



HAL
open science

Surrogate model-based multi-objective MDO approach for partially Reusable Launch Vehicle design

Loïc Brevault, Mathieu Balesdent, Ali Hebbal, Antoine Patureau de Mirand

► **To cite this version:**

Loïc Brevault, Mathieu Balesdent, Ali Hebbal, Antoine Patureau de Mirand. Surrogate model-based multi-objective MDO approach for partially Reusable Launch Vehicle design. AIAA Scitech 2019 Forum, Jan 2019, San Diego, United States. 10.2514/6.2019-0704 . hal-02913046

HAL Id: hal-02913046

<https://hal.science/hal-02913046>

Submitted on 31 May 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Surrogate model-based multi-objective MDO approach for partially Reusable Launch Vehicle design

Loic Brevault* and Mathieu Balesdent †
ONERA/DTIS, Université Paris Saclay, F-91123 Palaiseau Cedex, France

Ali Hebbal ‡
ONERA/DTIS, Université Paris Saclay, Université de Lille, CNRS/CRISTAL, INRIA Lille

Antoine Patureau De Mirand §
CNES, Direction des lanceurs – Avant-Projets et Prototypes, 75612 Paris Cedex

Reusability of the first stage of launch vehicles may offer new perspectives to lower the cost of payload injection into orbit if sufficient reliability and low refurbishment costs can be achieved. One possible option that may be explored is to design the launch vehicle first stage for both reusable and expendable uses, in order to increase the flexibility and adaptability to different target missions. This paper proposes a multi-level MDO approach to design aerospace vehicles addressing multi-mission problems. The proposed approach is focused on the design of a family of launchers for different missions sharing commonalities using multi-objective Bayesian Optimization to account for the computational cost associated with the discipline simulations. The multi-mission problem addressed in this paper considers two missions: a reusable configuration for a SSO orbit with a medium payload range and recovery of the first stage using a glider strategy; and an expendable configuration for a medium payload injected into a Geostationary Transfer Orbit (GTO). A dedicated MDO formulation introducing couplings between the missions is proposed in order to efficiently solve the multi-objective MDO problem while limiting the number of calls to the exact MDA thanks to the use of Gaussian Processes and multi-objective Efficient Global Optimization.

I. Introduction

REUSABILITY of the first stage of launch vehicles may offer new perspectives to lower the cost of payload injection into orbit if sufficient reliability and low refurbishment costs can be achieved. In function of various hypotheses on addressable institutional and commercial markets, different strategies for reusability of launch vehicle first stage may be investigated. One option that may be explored [1] is to design the launch vehicle first stage for both reusable and expendable uses, in order to increase the flexibility and adaptability to different target missions.

Following this strategy, this paper proposes a multi-level MDO approach to design aerospace vehicles addressing multi-mission problems. This category of design problems can be formulated as multi-objective MDO problems which involve numerous evaluations of MultiDisciplinary Analysis (MDA) used to assess the performance of the vehicles. This MDA involves computationally expensive coupled models (aerodynamics, propulsion, structural design, trajectory, *etc.*) [2].

Several multi-objective MDO approaches for launch vehicle design have been investigated in the literature [3–5]. Castellini *et al.* [3] proposed a comparison of seven population-based algorithms (*e.g.* NSGA-II [6], MOPSO [7], PAES [8]) dealing with multi-objective problems and compared them on expendable launch vehicle design problems. Two types of problem have been considered, ascent trajectory optimization with a fixed launcher design and launcher design optimization (with optimization of architectures: number of boosters, type of propulsion, type of engine cycle, *etc.*). In this work, MultiDiscipline Feasible (MDF) formulation is carried out to couple the different disciplines and conceptual design models are used. The two considered objectives are the sum of square errors on semiaxis, eccentricity and inclination and the payload mass (or Gross Lift-Off Weight - GLOW). Fazeley *et al.* [4] proposed to compare MDF and

* Research Engineer, ONERA/DTIS, Université Paris Saclay, loic.brevault@onera.fr

† Research Engineer, ONERA/DTIS, Université Paris Saclay, mathieu.balesdent@onera.fr

‡ Ph.D. student, ONERA/DTIS, Université Paris Saclay, Université de Lille, CNRS/CRISTAL, INRIA Lille, ali.hebbal@onera.fr

§ Project manager A6 Evolution and Ariane Next, CNES, Direction des lanceurs, antoine.patureau@cnese.fr

Collaborative Optimization (CO) for the multi-objective optimization of a bi-propellant space propulsion system design for an expendable launcher. The two objectives for the design of the propulsion systems are the total wet mass and the total impulse. NSGA-II algorithm is used to perform multi-objective optimization. In comparison, for this specific problem, the authors concluded that MDF requires less calls to the disciplines than CO formulation. Fujikawa *et al.* [5] performed a conceptual design study for a Two-Stage-To-Orbit space plane with ethanol-fueled rocket-based combined cycle engine using a multi-objective MDO approach. An All-At-Once MDO formulation is employed in the proposed process. Three objectives are considered for the problem: the payload mass, the GLOW and the take-off velocity. At each iteration, the multi-objective problem is transformed into its relevant single-objective problem via min-max goal programming, and is subsequently solved using a gradient-based optimization method (SQP) [9]. Kosugi *et al.* [10] used a multi-objective genetic algorithm to design a hybrid rocket. The two considered objectives are the GLOW and the maximal reached altitude by the sounding rocket and a MDF formulation is carried out.

To the best of the author knowledge, most of the existing papers in the literature focus on multi-objective problem for a single launch vehicle using population-based algorithms. In contrast, this paper is focused on the design of a family of launchers for different missions sharing design commonalities (one reusable mission dedicated to Sun Synchronous Orbit and one expendable mission to Geostationary Transfer Orbit) using Bayesian Optimization to account for the computational cost associated with the discipline simulations.

In general, due to the repeated evaluations of the MDA, multi-objective MDO solving using a classical optimization algorithm such as evolutionary approaches (*e.g.* NSGA-II [6], OMOPSO [11], PAES [8]) is intractable. In this paper, an alternative technique is used based on Bayesian multi-objective optimization. Bayesian optimization algorithms are employed to deal with computationally intensive black-box function problems. They rely on surrogate models to build a statistical relationship between the input design variables and the quantities of interest (objective and constraint functions given by the MDA), to predict these latter using a limited number of exact MDA simulations. The evaluation cost of the surrogate models is much cheaper compared to the exact MDA. Bayesian optimization methods have been extended to solve such multi-objective problems [12]. Specific infill criteria such as the Expected HyperVolume Improvement [13, 14] can be used to determine the candidates for which the exact MDA needs to be evaluated. These candidates are added to the dataset and the surrogate models are updated. This process is repeated until a given criterion is reached. In this paper, a surrogate model-based MDO approach using a multi-level decomposition and relying on Gaussian Processes (GP) is proposed to estimate with a reduced computational cost the optimal Pareto Front for multi-mission aerospace vehicle design problems.

The proposed multi-level MDO formulation is illustrated on the design of a family of conceptual Two-Stage-To-Orbit launch vehicles. The multi-mission problem considers two missions: a reusable configuration for an SSO orbit with a medium payload range and recovery of the first stage using a glider strategy; and an expendable configuration for a payload injected into a Geostationary Transfer Orbit (GTO). Glider strategy (also known as deadleaf) belongs to the family of Vertical Take-Off and Horizontal Landing approaches and is an alternative strategy to toss-back configuration by taking advantage of aeronautical technologies (*e.g.* addition of lifting surface). After the separation of the first stage, this later is turned around and the main engines are reignited in order to inverse the horizontal velocity. Then, a reentry into the atmosphere is performed and the lifting surfaces equipping the stage are used to perform an aerodynamic resource and to produce lift to enable a gliding phase up to horizontal landing near the launch site.

The following of the paper is organized as follows. In Section 2, an overview of Multi-objective Bayesian optimization is presented along with Gaussian Process principles and infill criterion. In Section 3, the application of the proposed MDO methodology to the design of multi-mission reusable launch vehicles is described with a presentation of the expandable and reusable missions along with the vehicle concept and design requirements. Then, the multi-level MDO formulation for the multi-mission problems is introduced and the disciplinary models are presented with the MultiDisciplinary Analysis. In Section 4, the results of the proposed multi-objective MDO process on the launch vehicle design problem are presented, compared to reference approaches and discussed in details.

II. Multi-objective Bayesian optimization

The design of a reusable launch vehicle for multi-mission can be described as a multi-objective MDO problem. A general multi-objective problem characterized by n objective functions and q constraints may be formulated as:

$$\min \quad \mathbf{f}(\mathbf{z}) = [f_1(\mathbf{z}), \dots, f_n(\mathbf{z})] \quad (1)$$

$$\text{w.r.t.} \quad \mathbf{z} \in \mathbb{R}^m$$

$$\text{s.t.} \quad g_i(\mathbf{z}) \leq 0, \quad i = 1, \dots, q \quad (2)$$

$$\mathbf{z}_{\min} \leq \mathbf{z} \leq \mathbf{z}_{\max} \quad (3)$$

with \mathbf{z} the vector of design variables of dimension m , $\mathbf{f}(\cdot)$ the objective function vector of size n and $g_i(\cdot)$ the i^{th} constraint function of the vector of constraint $\mathbf{g}(\cdot)$ of size q . Evolutionary algorithms such as NSGA-II (Non-dominated Sorting Genetic Algorithm II) [6] or SMPSO (Speed-constrained Multi-objective Particle Swarm Optimization) [15] are classically used to solve multi-objective optimization problems. However, due to the computationally intensive nature of the MultiDisciplinary Analysis involved in launch vehicle design, this type of algorithms is not suited for such problems.

In this work, an alternative technique is used based on Bayesian multi-objective optimization. It relies on surrogate models to build a statistical relationship between the input design variables and the quantities of interest (objective and constraint functions), to predict these latter using a limited number of data also called Design of Experiments (DoE). The evaluation cost of the surrogate models is much cheaper compared to the exact functions. In this paper, Efficient Global Optimization (EGO) developed by Jones *et al.* [16] is considered. It relies on Kriging surrogate model [17] which is based on the Gaussian Process (GP) theory. The main advantage of Kriging compared to alternative surrogate modeling techniques is that in addition to the prediction, it provides uncertainty estimation of the model response. Based on these two information, an infill criterion (such as the Expected Improvement adapted to multi-objective problems) is used to iteratively add the most promising candidate to the existing DoE. This solution is then evaluated on the exact functions (MDA providing objective and constraint function values) and the GPs are updated and so on, until a stopping criterion is reached.

A. Gaussian Process (GP)

Let us initially consider a single objective unconstrained optimization problem. A surrogate model based on Gaussian Process can be built from a set of samples (DoE) of size N , $\mathcal{Z}_N = \{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(N)}\}$ the input data set ($\mathbf{z} \in \mathbb{R}^m$) and the corresponding objective function responses $\mathcal{Y}_N = \{y^{(1)} = f(\mathbf{z}^{(1)}), \dots, y^{(N)} = f(\mathbf{z}^{(N)})\}$. Then, it is possible to use this surrogate model to predict the exact function response $f(\cdot)$ at a new point without evaluating it. The advantage of this approach is relative to the computational cost reduction. Indeed, instead of evaluating the expensive black-box function $f(\cdot)$, the surrogate model (which is much cheaper to evaluate) is used. The interest of the GP surrogate model is that it provides a variance estimation of the prediction in addition to this latter, which makes it suitable in a global optimization framework.

A GP is used to describe a distribution over functions, it corresponds to a collection of infinite random variables, any finite number of which has a joint Gaussian distribution. A GP is characterized by its mean and covariance functions. In a GP regression, a GP prior is placed on the unobserved function $f(\cdot)$ using a prior covariance function $k^\Theta(\mathbf{z}, \mathbf{z}')$ that depends on several hyper-parameters Θ and mean function μ . As the trend of the response is *a priori* unknown a constant mean function μ is assumed (then GP is named ordinary Kriging). Therefore, the GP may be written as $\hat{f}(\mathbf{z}) \sim \mathcal{N}(\mu, k^\Theta(\mathbf{z}, \mathbf{z}'))$ and has a multivariate distribution on any finite subset of variables, in particular on the DoE \mathcal{Z}_N (noted \mathbf{f}_N), $\mathbf{f}_N | \mathcal{Z}_N \sim \mathcal{N}(\mathbf{1}\mu, \mathbf{K}_{NN}^\Theta)$ where \mathbf{K}_{NN}^Θ is the covariance matrix constructed from the parametrized covariance function $k(\cdot)$ on \mathcal{Z}_N (in the rest the dependence on Θ is dropped for notation simplicity). The choice of the covariance function determines our prior assumptions of the function to be modeled. A Gaussian noise variance is considered, such that the relationship between the latent function values $f(\mathcal{Z}_N)$ and the observed responses \mathcal{Y}_N is given by: $p(\mathbf{y} | \mathbf{f}_N) = \mathcal{N}(\mathbf{y} | \mathbf{f}_N, \sigma^2 \mathbf{I})$. This Gaussian noise is necessary when the objective function or the constraints are non deterministic. The marginal likelihood is obtained by integrating out the latent function $f(\cdot)$:

$$p(\mathbf{y} | \mathcal{Z}_N, \Theta) = \mathcal{N}(\mathbf{y} | \mu, \mathbf{K}_{NN} + \sigma^2 \mathbf{I}) \quad (4)$$

In order to train the GP, it is possible to maximize the negative log marginal likelihood to find the optimal values of the hyper-parameters Θ, μ, σ . All the kernel matrices implicitly depend on the hyperparameters Θ and the negative log

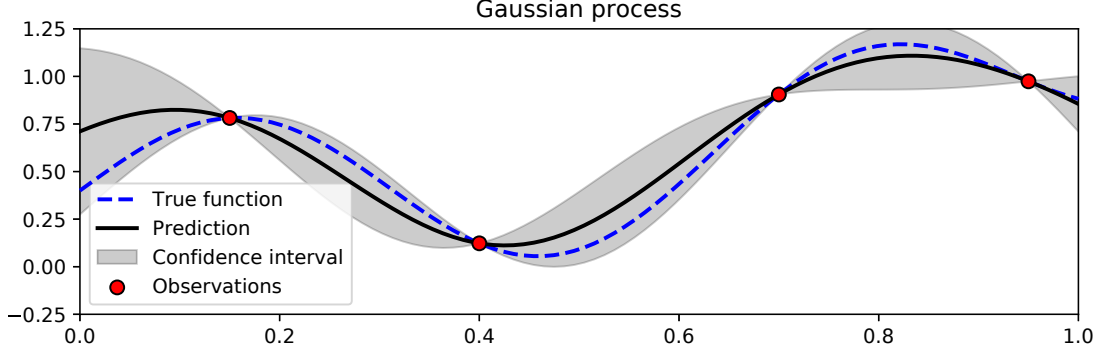


Fig. 1 Example of Gaussian process prediction and associated confidence interval

marginal likelihood (and its derivative) is given by:

$$L(\Theta | \mathcal{Z}_N, \mathcal{Y}_N) = \log(p(\mathbf{y} | \mathcal{Z}_N, \mathcal{Y}_N, \Theta)) \propto \log(|\hat{\mathbf{K}}_{NN}|) - \mathbf{y}^T \hat{\mathbf{K}}_{NN}^{-1} \mathbf{y} \quad (5)$$

$$\frac{dL}{d\Theta} = \mathbf{y}^T \hat{\mathbf{K}}_{NN}^{-1} \frac{d\hat{\mathbf{K}}_{NN}}{d\Theta} \hat{\mathbf{K}}_{NN}^{-1} \mathbf{y} + \text{Tr} \left(\hat{\mathbf{K}}_{NN}^{-1} \frac{d\hat{\mathbf{K}}_{NN}}{d\Theta} \right) \quad (6)$$

Gradient-based optimizers may be used to minimize the negative log marginal likelihood in order to determine the hyperparameter values of the trained GP.

After the training, the prediction at a new point \mathbf{z}^* is made by using the conditional properties of a multivariate normal distribution (Figure 1):

$$p(y^* | \mathbf{z}^*, \mathcal{Z}_N, \mathcal{Y}_N, \Theta) = \mathcal{N}(y^* | \hat{y}^*, \hat{\delta}^{*2}) \quad (7)$$

with \hat{y}^* , $\hat{\delta}^{*2}$ the mean prediction and the associated variance given by:

$$\hat{y}^* = \mu + \mathbf{k}_{\mathbf{z}^*}^T (\mathbf{K}_{NN} + \sigma^2 \mathbf{I})^{-1} (\mathbf{y} - \mathbf{1}\mu) \quad (8)$$

$$\hat{\delta}^{*2} = k_{\mathbf{z}^*, \mathbf{z}^*} - \mathbf{k}_{\mathbf{z}^*}^T (\mathbf{K}_{NN} + \sigma^2 \mathbf{I})^{-1} \mathbf{k}_{\mathbf{z}^*} + \sigma^2 \quad (9)$$

where $k_{\mathbf{z}^*, \mathbf{z}^*} = k(\mathbf{z}^*, \mathbf{z}^*)$ and $\mathbf{k}_{\mathbf{z}^*} = [k(\mathbf{z}^{(i)}, \mathbf{z}^*)]_{i=1, \dots, N}$

An example of a 1D function and a GP build based on four observations (samples) is represented in Figure 1. The confidence interval is based on the associated GP variance ($\pm 3\hat{\delta}^{*2}$). The variance is null at the observation and increases as the distance from an existing data sample increases. GP is very interesting in the complex design because it provides the design with both the prediction of the model and its estimated uncertainty. This possibility to quantify the surrogate model prediction uncertainty is exploited in optimization to balance between global search exploration and exploitation. To solve an optimization problem with Bayesian optimization, an infill criterion such as the Expected Improvement [16] is used to iteratively add the most promising candidate to the existing DoE. This solution is then evaluated on the exact objective function and the GP is updated. This process is repeated until a stopping criterion is reached. Bayesian optimization has been extended to solve multi-objective problems as discussed in the next Section.

B. Multi-objective Infill Sampling Criteria

For multi-objective problem, different approaches have been proposed to select the infill sample candidates to be added to the current DoE such as aggregation-based techniques [18] or domination-based techniques [13, 14]. In this study the approach proposed by Emmerich *et al.* [14] based on the Expected HyperVolume Improvement is used. The notion of Expected HyperVolume Improvement (EHVI) is an adaptation of the Expected Improvement (EI) to the multi-objective case. Consider an unconstrained multi-objective optimization problem characterized by n objectives, $\mathbf{y} = \mathbf{f}(\mathbf{z}) = [y_1, \dots, y_n]$ stands for the objective function outputs. Considering a DoE of size N , $\mathcal{Z}_N = \{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(N)}\}$ the input data set and the corresponding objective function responses $\mathcal{Y}_N = \{\mathbf{y}^{(1)} = \mathbf{f}(\mathbf{z}^{(1)}), \dots, \mathbf{y}^{(N)} = \mathbf{f}(\mathbf{z}^{(N)})\}$. Let

$\mathbb{V} = \{\mathbf{y} \in \mathbb{R}^n \mid \mathbf{y}^L \leq \mathbf{y} \leq \mathbf{y}^U\}$ be a finite hypervolume of the objective space where all the possible solutions lie, with $\mathbf{y}^L = [\min f_1(\mathbf{z}), \dots, \min f_n(\mathbf{z})]$ the ideal objective candidate and \mathbf{y}^U a chosen upper point (nadir point). The dominated hypervolume $H_{\mathcal{Y}_N}$ of the DoE responses \mathcal{Y}_N is defined as:

$$H_{\mathcal{Y}_N} = \{\mathbf{y} \in \mathbb{V} \mid \exists i \in \{1, \dots, N\}, \mathbf{f}(\mathbf{z}^{(i)}) < \mathbf{y}\} \quad (10)$$

$H_{\mathcal{Y}_N}$ is a subset of \mathbb{V} whose points are dominated by the DoE responses \mathcal{Y}_N (the symbol $<$ stands for Pareto domination). Considering a new candidate $\mathbf{z}^{(N+1)}$, since $H_{\mathcal{Y}_N} \subseteq H_{\mathcal{Y}_{N+1}}$, the hypervolume improvement by adding this candidate is given by: $I_N(\mathbf{z}^{(N+1)}) = |H_{\mathcal{Y}_{N+1}}| - |H_{\mathcal{Y}_N}|$. Figure 2 illustrates the concepts introduced previously in the two objective case.

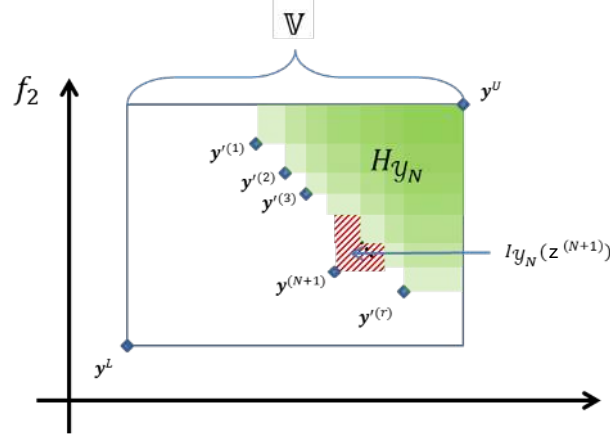


Fig. 2 Hypervolume improvement (hatched area) in a two objective case

Considering the collection of GP $\hat{\mathbf{Y}} = \{\hat{Y}_1(\mathbf{z}) \sim \mathcal{N}(\hat{y}_1, \hat{\sigma}_1^2), \dots, \hat{Y}_n(\mathbf{z}) \sim \mathcal{N}(\hat{y}_n, \hat{\sigma}_n^2)\}$ modeling the exact functions f_1, \dots, f_n . The Expected HyperVolume Improvement for a candidate solution \mathbf{z} is given by:

$$EHVI_{\mathcal{Y}_N}(\mathbf{z}) = \mathbb{E} [|H_{\mathcal{Y}_{N+1}}| - |H_{\mathcal{Y}_N}|] = \int_{\mathbb{V} \setminus H_{\mathcal{Y}_N}} \mathbb{P} [\hat{\mathbf{Y}}(\mathbf{z}) < \mathbf{p}] d\mathbf{p} \quad (11)$$

In the constrained case, a constrained infill criterion is considered which combines EHVI with the probability of feasibility [19] or the expected violation of the constraints [20]. Schonlau [19], proposed to integrate a measure of the feasibility of a candidate point into the unconstrained infill criterion (initially for single objective problems but extended to multi-objective problems here). The concept of probability of feasibility is defined as the probability that a point satisfies all the constraints (independent):

$$P_f(\mathbf{z}) = \prod_{i=1}^q \Phi \left(\frac{0 - \hat{g}_i(\mathbf{z})}{\hat{\sigma}_{g_i}(\mathbf{z})} \right) \quad (12)$$

with $\hat{g}_i(\cdot)$ the Kriging mean of the constraint function $g_i(\cdot)$, $\hat{\sigma}_{g_i}^2(\cdot)$ the prediction variance and $\Phi(\cdot)$ the cumulative distribution of the normal law. By multiplying this probability of feasibility to the Expected Hypervolume Improvement, the magnitude of the resulting quantity tends to zero in the design space areas where there is a low likelihood of the constraint feasibility and it tends to the EHVI value where there is a high likelihood of the constraint feasibility.

At each iteration of the Multi-Objective EGO (MO-EGO), this infill criterion $C(\mathbf{z}) = EHVI(\mathbf{z}) \times P_f(\mathbf{z})$ is maximized in order to identify the most promising candidate. This latter is evaluated on the exact objective and constraint functions and added to the DoE. All the surrogate models are updated and a new iteration can start if the convergence criterion is not satisfied. The different steps of MO-EGO are summarized in Figure 3.

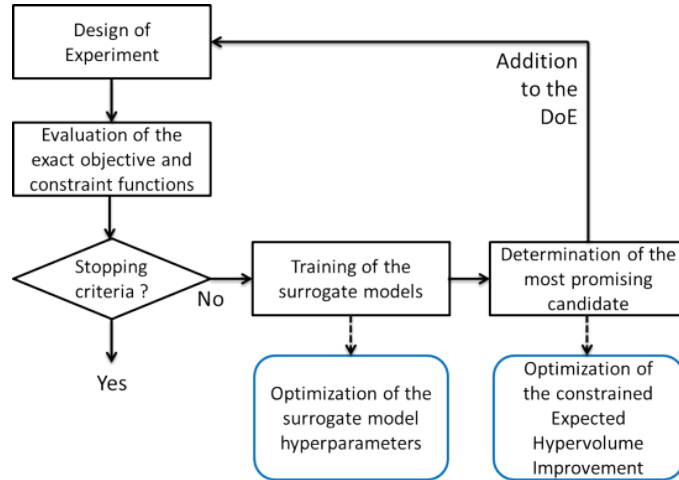


Fig. 3 Steps of Multi-objective EGO (MO-EGO)

III. Application to reusable launch vehicle design

In order to illustrate the proposed method to solve multi-mission launch vehicle design problem, an application to reusable launch vehicle design is presented in the following Section.

A. Description of vehicle concept, mission and design requirements

In function of different assumptions on addressable institutional and commercial satellite markets, different strategies for the reusability of the first stage of launch vehicles may be investigated. One option that may be explored [1] is to design the first stage for both reusable and expendable configurations in order to address different missions. Following this strategy, two baseline missions are considered in this paper:

- SSO mission, with recovery, refurbishment and reuse of the launch vehicle first stage,
- GTO mission, in expandable mode with no recovery of the vehicle first stage.

The characteristics of the SSO mission are:

Target orbit	SSO, 800km
Launch site	CSG, Kourou, French Guiana
Payload mass	6t
Vehicle type	Two-Stage-To-Orbit (TSTO) reusable vehicle

The characteristics of the GTO mission are:

Target orbit	GTO, 250km-36000km
Launch site	CSG, Kourou, French Guiana
Payload mass	To be determined
Vehicle type	Two-Stage-To-Orbit (TSTO) expendable vehicle

The proposed launch vehicle architecture should be able to perform these two missions. In order to identify the possible optimal architectures, the design of such a vehicle may be obtained through the solving of a multi-objective problem and determining the optimal Pareto front.

The considered launchers use Prometheus engines in both stages with adaptation of the nozzle for the first and the second stages [21]. For the reusable SSO mission, all the additional subsystems that are added to the vehicle first stage to enable recovery are part of a reusability kit. This kit is removed for the expendable GTO mission and can be added to another first stage for a future reusable missions. Moreover, for the expendable mission, two solid rocket boosters are added to the central core. For the reusable SSO mission, a glider strategy is used. The first stage is equipped with several subsystems such as lifting surfaces (wing and control surfaces) which are added to the stage to enable a glided return to the launch site (Figures 6, 7).

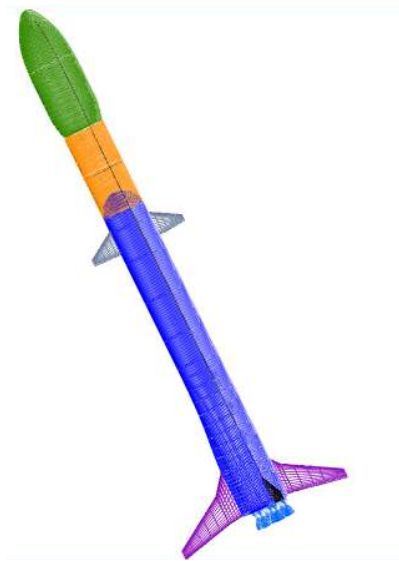


Fig. 4 Illustration of the reusable launch vehicle architecture

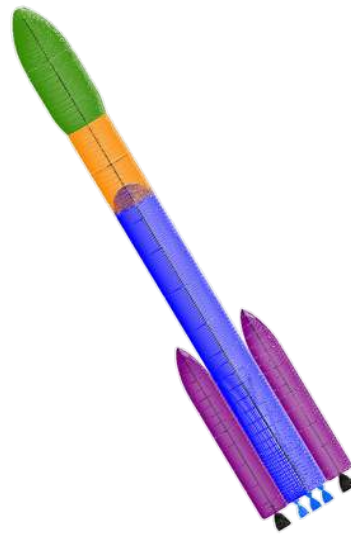


Fig. 5 Illustration of the expendable launch vehicle architecture

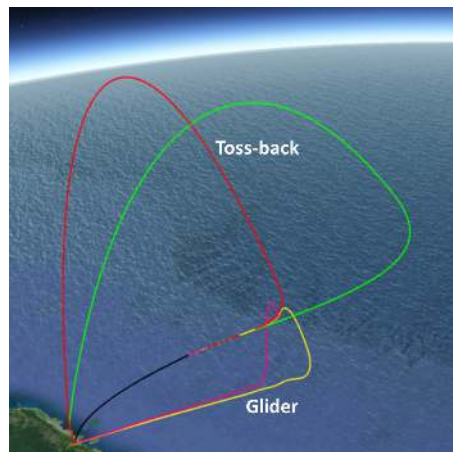


Fig. 6 Comparison of two toss-back trajectories (red, green) and two glider trajectories (yellow, pink)

The sequences of the glider strategy are presented in Figure 7. After the lift-off (step 1) and the separation of the first stage and second stage (step 2), the first stage is turned around and the main engines (one or several depending on the return strategy) are reignited (step 3) in order to inverse the horizontal velocity. This operation is performed usually in high altitude (above 50km) with limited drag. Then, a reentry into the atmosphere is performed (step 4) and the wings equipping the stage are used to perform an aerodynamic resource (step 5) and to produce lift to enable a glided phase up to horizontal landing near the launch site (step 6).

The reusability kit for the glider version of the 1st stage is composed of: lifting surfaces (main wings and canards), a fairing, a skirt between the fairing and the upper tank, a vertical tail, and additional power and avionics. Moreover, compared to classical expendable stage, the structures are reinforced to account for additional loads during the rocket engine boost to inverse horizontal velocity, the reentry and the aerodynamic resource, especially with respect to transverse loads which are not usual for such type of vehicles.

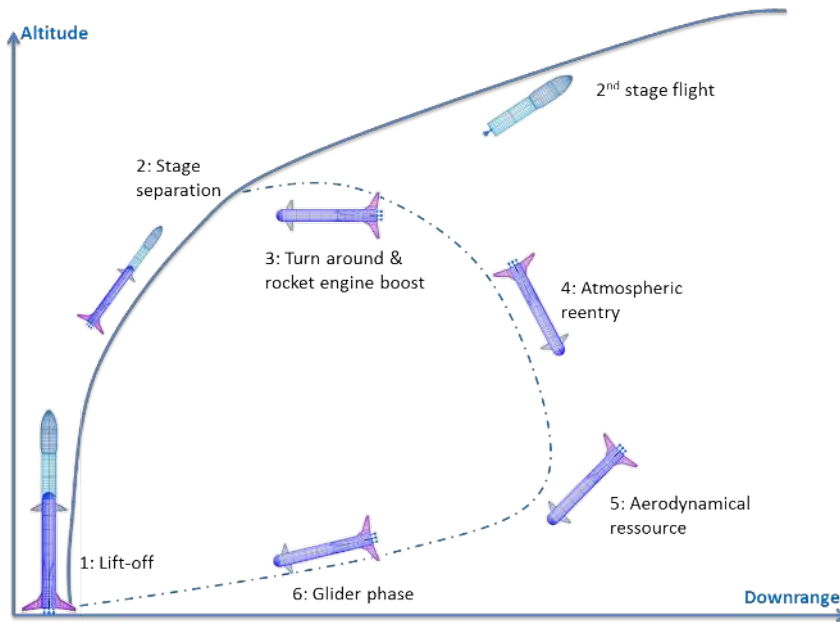


Fig. 7 Steps of the glider mode recovery

B. Multi-objective and Multidisciplinary Design Optimization

1. Single-level, multi-objective MDO formulation

In order to design a reusable launch vehicle for the two missions described in the previous Section, a multi-objective MDO problem is set. The reusable and expendable launch vehicles share the same first and second stages, the differences between the two is the absence of the reusability kit for the expendable mode, the presence of solid rocket boosters for the expendable strategy and the possibility to use all the propellants for the payload injection. The optimization objective for the reusable vehicle is to minimize its Gross Lift-Off Weight whereas for the expendable launcher, the optimization objective is to maximize the mass of the injected payload (Figure 9). Due to the fact that the two missions share the same first and second stages (except for the reusability kit) these two stages are sized with respect to the mission that leads to the maximal trajectory loads. Therefore a MDO coupling exists between the two missions and consequently design problems (Figure 8).

The propulsion and the geometry and structural sizing of the main core disciplines are commune to the two configurations. A specific module for the structural sizing of the reusability kit is present in the reusable configuration. A module called "maximal trajectory load" is in charge to compare the loads undertaken by the two vehicles during the trajectory and to take the maximum value in order to size the vehicle with respect to the most constrained one. In practice, in order to avoid loops between the MDA of the two missions, the coupling variables between the two configurations are removed and controlled at the system-level by the optimizer along with the design variables. The single-level, multi-objective decoupled MDO problem formulation is given by (Figure 9):

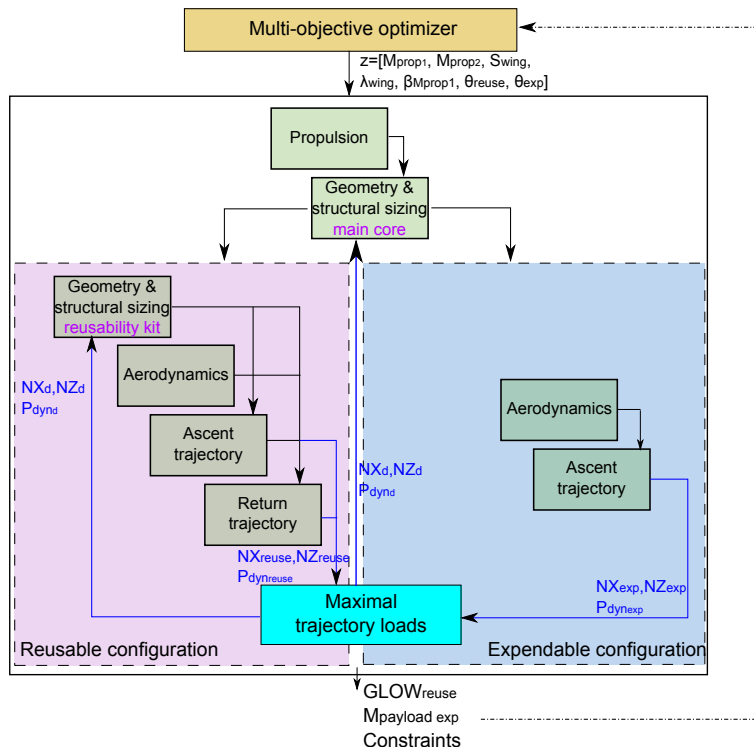


Fig. 8 Single-level coupled MDO formulation for multi-mission launch vehicle design

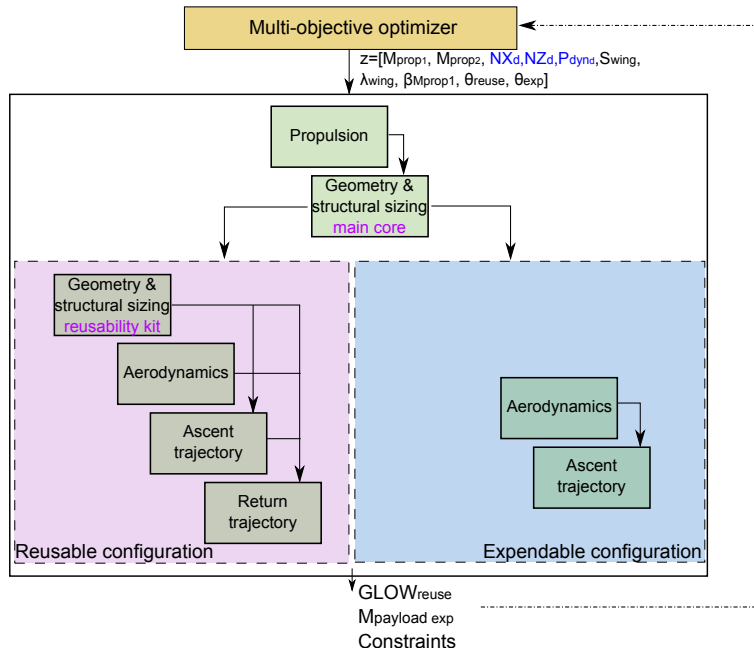


Fig. 9 Single-level decoupled MDO formulation for multi-mission launch vehicle design

$$\begin{aligned}
\min \quad & \mathbf{f}(\mathbf{z}) = [GLOW_{reuse}(\mathbf{z}), -M_{payload_{exp}}(\mathbf{z})] & (13) \\
\text{w.r.t.} \quad & \mathbf{z} = [M_{prop1}, M_{prop2}, S_{wing}, \lambda_{wing}, \\
& \beta_{M_{prop1}}, \boldsymbol{\theta}_{reuse}, \boldsymbol{\theta}_{exp}, NX_d, NZ_d, P_{dyn_d}] \\
\text{s.t.} \quad & |h_{apogee_{exp}}(\mathbf{z}) - h_{apogee_{exp}}^t| \leq \epsilon_{ha} & (14) \\
& |h_{perigee_{exp}}(\mathbf{z}) - h_{perigee_{exp}}^t| \leq \epsilon_{hp} & (15) \\
& NX_{\max_{exp}}(\mathbf{z}) \leq NX_{\max_{exp}}^t & (16) \\
& P_{dyn_{\max_{exp}}}(\mathbf{z}) \leq P_{dyn_{\max_{exp}}}^t & (17) \\
& Flux_{\max_{exp}}(\mathbf{z}) \leq Flux_{\max_{exp}}^t & (18) \\
& \alpha_{\max_{exp}}(\mathbf{z}) \leq \alpha_{\max_{exp}}^t & (19) \\
& |h_{apogee_{reuse}}(\mathbf{z}) - h_{apogee_{reuse}}^t| \leq \epsilon_{ha} & (20) \\
& |h_{perigee_{reuse}}(\mathbf{z}) - h_{perigee_{reuse}}^t| \leq \epsilon_{hp} & (21) \\
& NX_{\max_{reuse}}(\mathbf{z}) \leq NX_{\max_{reuse}}^t & (22) \\
& NZ_{\max_{reuse}}(\mathbf{z}) \leq NZ_{\max_{reuse}}^t & (23) \\
& P_{dyn_{\max_{reuse}}}(\mathbf{z}) \leq P_{dyn_{\max_{reuse}}}^t & (24) \\
& Flux_{\max_{reuse}}(\mathbf{z}) \leq Flux_{\max_{reuse}}^t & (25) \\
& d_{\text{site}_{\max_{reuse}}}(\mathbf{z}) \leq d_{\text{site}_{\max_{reuse}}}^t & (26) \\
& \alpha_{\max_{reuse}}(\mathbf{z}) \leq \alpha_{\max_{reuse}}^t & (27) \\
& \mathbf{z}_{\min} \leq \mathbf{z} \leq \mathbf{z}_{\max} & (28)
\end{aligned}$$

with $\mathbf{z} \in \mathbb{R}^{38}$ the design variable vector. $\boldsymbol{\theta}_{reuse}$ and $\boldsymbol{\theta}_{exp}$ are the control vector for the trajectory including pitch, angle of attack and yaw controls during the different phases of the trajectories (ascent and return for the reusable mission). S_{wing} and λ_{wing} are the surface area and the aspect ratio of the main wings for the reusable launcher, while $\beta_{M_{prop1}}$ is the allocation of the propellant mass for the first stage M_{prop1} that is not used for the ascent trajectory and dedicated to the rocket engine boost to inverse horizontal velocity during the return to launch site phase. Several inequality constraints are considered in this multi-objective problem, including the altitudes of apogee and perigee for the two missions ($h_{apogee}, h_{perigee}$), the maximal acceptable loads (axial (NX), transverse (NZ), dynamic pressure (Pdyn) and flux). Moreover, a constraint representing the distance to the landing site (both in altitude and range) is added in the optimization problem for the return trajectory. All the constraints are specified with respect to a maximal exceedance threshold. $M_{prop1}, M_{prop2}, NX_d, NZ_d$ and P_{dyn_d} are common to the two vehicles. The loads are considered for the sizing of the stages (see Section III.C.1 for more details).

2. Multi-level and multi-objective MDO formulation

Due to the design space dimension ($\mathbf{z} \in \mathbb{R}^{38}$), the non-convexity and the proportion of the feasible region over the entire design space for such multi-objective problems, it is complicated to solve with a single-level (see Section IV.B for the application of the single-level MDO formulation). Indeed, the injection into orbit (altitudes of apogee and perigee) and the return trajectory constraints (final distance to site, admissible loads, *etc.*) are difficult to satisfy. In order to efficiently solve this design problem, a multi-level MDO process is proposed (Figure 10). At the system-level, a Bayesian optimizer controls the decision variables that are shared between the two configurations, whereas, at the subsystem-level, two MDO problems are solved, one for the expendable configuration and one for the reusable launcher in order to facilitate the satisfaction of the constraints of the system-level. The two subsystem optimizers control the design variables specific to the considered configuration. The multi-level approach enables to distribute the problem complexity over different dedicated subproblem optimizations.

The proposed multi-level formulation is given by:

This formulation decomposes the global problem into three optimization problems that are easier to solve. At the system-level, the propellant masses for the two launchers are controlled as well as the sizing loads (axial, transverse and

System-level:

$$\min \quad \mathbf{f}(\mathbf{z}) = [GLOW_{reuse}(\mathbf{z}^*), -M_{payload_{exp}}(\mathbf{z}^*)] \quad (29)$$

$$\text{w.r.t.} \quad \mathbf{z}^* = [M_{prop1}, M_{prop2}, NX_d, NZ_d, P_{dyn_d}]$$

$$\text{s.t.} \quad J_{consexp}(\mathbf{z}^*) \leq \epsilon_{exp} \quad (30)$$

$$J_{consreuse}(\mathbf{z}^*) \leq \epsilon_{reuse} \quad (31)$$

$$\mathbf{z}_{min}^* \leq \mathbf{z}^* \leq \mathbf{z}_{max}^* \quad (32)$$

Subsystem-level:

$$\min \quad -M_{payload_{exp}}(\mathbf{z}, \mathbf{z}^*)$$

$$\text{w.r.t.} \quad \mathbf{z} = [\delta M_{prop1}, \delta M_{prop2}, \theta_{exp}]$$

$$\text{s.t.} \quad |h_{apogee_{exp}}(\mathbf{z}, \mathbf{z}^*) - h_{apogee_{exp}}^t| \leq \epsilon_{ha}$$

$$|h_{perigee_{exp}}(\mathbf{z}, \mathbf{z}^*) - h_{perigee_{exp}}^t| \leq \epsilon_{hp}$$

$$NX_{max_{exp}}(\mathbf{z}, \mathbf{z}^*) \leq NX_{max_{exp}}^t$$

$$P_{dyn_{max_{exp}}}(\mathbf{z}, \mathbf{z}^*) \leq P_{dyn_{max_{exp}}}^t$$

$$Flux_{max_{exp}}(\mathbf{z}, \mathbf{z}^*) \leq Flux_{max_{exp}}^t$$

$$\alpha_{max_{exp}}(\mathbf{z}) \leq \alpha_{max_{exp}}^t$$

$$\mathbf{z}_{min} \leq \mathbf{z} \leq \mathbf{z}_{max}$$

$$\min \quad GLOW_{reuse}(\mathbf{z}, \mathbf{z}^*)$$

$$\text{w.r.t.} \quad \mathbf{z} = [\delta M_{prop1}, \delta M_{prop2}, S_{wing}, \lambda_{wing}, \beta_{M_{prop1}}, \theta_{reuse}]$$

$$\text{s.t.} \quad |h_{apogee_{exp}}(\mathbf{z}, \mathbf{z}^*) - h_{apogee_{exp}}^t| \leq \epsilon_{ha}$$

$$|h_{apogee_{reuse}}(\mathbf{z}, \mathbf{z}^*) - h_{apogee_{reuse}}^t| \leq \epsilon_{ha}$$

$$|h_{perigee_{reuse}}(\mathbf{z}, \mathbf{z}^*) - h_{perigee_{reuse}}^t| \leq \epsilon_{hp}$$

$$NX_{max_{reuse}}(\mathbf{z}, \mathbf{z}^*) \leq NX_{max_{reuse}}^t$$

$$NZ_{max_{reuse}}(\mathbf{z}, \mathbf{z}^*) \leq NZ_{max_{reuse}}^t$$

$$P_{dyn_{max_{reuse}}}(\mathbf{z}, \mathbf{z}^*) \leq P_{dyn_{max_{reuse}}}^t$$

$$Flux_{max_{reuse}}(\mathbf{z}, \mathbf{z}^*) \leq Flux_{max_{reuse}}^t$$

$$d_{sitemax_{reuse}}(\mathbf{z}, \mathbf{z}^*) \leq d_{sitemax_{reuse}}^t$$

$$\alpha_{max_{reuse}}(\mathbf{z}) \leq \alpha_{max_{reuse}}^t$$

$$\mathbf{z}_{min} \leq \mathbf{z} \leq \mathbf{z}_{max}$$

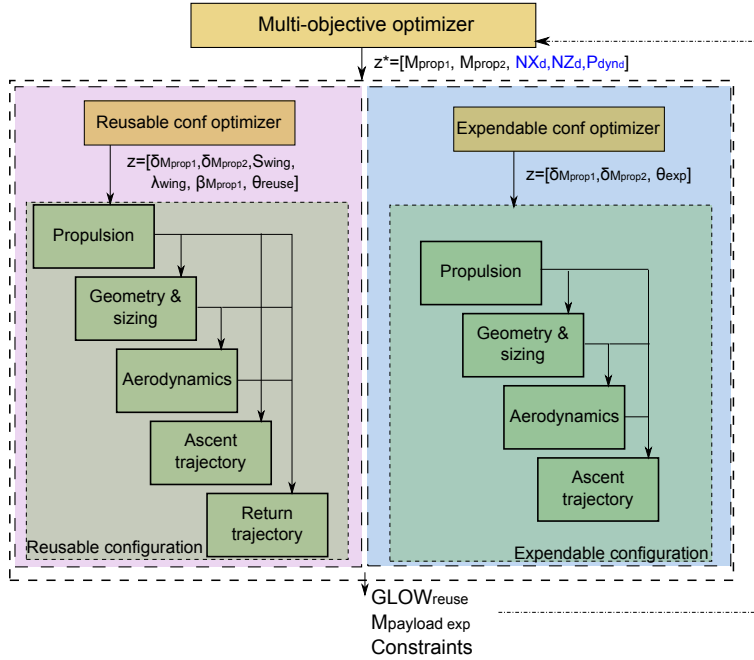


Fig. 10 Multi-level MDO formulation for multi-mission launch vehicle design

dynamic pressure) that must be shared by the two vehicles. The first and second stages are sized with respect to the most important loads of the two missions. Indeed, as the first stage will be used both in expendable and reusable missions, it must be sized for these two types of trajectories. Therefore, by controlling these variables at the system-level, it ensures that the dry mass (*e.g.* tanks, thrust frame, skirt, intertanks, interstage, engines, subsystems) of the first and second stages are identical for the two missions (except for the reusability kit that is removed for the expendable mission). In addition, by controlling the sizing loads at the system-level, it enables to decouple the trajectory and the sizing disciplines and therefore to avoid a Fixed Point Iteration (FPI) between the disciplines to solve the MDA (see next Section). At the system-level, two constraints ($J_{\text{cons}_{exp}}$, $J_{\text{cons}_{reuse}}$) are handled and aggregate the constraints of the subsystem-levels to ensure that the found solutions at the subsystem-level are feasible with respect to the sizing constraints and the couplings between the trajectory and sizing. At the subsystem-level, each optimization problem controls the specific decision variables associated to the corresponding mission. $\delta_{M_{prop1}} \in [0.9, 1.0]$ and $\delta_{M_{prop2}} \in [0.9, 1.0]$ are auxiliary propellant mass variables to provide some flexibility in the optimization problem solving and enabling to not use the entire propellant masses provided by the system-level optimizer.

This multi-level MDO process enables to find candidate solutions satisfying the constraints, however it increases the computational cost at each iteration at the system-level, justifying the need for Bayesian optimization and the use of EGO derivation for multi-objective problems. Indeed, an iteration at the system-level involves two optimization problem solving at the subsystem-level. Even if these subsystem problems are easier to solve than the complete design problem, they still require important computational calculations to find the optimal subsystem design \mathbf{z} given the system-level design variable vector \mathbf{z}^* .

In order to estimate the objective function and the constraints for the two missions, a MultiDisciplinary Analysis is carried out with the discipline models that are described in the next Section.

C. MultiDisciplinary Analysis and disciplinary models

In order to assess the performance of the launch vehicle for the two different missions, a MultiDisciplinary Analysis (MDA) is used. The different disciplinary models (aerodynamics, propulsion, geometry, sizing, trajectory) are coupled through a Fixed Point Iteration method [22] (Figure 11).

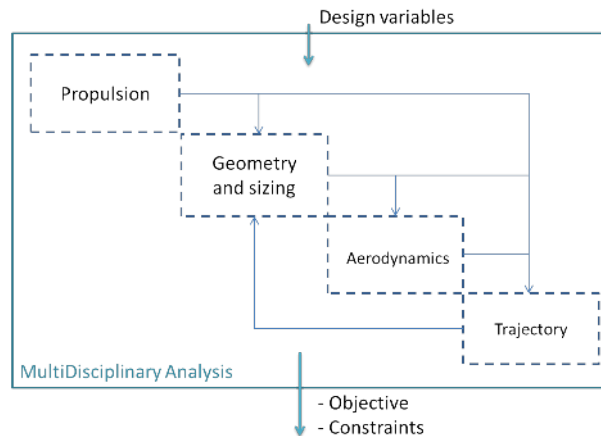


Fig. 11 Illustrative MultiDisciplinary Analysis for launch vehicle design

In the proposed approach, in order to avoid FPI, the feedback coupling variables between the trajectory discipline and the sizing discipline are removed and controlled at the system-level (Figure 11). The feedback couplings correspond to the maximal loads undertaken during the trajectory.

1. Geometry and sizing

In this test case, the geometry and sizing discipline aims at estimating the mass budget of the launcher and its geometry. The 1st and the 2nd stages are LOx/CH4 stages. The dry mass of the stages is the sum of the masses of the structural masses (tanks, skirt, intertanks, thrust frame, interstages), turbopumps, combustion chamber, nozzle,

pressurization system and avionics. The mass and geometry models are derived from the engineering models for the conceptual design of launch vehicle developed by Castellini [23]. For the reusable mission and especially the recovering subsystems (*e.g.* lifting and control surfaces) dedicated engineering models [24] are used to estimate their masses. An interdisciplinary coupling between the sizing discipline and the trajectory is required to model the dependencies between the dry mass and the loads undergone during the flight. Three types of loads are considered for the sizing of the stages: the axial (NX_d), transverse (NZ_d) loads and the dynamic pressure (P_{dyn_d}). These loads are provided by the trajectory simulation to the sizing discipline. These three variables stand for the feedback couplings in the MDA. In order to avoid the use of FPI, these three variables are controlled at the system-level as explained in the previous Section. Taking into account these loads is essential for the sizing of the launch vehicle stage. Indeed, the reuse of a launch vehicle with a glider strategy leads to more important transverse load than conventional expendable launchers. These loads have to be taken into account in order to ensure that the vehicle will be able to perform the mission. The increase in the undertaken loads leads to an increase in the stage dry mass and the structural index (ratio of dry mass to propellant mass). In Figure 12, an illustration of the variation of the structural index with the maximal transversal load encountered during the flight is represented.

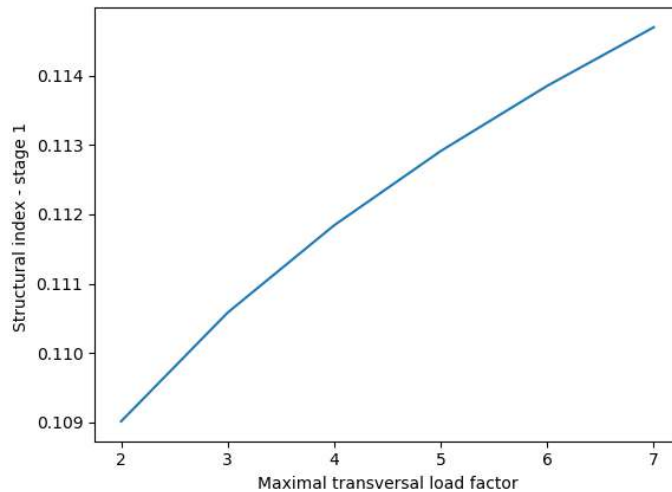


Fig. 12 Variation of the structural index for stage 1 as a function of the maximal transversal load for a given propellant mass

2. Aerodynamics

The aerodynamics discipline consists in computing the aerodynamics coefficients such as the drag and lift coefficients required to compute the aerodynamics loads during the launcher atmospheric flight. The calculations of the drag and lift coefficients are based on the ONERA code MISSILE [25] which relies simplified aerodynamics theory and on an experimental data base to determine the aerodynamics forces and coefficients of complex launcher geometries. Drag and lift coefficients tables as a function of the Mach number and angle of attack are directly given to the trajectory discipline allowing to remove the feedback loop between the trajectory and the aerodynamics. This model is generally sufficient in the early design studies. In order to decrease the computational cost associated with the MISSILE execution, a Neural Network relying on a Multi-Layer Perceptron [26] is used to generate the aerodynamics coefficients from an original database. In Figure 13, a comparison between the Neural Network prediction and the exact MISSILE aerodynamics coefficients is presented for the lift to drag ratio of the reusable configuration.

3. Trajectory

A three dimensional model with rotating round Earth is used. In general, the trajectory discipline consists in solving an optimization problem. The objective of ascent trajectory optimization is to minimize the distance between the

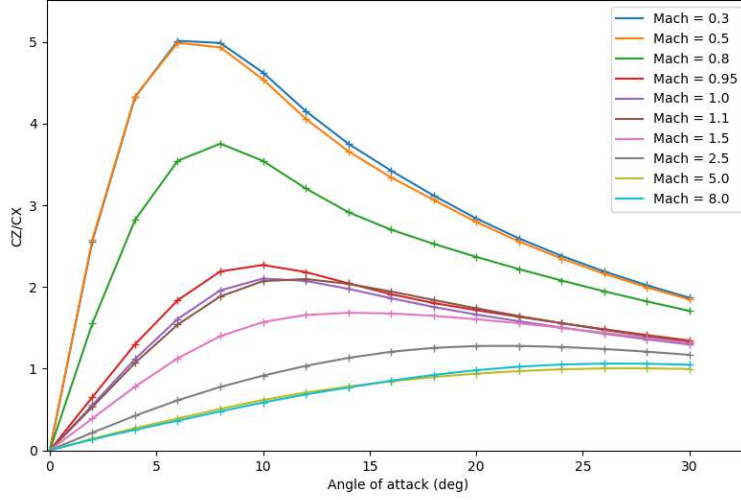


Fig. 13 Comparison of the Neural Network prediction (line) and the exact MISSILE function evaluations (crosses)

injection point and the given target. The ascent trajectory optimization consists of different flight phases: vertical take-off, pitch-over, gravity turn, bilinear tangent law, upper stage's coast and circularization burn for the SSO or injection into the GTO for the expendable mission. The vertical take-off phase enables to leave the launch pad before doing a pitch-over maneuver constituted of a pitch-down phase of duration Δt_{po} with a linearly increasing angle of attack $\gamma - \theta$ (flight path angle, pitch) up to a maximum $\Delta\theta_{po}$. Next, to reach the gravity turn conditions, an exponential decay of $\gamma - \theta$ is performed. For the azimuth, a target inclination law is assumed $\Psi(t) = \sin\left(\frac{\cos(i_T)}{\cos(\delta(t))}\right)^{-1}$ depending on the target inclination and the current latitude (declination). During the gravity turn phase, the yaw is constrained to follow the target inclination law and the pitch is equal to the flight path angle. Once the exoatmospheric conditions have been reached, it is possible to control the pitch with a piece-wise linear interpolation of optimizable nodal value $\theta_{gt}(t)$. For the second stage's flight, a bilinear tangent law [23] is assumed for the pitch angle and its parameters are decision variables. Finally, upper stage's coast and circularization burn are considered since the considered upper stage engine is reignitable. The engine is shut down once the current orbit apogee matches the target apogee. Then, the propellant mass required to circularize the orbit is determined based on the final conditions of velocity and inclination. During the ascent trajectory, several constraints are taken into account such as the maximal axial loads, the maximal dynamic pressure, the maximal flux, the maximal angle of attack during the atmospheric flight and the payload injection conditions on the perigee and apogee. All the previous described trajectory parameters are decision variables noted θ_{exp} and θ_{reuse} for the two missions and are optimized along with launcher sizing variables.

For the return trajectory of the glider strategy, the objective of the optimization problem is to minimize the distance between the actual landing point of the vehicle and the landing site under some maximal admissible load constraints (*e.g.* axial and transverse load factors, dynamic pressure, heat flux). The decision variables for the return phase are β_{Mprop1} (allocation of the propellant mass for the engine boost), a piece-wise linear interpolation of optimizable nodal value $\alpha(t)$ for the guidance during the return phase.

IV. Multi-objective and Multidisciplinary Design Optimization: results and discussions

A. Multi-level, multi-mission MDO approach settings

The proposed multi-level and multi-mission MDO approach requires to optimize the subsystem problems which controls the design variables specific to each mission. In order to optimize the subsystem problems, Covariance Matrix

Adaptation - Evolution Strategy (CMA-ES) optimization algorithm [27] is used due to the non-convex objective and constraint functions and the presence of numerous minima. Indeed, by using a classical SQP algorithm with different initializations, it converges to different local minima leading to a non robust convergence of the lower level (Figure 14). Figure 14 represents a box-plot of 20 repetitions of expendable subsystem problem solving with different initializations of SQP for a given system-level design variable vector. The range of convergence for the objective function varies between 2.75t and 4.25t for the maximal payload mass which is clearly non robust to the initialization.

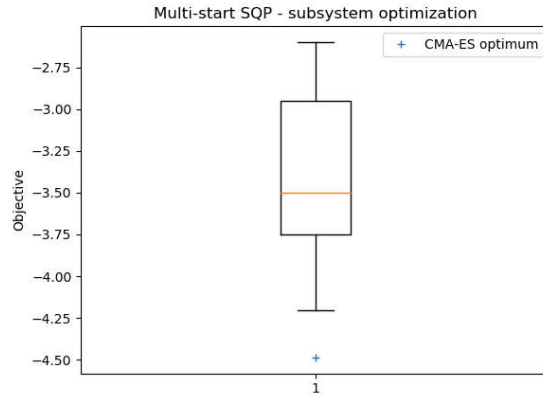


Fig. 14 Boxplot of the optimal objective function for the expendable subproblem optimization using a SQP algorithm - 20 repetitions with random start

On the contrary, by exploring the design space, CMA-ES succeeds to find a better optimum than a gradient-based approach and is more robust with respect to the initialization (that is unknown and set in the middle of the design space). CMA-ES is parallelized in order to evaluate each candidate solution with a multiprocessing approach to increase the computational efficiency of the subsystem problem solving. A penalization approach is carried out to control the optimization problem constraints at the subproblem-level.

Figures 15,16,17 represent a run of CMA-ES algorithm for the expendable optimization subproblem with the convergence plots of the objective function (Figure 15), the design variables (Figure 16) and the associated parallel plot (Figure 17). The convergence of the design variable values and the penalized objective function is reached by CMA-ES at around 250 iterations at the subsystem-level. With the exploration of the design space, CMA-ES algorithm succeeds to find the optimum for the lower-level for both the expendable and the reusable configurations given the system-level design variables.

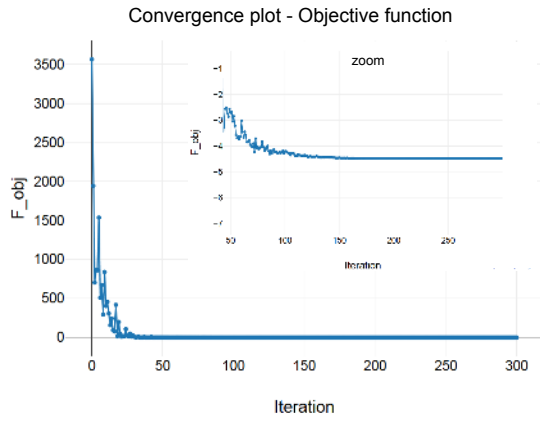


Fig. 15 Convergence curve for the objective function (CMA-ES) - expendable subproblem

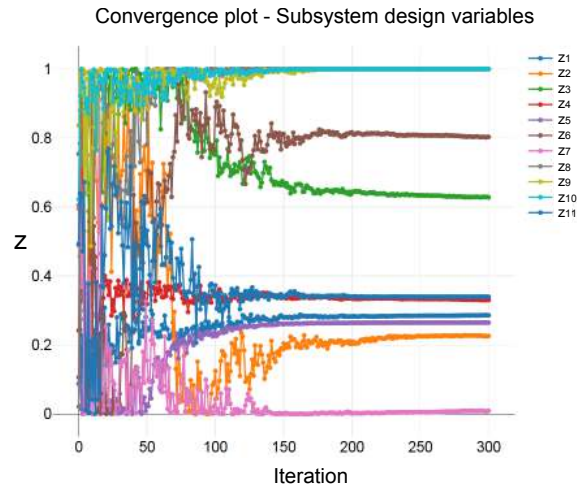


Fig. 16 Convergence curve for the design variables (CMA-ES) - expendable subproblem

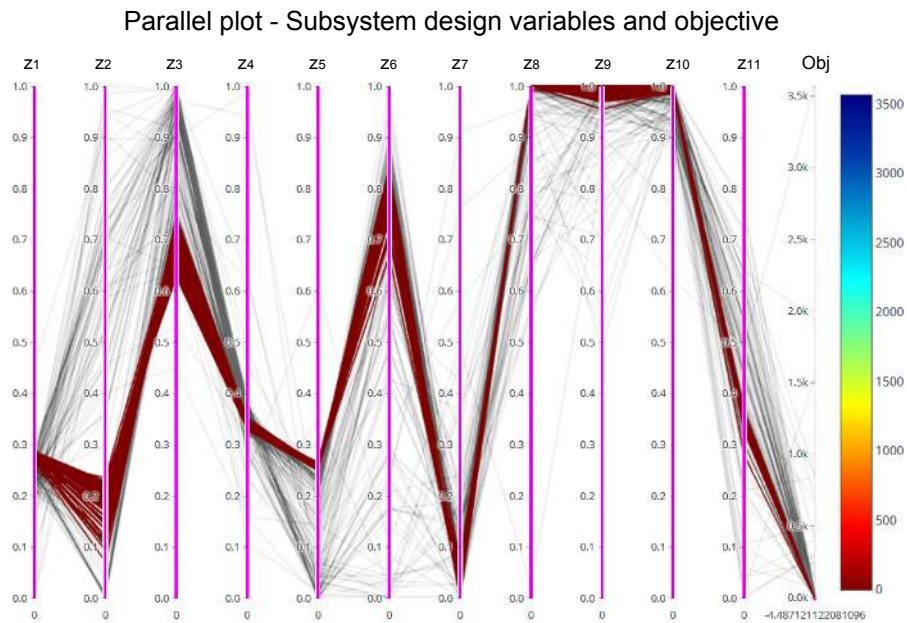


Fig. 17 Parallel plot (CMA-ES) - expendable subproblem

B. Result analysis and discussions

To illustrate the need for a multi-level MDO formulation, the single-level decoupled approach presented in Section III.B.1 is implemented using an OMOPSO algorithm [11]. Particle Swarm Optimization (PSO) [28] is a bio-inspired meta-heuristic mimicking the social behavior of bird flocking or fish schooling. In PSO, the solutions are represented as particles inside a swarm, with a certain position and velocity. The position of a particle at a given generation defines the value of the decision variables at this generation, and it is calculated using a so-called velocity equation. This equation expresses the variation in the position between two generations. It is calculated as a balance between the best position of the particle and the global best position of all the particles. In the adaptation of a PSO to the multi-objective case, the main difference is the notion of the best individual. While in the single objective case only one best value may be obtained, in multi-objective problems, several equally good solutions may be obtained. A natural way to extend this notion is to consider every non-dominated solution over the generations as a leader. In the process of updating the leaders, the particles that are non dominated are kept in the set of leaders. In OMOPSO, the idea to choose for each particle a leader using a binary tournament selection based on the crowding distance. That is for each particle, two solutions from the set of leaders are randomly selected, then the solution with the greatest crowding distance is chosen. For more details on OMOPSO see [11]. GPflow is used to model GP in the proposed multi-objective Bayesian Optimization approach [29].

A population of 20 individuals are considered and OMOPSO runs for 500 iterations corresponding to 10000 evaluations of the subsystem coupled models (for the reusable and expendable configurations). Among all the evaluations, only around 10% of solutions are feasible with respect to the system-level constraints (Figure 18). Moreover, the Pareto front found by OMOPSO with the single-level formulation is far from the Pareto front obtained with the multi-level formulation using EGO (Figure 19).

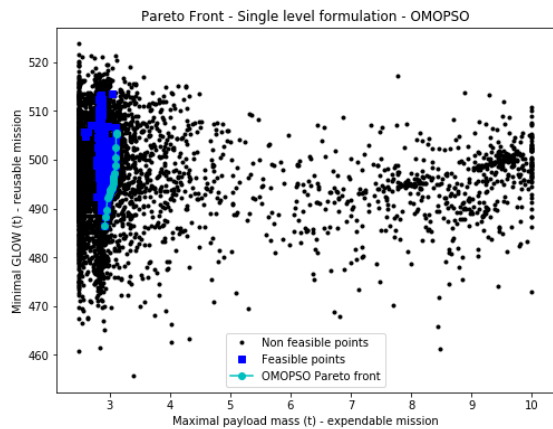


Fig. 18 Feasible and non feasible solutions - single-level formulation and OMOPSO

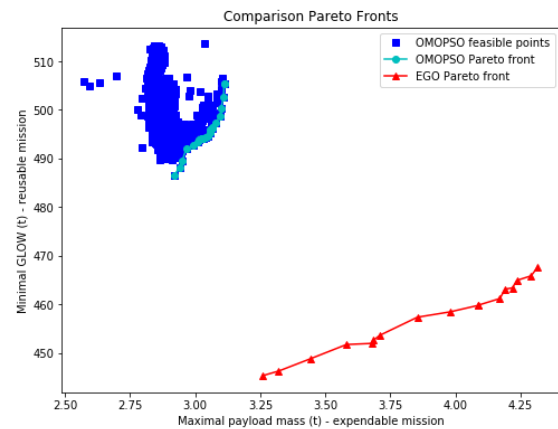


Fig. 19 Comparison of Pareto fronts: in cyan with OMOPSO (single-level formulation), in red with EGO (multi-level formulation)

The proposed multi-level MDO approach starts with an initial DoE size of 20 generated by Latin Hypercube Sampling (in dimension 5) and 200 iterations are run before stopping the algorithm. The proposed multi-level MDO approach is compared to a reference OMOPSO optimization algorithm [11] in terms of quality of the obtained Pareto front (Figures 22 and 24). The OMOPSO has a population of 20 individuals and runs for 11 iterations before stopping (therefore OMOPSO and EGO have the same number of exact MDA evaluations) and be compared to the multi-objective EGO. To evaluate the robustness to the initial DoE (or initial population of OMOPSO), three repetitions of the multi-level MDO approach and OMOPSO optimization are carried out.

From the three repetitions, the proposed multi-level MDO approach enables to find a better Pareto front after 220 evaluations (20 samples for the initial DoE and 200 multi-objective EGO iterations) of the exact MDA compared to OMOPSO. Moreover, the proposed methodology is more robust to the initial DoE samples than the initial population of the OMOPSO. Indeed, over the three repetitions, the multi-level MDO formulation ends at the same value of hypervolume (around 0.287) whereas OMOPSO algorithms present a large variability in the final hypervolume (between 0.275 and 0.287) as illustrated in Figure 24. The resulting Pareto fronts for the three repetitions are represented in Figure 25.

In order to further analyze the results, the initial Pareto front determined from the initial DoE of size 20 for the first repetition is represented in Figure 20. The proposed approach succeeded to add design points to improve the hypervolume (Figure 24) and the final Pareto front (Figure 21). The final Pareto front of EGO is better than the OMOPSO one. Moreover, for the Pareto front, no OMOPSO point dominates EGO points. From Figure 24, it can be seen that the hypervolume of EGO is always better than of OMOPSO even if the gap tends to decrease with the iteration there is still an important difference at the end of the optimization process.

Figure 26 illustrates the evolution of the EGO Pareto front through the iterations (at iteration 50, 100, 150 and 200). The EGO Pareto front spreads and moves forward to tend to an optimal Pareto front (which is unknown). The hypervolumes increases by adding appropriate design points which improve the Pareto front. In Figure 22, the evolution of the Pareto front for OMOPSO is depicted.

Compared to OMOPSO, multi-level MDO approach is able to provide a better Pareto front with a limited number of exact function evaluations. Moreover, using a multi-level approach allows to overcome the difficulty of constraint satisfaction in reusable launch vehicle design. The final Pareto front provides key elements to make an informed decision in terms of multi-mission launch vehicle design. Indeed, the proposed design process offers a quantification of the repercussions on the reusable GLOW for a SSO mission due to an increase of the payload in expendable mission to GTO orbit.

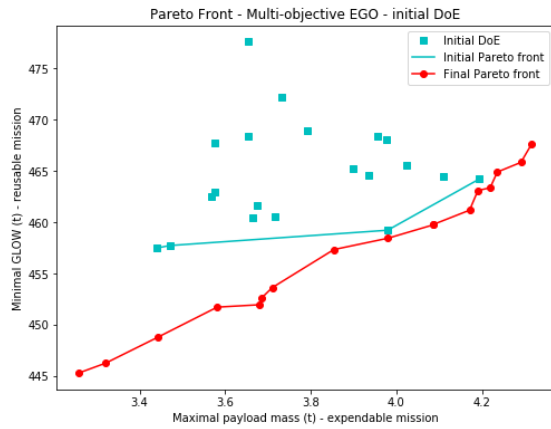


Fig. 20 Initial DoE samples, initial and final Pareto fronts - EGO multi-level formulation

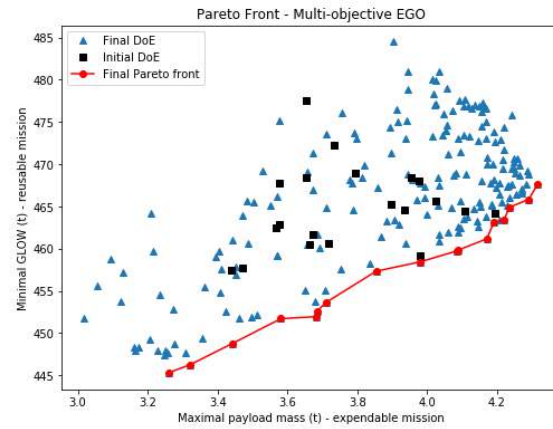


Fig. 21 Initial and final DoE samples, final Pareto front - EGO multi-level formulation

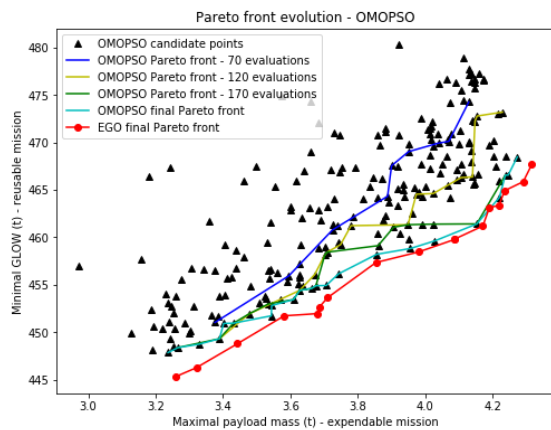


Fig. 22 Pareto front evolution - OMOPSO, multi-level formulation

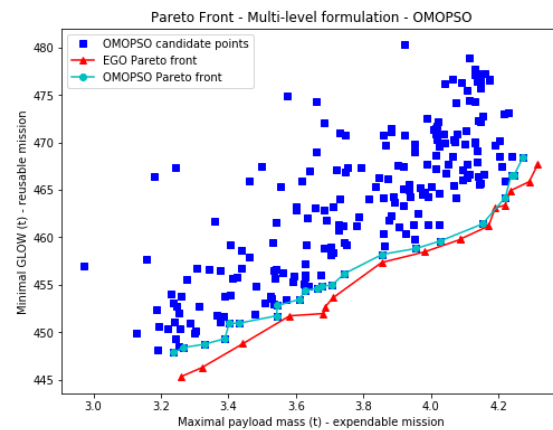


Fig. 23 Comparison final Pareto fronts - EGO and OMOPSO, multi-level formulation

Figure 27 displays the parallel coordinate plots for the multi-objective EGO (corresponding to the added design points in the DoE). The colors of the upper graph correspond to the maximal payload mass objective of the expendable

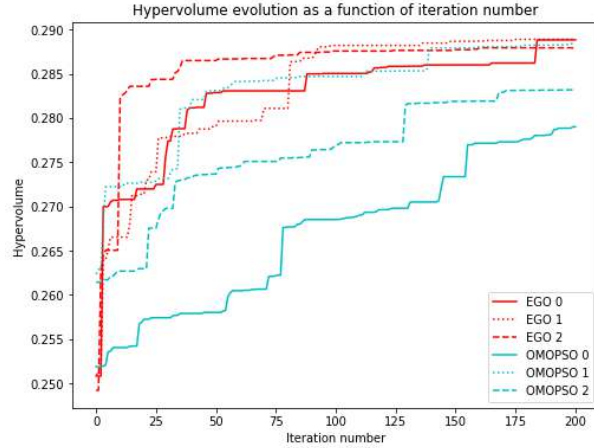


Fig. 24 Comparison hypervolume evolution - multi-objective EGO and OMOPSO

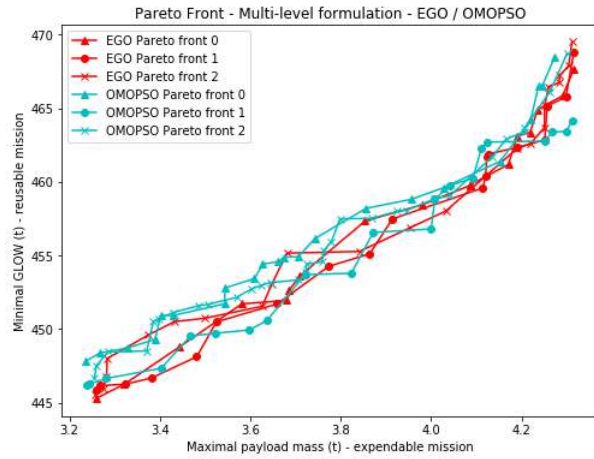


Fig. 25 Comparison Pareto fronts - multi-objective EGO and OMOPSO

mission while the lower graph correspond to the minimal GLOW objective of the reusable mission. The value for the system-level design variables are normalized (propellant masses of stage 1 M_{prop1} , propellant masses of stage 2 M_{prop2} , design maximal axial loads NX_d , design maximal dynamic pressure $P_{dyn,d}$, design maximal transverse loads NZ_d). From these graphs, it can be seen that in order to send the maximal payload mass into orbit for the expendable mission (around 4.3t) it is necessary to load the maximal amount of propellant masses for the stage 1 while to send the minimal payload mass (around 3.0t) it is necessary to take the minimal amount of propellant masses. The same observations for the minimal GLOW of the reusable launcher may be done and extended to the mass of the propellant second stage.

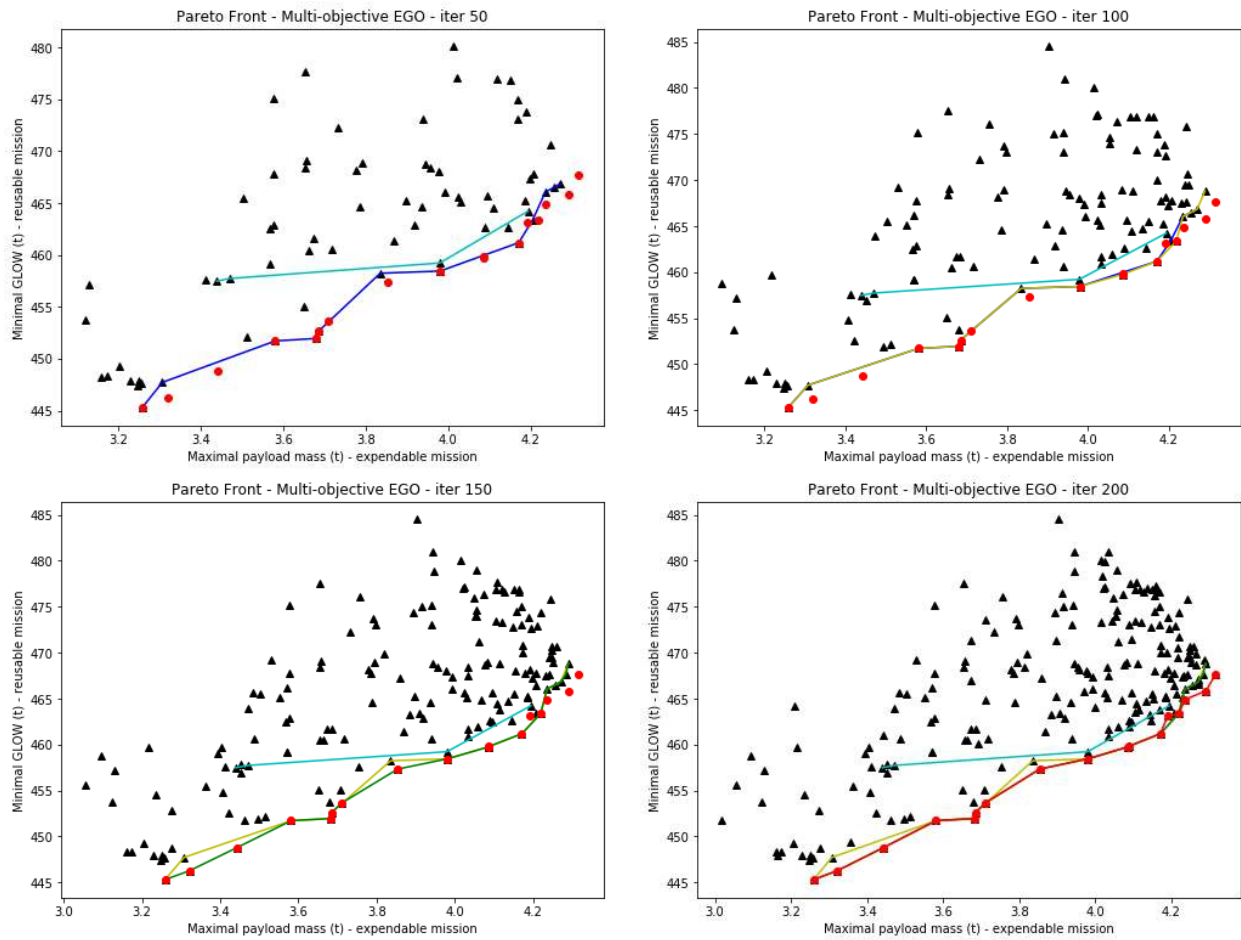


Fig. 26 Evolution of Pareto front for EGO at iteration 50, 100, 150 and 200. Triangles are current DoE, line current Pareto front, circles final Pareto front

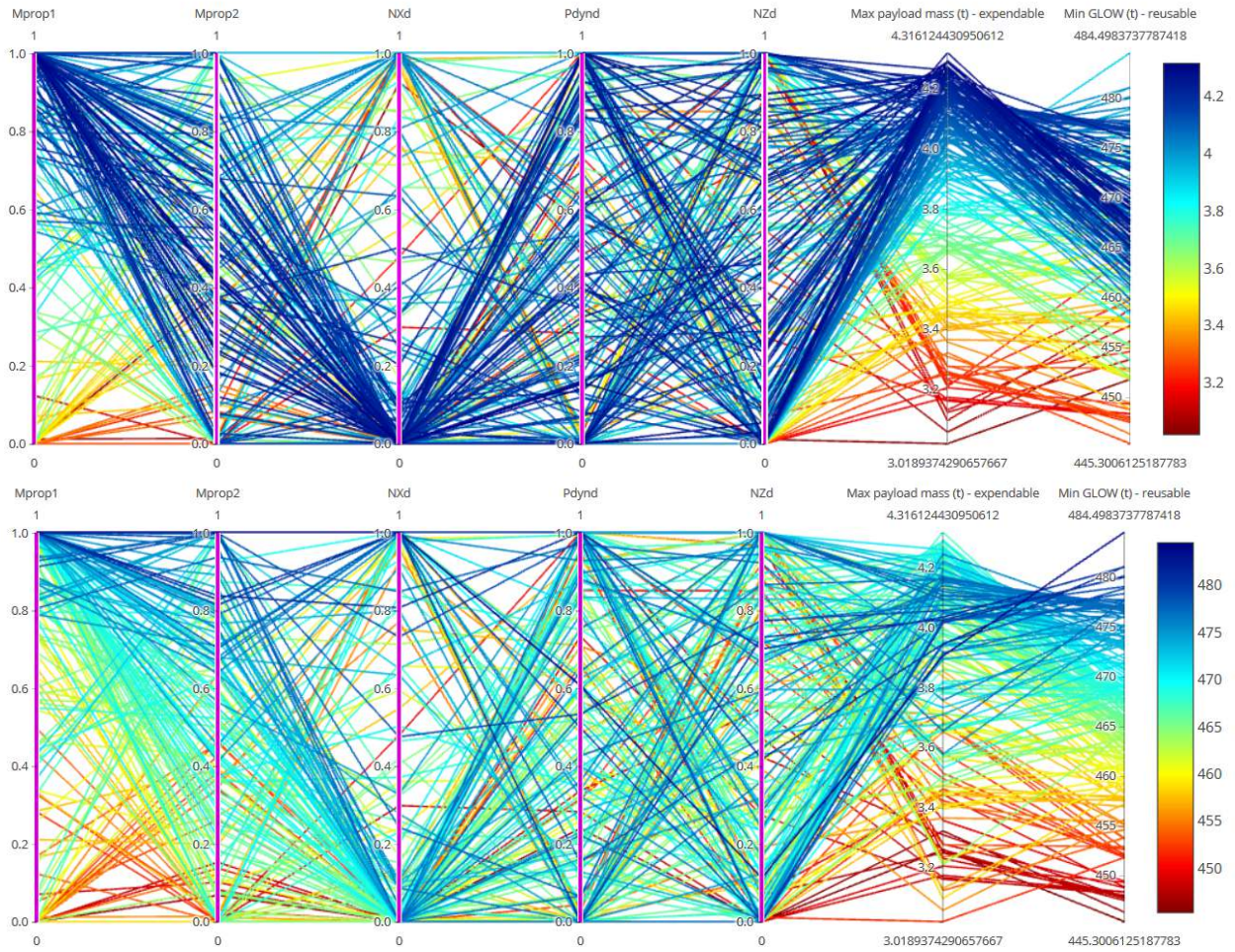


Fig. 27 Parallel plots with respect to system-level design variables - multi-objective EGO (in upper graph, colors correspond to maximal payload mass objective; in lower graph, colors correspond to minimal GLOW objective)

V. Conclusions

In this paper, a multi-level multi-mission decoupled MDO formulation has been proposed in order to design a family of reusable launchers to address two specific missions: a medium range mass payload into SSO with return to the launch site of the first stage using a glider architecture, and a medium range mass payload into GTO with an expendable architecture. The proposed approach relies on multi-objective EGO to efficiently solve the multi-mission problem and to limit the number of calls to the exact MDA thanks to the use of Gaussian Processes. The multi-mission aspect introduces couplings between the different missions in order to ensure the appropriate structural sizing of the launch vehicle. The multi-level MDO formulation allows to find feasible solutions with respect to trajectory and sizing constraints that are difficult to satisfy with using classical single-level approaches. The proposed strategy has been compared on a reusable launch vehicle design problem to a multi-level formulation using OMOPSO optimization algorithm. The proposed approach converges faster to a better Pareto front than OMOPSO. In future works, a more complex multi-mission launch vehicle design problem involving non-stationary discipline models will be investigated using advanced Deep Gaussian Processes [30].

Acknowledgments

This work is funded by the ONERA (Office National d'Etudes et de Recherches Aéropatiales - The French Aerospace Lab) project HERACLES (Hypersonic Efficient and Reusable Aerospace Concepts for Launcher Evolution Strategies, 2017-2020). Support of CNES is gratefully acknowledged. The PhD thesis of A. Hebbal is funded by ONERA and the university of Lille.

References

- [1] Vila, J., and Patureau de Mirand, A., "Weighting options for the next generation of Ariane launchers," *68th International Astronautical Congress*, Adelaide, Australia, 25-29 Sept. 2017.
- [2] Balesdent, M., Brevault, L., Price, P., Defoort, S., Le Riche, R., Kim, N., Haftka, R., and Berend, N., "Advanced space vehicle design taking into account multidisciplinary couplings and mixed epistemic / aleatory uncertainties," *Space Engineering: Modeling and Optimization with Case Studies*, editors: G. Fasano and J. Pinter, Vol. 2, 2016, pp. 1–48.
- [3] Castellini, F., and Lavagna, M., "Comparative analysis of global techniques for performance and design optimization of launchers," *Journal of Spacecraft and Rockets*, Vol. 49, No. 2, 2012, pp. 274–285.
- [4] Fazeley, H., Taei, H., Naseh, H., and Mirshams, M., "A multi-objective, multidisciplinary design optimization methodology for the conceptual design of a spacecraft bi-propellant propulsion system," *Structural and Multidisciplinary Optimization*, Vol. 53, No. 1, 2016, pp. 145–160.
- [5] Fujikawa, T., Tsuchiya, T., and Tomioka, S., "Multi-objective, multidisciplinary design optimization of TSTO space planes with RBCC engines," *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2015, p. 0650.
- [6] Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T., "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II," *International Conference on Parallel Problem Solving From Nature*, Springer, 2000, pp. 849–858.
- [7] Coello Coello, C., and Lechuga, M., "MOPSO: a proposal for multiple objective particle swarm optimization," *Proceedings of the 2002 Congress on Evolutionary Computation*, 2002. CEC-02., pp. 1051–1056.
- [8] Knowles, J. D., and Corne, D. W., "Approximating the nondominated front using the Pareto archived evolution strategy," *Evolutionary computation*, Vol. 8, No. 2, 2000, pp. 149–172.
- [9] T., F., T. T., and S. T., "Multi-Objective, High-Accuracy Trajectory Optimization Using SQP and Sampling on the Manifold," *The 2013 Asia-Pacific International Symposium on Aerospace Technology*, Takamatsu, Japan, 2013.
- [10] Kosugi, Y., Oyama, A., Fujii, K., and Kanazaki, M., "Multidisciplinary and multi-objective design exploration methodology for conceptual design of a hybrid rocket," *Infotech@ Aerospace 2011*, 2011, p. 1634.
- [11] Sierra, M. R., and Coello, C. A. C., "Improving PSO-based multi-objective optimization using crowding, mutation and epsilon-dominance," *International Conference on Evolutionary Multi-Criterion Optimization*, Springer, 2005, pp. 505–519.
- [12] Beume, N., Naujoks, B., and Emmerich, M., "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, Vol. 181, No. 3, 2007, pp. 1653–1669.

- [13] Svenson, J. D., and Santner, T. J., "Multiobjective optimization of expensive black-box functions via expected maximin improvement," *The Ohio State University, Columbus, Ohio*, Vol. 32, 2010.
- [14] Emmerich, M. T., Giannakoglou, K. C., and Naujoks, B., "Single- and multiobjective evolutionary optimization assisted by Gaussian random field metamodelling," *IEEE Transactions on Evolutionary Computation*, Vol. 10, No. 4, 2006, pp. 421–439.
- [15] Nebro, A. J., Durillo, J. J., Garcia-Nieto, J., Coello, C. C., Luna, F., and Alba, E., "Smpso: A new pso-based metaheuristic for multi-objective optimization," *Computational intelligence in multi-criteria decision-making, 2009. mcdm'09. iee symposium on*, IEEE, 2009, pp. 66–73.
- [16] Jones, D. R., Schonlau, M., and Welch, W. J., "Efficient global optimization of expensive black-box functions," *Journal of Global optimization*, Vol. 13, No. 4, 1998, pp. 455–492.
- [17] Rasmussen, C. E., "Gaussian processes in machine learning," *Advanced lectures on machine learning*, Springer, 2004, pp. 63–71.
- [18] Zhang, Q., Liu, W., Tsang, E., and Virginas, B., "Expensive multiobjective optimization by MOEA/D with Gaussian process model," *IEEE Transactions on Evolutionary Computation*, Vol. 14, No. 3, 2010, pp. 456–474.
- [19] Schonlau, M., Welch, W. J., and Jones, D. R., "Global versus local search in constrained optimization of computer models," *Lecture Notes-Monograph Series*, 1998, pp. 11–25.
- [20] Audet, C., Denni, J., Moore, D., Booker, A., and Frank, P., "A surrogate-model-based method for constrained optimization," *8th Symposium on Multidisciplinary Analysis and Optimization*, 2000, p. 4891.
- [21] Iannetti, A., Girard, N., Tchou-kien, D., Bonhomme, C., Ravier, N., and Edeline, E., "PROMETHEUS, A LOX/LCH4 REUSABLE ROCKET ENGINE," 2017.
- [22] Ortega, J. M., "Stability of difference equations and convergence of iterative processes," *SIAM Journal on Numerical Analysis*, Vol. 10, No. 2, 1973, pp. 268–282. Doi: 10.1137/0710026.
- [23] Castellini, F., "Multidisciplinary design optimization for expendable launch vehicles," Ph.D. thesis, Politecnico de Milano, 2012.
- [24] Rohrschneider, R., and Olds, J., "A comparison of modern and historic mass estimating relationships on a two stage to orbit launch vehicle," *AIAA Space 2001 Conference and Exposition*, 2001, p. 4542.
- [25] Denis, P., "ONERA's aerodynamic prediction code- MISSILE," *RTO/AGARD, Symposium on Missile Aerodynamics, Sorrento, Italy, May 11-14, 1998, ONERA, TP*, 1998.
- [26] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al., "Scikit-learn: Machine learning in Python," *Journal of machine learning research*, Vol. 12, No. Oct, 2011, pp. 2825–2830.
- [27] Hansen, N., and Ostermeier, A., "Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation," *Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*, IEEE, 1996, pp. 312–317.
- [28] Eberhart, R., and Kennedy, J., "A new optimizer using particle swarm theory," *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*, IEEE, 1995, pp. 39–43.
- [29] Matthews, D. G., Alexander, G., Van Der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrà, P., Ghahramani, Z., and Hensman, J., "GPflow: A Gaussian process library using TensorFlow," *The Journal of Machine Learning Research*, Vol. 18, No. 1, 2017, pp. 1299–1304.
- [30] Damianou, A., and Lawrence, N., "Deep gaussian processes," *Artificial Intelligence and Statistics*, 2013, pp. 207–215.