in terms of variability or proximity to the reference values. In some cases, the values obtained with the classical methods are slightly better and in other cases the values obtained with the GA are better.

After performing independently the correlations of the hot and cold cases, it could be expected that the values of each variable should be the same in the hot and cold cases, but, as can be observed, the values obtained in the correlation of the hot case do not match with those obtained in the cold case correlations. In fact, if the values obtained in the correlation
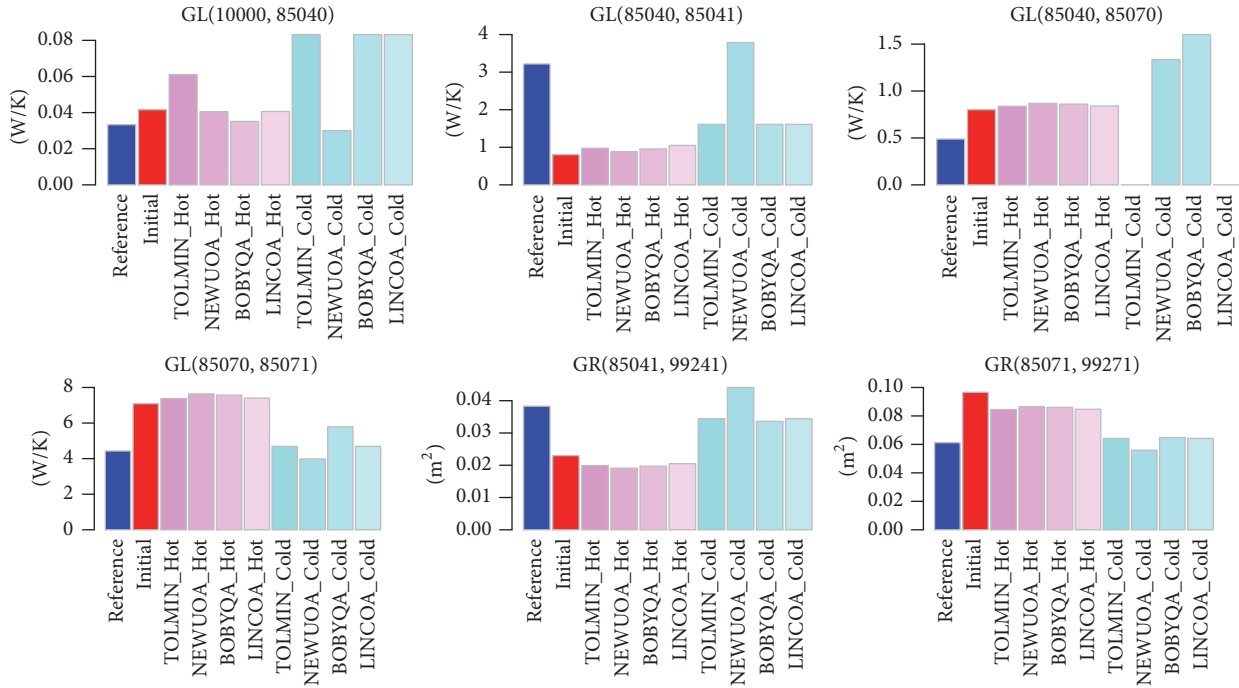
FIGURE 4: Steady-state, classical methods. Variable values obtained from correlations.
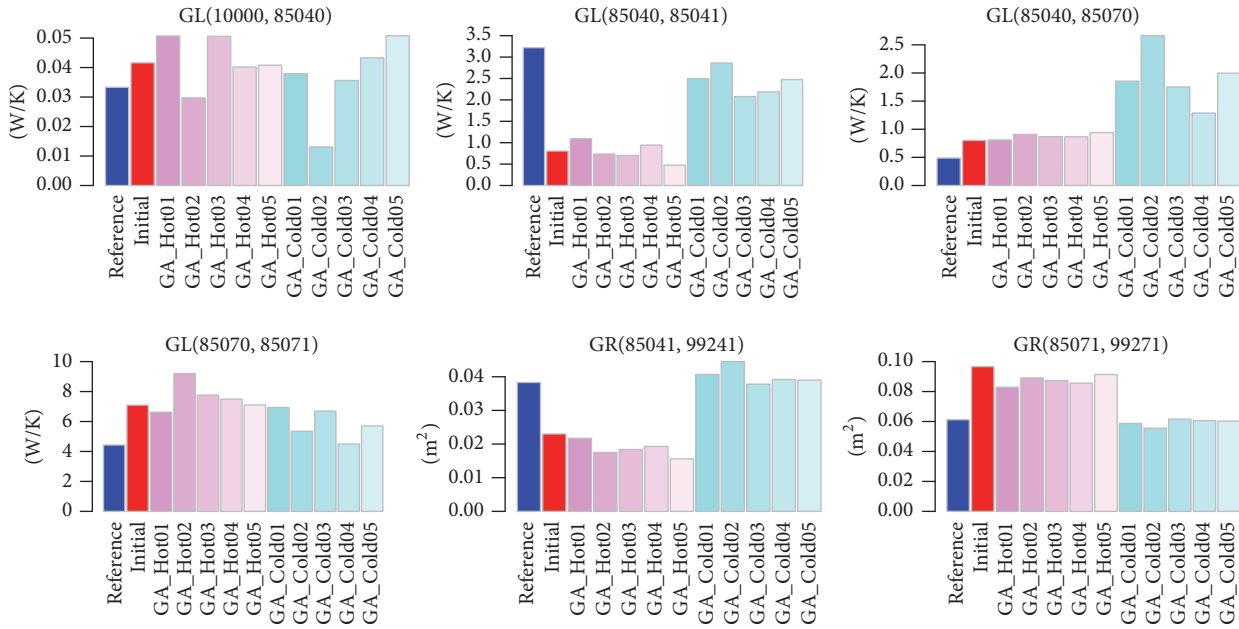

FIGURE 5: Steady-state GA. Variable values obtained from correlations.

of the hot case, for example, with NEWUOA algorithm, are used in the cold case, the temperatures obtained present an error equal to 3.2°C, lower than the initial error (9.3°C) but not very accurate. In the alternative case, when the values obtained in the correlation of the cold case (NEWUOA) are used with the hot case, the error is equal to 9.6°C, higher indeed than the error of the base model (8.07°C). This fact was also studied in a previous work devoted to the TMM correlations with GAs [8], where the same circumstance was observed. In that work, it was shown that the simultaneous correlation of the hot and cold cases solved this problem, obtaining variable values that fit well for the hot and cold cases. The limitation here is that the implemented versions of Powell's algorithms do not permit performing the simultaneous correlation of two different steady-state cases as are the hot and cold cases.

TABLE 5: Error values achieved with Powell's algorithms for transient cases (°C).

	TOLMIN	NEWUOA	BOBYQA	LINCOA
Hot and cold cases combined (°C)	—	0.400	1.965	0.885

TABLE 6: Error values achieved with the GA for transient cases (°C).

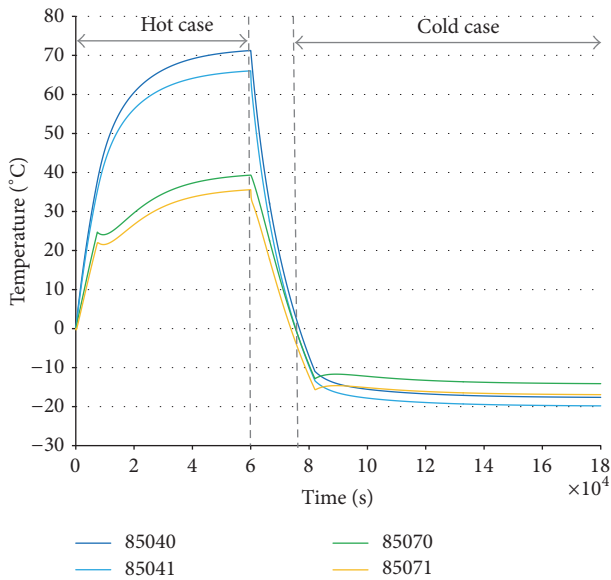|	GA Try 01	GA Try 02	GA Try 03	GA Try 04	GA Try 05	GA Average
Hot and cold cases combined (°C)	0.202	0.118	0.191	0.163	0.216	0.178

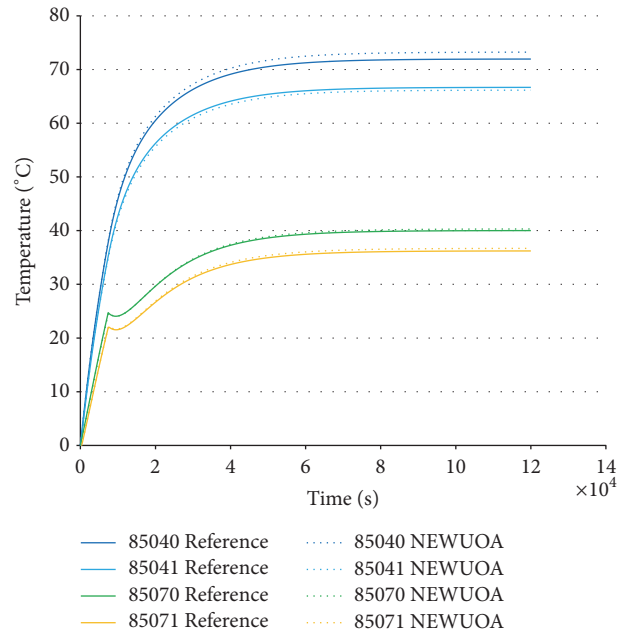FIGURE 6: Transient hot and cold cases combined sequentially.

FIGURE 7: Transient hot case. Temperatures obtained from correlations with NEWUOA.

4.2. Transient Cases. As it has been observed, the independent correlation of the hot and cold cases is not valid to obtain a set of variables useful for both cases. Therefore, it is needed to tackle the correlation considering the hot and cold cases simultaneously.

In the case of the GA, this is not a difficulty and both cases can be simultaneously correlated without problems. In the case of the present implementation of Powell's algorithms, this is not possible in a straightforward way, so the following workaround has been depicted. The cases have been combined consecutively: first the hot case and next the cold case. In this way, a new case to be used in the correlation has been created, combining the boundary conditions of each case sequentially. Figure 6 shows the temperature profile of this case.

Table 5 collects the error values calculated for the correlation of the combined hot and cold transient cases with the classical methods. In this case, the algorithm TOLMIN has not converged and does not give results. Considering that the initial error is equal to 6.8°C, the other three algorithms give good results, the best being those obtained with the NEWUOA algorithm (error: 0.4°C).

In the case of the GA, the results are collected in Table 6. In each of the 5 trials performed, the GA has reached an error

value lower than the NEWUOA algorithm, the average error being equal to 0.18°C.

Figures 7–10 show the comparison between the reference and calculated temperatures after the correlation performed with the NEWUOA algorithm and the GA. In the case of GA, the values correspond to the trial which has reached the lower error, Try 02. As can be observed, the results obtained with both alternatives are good, although those obtained with NEWUOA are slightly worse, especially in nodes 85040 and 85041.

If we consider the variable values (see Figure 11), the variability obtained with Powell's algorithms is significant. TOLMIN algorithm has not converged not providing results and BOBYQA clearly fails in GL(10000, 85040) and GL(85040, 85070). The values obtained with NEWUOA are quite good. The values obtained with the GA present lower variability. It is remarkable that, in this combined case, the values obtained are closer to the reference values than to the initial values for both approaches.

4.3. CPU Times. Table 7 collects the CPU times required for the different algorithms to perform the correlations. As can

TABLE 7: CPU times (in seconds).

	TOLMIN	NEWUOA	BOBYQA	LINCOA	GA*
Steady-state, hot case	10	3	2	5	1.5
Steady-state, cold case	13	13	5	5	15
Transient, hot and cold case	—	1020	89	104	179

*The values shown for the GA correspond to the average value obtained in the 5 executions of the algorithm.
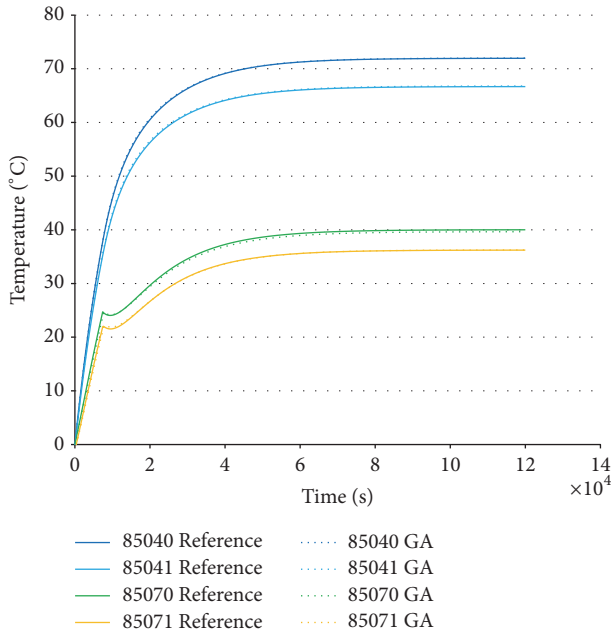

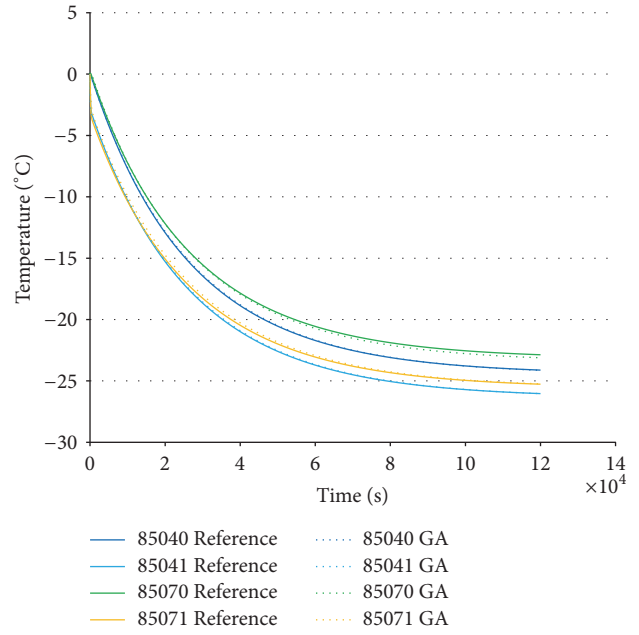FIGURE 8: Transient hot case. Temperatures obtained from correlations with GA.

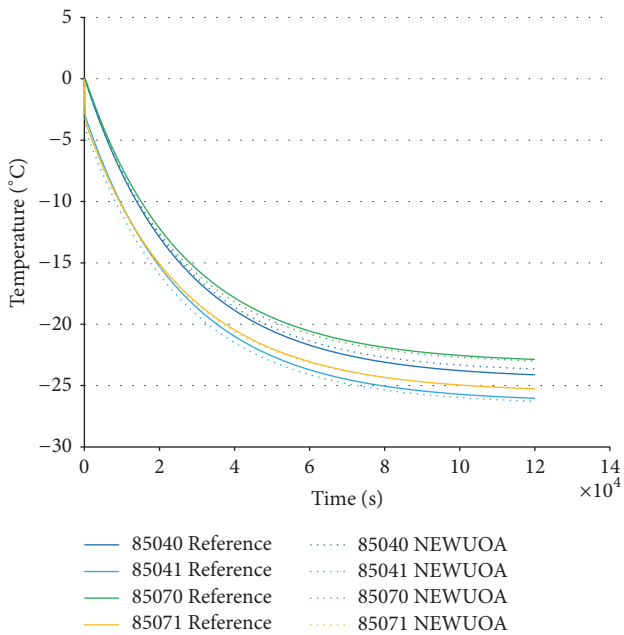FIGURE 10: Transient cold case. Temperatures obtained from correlations with GA.

FIGURE 9: Transient cold case. Temperatures obtained from correlations with NEWUOA.

be observed, the calculations are really fast for the steady-state cases, 15 seconds maximum. The times needed for the transient cases are higher but are also very fast. The longest time, which corresponds to the NEWUOA algorithm, is equal to 1020 seconds (17 minutes).

Numerical values of the elapsed CPU times presented in Table 7 should be read with some caution, as the computers used for Powell's algorithms and GA algorithms, although similar, were not exactly the same.

Although one of the drawbacks of the genetic algorithms is usually the high number of iterations needed to perform the correlation, as can be observed in this case, this does not imply long CPU times.

4.4. Considerations about Convergence. It is very convenient to study the convergence of the results obtained with the different correlation methods explained so far. In fact, this is a critical point that could make the difference between a method that is interesting and a method that could be used for practical applications. There are two main aspects that are to be studied: first the temperature distribution (those measured in the tests versus those predicted in the models) and second the value of the thermal parameters of the model (GLs, GRs, and MCs).

TABLE 8: GLs values for case 1 and case 2 [W/°C].

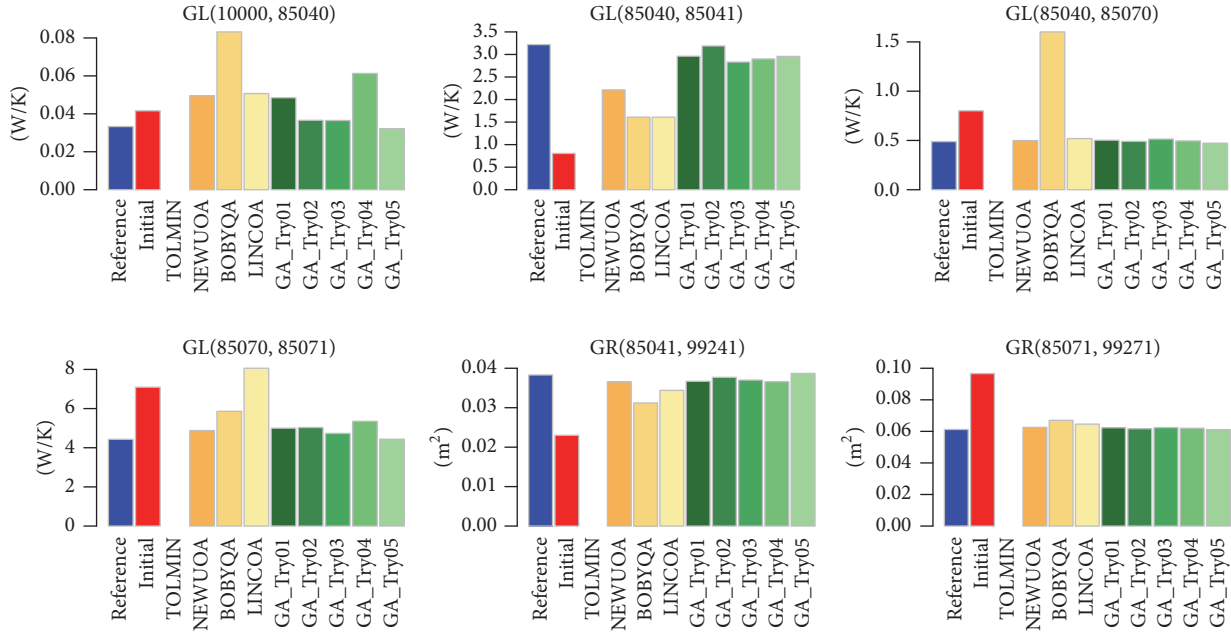| Parameter	GL(1, 2)	GL(1, 3)	GL(1, 4)	GL(2, 3)	GL(2, 4)	GL(3, 4)
Case 1 [W/°C]	8.00	6.00	5.00	4.00	8.00	3.00
Case 2 [W/°C]	6.00	7.00	5.57	8.00	6.94	2.97

FIGURE 11: Transient. Variable values obtained from correlations.

4.4.1. Convergence of Temperatures. The methods already presented (gradient-based and genetic algorithms-based) are, in general, capable of doing a good job in reproducing the temperatures measured in the tests. To do so, they change the values of the thermal parameters and reach a reasonable solution with moderate CPU times, as has been shown before. For bigger models, the CPU times increase significantly, but values can be affordable, at least with the examples studied with genetic algorithms (see [9] for more details). In fact, the objective of correlation, that is, the minimization of the sum of the differences between the measured temperatures and the calculated ones, is almost always achieved with both types of methods. As an exception, the TOLMIN subroutine has not achieved this objective (see Figure 11). We believe that this subroutine is more appropriate for seeking local minimums, those no "far" from the initial solution provided. So, the big mismatch between the base parameters and the reference ones (which could be as big as 75%) may be the reason of this bad result.

4.4.2. Convergence of Thermal Parameters. To reach the reference parameters when starting with the base parameters through the optimization of the differences of temperatures is a difficult task. The first and most important difficulty is the fact that the inverse problem does not have a unique solution; that is, different sets of thermal parameters can produce the same temperature distribution. If this is so, how could the optimization method distinguish between a set of
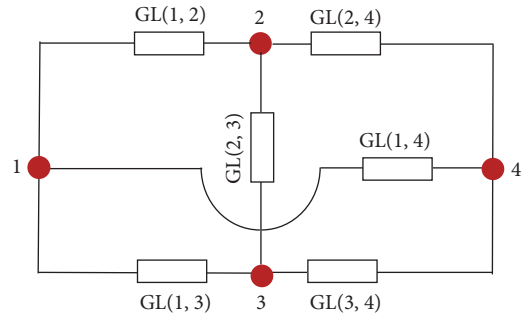

FIGURE 12: Complementary model.

"correct" thermal parameters and a set of "incorrect" ones if both produce a minimum in the temperature optimization method? There is not a simple answer to this question, but it is interesting to note that better results for the thermal parameters are obtained if both hot and cold cases are used simultaneously to do the temperature correlation. As a consequence, a complementary study has been developed in order to understand this fact.

(1) Complementary Model. A new model formed by 4 nodes and 6 GLs, shown in Figure 12, has been defined together with two study cases. Both cases have a source of power equal to 120 W applied in node 1 (q_1 = 120 W) and a fixed temperature of 20°C set in node 4 (T_4 = 20°C). The difference between the two study cases is the values assigned to the GLs, which are shown in Table 8.

Table 9: Temperature results (°C).

Node	1	2	3	4
Case 1 ($q_2 = 120$ W)	30.84	25.69	26.75	20.00
Case 2 ($q_2 = 120$ W)	30.84	25.69	26.75	20.00

Table 10: Heat flows (W).

Parameter	$q_{1\rightarrow2}$	$q_{1\rightarrow3}$	$q_{1\rightarrow4}$	$q_{2\rightarrow3}$	$q_{2\rightarrow4}$	$q_{3\rightarrow4}$
Case 1	41.24	24.53	54.22	−4.267	45.51	20.27
Case 2	30.93	28.62	60.44	−8.53	39.47	20.09

Table 11: Temperature results (°C).

Node	Case 1 $q_1 = 180$ W	Case 2 $q_1 = 180$ W	Case 1 $q_2 = 120$ W	Case 2 $q_2 = 120$ W
1	36.27	36.27	25.69	25.69
2	28.53	28.53	29.38	29.89
3	30.13	30.13	25.51	26.62
4	20.00	20.00	20.00	20.00

The GLs values of case 1 have been defined arbitrarily. The GLs of case 2 have been selected to be different from the values of case 1 but to produce the same temperature distribution. The procedure followed to obtain the GLs values of case 2 has been the following: the temperature results for case 1 have been calculated with the help of the TK software, obtaining the values shown in Table 9. Then, three arbitrary values have been assigned to three parameters of the model, in this case GL(1, 2), GL(1, 3), and GL(2, 3). Taking into account these values, the temperature results of case 1, (1), and the boundary conditions $q_1 = 120$ W and $T_4 = 20$°C, the values for the other three parameters GL(1, 4), GL(2, 4), and GL(3, 4) have been obtained.

Case 2 produces the same nodal temperatures as case 1 (see Table 9), although the flow distributions between nodes are different in both cases (see Table 10). This is just a simple example that shows that two different sets of thermal parameters can produce the same temperature distribution.

Moreover, if we change the power applied in node 1 and set it to 180 W ($q_1 = 180$ W) in both cases, the same temperature distribution is obtained for both cases 1 and 2 (see Table 11). This fact is explained because we have not altered the general "layout" of the model; we are applying the boundary conditions in the same nodes in a very simple model. Also, it is important to note that only linear conductances are present in this case (only GLs, not GRs).

However, if we modify original cases 1 and 2 applying 120 W in node 2 rather than node 1, that is, $q_2 = 120$ W, maintaining $T_4 = 20$°C, we obtain two different temperature distributions for each case as can be seen in Table 11.

This fact implies that two sets of thermal parameters that produce the same temperature distribution with a particular set of boundary conditions will produce a different distribution of temperatures if a different set of boundary conditions

is applied. This is a mechanism to discriminate between two sets of thermal parameters that produce the same result for one set of boundary conditions.

When the hot and cold cases are used at the same time for the correlation of temperatures, we are somehow applying different sets of boundary conditions. In doing so, we are "discriminating" between solutions that produce a bigger error than others and we are conducted by the optimization algorithms towards a solution closer to the reference values. The hot and cold cases "activate" different aspects of the model and the algorithms have more information in order to get the reference answer. Somehow, we are providing the optimization algorithms with more useful information. Finally, using transient hot and cold cases for the correlations means that more information is added in the intermediate steps, facilitating the finding of the reference values of the thermal parameters. If in these transient cases the boundary conditions change with time (which is quite usual, for example, heaters switch on or off depending on the temperatures, space sink temperatures change with time, etc.), we are even adding more useful information to the algorithm.

(2) TriboLab Model. The ideas expressed in the previous paragraphs have been applied to the TriboLab model (Figure 1). The situation is far more complicated in this case, mainly because of the presence of 2 radiative conductances GR(85041, 99241) and GR(85071, 99271) and because the power input takes place in 4 nodes (85040, 85041, 85070, and 85071). Table 12 shows the thermal parameters of the TriboLab model and of one alternative model that produces the same temperature distribution for the TriboLab hot case. The thermal parameters of the alternative model have been calculated in the same way as the previous complementary model.

TABLE 12: TriboLab model definition.

Parameter	TriboLab model	Alternative model
GL(10000, 85040) [W/°C]	0.033270	0.02567
GL(85040, 85070) [W/°C]	0.48830	0.2
GL(85040, 85041) [W/°C]	3.2190	5.0
GL(85070, 85071) [W/°C]	4.4309	2.0014
GR(85041, 99241) [m^2]	0.038300	0.0529
GR(85071, 99271) [m^2]	0.061200	0.0420

TABLE 13: Temperature results [°C].

Node	TriboLab model q' (W) = q/2	Alternative case q' (W) = q/2
85040	21.8251	24.1327
85041	19.4234	21.5761
85070	1.7115	1.5587
85071	−0.6405	−0.9970

The change applied has been to reduce the input powers to the half in these two cases. The new distributions of temperatures obtained for the two models (TriboLab and alternative) are different, as can be seen in Table 13. In the complementary model previously described, the change in the value of the source of power for the two sets of thermal variables did not produce different set of temperatures. But, in this model, the presence of GRs and the fact that the heat source is applied in four nodes and not in only one make the modification of the heat source values be enough to produce a different set of temperatures.

All these initial steps about the convergence of the thermal parameters towards the reference parameters need more in-depth investigation, trying to formulate in a consistent way the advantages of using hot and cold cases simultaneously. We believe that the presence of GRs in the TMMs facilitates the finding of the reference parameters when optimizing the correlation. This is so because of the nonlinearity implied in the radiation heat transfer process. If two sets of thermal parameters have the same temperature distribution and the applied power changes, the new distributions of temperatures for the two sets will probably be somehow different. We also believe that the addition of other thermal scenarios of the mission (not only the hot and cold cases but also the nominal functioning, for instance) will provide more chances to obtain better results.

5. Conclusions

Results obtained for steady-state cases with both types of algorithms, classical and GA, are good in terms of temperatures. The error levels have been reduced from 8.1 and 9.3°C to values below 0.44°C, and Powell's algorithms indeed have reached the exact value in some cases. The nodal temperature distributions obtained are also appropriate and the CPU times are extremely short, less than 15 seconds.

Unfortunately, the values assigned to the variables are not good. The main problem is the difference between the values obtained in the correlation of the hot and the cold cases. The values obtained in one case, hot case, for example, fail when used in the other case, cold case, and vice versa. To obtain a set of values appropriate for both cases, performing the simultaneous correlation of both cases is needed. This is not a problem for the GA but unfortunately it is not possible with the present implementation of Powell's algorithms.

In the transient case combining the hot and cold cases, the results are good in terms of temperature with the exception of TOLMIN algorithm which does not converge. Results obtained with BOBYQA and LINCOA are not exceptional but get a reduction of the initial error (6.8°C) to 1.97 and 0.88°C, respectively. NEWUOA algorithm reaches an error level equal to 0.4°C, which is a good value, and the GAs succeed in reaching values below 0.22°C. The nodal temperature distribution is also appropriate and the CPU times are very short with the GA, less than 3 minutes. The time needed by NEWUOA algorithm is longer, 17 minutes, but is a good value compared with the time needed for manual correlation.

Referring to the variable values, in this case, the sets of values obtained are valid for the hot and cold cases. With the exception of BOBYQA, which clearly fails in the calculation of some values, the rest provide reasonable values. It is remarkable that, in this combined case, the values obtained tend to converge to values closer to the reference values than to the initial values. A complementary model has been presented to explain the better convergence in this situation.

As a general conclusion, it can be said that the studied gradient-based solutions are not an appropriate tool for the steady-state correlation of TMMs of spacecrafts, with the use of a GA being a better solution. In the case of the correlation of transient TMMs, the use of NEWUOA algorithm is a valid option but the GA seems to be a better solution. The GA has been faster and the values provided for the thermal

parameters tend to converge a little better to the reference values.

Disclosure

This work has been performed as an independent research and no financial help has been received for it.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

 [1] R. D. Karam, *Satellite Thermal Control for Systems Engineers*, American Institute of Aeronautics and Astronautics, Reston, Va, USA, 1998.

 [2] D. G. Gilmore, *Spacecraft Thermal Control Handbook*, American Institute of Aeronautics and Astronautics, El Segundo, Calif, USA, 2nd edition, 2002.

 [3] J. F. Redor, *Introduction to Spacecraft Thermal Control*, ESA Publications Division, Noordwijk, The Netherlands, 1995.

 [4] K&K Associates, *Thermal Network Modeling Handbook*, Introduction to Spacecraft Thermal Control, K&K Associates, Westminster, Colo, USA, 2000.

 [5] J. Meseguer, I. Pérez-Grande, and A. Sanz-Andrés, *Spacecraft Thermal Control*, Woodhead Publishing, Cambridge, UK, 2012.

 [6] I. Garmendia, E. Anglada, H. Vallejo, and M. Seco, "Accurate calculation of conductive conductances in complex geometries for spacecrafts thermal models," *Advances in Space Research*, vol. 57, no. 4, pp. 1087–1097, 2016.

 [7] F. Jouffroy, D. Tochon, and A. Capitaine, Astrium Satellites; CNES & ESA, Bibliographical study of optimisation methods with focus on Genetic Algorithm techniques w.r.t post-test thermal model correlation problem, 2006 https://exchange.esa.int/restricted/model-correlation/.

 [8] E. Anglada and I. Garmendia, "Correlation of thermal mathematical models for thermal control of space vehicles by means of genetic algorithms," *Acta Astronaut*, vol. 108, pp. 1–17, 2015.

 [9] I. Garmendia and E. Anglada, "Thermal mathematical model correlation through genetic algorithms of an experiment conducted on board the International Space Station," *Acta Astronautica*, vol. 122, pp. 63–75, 2016.

[10] S. C. Chapra and R. P. Canale, *Numerical Methods for Engineers*, McGraw-Hill, New York, NY, USA, 6th edition, 2010.

[11] J. Klement, E. Anglada, and I. Garmendia, "Advances in automatic thermal model to test correlation in space industry," in *Proceeding of the 46th International Conference on Environmental Systems - ICES*, Vienna, Austria, 2016.

[12] HSL. A collection of Fortran codes for large scale scientific computation. Subroutine VA05, 1969, http://www.hsl.rl.ac.uk/.

[13] M. J. Powell, "A tolerant algorithm for linearly constrained optimization calculations," *Mathematical Programming*, vol. 45, no. 3, pp. 547–566, 1989.

[14] M. J. Powell, "The NEWUOA software for unconstrained optimization with derivatives," Tech. Rep. DAMTP/2004/NA05, University of Cambridge, Cambridge, UK, 2004.

[15] M. J. Powell, "The BOBYQA algorithm for bound constrained optimization without derivatives," Tech. Rep. DAMTP/2009/NA06, University of Cambridge, Cambridge, UK, 2009.

[16] M. J. Powell, "On fast trust region methods for quadratic models with linear constraints," Tech. Rep. DAMTP/2014/NA02, University of Cambridge, Cambridge, UK, 2014.

[17] I. Garmendia, E. Anglada, H. Vallejo, M. Brizuela, and N. Insausti, "Thermal control of tribolab, a materials experiment in the international space station," in *Proceeding of the 50th Anniversary Conference Engineering: Science and Technology*, Tecnun – Escuela de Ingenieros de la Universidad de Navarra, Ed., Servicio de Publicaciones de la Universidad de Navarra, Donostia-San Sebastian, Spain, 2012.

[18] D. R. Kincaid, J. R. Respess, D. M. Young, and R. R. Grimes, "ALGORITHM 586: ITPACK 2C: a FORTRAN package for solving large sparse linear systems by adaptive accelerated iterative methods," *ACM Transactions on Mathematical Software (TOMS)*, vol. 8, no. 3, pp. 302–322, 1982.

[19] Z. Zaikun, Software by Professor M. J. D. Powell, (n.d.), http://mat.uc.pt/~zhang/software.html#powell_software.

[20] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, AddisonWesley, New York, NY, USA, 1989.

[21] J. H. Holland, *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, Mich, USA, 1975.

[22] F. Herrera, M. Lozano, and J. L. Verdegay, "Tackling real-coded genetic algorithms: operators and tools for the behavioral analysis," *Artificial Intelligence Review*, vol. 12, no. 4, pp. 265–319, 1998.