MODELING AND PERFORMANCE EVALUATION OF MULTISTAGE LAUNCH VEHICLES THROUGH FIREWORK ALGORITHM

M. Pallone[*], *M. Pontani*[†], *P. Teofilatto*[‡]

Scuola di Ingegneria Aerospaziale, University of Rome "La Sapienza", Rome, Italy

ABSTRACT

Multistage launch vehicles of reduced size, such as Super Strypi or Sword, are currently investigated for the purpose of providing launch opportunities for microsatellites. A simple open-loop guidance strategy is proposed in this research and applied to the Scout rocket, a micro-launcher used in the past. Aerodynamics and propulsion are modeled with high fidelity through interpolation of available data. Unlike the original Scout, the terminal optimal ascent path is determined for the upper stage, using a firework algorithm in conjunction with the Euler-Lagrange equations and the Pontryagin minimum principle. Firework algorithms represent a recently-introduced heuristic technique inspired by the firework explosions in the night sky.

Index Terms— Multistage launch vehicles, trajectory optimization, firework algorithm optimization

1. INTRODUCTION

Currently, microsatellites can be launched according to the time and orbital requirements of a main payload. The limited costs of microsatellites and their capability to be produced and ready for use in short time make them particularly suitable to face an emergency (responsive space), therefore small launch vehicles dedicated to microsatellites would be very useful. On the other hand, in order to reduce the launcher size without increasing too much the launch cost per kg of payload it is necessary to simplify the launch system as much as possible, including the guidance algorithms.

In general, the numerical solution of aerospace trajectory optimization problems is not trivial and has been pursued with different approaches in the past. Indirect methods, such as the gradient-restoration algorithm [1, 2] and the shooting method [3], or direct techniques, such as direct collocation [4, 5], direct transcription [6, 7], and differential inclusion [8, 9], to name a few. However, only a relatively small number of publications are concerned with trajectory optimization of multistage launch vehicles [1, 2, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]. Calise et al. [10] and Gath and Calise [12] proposed and applied a hybrid analytic/numerical approach, based on homotopy and starting with the generation of the optimal solution in a vacuum. They adopted the approximate linear gravity model, and the same did Lu and Pan [13] and Lu et al. [14], who applied a multiple-shooting method to optimizing exoatmospheric trajectories composed of two powered phases separated by a coast arc. Weigel and Well [15] used a similar indirect, multiple-shooting approach to analyze and optimize the ascent trajectories of two launch vehicles with splash-down constraints. Miele [2] developed and applied the indirect multiple-subarc gradient restoration algorithm to optimizing the two-dimensional ascending trajectory of a three-stage rocket in the presence of dynamical and control constraints. The previously cited works [2, 10, 12, 13, 14, 15] make use of indirect algorithms and require a considerable deal of effort for deriving the analytical conditions for optimality and for the subsequent programming and debugging. Furthermore, these methods can suffer from a slow rate of convergence and an excessive dependence on the starting guess. This difficulty has been occasionally circumvented through homotopy [10, 12, 16], but this adds further complexity to the solution process. Other papers deal with direct numerical techniques applied to multistage rocket trajectory optimization. Roh and Kim [17] used a collocation method for optimizing the performance of a four-stage rocket, whose two-dimensional trajectory was assumed to be composed of three thrust phases and a coast arc of specified duration. Collocation was also employed by Jamilnia and Naghash [11], with the additional task of determining the optimal staging, and by Martinon et al. [16], for the purpose of validating the numexrical results attained through indirect shooting. This latter paper refers to the Ariane V launch vehicle and is specifically devoted to investigating singular arcs. Direct methods convert the optimal control problem into a nonlinear programming problem involving a large number of unknown parameters to optimize. The disadvantage is in the need of using specialized nlp solvers, such as SNOPT [20]. Recently, different methodologies appeared that do not belong to the category of either the indirect nor the direct techniques. These techniques are usually referred to as heuristics and have been sporadically applied to optimizing ascent trajectories of multistage rockets. Bayley et al. [18] used a genetic algorithm for the purpose of minimizing the overall rocket mass in the con-

[*]corresponding author: pallone.1420138@studenti.uniroma1.it

[†]mauro.pontani@uniroma1.it

[‡]paolo.teofilatto@uniroma1.it

text of a high fidelity model of the entire vehicle. Three- and four-stage rockets were considered, with ascending trajectories composed (respectively) of three or four powered arcs. Lastly, Qazi et al. [19] integrated neural networks, particle swarm optimization, and sequential quadratic programming for the simultaneous conceptual design and trajectory optimization of a new multistage launch vehicle.

The work that follows is concerned with a novel approach, which is intended to supply a fast performance evaluation for multistage rockets with given characteristics, under some simplifying assumptions. The technique described in this work is applied to a four-stage rocket, whose two-dimensional trajectory is composed of the following thrust phases and coast arcs:

1. first stage propulsion
2. second stage propulsion
3. third stage propulsion
4. coast arc (after the third stage separation)
5. fourth stage thrust phase

In general, the inclusion of a coast arc (between two powered phases) leads to substantial propellant savings and this circumstance justifies the trajectory structure assumed in this research. Usually the coast duration increases as the injection altitude increases, as remarked by Lu et al. [14]. In order to simplify the open loop guidance law employed for the first three stages, the aerodynamic angle of attack is assumed constant for each stage. In the last stage thrust phase the problem of minimizing the propellant is solved defining a Hamiltonian function which is minimized through the Pontryagin minimum principle. The optimization algorithm used in this work is the firework algorithm [21], a novel swarm intelligence algorithm inspired by the explosions of fireworks in a night sky. The concept that underlies this method is relatively simple: a firework explodes in the search space of the unknown parameters, with amplitude and number of sparks determined dynamically. The succeeding iterations preserve the best sparks. The method that is being presented requires a reduced deal of effort in programming, debugging, and testing the algorithmic codes, as existing routines are used, in conjunction with analytical developments and a simple implementation of firework algorithm. Hence, the methodology treated in this paper is intended to: (i) yield a reasonable solution for performance evaluation of multistage rockets and (ii) represent a technique for generating a suitable first-attempt guess trajectory to be employed by more refined algorithms tailored to optimizing the overall trajectory.

2. ROCKET MODELING

The four stage rocket that is being considered is the Scout launcher, which is a rocket designed to place small satellites into low orbit. It has specified structural, propulsive and aerodynamic characteristics and it is represented in Figure 1.

Fig. 1: Scout rocket geometry

To simplify the analysis at hand, the mass distribution of the launch vehicle can be described in terms of masses of subrockets: subrocket 1 is the entire rocket with all the four stages, subrocket 2 is the launch vehicle after the separation of the first stage, subrocket 3 is the launch vehicle after the separation of the first two stages and subrocket 4 is represented by the last stage only. Let $m_0^{(i)}$ denote the initial mass of the subrocket *i*, this mass is composed of a structural mass $m_S^{(i)}$, a propellant mass $m_P^{(i)}$ and a payload mass $m_U^{(i)}$:

$$m_0^{(i)} = m_S^{(i)} + m_P^{(i)} + m_U^{(i)} \tag{1}$$

For the first three subrockets $m_U^{(i)}$ (*i*=1,2,3) coincides with the initial mass of the subsequent subrocket (i.e. $m_U^{(i)} = m_0^{(i+1)}$). With regard to the fourth subrocket, its propellant mass is to be minimized. The mass distribution for the Scout rocket is in the Table 1. For the last stage the initial mass is 514 kg. Minimizing the propellant mass is equivalent to maximizing the payload mass.

Table 1: Mass distribution for the first three subrockets

i	$m_0^{(i)}$	$m_S^{(i)}$	$m_P^{(i)}$	$m_U^{(i)}$
1	21643 kg	1736 kg	12810 kg	6897 kg
2	6897 kg	915 kg	3749 kg	2033 kg
3	2033 kg	346 kg	1173 kg	514 kg

2.1. Propulsive thrust

The propulsive characteristic of the launch vehicle can be described in terms of thrust magnitude $T^{(j)}$ and specific impulse

Table 2: Specific impulse for the four stages

j	1	2	3	4
$Isp^{(j)}$	260 s	288 s	284 s	270 s

$I_{SP}^{(j)}$, with superscript j referring to the stage number. In Table 2 the specific impulses for the four stages are listed.

While the specific impulse is considered time-independent for all the four stages, the thrust is obtained through a linear interpolation of the experimental thrust data which are given at discrete times. Figure 2 portrays the thrust curves for each motor, whose burnout time is $t_{B(j)}$.

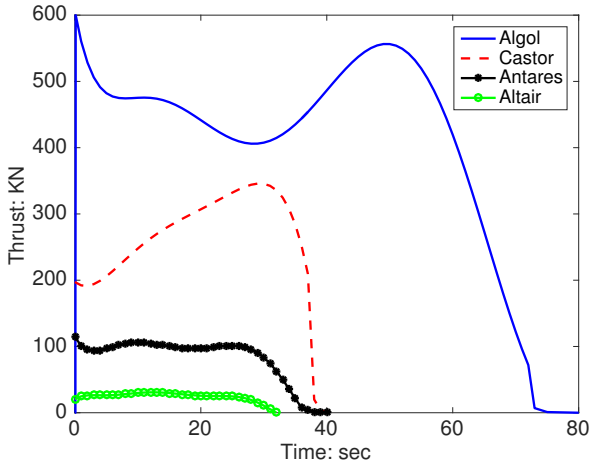


Fig. 2: Thrust curve for the four stages

2.2. Aerodynamics

Aerodynamic modeling is composed of two steps:

1. derivation of C_D and C_L at a relevant number of Mach numbers and angles of attack
2. fourth degree polynomial interpolation

Following the approach presented by Mangiacasale [22], the aerodynamics of the Scout rocket was modeled through the Missile DATCOM software [23] for the first three subrockets. The aerodynamic force is assumed to be composed of two components: lift force *L* and drag force *D*. Given the aerodynamic surface *S*, the atmospheric density ρ, the speed relative to the Earth atmosphere *v*, and the lift and drag coefficients, C_L and C_D respectively, the two components are:

$$L = \frac{1}{2} C_L(\alpha, M) S^{(i)} \rho v^2 \tag{2}$$

$$D = \frac{1}{2} C_D(\alpha, M) S^{(i)} \rho v^2 \tag{3}$$

where the coefficients C_L and C_D depend on the Mach number *M* and the aerodynamic angle of attack α. The aerodynamic surfaces used in the computation are the cross surfaces and are listed in the Table 3. The C_L and C_D coefficient

Table 3: Aerodynamic surfaces for the first three subrockets

j	1	2	3
$S^{(i)}$	1.026 m^2	0.487 m^2	0.458 m^2

are obtained using a four degree polynomial interpolation to speed up the computation time. The polynomial expression (valid both for C_L and C_D) is:

$$\begin{aligned} C_k = {} & C_{00,k} + C_{10,k}\alpha + C_{01,k}M + C_{20,k}\alpha^2 \\ & + C_{11,k}\alpha M + C_{02,k}M^2 + C_{30,k}\alpha^3 \\ & + C_{21,k}\alpha^2 M + C_{12,k}\alpha M^2 + C_{03,k}M^3 \\ & + C_{40,k}\alpha^4 + C_{31,k}\alpha^3 M + C_{22,k}\alpha^2 M^2 \\ & + C_{13,k}\alpha M^3 + C_{04,k}M^4 \end{aligned} \tag{4}$$

with $k = L, D$. To improve the quality of the interpolation, the distinction between subsonic ($M \in [0, 0.8]$), transonic ($M \in [0.8, 1.2]$) and supersonic ($M \in [1.2, 10]$) polynomial coefficients has been done. While the subsonic and the supersonic cases follow the Equation (4), in the transonic case an embedded Matlab interpolation routine was used due to the abrupt behavior of the coefficients as *M* varies around 1. Table 4 reports the coefficients used for each subrocket and an example of the interpolation is in Figure 3. The fourth stage usually flies over 120 km of altitude so the atmosphere is very rarefied and is neglected.

3. ROCKET DYNAMICS

As the rocket performance is being evaluated, the simulations are performed in the most favorable dynamical conditions, i.e. equatorial trajectory and launch toward the East direction. The four-stage rocket is modeled as a point mass, in the context of a two-degree-of-freedom problem.

The rocket motion is described more easily in a rotating (i.e. non inertial) reference frame. The *Earth Centered Earth Fixed* (ECEF) reference frame represents a reference system that rotates with the Earth and has the origin in its center. The ECEF system rotates with a speed $\omega_E = 7.292115 \times 10^{-5}\ s^{-1}$ with respect an inertial Earth-centered frame (ECI), denoted with $(\hat{c}_1, \hat{c}_2, \hat{c}_3)$. Both frames share the same origin *O*. $\hat{c}_1$ is the vernal axis and the vector $\hat{c}_3 = \hat{k}$ is aligned with the planet rotation axis and is positive northward. Therefore $\omega_E \hat{k}$ represents the (vector) rotation rate of the ECEF frame with respect to the ECI frame. The unit vector $\hat{i}$ intersects the Greenwich meridian at all times. The ECEF-frame is associated with $(\hat{i}, \hat{j}, \hat{k})$, which form a right-handed,

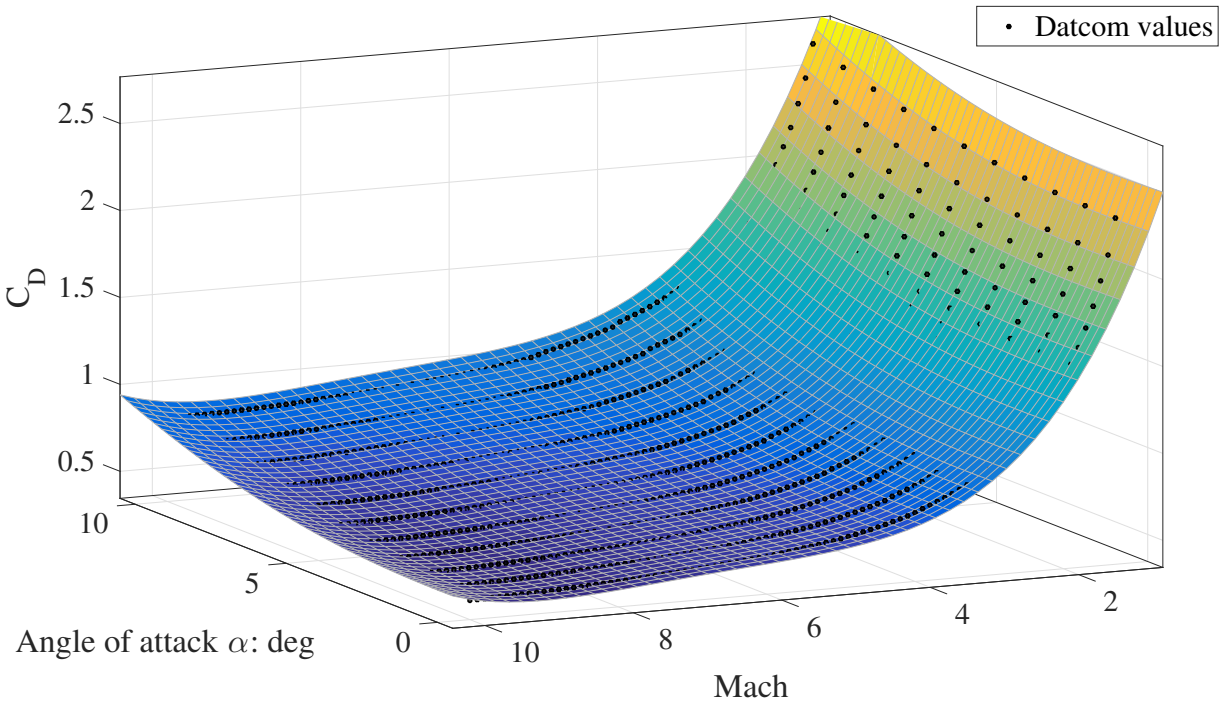


Fig. 3: Fourth degree polynomial interpolation curve for the C_D of the first subrocket in supersonic flight

time-dependent sequence of unit vectors. As the reference Greenwich meridian rotates with rotation rate ω_E, its angular position (with respect to the ECI-frame) is identified by its absolute longitude (usually termed Greenwich sidereal time) $\theta_g(t) = \theta_g(\bar{t}) + \omega_E(t - \bar{t})$, where $\bar{t}$ denotes a generic time instant. The position vector of the orbiting spacecraft in the ECEF-frame is denoted with $\mathbf{r}$, whereas the subscript *I* corresponds to a quantity in the ECI-frame. The inertial velocity is related to the (relative) velocity $\mathbf{v}$ through the following expression:

$$\mathbf{v}_I = \mathbf{v} + \omega_E \times \mathbf{r} \tag{5}$$

This means that, unlike the position vector, the velocity vector in rotating coordinates, $\mathbf{v}$, does not coincide with the inertial velocity vector, $\mathbf{v}_I$. As the entire trajectory lies in the equatorial plane, the flight path angle γ suffices to identify the velocity direction. The instantaneous position is defined through *r* (=$|\mathbf{r}|$) and the geographical longitude ξ. From inspection of Figure 4, it is apparent that

$$\mathbf{v} = v[\sin\gamma, \cos\gamma][\hat{r}, \hat{E}]^T \tag{6}$$

$$\mathbf{r} = r[\cos\xi, \sin\xi][\hat{r}, \hat{E}]^T \tag{7}$$

The overall aerodynamic force $\mathbf{A}$ is conveniently written in the $(\hat{n}, \hat{v}, \hat{h})$ frame (with $\hat{v}$ aligned with $\mathbf{v}$) as the sum of the lift and drag forces:

$$\mathbf{A} = L\hat{n} - D\hat{v} \tag{8}$$

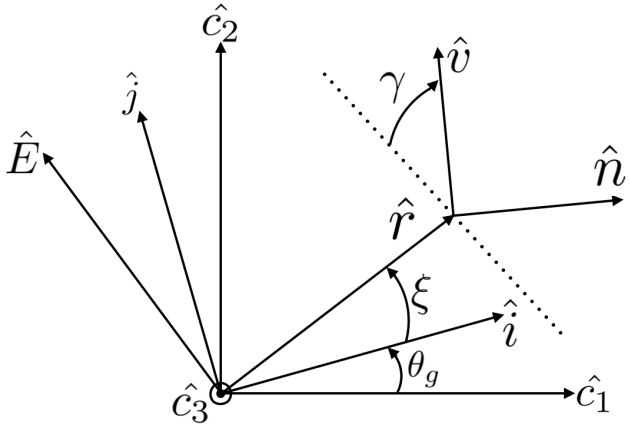


Fig. 4: $(\hat{r}, \hat{E})$ frame, $(\hat{n}, \hat{v})$ frame and related angles

3.1. Equations of motion

The equations of motion that govern the two dimensional rocket dynamics can be conveniently written in terms of its radius *r*, flight-path angle γ, velocity *v* and mass *m*. These variables refer to the relative motion in an Earth-centered rotating frame (ECEF). They form the state vector $\mathbf{x}_R$ ($[r\ \gamma\ v\ m]^T$) of the launch vehicle (in rotating coordinates). Omitting the superscript *i*, for each subrocket the equations of motion are:

Table 4: Polynomial coefficient for aerodynamic interpolation

	subsonic						supersonic					
	subrocket 1		subrocket 2		subrocket 3		subrocket 1		subrocket 2		subrocket 3	
	C_D	C_L	C_D	C_L	C_D	C_L	C_D	C_L	C_D	C_L	C_D	C_L
C_{00}	0.213	0.141	0.160	0.004	0.113	0.002	6.024	0.098	4.284	-0.515	4.814	-0.381
C_{10}	-0.306	5.638	-0.008	1.714	0.015	1.826	0.810	22.290	1.382	9.774	0.862	4.495
C_{01}	0.017	-1.857	0.961	-0.041	1.055	-0.020	-4.449	-0.202	-2.479	0.608	-2.855	0.466
C_{20}	7.398	6.636	1.959	9.338	1.908	3.333	-5.188	-98.280	-9.404	-86.950	-5.682	-39.830
C_{11}	2.781	10.440	0.092	0.294	-0.114	-0.237	-0.250	-7.475	-0.942	-4.251	-0.583	-1.072
C_{02}	-0.073	6.886	-3.484	0.120	-3.828	0.061	1.481	0.114	0.815	-0.260	0.974	-0.203
C_{30}	5.448	-6.733	7.732	-2.092	2.473	2.403	56.290	422.40	55.090	443.40	31.140	201.0
C_{21}	-4.116	-6.103	-0.330	-3.803	0.370	-2.110	5.577	41.750	3.890	34.230	2.599	15.590
C_{12}	-6.230	-26.20	-0.074	0.373	0.327	1.569	-0.031	1.059	0.201	0.705	0.132	0.044
C_{03}	-0.361	-9.520	3.804	-0.132	4.25	-0.070	-0.227	-0.024	-0.120	0.048	-0.155	0.037
C_{40}	-9.063	-32.270	-2.443	-7.244	-1.494	-15.90	-50.640	-656.60	-49.710	-765.50	-36.330	-353.70
C_{31}	-1.953	14.090	3.628	6.947	2.080	2.065	-3.315	-51.920	-3.605	-47.130	-1.794	-20.810
C_{22}	5.965	2.232	-0.517	5.463	-1.142	3.364	-0.608	-3.569	-0.323	-2.867	-0.263	-1.344
C_{13}	3.866	19.890	-0.008	-1.335	-0.229	-2.122	0.009	-0.043	-0.014	-0.029	-0.010	0.012
C_{04}	0.769	4.373	-0.492	0.048	-0.638	0.027	0.013	0.002	0.007	-0.003	0.009	-0.002

$$\dot{r} = v \sin\gamma \tag{9}$$

$$\dot{\gamma} = \frac{T}{mv}\sin\alpha_T + \left(\frac{v}{r} - \frac{\mu_E}{r^2 v}\right)\cos\gamma + \frac{L}{mv} + 2\omega_E + \frac{\omega_E^2 r}{v}\cos\gamma \tag{10}$$

$$\dot{v} = \frac{T}{mv}\cos\alpha_T - \frac{\mu_E}{r^2}\sin\gamma - \frac{D}{m} + \omega_E^2 r \sin\gamma \tag{11}$$

$$\dot{m} = -\frac{T}{I_{sp} g_0} \tag{12}$$

where α_T refers to the thrust angle, μ_E (=398600.4 km^3/sec^2) is the Earth gravitational parameter. As the thrust vector is assumed to be coplanar with the position vector $\mathbf{r}$ and the velocity vector $\mathbf{v}$, the angle α_T suffices to define its direction, which is taken clockwise from $\mathbf{v}$. With the exception of m, the state $\mathbf{x}_R$ is continuos across stage separations, which occur at time t_{b1} (first stage separation), t_{b2} (second stage separation) and t_{b3} (third stage separation). The initial condition for Equations (9) - (12) are listed in the Equation (13)-(14).

$$r(0) = R_E \;\; \gamma(0) = 86\ deg \tag{13}$$

$$v(0) = 0.001\ km/sec \;\; m(0) = m_0^{(1)} \tag{14}$$

where R_E (=6378.136 km) is the Earth radius. The fourth stage trajectory is assumed to be composed of two phases: a coast arc and a thrust phase. During the coast arc, the true anomaly variation Δf suffices to describe the rocket dynamics. In fact, if t_{CO} represents the ignition time of the fourth stage, then $f_4 \equiv f(t_{CO}) = f(t_{b3}) + \Delta f$. The orbital elements a and e do not vary during the coast arc. Hence, they can be computed at separation of the third stage through the following steps:

1. derivation of the inertial state variables (r_I, γ_I, v_I) from the relative state variables (r, γ, v)
2. derivation of the orbital elements (a, e, f) from the inertial state variables (r_I, γ_I, v_I)

The velocity vectors $\mathbf{v}_I$ and $\mathbf{v}$ have the following expression in the rotating frame $(\hat{r}, \hat{E}, \hat{N})$:

$$\mathbf{v}_I = [\sin\gamma_I, \cos\gamma_I][\hat{r}, \hat{E}]^T \tag{15}$$

and

$$\mathbf{v} = [\sin\gamma, \cos\gamma][\hat{r}, \hat{E}]^T \tag{16}$$

Due to this and the fact that $\omega_E \times \mathbf{r} = \omega_E r \hat{E}$, Equation (5) yields to two simple relations:

$$v_I = \sqrt{v^2 + (\omega_E r)^2 + 2v\omega_E r \cos\gamma} \tag{17}$$

$$\gamma_I = \arcsin\frac{v\sin\gamma}{v_I} \tag{18}$$

With regard to step (2), the in plane orbital elements (a, e, f) can be easily calculated from r, v_I and γ_I ([24]). In fact, the conservation of energy yields a:

$$a = \frac{\mu_E r}{2\mu_E - r v_I^2} \tag{19}$$

Then using the definition of the magnitude of the angular momentum in terms of orbital elements $h = \sqrt{\mu_E a(1-e^2)}$ and noticing that $h = r v_I \cos\gamma_I$, the eccentricity can be expressed as:

$$e = \sqrt{1 - \frac{r v_I \cos\gamma_I}{\mu_E a}} \tag{20}$$

The true anomaly f can be obtained using the polar equation of the ellipse,

$$r = \frac{a(1-e^2)}{1+e\cos f} \rightarrow \cos f = \frac{a(1-e^2)-r}{re} \tag{21}$$

in conjunction with the radial component of velocity:

$$\sqrt{\frac{\mu_E}{a(1-e^2)}}e\sin f = v\sin\gamma \rightarrow \sin f = \frac{v\sin\gamma}{e}\sqrt{\frac{a(1-e^2)}{\mu_E}} \tag{22}$$

So the inertial radius, velocity and the flight path angle at t_{CO} are given by:

$$r(t_{CO}) = \frac{a_3(1-e_3^2)}{1+e_3\cos f_4} \tag{23}$$

$$v_I(t_{CO}) = \sqrt{\frac{\mu_E}{a_3(1-e_3^2)}}\sqrt{1+e_3^2+2e_3\cos f_4} \tag{24}$$

$$\gamma_I(t_{CO}) = \arctan\frac{e_3\sin f_4}{1+e_3\cos f_4} \tag{25}$$

where a_3 and e_3 are the semimajor axis and eccentricity at t_{B3}. During the propulsion phase, the fourth stage motion can be described through the use of the following equations that regard r, v_I and γ_I, using the initial conditions reported in (23)-(25).

$$\dot{r_I} = v_I\sin\gamma_I \tag{26}$$

$$\dot{\gamma_I} = \frac{T}{m}\frac{\sin\alpha_T}{v} + \frac{\mu_E}{r_I^2 v_I}\cos\gamma_I + \frac{L}{mv_I} \tag{27}$$

$$\dot{v_I} = \frac{T}{m}\frac{\cos\alpha_T}{v_I} - \frac{\mu_E}{r_I^2}\sin\gamma_I - \frac{D}{m} \tag{28}$$

$$\dot{m} = -\frac{T}{I_{sp}g_0} \tag{29}$$

4. PERFORMANCE EVALUATION

As remarked previously, the entire trajectory is assumed to be planar. Regarding the control law for all the subrockets, this work employs two distinct approaches for determining the control law of the first three stages and that of the last stage.

4.1. Formulation of the problem

The desired orbit is assumed to be a circular orbit of radius R_d, therefore

$$r_{I,d}(t_f) = R_d \;\; v_{I,d}(t_f) = \sqrt{\frac{\mu_E}{R_d}} \;\; \gamma_{I,d}(t_f) = 0 \tag{30}$$

The objective is minimizing fuel consumption for the upper stage while injecting the spacecraft into the desired orbit. Hence the objective function for the entire rocket optimization is:

$$J = t_f - t_{CO} \tag{31}$$

The determination of the trajectory of the first three stages is based on maintaining constant the angle of attack during the flight, except for the first 5 seconds of flight of the first subrocket when the thrust direction is radial. The optimal control strategy for the last stage is instead obtained through minimizing the Hamiltonian function.

4.2. Method of solution

As remarked before, the angle of attack of the first three stages is constant and the thrust angle is assumed to be aligned with the rocket longitudinal axis, thus:

$$\alpha_{T,i} = \alpha_i \tag{32}$$

To obtain the fourth stage optimal control law, the following optimization problem is defined: find the optimal $\alpha_T(t)$ and the optimal true anomaly f_4 such that J is minimized. The ignition time t_{CO} is computed through the Kepler's law,

$$\begin{aligned} t_{CO} = t_{b3} + \sqrt{\frac{a_3^3}{\mu_E}}[&E(t_{CO}) - E(t_{b3}) - e_3\sin E(t_{CO}) \\ &+ e_3\sin E(t_{B3})] \end{aligned} \tag{33}$$

where $E(t_{b3})$ and $E(t_{CO})$ are the eccentric anomalies associated with $f(t_{b3})$ and $f(t_{CO})$. Letting $\mathbf{x} = [x_1, x_2, x_3]^T = [r, v_I, \gamma_I]^T$, to obtain necessary conditions for an optimal solution, a Hamiltonian H and a function of the boundary condition Φ are introduced as

$$\begin{aligned} H \equiv \lambda_1 x_2\sin x_3 + \lambda_2\left[\frac{T}{m}\frac{\cos\alpha_T}{x_2} - \frac{\mu_E}{x_1^2}\sin x_3\right] \\ \lambda_3\left[\frac{T}{m}\frac{\sin\alpha_T}{x_2} + \left(\frac{x_2}{x_1} - \frac{\mu_E}{x_1^2 x_2}\right)\cos x_3\right] \end{aligned} \tag{34}$$

$$\begin{aligned} \Phi \equiv (t_f - t_{CO}) &+ \nu_1\left[x_{10} - \frac{a_3(1-e_3^2)}{1+e_3\cos f_4}\right] \\ &+ \nu_2\left[x_{20} - \sqrt{\frac{\mu_E}{a_2(1-e_3^2)}}\sqrt{1+e_3^2+2e_3\cos f_4}\right] \\ &+ \nu_3\left[x_{30} - \arctan\frac{e_3\sin f_4}{1+e_3\cos f_4}\right] + \nu_4\left[x_{1f} - R_d\right] \\ &+ \nu_5\left[x_{2f} - \sqrt{\frac{\mu_E}{R_d}}\right] + \nu_6 x_{3f} \end{aligned} \tag{35}$$

where $x_{k0} = x_k(t_{CO})$ and $x_{kf} = x_k(t_f)$ (k=1,2,3); $\boldsymbol{\lambda}$ ($\equiv [\lambda_1, \lambda_2, \lambda_3]^T$) and $\boldsymbol{\nu}$ ($\equiv [\nu_1, \nu_2, \nu_3, \nu_4, \nu_5, \nu_6]^T$) represent, respectively, the adjoint variable conjugate to the dynamics Equation (26)-(29) and to the boundary conditions (Equation (30)). The necessary conditions for optimality

(Equation (34)) yield the following adjoint equations for the costate $\boldsymbol{\lambda}$:

$$\dot{\lambda_1} = (x_2\lambda_3\cos x_3)\,\frac{1}{x_1^2} - \left(2\mu_E\lambda_2\sin x_3 + 2\mu_E\lambda_3\frac{\cos x_3}{x_2}\right)\frac{1}{x_1^3} \tag{36}$$

$$\dot{\lambda_2} = -\lambda_1\sin x_3 - \lambda_3\left[\cos x_3\left(\frac{1}{x_1} + \frac{\mu_E}{x_1^2x_2^2}\right) - \frac{T}{mx_2^2}\sin\alpha_T\right] \tag{37}$$

$$\dot{\lambda_3} = -x_2\lambda_1\cos x_3 + \mu_E\lambda_2\frac{\cos x_3}{x_1^2} + \lambda_3\sin x_3\left(\frac{x_2}{x_1} - \frac{\mu_E}{x_1^2x_2}\right) \tag{38}$$

in conjunction with the respective boundary conditions,

$$\lambda_{k0} = -\nu_k \;\; and \;\; \lambda_{kf} = \nu_{k+3} \;\; (k = 1, 2, 3) \tag{39}$$

In the presence of initial conditions depending on the parameter f_4 a pair of additional necessary conditions must hold,

$$\frac{\partial\Phi}{\partial f_4} = 0 \;\text{ and }\; \frac{\partial^2\Phi}{\partial f_4^2} \geq 0 \tag{40}$$

The first equation yields a relation that express λ_{30} as a function of λ_{10}, λ_{20} and f_4,

$$\begin{aligned}\lambda_{30} = \lambda_{20}\frac{\sin f_4\sqrt{1 + 2_3^2 + 2e_3\cos f_4}}{e_3 + \cos f_4} \\ - \lambda_{10}\frac{a_3(1 - e_3^2)}{e_2 + \cos f_4}\frac{1 + e_3^2 + 2e_3\cos f_4}{(1 + e_3\cos f_4)^2}\end{aligned} \tag{41}$$

In addition, the optimal control α_T^* can be expressed as a function of the costates through the Pontryagin minimum principle:

$$\alpha_T^* = \arg\min_{\alpha_T} H \tag{42}$$

implying

$$\sin\alpha_T^* = -\frac{\lambda_3}{x_2}\left[\left(\frac{\lambda_3}{x_2}\right)^2 + \lambda_2^2\right]^{-1/2} \tag{43}$$

and

$$\cos\alpha_T^* = -\lambda_2\left[\left(\frac{\lambda_3}{x_2}\right)^2 + \lambda_2^2\right]^{-1/2} \tag{44}$$

Lastly, as the final time is unspecified, the following transversality condition must hold:

$$H(t_f) + \frac{\partial\Phi}{\partial t_f} = 0 \tag{45}$$

implying

$$\frac{n_0^{(4)}}{1 - n_0^{(4)}(t_f - t_{CO})/(g_0I_{sp}^{(4)})}\sqrt{\left[\frac{\lambda_{3f}}{x_{2f}}\right]^2 - \lambda_{2f}^2} - 1 = 0 \tag{46}$$

where $n_0^{(4)}$ is the initial T/m for the fourth stage. The necessary conditions for optimality allow translating the optimal control problem into a two-point boundary-value problem involving Equation (36)-(46), with unknowns represented by the initial values of $\boldsymbol{\lambda}$, f_4 and t_f. The equality constraints reduce the search space where the solution can be located. However, Pontani et al.[25] demonstrated that for the problem at hand the transversality condition can be neglected by the firework algorithm and transformed into an inequality constraint. Therefore, the optimal control α_T^* can be determined without considering the transversality condition, which is in fact ignorable in this context. Thus, this condition in discarded in order to reduce the equality condition considered by the firework algorithm with the intent of improving its performance.

In short, the following parameter set can be employed in the solution process: $\{\alpha_1, \alpha_2, \alpha_3, \lambda_{10}, \lambda_{20}, \Delta f, t_f\}$. The remaining parameter λ_{30} can be easily obtained by means of Equation (41). Specifically, the technique is based on the following points:

1. given the initial condition ((13)-(14)) and $\{\alpha_1, \alpha_2, \alpha_3\}$, the state Equations (9)-(12) are integrated numerically for each subrocket until the third stage burnout time
2. the coast arc is computed analytically using Equations (23)-(25)
3. for the upper stage the control variable is expressed as a function of the costate through the Equations (43)-(44)
4. the value of λ_{30} is calculated by means of Equation (41), after picking the unknown values of the remaining Lagrange multipliers at the initial time (λ_{10} and λ_{20}), and the true anomaly f_4
5. Equations (26)-(29) are used together with Equations (36)-(38). The respective initial conditions are known once the parameters f_4 (for the state equations) and $\{\lambda_{10}, \lambda_{20}, \lambda_{30}\}$ (for the adjoint equations) are specified
6. the inequality condition in Equation (40) (not expanded for the sake of brevity) and the conditions at injection (Equation (30)) are evaluated.

In summary, the problem reduces to the determination of seven unknown parameters, $\{\alpha_1, \alpha_2, \alpha_3, \lambda_{10}, \lambda_{20}, \Delta f, t_f\}$, that lead the dynamical system to satisfying three conditions (Equation (30)).

4.3. Numerical results

The ascending trajectories are determined by employing canonical units: the distance unit (DU) is the Earth radius, whereas the time unit (TU) is such that $\mu_E = 1$ DU^3/TU^2. Thus DU=6378.165 km and TU=806.8 sec.

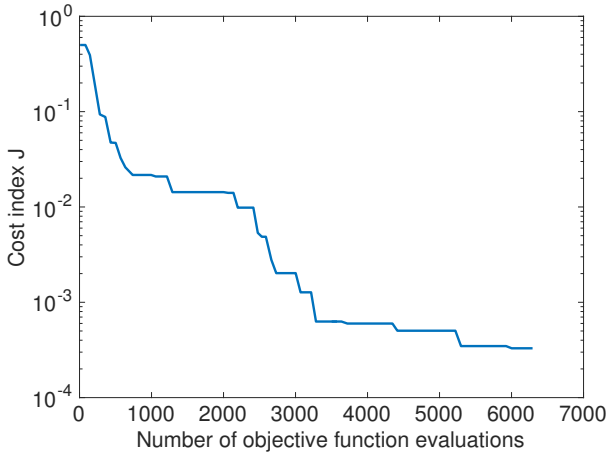


Fig. 5: Objective evolution as a function of the number of objective function evaluations

With regard to the swarming optimizer, it is employed with the following settings: $N_F = 10$ (numbers of fireworks), $N_S^{(max)} = 30$ (maximum number of sparks for each firework) and $N_{evals} = 50000$ (maximum number of objective function evaluations). The optimization was performed with an 2,3 GHz Intel Core i7 with a runtime of 10822.24 sec. Figure 5 shows the objective evolution as a function of the number of evaluations. The computation ended when J reached the value 3.3×10^{-4}.

To limit the dynamic pressure in the first three stages, the optimal values of the unknown angles of attack are sought in the range $0 \leq \alpha_i \leq 10$ deg (i=1,2,3) while for the last stage $-1 \leq \lambda_{k0} \leq 1$ (k=1,2,3), $0 \leq \Delta f \leq \pi$, and 1/TU $\leq t_f \leq$30/TU. It is worth remarking that ignorability of the transversality condition allows defining arbitrarily the search space for the initial value of the Lagrange multipliers. This means that they can be sought in the interval $-1 \leq \lambda_{k0} \leq 1$ by the firework algorithm and only a posteriori their correct values (that also fulfill the transversality condition in Equation (46)) can be recovered. The case test that has been considered has R_d=R_E+300 km. The main optimization results are reported in the Table 5:

Table 5: Optimal set of parameters

α_1	3.65 deg
α_2	8.86 deg
α_3	9.85 deg
λ_{01}	-0.17958
λ_{02}	-0.41587
Δf	15.33 deg
t_f	24.44 sec

So the coast arc duration is Δt_{CO}=295.21 sec. Figure 7 portrays the state components (radius, velocity, flight path angle and mass) for each subrocket obtained with the optimized parameters in Table 5, while Figure 6 shows the optimal thrust pointing angle for the last stage, where a near horizontal burn circularizes the orbit at the desired altitude.

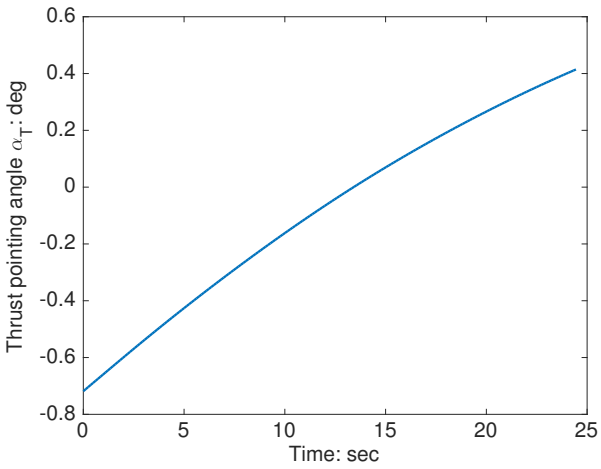
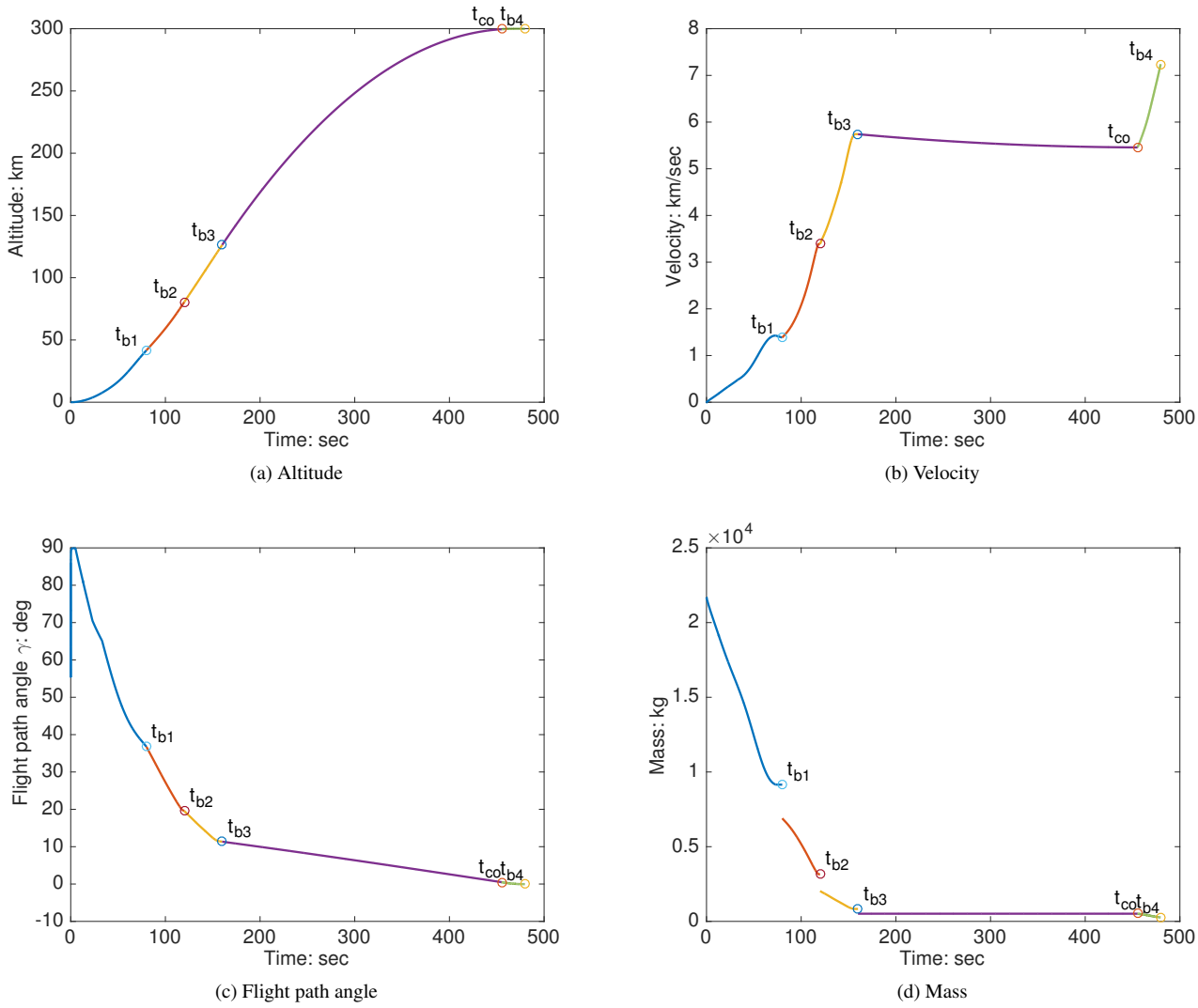


Fig. 6: Control law for the last stage

From inspecting the results summarized in Figure 7 it is apparent that the final conditions at injection are fulfilled in a satisfactory way. The final payload mass obtained through the optimization is 223.87 kg, including 38.33 kg of structural mass. This results is very similar to the one presented in Figure 8, taken from the Scout manual ([26]). Despite its simplicity with respect to alternative existing approaches, the technique applied in this work yields a trajectory that resembles qualitatively the ones reported in [2, 13, 14, 15, 19, 27, 25, 26].

5. CONCLUDING REMARKS

The generation of an optimal trajectory for multistage launch vehicles is a challenging problem, treated with different approaches in the past. This work proposes and successfully applies a simple technique for generating near-optimal two-dimensional ascending trajectories for multistage rockets, for the purpose of performance evaluation. Only existing routines and a simple implementation of firework algorithm are employed, in conjunction with the analytical necessary conditions for optimality, applied to the upper stage trajectory. With regard to the problem at hand, the unknown parameters are (i) the aerodynamic angles of attack of the first three stages, (ii) the coast time interval, (iii) the initial values of the adjoint variables conjugate to the upper stage dynamics, and (iv) the thrust duration of the upper stage. The numerical results unequivocally prove that the methodology at hand is rather robust, effective, and accurate, and definitely allows evaluating the performance attainable from multistage launch



(a) Altitude

(b) Velocity

(c) Flight path angle

(d) Mass

Fig. 7: State variables for all the four stages

vehicles with very accurate aerodynamic and propulsive modeling. The solutions found in this work resemble the optimal trajectories found in the past for alternative launch vehicles through different techniques, and appear suitable also for being employed as guesses for more refined optimization algorithms.

6. APPENDIX: FIREWORK ALGORITHM

The fireworks algorithm is a novel swarm intelligence method for parameter optimization problems [21] . It is a technique inspired by the firework explosions in the night sky. The concept that underlies this methodology is relatively simple: a firework explodes in a point of a n-dimensional space, with an amplitude and number of sparks that are determined dynamically through evaluation of the objective function at that point. The succeeding generation is chosen by including the best sparks (in relation with the objective function), and these become the new fireworks. This process is iterated until the termination criterion is met. For instance, this can be reaching the maximum number of function evaluations or reaching the desired fitness. If χ represents the parameter set, as a starting point N_F fireworks are generated randomly in the feasible space,

$$a \leq \chi \leq b \tag{47}$$

At iteration j, each firework and spark, denoted respectively with the index i and l, is associated with a particular determination of the parameter set, corresponding to a solution of the problem, with a specific value of the objective function,

$$\chi^{(j)}(i) = [\chi_1^{(j)}(i), \ldots, \chi_n^{(j)}(i)], \;\; j = 1, \ldots, N_F \tag{48}$$

$$\chi^{(j)}(l) = [\chi_1^{(j)}(l), \ldots, \chi_n^{(j)}(l)], \;\; l = 1, \ldots, N_{S,tot} \tag{49}$$

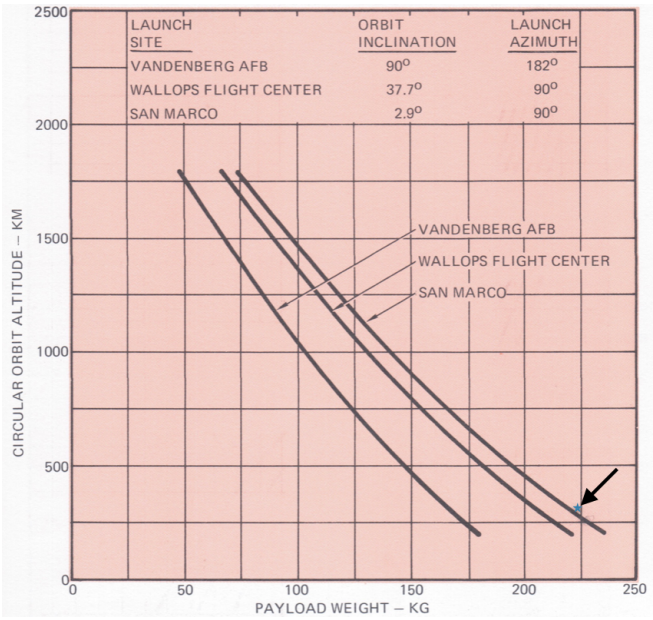


Fig. 8: Circular orbit performances from the Scout manual [26], the $\star$ indicates the result of the optimization

The actual number of sparks, $N_{S,tot}$, varies dynamically iteration after iteration, never exceeding the maximal value $2N_F N_S^{(max)}$, where $N_S^{(max)}$ represents the maximum number of sparks for each firework. In greater detail, the following steps compose the generic iteration j:

1. update the minimal spark amplitude, $A_{min}^{(j)}$

$$A_{min}^{(j)} = A_i - \frac{A_i - A_f}{N_{EV}^{(max)}} \sqrt{N_{EV}^{(j)} \left(2N_{EV}^{(max)} - N_{EV}^{(j)}\right)} \tag{50}$$

where $N_{EV}^{(max)}$ is the maximum number of objective function evaluations, $N_{EV}^{(j)}$ is the number of evaluations performed prior to iteration j, whereas A_i and A_f are two reference minimum amplitudes, with $A_i > A_f$;

2. for i=1, …, N_F: evaluate the objective function associated with firework i, $J^{(j)}(i)$;

3. in relation to the objective function, determine the global best and worst parameter set, $\mathbf{Y}^{(j)}$ and $\mathbf{Z}^{(j)}$, associated with the current fireworks,

$$i_m = \arg\min_i J^{(j)}(i) \;\; i_M = \arg\max_i J^{(j)}(i) \tag{51}$$

which implies

$$\mathbf{Y}^{(j)} = J^{(j)}(i_m) \tag{52}$$

$$\mathbf{Z}^{(j)} = J^{(j)}(i_M) \tag{53}$$

4. for i=1, …, N_F: calculate the number of sparks

$$N_S(i) = \text{round}\left\{ N_S^{(max)} \frac{J^{(j)}(i_M) - J^{(j)}(i) + \epsilon}{\sum_{r=1}^{N_F} [J^{(j)}(i_M) - J^{(j)}(i)] + \epsilon} \right\} \tag{54}$$

where ϵ is a tiny constant, whereas the operator round selects the integer part of the expression in parenthesis; if $N_S(i) = 0$, then $N_S(i)$ is set to 1;

for each component of χ (associated with k), calculate the explosion amplitude

$$A_k(i) = A_k^{(max)} \frac{J^{(j)}(i) - J^{(j)}(i_m) + \epsilon}{\sum_{r=1}^{N_F} [J^{(j)}(i) - J^{(j)}(i_m)] + \epsilon} \tag{55}$$

where $A_k^{(max)}$ is the maximal allowed amplitude, typically the search space amplitude $(b_k - a_k)$;

for each component of $\chi(i)$, generate two random numbers with uniform distribution (with limiting values indicated in parenthesis), $r_1(0,1)$ and $r_2(-1,1)$. Then

$$\begin{cases} \text{if } r_1 \leq 0.5 \rightarrow \chi_k^{(j)}(l) = \chi_k^{(j)}(i) + A_k(i) r_2 \\ \text{if } r_1 > 0.5 \rightarrow \chi_k^{(j)}(l) = \chi_k^{(j)}(i) \end{cases} \tag{56}$$

with l=1, …, $N_S(i)$;

for each component of $\chi(l)$ (l=1, …, $N_S(i)$), generate a random number with Gaussian distribution (zero average value, standard deviation equal to 1), $n_1(0,1)$, and a random number with uniform distribution, $r_3(0,1)$. Then

$$\begin{cases} \text{if } r_3 \leq 0.5 \rightarrow \chi_k^{(j)}(N_s(i) + l) = \chi_k^{(j)}(l) \\ \quad + n_1[Y_k^{(j)} - \chi_k^{(j)}(l)] \\ \text{if } r_3 > 0.5 \rightarrow \chi_k^{(j)}(N_s(i) + l) = \chi_k^{(j)}(l) \end{cases} \tag{57}$$

with l=1, …, $N_S(i)$;

for each component of $\chi(l)$ (l=1, …, $2N_S(i)$),

$$\begin{aligned} &if\ \chi_k^{(j)}(l) < a_k \ or\ \chi_k^{(j)}(l) > b_k \\ &\text{then } \chi_k^{(j)}(l) = a_k + r_4(0,1)(b_k - a_k) \end{aligned} \tag{58}$$

5. the fireworks and the sparks are ordered in relation to their respective objective function value. Then, N_R fireworks for the new generations are selected among the best elements in $\{\chi(i), \chi(l)\}_{i=1,\ldots,N_F; l=1,\ldots,2N_S(i)}$ whereas the remaining $(N_F - N_R)$ fireworks are selected randomly among the remaining elements.

The *explosion operator* associates a reduced number of sparks to the fireworks that have higher values of the objective function to minimize, thus improving local search in the proximity

of the most promising solutions. The *explosion amplitude* depends also on the objective function associated with each firework, and assumes greater values as the objective decreases. However, excessive contraction of the amplitude is avoided through adoption of the minimal amplitude , which is also dynamically updated, without reaching 0. The *displacement operator* creates new sparks according to the amplitude calculated at the previous step. The *Gaussian operator* introduces a displacement with random distribution toward the best position yet located up to the current iteration. The *mapping operator* avoids violation of the parameter bounds . Finally, the *elitism-random selection mechanism* preserves the best sparks and fireworks in the generation to come, while introducing new fireworks through random selection, in order to avoid premature convergence. In the end, the firework algorithm has several features that ensure satisfactory performance in parameter optimization problems, because both local search and global search are effectively performed through combination of the various operators described in this appendix.

7. REFERENCES

[1] A. Miele, "Multiple-subarc gradient-restoration algorithm, part 1: Algorithm structure.," *Journal of Optimization Theory and Applications*, vol. 116, no. 1, pp. 1–17, 2003.

[2] A. Miele, "Multiple-subarc gradient-restoration algorithm, part 1: Application to a multistage launch vehicle design.," *Journal of Optimization Theory and Applications*, vol. 116, no. 1, pp. 19–39, 2003.

[3] E. F. Harrold K. R. Brown and G. W. Johnson, "Rapid optimization of multiple-burn rocket flights.," Tech. Rep. CR-1430, NASA, 1969.

[4] C. R. Hargraves and S. W. Paris, "Direct trajectory optimization using nonlinear programming and collocation.," *Journal of Guidance, Control and Dynamics*, vol. 10, no. 4, pp. 338–342, 1987.

[5] A. L. Herman and B. A. Conway, "Direct optimization using collocation based on high-order gauss-lobatto quadrature rules," *Journal of Guidance, Control and Dynamics*, vol. 19, no. 3, pp. 592–599, 1996.

[6] P. J. Enright and B. A. Conway, "Discrete approximations to optimal trajectories using direct transcription and nonlinear programming," *Journal of Guidance, Control and Dynamics*, vol. 15, no. 4, pp. 994–1002, 1992.

[7] J. T. Betts, "Optimal interplanetary orbit transfers by direct transcription," *Journal of the Astronautical Sciences*, vol. 42, no. 3, pp. 480–487, 1994.

[8] H. Seywald, "Trajectory optimization based on differential inclusion," *Journal of Guidance, Control and Dynamics*, vol. 17, no. 3, pp. 480–487, 1994.

[9] V. Coverstone-Carroll and S. N. Williams, "Optimal low thrust trajectories using differential inclusion concepts," *Journal of the Astronautical Sciences*, vol. 42, no. 4, pp. 379–393, 1994.

[10] D. H. Young A. J. Calise, S. Tandon and S. Kim, "Further improvements to a hybrid method for launch vehicle ascent trajectory optimization," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Denver, CO, 2000.

[11] R. Jamilnia and A. Naghash, "Simultaneous optimization of staging and trajectory of launch vehicles using two different approaches," *Aerospace Science and Technology*, 2011.

[12] P. F. Gath and A. J. Calise, "Optimization of launch vehicle ascent trajectories with path constraints and coast arcs," *Journal of Guidance, Control and Dynamics*, vol. 24, no. 2, pp. 296–304, 2001.

[13] P. Lu and B. Pan, "Trajectory optimization and guidance for an advanced launch system," in *30th Aerospace Sciences Meeting and Exhibit*, Reno,NV, 1992.

[14] G. A. Dukeman P. Lu, B. J. Griffin and F. R. Chavez, "Rapid optimal multiburn ascent planning and guidance," *Journal of Guidance, Control and Dynamics*, vol. 31, no. 6, pp. 45–52, 2008.

[15] N. Weigel and K. H. Well, "Dual payload ascent trajectory optimization with a splash-down constraint," *Journal of Guidance, Control and Dynamics*, vol. 23, no. 1, pp. 45–52, 2000.

[16] J. Laurent-Varin P. Martinon, F. Bonnans and E. Trélat, "Numerical study of optimal trajectories with singular arcs for an ariane 5 launcher," *Journal of Guidance, Control and Dynamics*, vol. 32, no. 1, pp. 51–55, 2009.

[17] W. Roh and Y. Kim, "Trajectory optimization for a multi-stage launch vehicle using time finite element and direct collocation methods," *Engineering Optimization*, vol. 34, no. 1, pp. 15–32, 2002.

[18] J. E. Burkhalter D. J. Bayley, R. J. Hartfield Jr. and R. M. Jenkins, "Design optimization of a space launch vehicle using a genetic algorithm," *Journal of Spacecraft and Rockets*, vol. 45, no. 4, pp. 733–740, 2008.

[19] H. Linshu M. D. Qazi and T. Elhabian, "Rapid trajectory optimization using computational intelligence for guidance and conceptual design of multistage launch vehicles," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, San Francisco, CA, 2005.

[20] W. Murray P. E. Gill and M. A. Saunders, *User's Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming*, Stanford University, Stanford, CA.

[21] Y. Tan, *Fireworks Algorithm*, Springer, 2015.

[22] L. Mangiacasale, *Meccanica del volo atmosferico. Terne di riferimento, equazioni di moto, linearizzazione, stabilità*, Ingegneria 2000, 2008.

[23] W. B. Blake, *Missile DATCOM User's Manual – 1997 Fortran 90 Revision*, United States Air Force, 1998.

[24] J. E. Prussing and B. A. Conway, *Orbital Mechanics*, Oxford University Press, New York, NY, 1993.

[25] M. Pontani and P. Teofilatto, "Simple method for performance evaluation of multistage rockets," *Acta Astronautica*, vol. 94, pp. 434–445, 2014.

[26] Vought Corporation, *SCOUT Planning Guide*, 10 1976.

[27] Mauro Pontani, "Particle swarm optimization of ascent trajectories of multistage launch vehicles," *Acta Astronautica*, vol. 94, no. 2, pp. 852 – 864, 2014.