Adapted Genetic Algorithm for the Design Optimization of the Multi-Pulse Solid Rocket Motor

Tiago Neves
Dept. T3, Bayern-Chemie GmbH
P.O. Box 1131 Aschau, Germany
Instituto Superior Técnico
Lisbon, Portugal
tiago.neves@mbda-systems.de

Karim El Sioufy
Dept. T3, Bayern-Chemie GmbH
P.O. Box 1131 Aschau, Germany
karim.sioufy@mbda-systems.de

Stephan Hase
Dept. T3, Bayern-Chemie GmbH
P.O. Box 1131 Aschau, Germany
stephan.hase@mbda-systems.de

Pedro C. Pinto
Dept. T5, Bayern-Chemie GmbH
P.O. Box 1131 Aschau, Germany
pedro.caldas-pinto@mbda-systems.de

Christoph Bauer
Technical Director, Bayern-Chemie GmbH
P.O. Box 1131 Aschau, Germany
christoph.bauer@mbda-systems.de

João M. C. Sousa
IDMEC
Instituto Superior Técnico
Lisbon, Portugal
jmsousa@tecnico.ulisboa.pt

***Abstract*—Design optimization of solid rocket motors can be formulated as an optimization problem. The multi-pulse variant of this design offers many advantages compared to other variants, making it a valuable design choice. However, its design problem lacks a known formulation, to the best of our knowledge. Therefore, this paper proposes a mixed integer programming formulation for this problem. The formulation contains equality and inequality constraints, which bring forth many challenges, often causing results yielded by evolutionary algorithms to have feasibility issues, without specialized constraint handling. This paper proposes a constraint handling technique that is suitable for the design optimization problem of the multi-pulse solid rocket motor, and serves as the backbone for a novel gradient-based algorithm, the hybrid random restart hill climbing algorithm, to be compared with an adapted genetic algorithm. A brief overview of this problem is given in terms of modelling and simulation. On average across all seeds, the hybrid random restart hill climbing algorithm yields better solutions compared to the genetic algorithm. However, in terms of best across all seeds, the latter outperforms the former. This indicates that the solution space of the multi-pulse solid rocket motor design problem is nonconvex, thus justifying the usage of metaheuristic algorithms.**

***Index Terms*—Design optimization, genetic algorithm, solid rocket motor, multi-pulse, constraint handling.**

I. Introduction

Mixed integer programming (MIP) problems with equality and inequality constraints are optimization problems which bring many challenges in terms of constraint handling. Previous authors have mentioned how equality constraints lead to feasibility problems for evolutionary algorithms without specialized constraint handling techniques [1]. For problems with nonconvex solution spaces, this becomes particularly important, since metaheuristic usage, while justified, becomes compromised with such constraints. In certain real-world applications, even the slightest constraint violation may lead to catastrophic consequences and render entire designs obsolete, which makes the matter of constraint handling paramount.

The present work proposes a constraint handling technique that is suitable for tackling the equality and inequality constraints of an MIP problem, at different levels of precision across each constraint. This technique is then used to develop a novel algorithm, the hybrid random restart hill climbing algorithm, which reduces the overall MIP problem to a continuous one. An adapted genetic algorithm is also developed with the aforementioned constraint handling technique.

The aforementioned techniques are applied to the design optimization of a multi-pulse solid rocket motor, in which certain design constraints are critical from a physical standpoint, making this example adequate in light of constraint handling importance. Although certain optimization techniques such as genetic algorithms have been used in solid rocket motor design contexts by other authors [2], [3], no application of these methods in a multi-pulse setting have been found in the literature. Indeed, multi-pulse solid rocket motor designs are a technology that has gained attention recently, making it a particularly promising area to which many optimization techniques have not yet been explored. In that sense, this paper contributes to the state of the art by successfully applying the proposed algorithms to the optimization problem of this solid rocket motor technology.

This paper offers an overview of multi-pulse solid rocket motors in Sec. II in terms of advantages, disadvantages, as well as design modelling and simulation. In Sec. III-A, its optimization problem is formulated. The two algorithms developed for this problem are found in Sec. III-B, with the results being discussed and compared throughout Sec. III-C. Conclusions are drawn in Sec. IV.

II. Multi-pulse solid rocket motor

A solid rocket motor (SRM) is a type of rocket motor that uses a solid propellant in which both the fuel and the oxidizer are mixed together. The solid propellant is placed inside a chamber within the rocket motor structure, where it is ignited, causing a subsequent combustion along its surface area. The hot gases produced in this process pass through a nozzle of a certain shape, where the corresponding flow is accelerated. The smallest cross-sectional area of the nozzle is known as the nozzle throat, which has a decisive impact on the overall performance [4]. Fig. 1 illustrates a generic SRM design and its main components. This definition consists of a rocket motor with a single structure, chamber, propellant and nozzle.

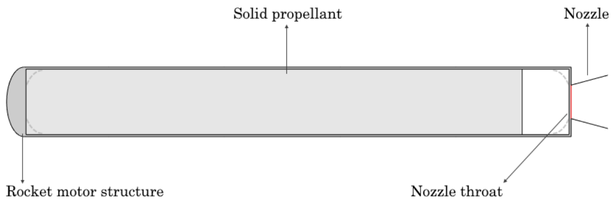

Fig. 1. Example of an SRM design

The multi-pulse variant consists of an SRM design which contains multiple chambers that are all part of a single structure. A pulse separation device (PSD) is used to separate two subsequent chambers, thereby allowing for the independent ignition of each chamber, from closest to farthest from the nozzle. Fig. 2 illustrates a 2-pulse SRM design.

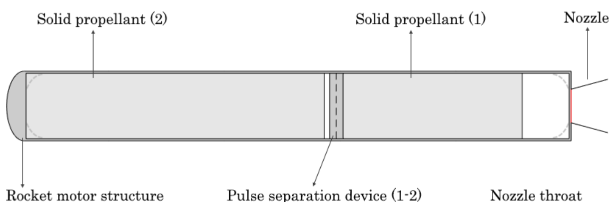

Fig. 2. Example of a multi-pulse SRM design

As opposed to other SRM design variants such as the multi-stage one, in which each chamber is housed in a separate, detachable, structure, the multi-pulse variant houses all the separate chambers under a single structure [5], thus overcoming the difficulty of reliability that comes from the former variant, whilst also leading to simpler designs that involve less moving parts, or none at all, requiring only the addition of the PSD [6]. Other advantages include its readiness and ease of use, due to its compact nature, as well as designs that lead to high specific impulse and thrust density values. The multi-pulse variant has also been shown to be suitable for optimal control strategies [7]. A major setback here is carrying the weight of emptied chambers throughout the remaining rocket flight, which marks the tradeoff for the performance and design benefits that come from it. These factors make this variant particularly attractive for civil and military applications.

The design layout for the multi-pulse SRM is shown in Fig. 3, where L and D are the overall length and diameter, respectively, and $d_{\text{eff,ex,noz}}$ the effective nozzle exit diameter. As for the chambers, $\text{prop}^{(i)}$ is the i-th chamber propellant, $l_{\text{ch}}^{(i)}$ its length and $s_{\text{PSD}}^{(i,i+1)}$ the length of the PSD that separates the i-th chamber from the $(i+1)$-th. Each chamber length is specified through an input parameter called the chamber proportion factor $\Phi_{\text{ch}}^{(i)}$, which represents the normalized cumulative position of each pulse separation device, relative to the total chamber structure length.

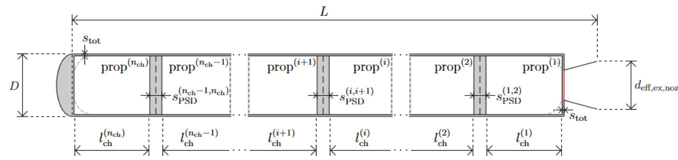

Fig. 3. Multi-pulse SRM layout

From a set of input parameters, the SRM design parameters, along with certain outputs, are computed using the methodology illustrated by the flowchart of Fig. 4. As it is seen, two iterative processes are present, each with respect to a certain variable. Once it converges, its respective loop terminates. The first yields the nozzle diameter d_{noz} which allows for the calculation of all required nozzle and chamber quantities, as well as other propellant burn information that is required for the second iterative process. There, the chamber pressure vector $\mathbf{p}^{(i)}$ is built, which is then used to compute the performance of the SRM design.

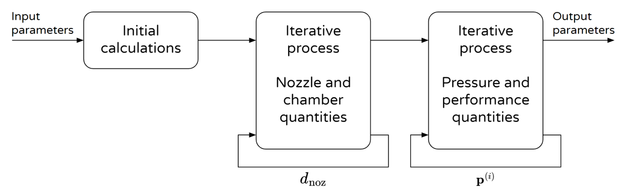

Fig. 4. Multi-pulse SRM design methodology

Two output parameters are used throughout this work, and are now presented. The first is the total impulse at sea level, shown in (1). The second is the potential range, a custom performance metric, defined as the integral the Tsiolkovsky rocket equation over time, given by (2).

$$I = \sum_{i=1}^{n_{\text{ch}}} \left(\int_0^{t_f^{(i)}} \mathbf{F}_{\text{sl}}^{(i)} dt \right), \tag{1}$$

where n_{ch} is the number of chambers, $t_f^{(i)}$ the final simulation time for the i-th chamber and $\mathbf{F}_{\text{sl}}^{(i)}$ its thrust vector at sea level.

$$\mathcal{R} = \int_0^{t_f} \mathbf{I}_{\mathbf{sp},\text{sl}} g \ln \left(\frac{m_{\text{tot},0}}{m_{\text{tot},0} - m_{\text{prop},0}} \right) dt, \tag{2}$$

where t_f is the cumulative final simulation time (i.e., considering all chambers in sequence), $\mathbf{I}_{\mathbf{sp},\text{sl}}$ the concatenated specific impulse vector at sea level, $m_{\text{tot},0}$ and $m_{\text{prop},0}$ the initial total and propellant masses, respectively. $\mathcal{R}$ is to be interpreted as the maximum attainable range at zero drag during a time interval of t_f, and not as the one obtained through real thrust.

Throughout the process, some constraints are considered so as to ensure the feasibility of the final design obtained. The sum of all chamber proportion factors $\Phi_{\text{ch}}^{(i)}$ needs to equal 1 so that the chambers fill the entirety of the available length. To ensure combustion stability throughout the propellant burning process, the length over diameter $L/D^{(i)}_{\text{prop}}$, volumetric $v^{(i)}_{f,\text{prop}}$ and web fractions $w^{(i)}_{f,\text{prop}}$ of each propellant are bounded by values that can be found in the relevant literature [4], [8]. Finally, the pressure $\mathbf{p}^{(i)}_{\text{iter}}(k)$ computed at each time step and sample of the iterative process must not be below the atmospheric pressure p_{atm}, nor above a prescribed maximum p_{max}. These constraints are considered in the design optimization problem shown in Sec. III, as seen in (5)–(9).

The methodology developed for the multi-pulse SRM modelling and simulation shown in this section has been validated with the Solid Performance Program (SPP), against which the performance results obtained were satisfactorily comparable. This software is widely employed in the aerospace field as a validated benchmark [9], [10], having been used in National Aeronautics and Space Administration (NASA) projects for solid booster design [11], [12].

III. Design Optimization

In this section, the multi-pulse SRM design optimization problem is formulated. The particular case of two chambers, a dual-pulse design, is then optimized. It is desired to maximize a certain objective function $y(X)$, which, in the present work, will be considered either the total impulse at sea level I or the potential range $\mathcal{R}$, given by (1) and (2), respectively.

A. Problem formulation

The decision vector is constructed as follows. Structural and thermodynamical properties are arbitrated in this instance. As such, the propellant grain types $\text{geo}^{(i)}_{\text{prop}}$, as well as chamber proportion factors $\Phi^{(i)}_{\text{ch}}$, are decision variables. The number of nozzles n_{noz}, their geometry noz_{geo} and reduction factor $\phi_{\text{cone,noz}}$ are also considered. The decision vector X is thus built as per (3). It is important to note that some decision variables are categorical, making this an MIP problem.

$$X = \Big[\text{geo}^{(1)}_{\text{prop}}, \cdots, \text{geo}^{(n_{\text{ch}})}_{\text{prop}}, \Phi^{(1)}_{\text{ch}}, \cdots, \Phi^{(n_{\text{ch}})}_{\text{ch}}, \\ n_{\text{noz}}, \text{noz}_{\text{geo}}, \phi_{\text{cone,noz}}\Big] \tag{3}$$

As it has been mentioned in Sec. II, some constraints have been considered during the design problem, and shall be part of the optimization problem as well. Therefore, the multi-pulse SRM design optimization problem can be formulated as follows. The bounds for the quantities in the constraints (6)–(8) depend on $\text{geo}^{(i)}_{\text{prop}}$, making the constraints vary with relation to the decision vector.

$$\max_{X} \quad y(X) \tag{4}$$

$$\text{s.t.} \quad \sum_{i=1}^{n_{\text{ch}}} \Phi^{(i)}_{\text{ch}} = 1 \tag{5}$$

$$a_1 \leq L/D^{(i)}_{\text{prop}} \leq a_2 \tag{6}$$

$$b_1 \leq v^{(i)}_{f,\text{prop}} \leq b_2 \tag{7}$$

$$c_1 \leq w^{(i)}_{f,\text{prop}} \leq c_2 \tag{8}$$

$$p_{\text{atm}} < \mathbf{p}^{(i)}_{\text{iter}}(k) < p_{\text{max}} \tag{9}$$

To study this optimization problem, two algorithms shall be used. The first is a hybrid random restart hill climbing (HRRHC) algorithm, presented in Sec. III-C1, and the second is an adapted genetic algorithm (GA), in Sec. III-C2. These two types of algorithms, widely used as a benchmark [13], were chosen to assess whether the usage of a metaheuristic algorithm is justified in the context of a multi-pulse SRM design optimization problem. Results for each algorithm are included. Random number generator seeds ranging from 1 to 50 were set for all cases, with a uniform distribution being used. The results are discussed in Sec. III-C3. The particular case of $n_{\text{ch}} = 2$, a dual-pulse design, is studied, with the same propellant used for both chambers. The decision variable bounds and types are specified in Tab. I.

TABLE I
Decision variable bounds

Decision variable (Type)	**Bounds**
$\text{geo}^{(1)}_{\text{prop}}$ (Categorical)	{'END','FYN','IBT'}
$\text{geo}^{(2)}_{\text{prop}}$ (Categorical)	
$\Phi^{(1)}_{\text{ch}}$ (Continuous)	$[0, 1]$
$\Phi^{(2)}_{\text{ch}}$ (Continuous)	
n_{noz} (Categorical)	$\{1, 2, 4\}$
noz_{geo} (Categorical)	{'cone','bell'}
$\phi_{\text{cone,noz}}$ (Continuous)	$[0, 1]$

B. Algorithms used

To adapt the developed algorithms to the constraints shown in (4), a constraint residual z_i is computed as per (10), for a given constraint. Given k constraints, a constraint residual vector $Z = z_1, \cdots, z_k$ is built.

$$z_i = \begin{cases} |a_{ij}X_i - b| & \text{if} \quad a_{ij}X_i = b \\ a_{ij}X_i - b & \text{if} \quad a_{ij}X_i \leq b \\ b - a_{ij}X_i & \text{if} \quad a_{ij}X_i \geq b \end{cases} \tag{10}$$

Based on a constraint residual z_i and a corresponding tolerance z_i^{tol}, its penalty p_i is formulated as per (11), from which the constraint penalty vector $P = p_1, \cdots, p_k$ is constructed.

$$p_i \coloneqq \left(z_i > z_i^{\text{tol}}\right) \iff p_i = \begin{cases} 1 & \text{if} \quad z_i > z_i^{\text{tol}} \\ 0 & \text{otherwise} \end{cases}, \tag{11}$$

From these definitions, Alg. 1 is conceived to tackle both equality and inequality constraints in the optimization loop of the algorithms to be developed, for a maximization scenario. This pipeline allows certain constraints to be handled at increased computational precision with relation to others, which is useful for ensuring the feasibility of the solutions obtained, when a certain constraint is more important compared to others in a physical sense.

Algorithm 1 Constraint handling strategy for maximization

Inputs: Constraint tolerance vector Z^{tol}, penalty coefficient p_{coeff}, unconstrained objective function y
Outputs: Constrained objective function $\tilde{y}$
Compute constraint vector $Z = (z_1, \cdots, z_k)$
Initialize number of violations
for each $z_i \in Z$ **do**
    Compute constraint penalty p_i
end for
if $\exists p_i \in P : p_i = 1$ **then**
    return $\tilde{y} = y - p_{\text{coeff}}$
else
    return $\tilde{y} = y$
end if

1) Hybrid random restart hill climbing algorithm: The HRRHC algorithm was conceived with the intent to reduce the solution space of an MIP problem to one with only continuous decision variables, in which a gradient-based method is employed. The underlying idea of the algorithm consists in computing all possible combinations of the categorical decision variables and, for each combination, the problem is solved for the remaining continuous decision variables. The pseudocode of this algorithm can be found in Alg. 2. Here, a sequential least squares quadratic programming (SLSQP) solver is chosen as the gradient-based method, whose constraint handling follows the strategy shown in Alg. 1.

Algorithm 2 Hybrid random restart hill climbing algorithm

Inputs: Decision vector X and bounds, initial population size n, objective function
Outputs: s_{best}, x_{best}
Generate initial population X_{init} with n individuals
Create set C with all combinations of categorical decision variables
while $i \leq n$ **do**
    $x_{\text{cont}} \leftarrow$ Continuous variables from $X_{\text{init}}(i)$
    for each $x_{\text{cat}} \in C$ **do**
        $x_{\text{current}} \leftarrow \{x_{\text{cont}} \cup x_{\text{cat}}\}$
        Solve problem with SLSQP and x_{current} as the initial guess, with constraint handling strategy of Alg. 1
        $s_{\text{opt}} \leftarrow$ Optimal objective function value
        $x_{\text{opt}} \leftarrow$ Individual of s_{opt}
        if $s_{\text{opt}} > s_{\text{best}}$ **then**
            $s_{\text{best}} \leftarrow s_{\text{opt}}$
            $x_{\text{best}} \leftarrow x_{\text{opt}}$
        end if
    end for
    $i \leftarrow i + 1$
end while
return $s_{\text{best}}, x_{\text{best}}$

2) Adapted genetic algorithm: An adapted GA was chosen as the metaheuristic algorithm to solve the overall MIP problem. It was implemented as per the pseudocode found in Alg. 3. The idea behind the usage of this algorithm is to verify whether it can explore the solution space more thoroughly when compared to HRRHC, in which, as aforementioned, the exploration is conditioned by the generated initial population. It uses the constraint handling strategy developed in Alg. 1 at the beginning of every generation, ensuring that penalties are applied to the objective function prior to generating any offspring. This should cause the algorithm to more quickly deviate from unfeasible regions of the solution space, thus causing it to be more extensively explored when compared to GA implementations where constraint handling is only performed after offspring generation. Should the solution space of the problem be nonconvex, it is expected that this adapted GA yields better solutions when compared to HRRHC.

Algorithm 3 Adapted genetic algorithm

Inputs: Decision vector X and bounds, initial population size n, number of generations n_{gen}, objective function
Outputs: s_{best}, x_{best}
Generate initial population X_{init} with n individuals
$X_{\text{current}} \leftarrow X_{\text{init}}$
while $i \leq n_{\text{gen}}$ **do**
    Evaluate objective function for all $x \in X_{\text{current}}$ and apply constraint handling strategy of Alg. 1
    $X_{\text{parents}} \leftarrow$ Select individuals from X_{current}
    $X_{\text{offspring}} \leftarrow$ Apply crossover to X_{parents}
    $X_{\text{offspring}} \leftarrow$ Apply mutation to $X_{\text{offspring}}$
    $X_{\text{elite}} \leftarrow$ Elite individuals from X_{current}
    $X_{\text{next}} \leftarrow \{X_{\text{offspring}} \cup X_{\text{elite}}\}$
    $X_{\text{current}} \leftarrow X_{\text{next}}$
    $i \leftarrow i + 1$
end while
$s_{\text{best}} \leftarrow$ Best objective function obtained
$x_{\text{best}} \leftarrow$ Individual of s_{best}
return $s_{\text{best}}, x_{\text{best}}$

C. Result discussion

The results obtained for the two aforementioned algorithms are presented and discussed in this section. A constraint tolerance of 10^{-6} was used for all constraint residuals z_i, with a penalty coefficient of 10^{10}, which is adjusted to the order of magnitude of the objective functions considered. The total impulse at sea level (1) and the potential range (2) are used as objective functions. An initial population of 200 individuals was chosen. It was verified that a population size higher than this value does not lead to better solutions nor a substantial increase in solution space exploration, hence this choice. The decision vector is formed and bounded as specified in Tab. I. Convergence results are presented in terms of best objective function value obtained versus number of objective function evaluations, for a matter of result comparison to be performed in Sec. III-C3, including a visualization of separate seed runs, mean objective function value and respective standard deviation. For the case of total impulse at sea level I, 100000 objective function evaluations are required to reach a plausible

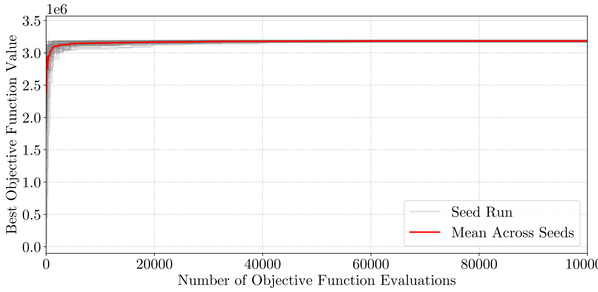

Fig. 5. Convergence for the HRRHC algorithm for total impulse at sea level

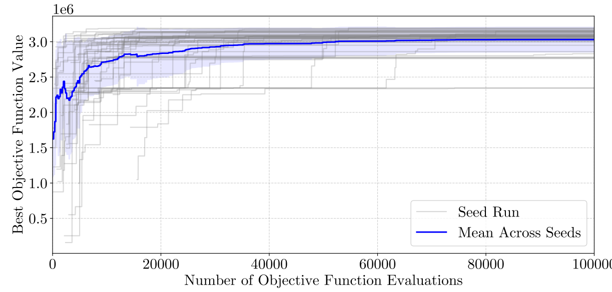

Fig. 7. Convergence for the GA for total impulse at sea level

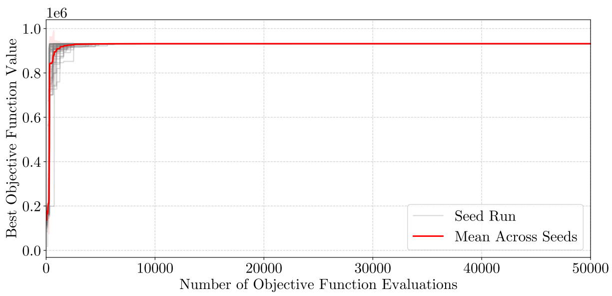

Fig. 6. Convergence for the HRRHC algorithm for potential range

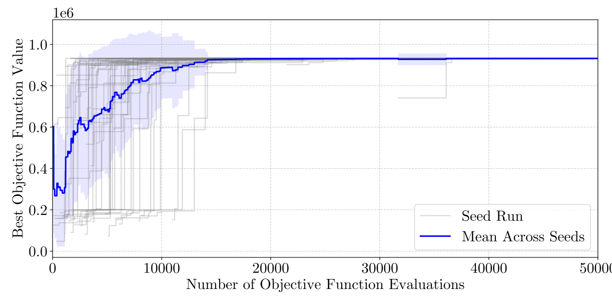

Fig. 8. Convergence for the GA for potential range

steady state in the convergence, whereas, for the case of potential range $\mathcal{R}$, 50000 evaluations suffice.

1) HRRHC results: With the setup information previously described, Alg. 2 was applied. Fig. 5 shows the convergence for total impulse at sea level, and Fig. 6 for potential range. The best objective function values and respective individuals obtained, across all seeds, are shown below:

$$I = 3194193.3759$$
$$X = [\text{'IBT'}, \text{'FYN'}, 0.44365, 0.55635, 4, \text{'bell'}, 0.98603]$$

$$\mathcal{R} = 931974.7150$$
$$X = [\text{'END'}, \text{'END'}, 0.67276, 0.32724, 2, \text{'bell'}, 0.97219]$$

Unfeasible solutions have been omitted from Fig. 5 and 6. It is important to note that, as Alg. 2 evidences, for the case of the continuous variables, the solution space exploration is conditioned by the generated initial population, which, in a nonconvex setting, may cause the HRRHC algorithm to get stuck in a local optimum [13]. Indeed, as the two convergence plots reveal, the solution space exploration is relatively narrow, with standard deviation being low and steady state reached at roughly 40000 evaluations for I and roughly 5000 for $\mathcal{R}$.

2) GA results: With the setup information previously described, 500 generations were chosen so as to have a similar total number of objective function evaluations to that which was prescribed in the beginning of this section, allowing for posterior result comparison. A manual hyperparameter tuning procedure was performed to reach reasonable values for the crossover and mutation rates, so as to prevent the algorithm from converging within too few objective function evaluations and from having a narrow exploration of the solution space. This was done around standard values often used in the literature [13], having reached crossover and mutation rates of 0.80 and 0.15, respectively. Alg. 2 was then applied, leading to the results seen in Fig. 7 and 8. The best objective function values and respective individuals obtained, across all seeds, are displayed below:

$$I = 3194712.1212$$
$$X = [\text{'IBT'}, \text{'FYN'}, 0.44378, 0.55634, 4, \text{'bell'}, 0.97810]$$

$$\mathcal{R} = 933113.8243$$
$$X = [\text{'END'}, \text{'END'}, 0.70680, 0.29372, 2, \text{'bell'}, 0.96505]$$

Unfeasible solutions are omitted as before. Convergence reached steady state at roughly 80000 evaluations for I, and 20000 for $\mathcal{R}$. The small fluctuation observed in Fig. 8 at around 32000 evaluations occurs due to the fact that, in one of the seeds, the first feasible solution is only found here, which causes the mean to slightly shift and, consequently, the standard deviation to slightly increase. Therefore, this phenomenon occurs for a matter of randomness and is disregarded in the remaining discussion.

Genetic algorithms often require more function evaluations to reach a certain solution quality, when compared to gradient-based methods [14]. Consequently, a question is raised. Given a certain maximum number of function evaluations, which algorithm yields the best solution? This matter is explored in Sec. III-C3.

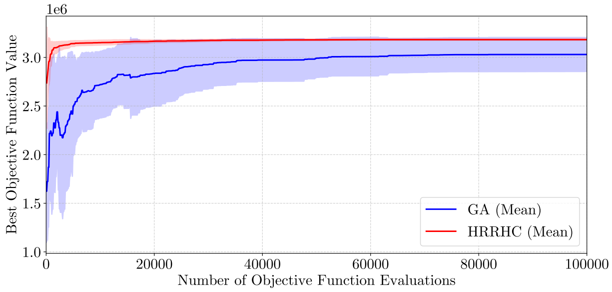

Fig. 9. Comparison between GA and HRRHC (total impulse at sea level)

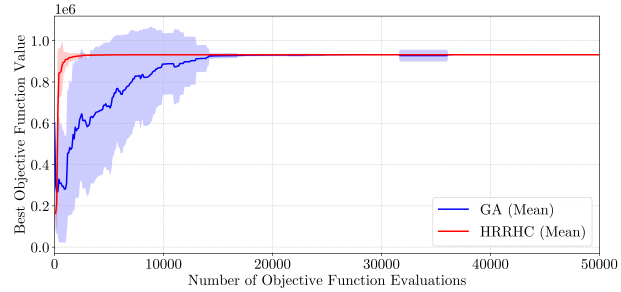

Fig. 10. Comparison between GA and HRRHC (potential range)

3) Performance comparison: As mentioned in Sec. III-C1 and III-C2, the HRRHC algorithm is likely to get stuck in local optima of a nonconvex solution space, whereas the GA is expected to more widely explore said solution space. At the same time, the latter is likely to require more objective function evaluations to yield solutions as good as those yielded by the HRRHC algorithm. In order to compare them, the convergence results obtained through both algorithms, across all seeds and for both objectives, are now visualized together in Fig. 9 and 10. Individual seed runs are omitted for readability.

The comparison plots reveal that, in relation to the HRRHC algorithm, convergence took many more objective function evaluations for the GA. Indeed, the results show that, overall, the GA took approximately 4 times more objective function evaluations than the HRRHC algorithm to converge, for both objectives compared. Therefore, as verified in Sec. III-C1 and III-C2, despite the slight improvements by the GA in the best objective function values obtained compared to the HRRHC algorithm, the latter outperforms, on average, the former. Nevertheless, the fact that the GA managed to obtain a better solution in comparison to the gradient-based method indicates that the solution space of the dual-pulse SRM design optimization problem is nonconvex to an extent [13]. In this problem, the categorical solution space is sufficiently small, to the point it can be efficiently bruteforced by the HRRHC algorithm, allowing it to reach a high-quality solution in much fewer objective function evaluations compared to the GA. Even then, the continuous solution space was, in at least one of the seeds, better solved by the latter. Indeed, as the number of pulses in the generic multi-pulse SRM design increases, the nonconvexity of the solution space is expected to increase as well. Therefore, in limit, it is expected that the HRRHC algorithm would be outperformed, including on average, by the GA, at which point metaheuristic usage becomes even more advantageous in relation to gradient-based methods.

IV. Conclusions

The present work proposed an MIP formulation for the design optimization problem of the multi-pulse solid rocket motor. A proposed technique which allows for constraint handling to be performed at different levels of precision across different constraints was used. This served as the backbone for the novel HRRHC algorithm, which reduces the overall MIP problem to a continuous one, as well as an adapted GA. The results confirm the suitability of the proposed technique and algorithms for the problem tackled in this paper.

The HRRHC algorithm yielded better results, on average across all seeds, compared to the GA, for both objective functions considered. However, in terms of best across all seeds, the latter managed to outperform the former, which indicates that the solution space of the design optimization problem of the dual-pulse solid rocket motor is nonconvex to an extent. As the number of pulses increases, so does the nonconvexity of the solution space, thereby justifying metaheuristic usage for the generic design optimization problem of the multi-pulse solid rocket motor. This consists in potential future work in the area, including the expansion to a multi-objective setting.

References

[1] A. S. S. M. Barkat Ullah, R. Sarker, and C. Lokan, “Handling equality constraints in evolutionary optimization,” European Journal of Operational Research, vol. 221, no. 3, pp. 480–490, 2012, doi:10.1016/j.ejor.2012.01.047.

[2] A. Rafique, H. LinShu, Q. Zeeshan, et al., “Integrated system design of an air-launched small space launch vehicle using genetic algorithms,” in Proc. AIAA Joint Propulsion Conf., 2009.

[3] M. I. Abyan, A. Nuryaman, B. H. Jihad, et al., “Design optimization of propellant grain and nozzle contour,” J. Eng. Technol. Sci., vol. 54, no. 5, 2022.

[4] G. P. Sutton and O. Biblarz, Rocket Propulsion Elements, 7th ed. Wiley, 2013.

[5] D. W. Netzer and G. Jensen, Tactical Missile Propulsion. AIAA, 1996.

[6] K. W. Naumann and L. Stadler, Bayern-Chemie, “Double-pulse solid rocket motor technology—applications and technological solutions,” in Proc. AIAA Joint Propulsion Conf., 2010.

[7] C. A. Phillips, “Energy management for a multiple-pulse missile,” M.S. thesis, Virginia Polytechnic Institute and State Univ., 1986.

[8] NASA, Solid Propellant Grain Design and Internal Ballistics, NASA SP-8076, 1972.

[9] H. Navaz, A. L. Dang, R. Chamberlain, et al., “A new particle break-up model for solid rocket motors,” in Proc. JANNAF, 2017.

[10] H. K. Navaz, A. L. Dang, S. S. Dunn, et al., “Parallel processing VIPER and code validation,” JANNAF, Paper 9405, 2023.

[11] G. E. Jensen, A. L. Holzman, S. O. Leisch, et al., Hybrid Propulsion Technology Program, vol. II, NASA Marshall Space Flight Center, 1989.

[12] H. K. Navaz, D. E. Coats, G. Davis, and A. Pourmovahed, “An innovative method of NOx reduction through fuel additives,” NASA Glenn Research Center, 2002.

[13] S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 3rd ed. Prentice Hall, 2016.

[14] D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, 1989.