CEAS EuroGNC 2019
"Conference on Guidance, Navigation and Control"
3-5 April 2019 @ Politecnico di Milano, Milan, Italy.

CEAS-GNC-2019-044

Fast Trajectory Optimization Using Sequential Convex Method for Guided Missiles

Heekun Roh, Young-Jae Oh, Min-Jea Tahk, and Chang-Hun Lee

Abstract This paper proposes a novel trajectory optimization method for air-launched missiles. The suggested L1-Penalized Sequential Convex Programming (LPSCP) approach reduces the order of magnitude of computation time by 2, compared to the pseudospectral approach. The new approach is directly applicable to offline trajectory planning with convergence less than 0.5 second on Intel i7-6700 cpu. Furthermore, the suggested LPSCP method has the potential to be implemented onboard, which will enable autonomous real-time guidance in the future.

Throughout the paper, a convex approximation method for a generic air-launched missile guidance problem is outlined. The missile model considers thrust cut-off after burn time, which is not commonly considered in the domain of sequential convex methods. After the convexification process, given optimal guidance problem is locally approximated to form subproblems in conic form, then solved iteratively using LPSCP algorithm. The proposed method is applied to series of numerical examples to demonstrate its advantages, compared to classic pseudospectral approach. The simulation results show clear evidence of effectiveness and versatility of LPSCP algorithm on optimal missile guidance problems.

Heekun Roh
Korea Advanced Institute of Science and Technology (KAIST), 34141 Daejeon, Republic of Korea, e-mail: heekunroh@kaist.ac.kr

Young-Jae Oh
Korea Advanced Institute of Science and Technology (KAIST), 34141 Daejeon, Republic of Korea, e-mail: yjoh@fdcl.kaist.ac.kr

Min-Jea Tahk
Korea Advanced Institute of Science and Technology (KAIST), 34141 Daejeon, Republic of Korea, e-mail: mjtahk@kaist.ac.kr

Chang-Hun Lee
Korea Advanced Institute of Science and Technology (KAIST), 34141 Daejeon, Republic of Korea, e-mail: chlee@kaist.ac.kr

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

1 Introduction

Guidance technique for homing missiles gained interests after the age of World War II. The goal of homing missile guidance is to intercept the target by missile. For example, famous Proportional Navigation (PN) guidance law[18], and its variants[8, 16], are often utilized for homing guidance. Although started as a heuristic, emergence of optimal control theory eventually lead to a proof that PN guidance law is actually optimal under certain assumptions.[18] Furthermore, special closed-form guidance laws even with terminal constraints were derived from optimal control theories. [7, 9, 16]

However, in order to derive closed-form guidance laws, it was impossible to consider every operational constraints. For example, even maximum field-of-view constraint, which is crucial for feasible missile operation, cannot be included in the derivation of optimal guidance laws due to its nonlinear nature. [5] Even till modern days, because of limitations on onboard computational capability, missiles are equipped with closed-form guidance laws.

Nowadays, development of high-speed embedded computers adjoined with state-of-art optimization techniques are bringing the new wave of missile guidance. Computational guidance, which emphasizes utilization of numerical techniques in guidance,[13] has obtained certain interests across the literature. By numerically solving the missile guidance problem in optimal control sense, it is possible to incorporate various nonlinear dynamics and constraints in real-time, while maintaining optimality at the same time. Provided that enough computational capability is present, advantages of computational guidance technique over classic guidance laws are evident.

Sequential convex method is one of the high-speed optimization method which exploits well-defined properties of convex optimization.[4, 10, 13] Interests over sequential convex techniques are growing after its successful employment in Space X's Falcon 9 landing guidance.[3] Sequential convex approach is expanding its frontier from landing guidance problem[1, 17] to more general aerospace guidance problems. [2, 11, 12]

In this paper, Sequential Convex Programming (SCP) is implemented for general homing missile guidance problem. It is worth noting that a thrust component is considered, even with cut-off after burn time. Furthermore, time-varying mass, variable air density, and varying aerodynamic coefficients are also incorporated in the dynamic model.

In order to solve highly nonlinear optimal control problem, stability improvement on generic SCP method is necessary. Instability due to subproblem infeasibility is handled using L1 penalty method,[6] which do not interfere with the convexity of original subproblem. We propose L1-Penalized Sequential Convex Programming (LPSCP), which combined L1 penalty method with SCP method.

This paper is organized as follows: chapter 2 introduces a novel LPSCP method, which improved stability and versatility of general sequential convex method. In chapter 3, we define the optimal missile control problem, and transform it to approximate convex subproblem. Numerical simulation is performed in chapter 4 to

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

show the clear computational advantage of LPSCP method over classic pseudospectral optimal control method. Chapter 5 concludes the paper.

2 L1-Penalized Sequential Convex Programming

For a long time, nonlinear parameter optimization problems were impossible to solve efficiently. But in the last few decades, certain series of nonlinear programming problems, namely convex programming problems, proved to be globally optimizable in an efficient manner using interior point algorithm.[4] Convex programming problems are defined with convex objective function, convex inequality constraints, and affine equality constraints. Nowadays, fast and stable convex optimization algorithms are easily accessible via several implementations, from state-of-art commercial solvers to open-source free solvers.[14]

Still, convex problems only cover limited numbers of nonlinear optimization problems. Yet it would be desirable if one can exploit enjoyable properties of convex optimization problems (proved global convergence, with polynomial complexity) while solving general nonlinear problems. Inspired from these ideas, Sequential Convex Programming (SCP) algorithms are developed.[10]

SCP algorithms approximate given nonlinear problems into convex subproblems in the proximity of current solution estimate. The solution of current convex subproblem is utilized to generate an updated solution estimate, which defines a new convex subproblem. Iterations of successive convexification and substitution eventually converges to certain solution estimate, which is a local optimal solution to the original nonlinear problem. Note that SCP algorithm is a heuristic. However, due to fast and stable properties of convex algorithms, SCP heuristics converges to the local optimal solution in a rapid manner.[4]

Let us concentrate on a general nonlinear problem eq. (1) defined with nonlinear objective function $f_0(x)$, nonlinear inequality constraints $f_i(x) \leq 0$, and nonlinear equality constraints $h_i(x) = 0$.

$$\begin{aligned} \text{minimize} \quad & f_0(x) \\ \text{subject to :} \quad & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \end{aligned} \tag{1}$$

Given current solution estimate $x^{(k)}$, we may rearrange, expand, or linearize given nonlinear functions to convexify eq. (1). General convexification process of nonlinear functions are not well established in the literature. Yet, the simplest convexification method is to evaluate the nonconvex part of problem using current solution estimate, which renders every nonlinearity into constants. Let us define the convexified subproblem eq. (2) with convex objective function $\hat{f}_0(x)$, convex inequality constraints $\hat{f}_i(x) \leq 0$, and affine equality constraints $\hat{h}_i(x) = 0$. In order to limit the convex approximation to hold in the proximity of current solution estimate, additional trust-region constraint $x \in \mathscr{T}^{(k)}$ can be added.

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

$$\begin{aligned} \text{minimize} \quad & \hat{f}_0(x) \\ \text{subject to :} \quad & \hat{f}_i(x) \le 0, \quad i = 1, ..., m \\ & \hat{h}_i(x) = 0, \quad i = 1, ..., p \\ & x \in \mathscr{T}^{(k)} \end{aligned} \tag{2}$$

Using the provided convex subproblem eq. (2), SCP algorithm can be generated. SCP algorithm iterates until the solution estimate reaches a stationary point. That is, it terminates when the new solution estimate is close enough to the previous one by ε_{tol}. Pseudocode of generic SCP algorithm is provided in algorithm 1.

Algorithm 1 Sequential Convex Programming

```
Provide an initial guess of solution x^(0)
for k = 0 : k_max do
    set convex subproblem eq. (2) with x^(k)
    solve the subproblem to obtain x^(k+1)
    if |x^(k+1) - x^(k)| ≤ ε_tol then
        return x^(k+1)
    end if
end for
```

Generic SCP algorithm assumes every convex subproblems to be feasible. If not, the algorithm halts. However, this assumption is not always valid. For certain problems, its convex subproblems can be infeasible, regardless of feasibility of original nonlinear problem. Mainly the subproblem infeasibility is due to poor solution estimate provided. If the solution estimate is infeasible to the original nonlinear problem, it is highly possible that the expanded convex subproblem in the neighborhood of the guess can be also infeasible. In order to resolve this issue, a L1 penalized approach can be utilized.

L1 penalty method[6] renders hard constraints in the given problem into soft constraints by adding several nonnegative slack variables, which transforms the given problem to be always strictly feasible. (eq. (3)) Current value of slack variables directly contains the current constraint violation. Augmenting the sum of slack variables on the objective function is equivalent to penalizing L1 norm of total constraint violation, which gives the name L1 penalty method. Since additional slack variables are introduced as a plain linear term, *L1 penalty method always preserves convexity of the original problem.* Thus, L1 penalty method can be successfully integrated with SCP method to ameliorate its stability. The modified SCP method, namely L1-Penalized SCP(LPSCP) is briefly summarized in algorithm 2.

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

$$
\begin{aligned}
\text{minimize} \quad & f_0(x) + \xi\nu \\
& \Big(\nu = \sum_{i=1}^{m}(v_{1i} + v_{2i}) + \sum_{i=1}^{p}(w_{1i} + w_{2i})\Big) \\
\text{subject to :} \quad & f_i(x) + v_{1i} - v_{2i} \le 0, \quad i = 1, \dots, m \\
& v_{1i}, v_{2i} \ge 0, \quad i = 1, \dots, m \\
& h_i(x) + w_{1i} - w_{2i} = 0, \quad i = 1, \dots, p \\
& w_{1i}, w_{2i} \ge 0, \quad i = 1, \dots, p
\end{aligned} \tag{3}
$$

Algorithm 2 L1 Penalized Sequential Convex Programming

Provide an initial guess of solution $x^{(0)}$
Set initial value of $\xi = \xi_0$
for $k = 0 : k_{max}$ **do**
  set L1 penalized subproblem eq. (3) with $x^{(k)}$
  solve the subproblem to obtain $x^{(k+1)}, \nu$
  if $\nu \ge \varepsilon_{CVtol}$ **then**
    $\xi \leftarrow \alpha\xi$
  end if
  if $|x^{(k+1)} - x^{(k)}| \le \varepsilon_{tol}$ and $\nu < \varepsilon_{CVtol}$ **then**
    return $x^{(k+1)}$
  end if
end for
Start SCP (algorithm 1) with initial guess $x^{(k+1)}$

3 Problem Formulation

3.1 Missile Dynamic Model

Throughout the paper, a simple two dimensional missile is utilized. Here, the motion of missile is described using four states, x as downrange, z as altitude[1], v as its speed, and γ as its Flight Path Angle (FPA). It is assumed that four forces are exerted on the missile: lift (L), drag (D), thrust (T), and gravity (g). The missile adjusts its Angle Of Attack (AOA, α) in order to control lift and drag. As a result, a simple 2D missile dynamic equations of motion can be summarized as eq. (4). Visual representation of missile dynamic model is given in fig. 1.

[1] For a full 6DOF missile model, positive z axis of missile is often defined with respect to nadir (downward positive). However, for sake of simplicity, here we denoted altitude as z (upward positive).

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

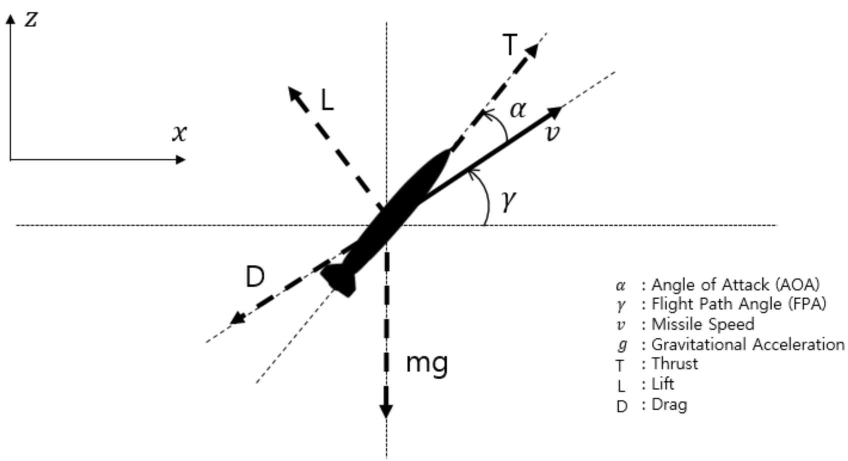

Fig. 1: Illustration of missile dynamic model

$$
\begin{aligned}
\dot{x} &= v\cos\gamma \\
\dot{z} &= v\sin\gamma \\
\dot{v} &= -\frac{D}{m} + \frac{T\cos\alpha}{m} - g\sin\gamma \\
\dot{\gamma} &= -\frac{L}{mv} + \frac{T\sin\alpha}{mv} - \frac{g\cos\gamma}{v}
\end{aligned}
\tag{4}
$$

$$
\begin{aligned}
L &= \frac{1}{2}\rho v^2 S_{ref} C_{L\alpha}\alpha \\
D &= \frac{1}{2}\rho v^2 S_{ref}(C_{D_0} + kC_{L\alpha}^2\alpha^2) \\
T &= \begin{cases} T_0, & t <= t_{burn} \\ 0, & t > t_{burn} \end{cases} \\
m &= \begin{cases} m_0 - \dot{m}t, & t <= t_{burn} \\ m_0 - \dot{m}t_{burn}, & t > t_{burn} \end{cases}
\end{aligned}
\tag{5}
$$

Here, $m = m(t)$ stands for total missile mass, $\rho = \rho(z, v)$ denotes air density, S_ref denotes missile reference area. $C_{L\alpha}$ denotes lift coefficient slope (=$\partial C_L/\partial\alpha$), C_{D_0} denotes zero-lift drag coefficient, and k stands for drag constant.

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

3.2 Optimal Missile Guidance Problem

In this paper, optimal missile guidance to Projected Impact Point (PIP) is considered. PIP is a pre-determined point which the missile desires to make an impact with the target, at final time t_f. If the position of PIP is given as (x_f, z_f), arrival at PIP gives a boundary condition in optimal control sense.

$$\begin{aligned} x(t_f) &= x_f \\ z(t_f) &= z_f \end{aligned} \tag{6}$$

Also, the initial launch condition at initial time t_0 must be considered as another set of boundary conditions.

$$\begin{aligned} x(t_0) &= x_0 \\ z(t_0) &= z_0 \\ v(t_0) &= v_0 \\ \gamma(t_0) &= \gamma_0 \end{aligned} \tag{7}$$

Furthermore, limitations on control input must be considered. In this paper, we considered maximum AOA limit α_{max}

$$|\alpha| \leq \alpha_{max} \quad \text{or} \quad -\alpha_{max} \leq \alpha \leq \alpha_{max} \tag{8}$$

In order to maximize the effectiveness of its warhead, it is often desired to arrive at the PIP with its maximum possible speed. In optimal control point-of-view, maximum speed objective can be mathematically stated as a minimization problem, comprising every constraints mentioned above. (eq. (9))

$$\begin{aligned} \text{minimize :} \quad & -v(t_f) \\ \text{subject to :} \quad & \text{eqs. (4) to (8)} \end{aligned} \tag{9}$$

3.3 Convexification of Dynamic Constraints

In order to convexify eq. (9), convexification of dynamic constraint eq. (4) is necessary. Other than that, suggested constraints are already linear, which do not require further convexification.[2]

First, change of variable from t to x is necessary in order to fix the domain of independent variable. Note that x is increasing monotonically in a physically possible missile guidance situation, which is required for independent variables.

[2] Note: Linear functions are subsets of convex functions

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

$$\begin{aligned}
\frac{dz}{dx} &= \tan\gamma \\
\frac{dv}{dx} &= -\frac{D}{mv\cos\gamma} + \frac{T\cos\alpha}{mv\cos\gamma} - \frac{g\tan\gamma}{v} \\
\frac{d\gamma}{dx} &= -\frac{L}{mv^2\cos\gamma} + \frac{T\sin\alpha}{mv^2\cos\gamma} - \frac{g}{v^2}
\end{aligned} \tag{10}$$

Since dynamic constraints are equality constraints, they have to be linearized to form affine equality constraints. A naive approach would directly linearize the dynamic equations of motion. However, separating control inputs can often accelerate the convergence and reduce control oscillations. [12, 11] Hence, control-related terms must be separated to generate a control-affine form eq. (11). Here, nonlinear terms of control input are substituted with new control variables, which are defined in eq. (12).

$$\begin{aligned}
\frac{dz}{dx} &= \tan\gamma \\
\frac{dv}{dx} &= \left[-\frac{g\tan\gamma}{v} - \frac{\frac{1}{2}\rho v^2 S_{ref} C_{D_0}}{mv\cos\gamma}\right] + \left[-\frac{\frac{1}{2}\rho v^2 S_{ref} k C_{L\alpha}^2}{mv\cos\gamma}\right] u_2 + \left[\frac{T}{mv\cos\gamma}\right] u_3 \\
\frac{d\gamma}{dx} &= \left[-\frac{g}{v^2}\right] + \left[\frac{\frac{1}{2}\rho v^2 S_{ref} C_{L\alpha}}{mv^2\cos\gamma}\right] u_1 + \left[\frac{T}{mv^2\cos\gamma}\right] u_4
\end{aligned} \tag{11}$$

$$\begin{aligned}
u_1 &= \alpha \\
u_2 &= \alpha^2 \\
u_3 &= \cos\alpha \\
u_4 &= \sin\alpha
\end{aligned} \tag{12}$$

Linearizing the control-affine dynamics in the current state estimate $(z^{(k)}, v^{(k)}, \gamma^{(k)})$ provides an approximated affine equality constraint to the convex subproblem. For example, $\frac{dz}{dx}$ can be linearized into: $\frac{dz}{dx} = \tan\gamma^{(k)} + \sec^2\gamma^{(k)}(\gamma - \gamma^{(k)})$. Closed-form expressions for $\frac{dv}{dx}$ and $\frac{d\gamma}{dx}$ can be also derived after tedious calculation, which are omitted in the text due to its excessive length. In order to show the outline of linearization only, let us denote eq. (11) in matrix form (eq. (14)), with state vector $\boldsymbol{\zeta} = (z^{(k)}, v^{(k)}, \gamma^{(k)})$ and control vector $\boldsymbol{u} = (u_1, u_2, u_3, u_4)$.

$$\frac{d\boldsymbol{\zeta}}{dx} = A(\boldsymbol{\zeta}) + B(\boldsymbol{\zeta})\boldsymbol{u} \tag{13}$$

Linearizing eq. (14) in the proximity of current state estimate $\boldsymbol{\zeta}^{(k)}$ gives the approximated affine dynamic constraint.

$$\frac{d\boldsymbol{\zeta}}{x} = f(\boldsymbol{\zeta}^{(k)})\zeta + B(\boldsymbol{\zeta}^{(k)})\boldsymbol{u} + c(\boldsymbol{\zeta}^{(k)}) \tag{14}$$

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

where $f(\boldsymbol{\zeta}^{(k)})) = \frac{\partial A(\boldsymbol{\zeta})}{\partial \zeta}|_{\boldsymbol{\zeta}=\boldsymbol{\zeta}^{(k)}}, \quad c(\boldsymbol{\zeta}^{(k)}) = A(\boldsymbol{\zeta}^{(k)}) - f(\boldsymbol{\zeta}^{(k)})\zeta^{(k)}$.

For additional control inputs, alternative expression of eq. (12) can be suggested as eq. (16). The new form removes α. Certain part of the new control constraints are actually convex.

$$\begin{aligned} u_1^2 &= u_2 \\ u_3^2 + u_4^2 &= 1 \end{aligned} \tag{15}$$

Relaxing equality to inequality transforms the given nonconvex expression into convex inequality constraint. Note that only half portion of the original nonconvex constraints are present in the new relaxed constraints.

$$\begin{aligned} u_1^2 &\leq u_2 \\ u_3^2 + u_4^2 &\leq 1 \end{aligned} \tag{16}$$

Because of relaxation, exact relation between control inputs cannot be assured to be satisfied. Practically, relaxation error in the optimization process stays in the acceptable range. Although relaxation degrades the quality of solution, advantage of rapid computation overwhelms the weakness in the practical domain.

Finally, the control inputs and state variables must be discretized to generate a convex parameter optimization problem. Any parameter discretization scheme can be used provided that it does not violate the convex property. First, let us divide the domain into N equispaced gridpoints, and map the corresponding states onto the grid.

$$\begin{aligned} x &\rightarrow [x_1, x_2, ..., x_i, ..., x_N] \\ \boldsymbol{\zeta} &\rightarrow [\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2, ..., \boldsymbol{\zeta}_i, ..., \boldsymbol{\zeta}_N] \\ \boldsymbol{u} &\rightarrow [\boldsymbol{u}_1, \boldsymbol{u}_2, ..., \boldsymbol{u}_i, ..., \boldsymbol{u}_N] \\ \Delta x &= (x_f - x_0)/(N-1) \\ x_i &= x_0 + (i-1)\Delta x \\ \boldsymbol{\zeta}_i &= \boldsymbol{\zeta}(x_i) \\ \boldsymbol{u}_i &= \boldsymbol{u}(x_i) \end{aligned} \tag{17}$$

Then, every objective and constraints can be described using the discretized state variables. Furthermore, differentiation associated with dynamic constraints can be discretized using trapezoidal rule, which preserves the affinity.

$$\begin{aligned} \dot{\boldsymbol{\zeta}}_i \approx \frac{\boldsymbol{\zeta}_{i+1} - \boldsymbol{\zeta}_i}{\Delta x} = &\frac{1}{2}[f(\boldsymbol{\zeta}_{i+1}^{(k)})\zeta + B(\boldsymbol{\zeta}_{i+1}^{(k)})\boldsymbol{u} + c(\boldsymbol{\zeta}_{i+1}^{(k)}) \\ &+ f(\boldsymbol{\zeta}_i^{(k)})\zeta + B(\boldsymbol{\zeta}_i^{(k)})\boldsymbol{u} + c(\boldsymbol{\zeta}_i^{(k)})] \end{aligned} \tag{18}$$

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

4 Numerical Example and Analysis

Table 1: Summary of Example Scenarios

Scenario Title	Objectives	x_f	z_f	Solution Method
3020_min_tf_ps	minimize t_f	30km	20km	Pseudospectral[a]
3020_max_vf_ps	maximize v_f	30km	20km	Pseudospectral
3020_max_vf_cvx[b]	maximize v_f	30km	20km	LPSCP[c]
3030_min_tf_ps	minimize t_f	30km	30km	Pseudospectral
3030_max_vf_ps	maximize v_f	30km	30km	Pseudospectral
3030_max_vf_cvx	maximize v_f	30km	30km	LPSCP
3040_min_tf_ps	minimize t_f	30km	40km	Pseudospectral
3040_max_vf_ps	maximize v_f	30km	40km	Pseudospectral
3040_max_vf_cvx	maximize v_f	30km	40km	LPSCP
5020_min_tf_ps	minimize t_f	50km	20km	Pseudospectral
5020_max_vf_ps	maximize v_f	50km	20km	Pseudospectral
5020_max_vf_cvx	maximize v_f	50km	20km	LPSCP

[a] Implemented using GPOPS-II[15]
[b] t_f minimization is omitted due to change of variable from t to x
[c] Implemented using MOSEK[14]

In order to perform numerical experiments, a guidance mission inspired from Air-Launched Hit-To-Kill (ALHTK) operation using PAC-3 missile is considered. The missile is launched mid-air at $x_0 = 0$ km, $z_0 = 10$ km, $v_0 = 300$m/s, $\gamma_0 = 0$ deg. Initial mass of the missile is m_0 = 320kg, with propellant mass m_p = 157.6 kg. The propellant burns for t_burn = 21.47 sec., which gives mass rate of $\dot{m}$ = 7.34 kg/s. The nominal thrust of missile is T_0 = 16900N. Furthermore, reference diameter is assumed to be 0.25m, which gives reference area S_{ref} = 0.0491 m^2. Finally, simple aerodynamic model is employed, with C_{D_0} = 0.2 0.66 depending on mach number and thrust, k = 0.1, and $C_{L\alpha}$ = 5.0/rad.

In order to demonstrate the advantage of suggested convex approach over classic pseudospectral approach, twelve example scenarios are considered. Each scenario is set with different final impact points, objectives, and solution methods. Table 1 summarizes details of example scenarios.

Pseudospectral optimization using GPOPS-II[15] is included to generate several nominal solutions for comparison purpose. While global optimality of these nominal solutions are not guaranteed,(which is nature of nonlinear optimization) their local optimality is assured. Furthermore, t_f minimization problem is also implemented in the pseudospectral approach. Practically, t_f minimization problem often generates a good alternative to v_f maximizing trajectory, with similar final velocity result.

LPSCP algorithm is implemented with MATLAB and MOSEK[14]. Convexified subproblems are implemented as conic optimization problems supported by MOSEK.

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

Identical initial estimates are supplied to both methods. A linear initial guess is used, which simply connects initial and final states linearly. It mimics the cold-start case, in which no prior information on optimal trajectory is given.

The optimized control and state trajectories are summarized in figs. 2 to 5. Final time and velocity results are summarized in table 2, with total cpu time consumed.

Table 2: Summary of Optimization Results

Scenario Title	t_f(s)	v_f (m/s)	Average Cpu Time[a] (s)
3020_min_tf_ps[b]	29.75	1464.3	13.7
3020_max_vf_ps	29.68	1492.7	12.3
3020_max_vf_cvx[c]	29.64	1456.9	0.33
3030_min_tf_ps	33.43	1440.8	17.6
3030_max_vf_ps	33.42	1451.9	11.2
3030_max_vf_cvx	33.09	1471.8	0.32
3040_min_tf_ps	38.93	1361.9	10.8
3040_max_vf_ps	38.94	1367.2	11.9
3040_max_vf_cvx	38.21	1422.5	0.42
5020_min_tf_ps	44.84	1200.5	5.9
5020_max_vf_ps	43.58	1325.4	11.1
5020_max_vf_cvx	43.39	1279.2	0.24

[a] Based on pc equipped with Intel i7-6700 cpu, with 32GB RAM.
[b] Based on GPOPS-II default setup, using ipopt nonlinear optimizer.
[c] Based on MOSEK default setup, with MATLAB fusion library

4.1 Analysis on Pseudospectral Results

Interestingly, min_tf series of scenarios converge into suboptimal t_f compared to max_vf series, although the suboptimality is very small. However, adjusting initial guess and maximum t_f bound for min_tf series scenarios does not provide similar results to max_vf series. While it contradicts common intuition, this seemingly impossible phenomenon occur due to the nonlinearity of missile guidance problem. Specifically, sharp transition of thrust from maximum to zero cannot be precisely accounted in the pseudospectral regime, which introduces certain amount of discretization error. The error unfortunately propagates into the optimization process, which produces suboptimality in the continuous domain. It is one of the reason why using several similar but different objective functions in practical situations.

While it seems it is hard to reach global optimality, still trajectories supplied from pseudospectral optimization shows two distinct types of nominal trajectories; one from min_tf series, and one from max_vf series. Both trajectories are feasible and efficient enough in practice.

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

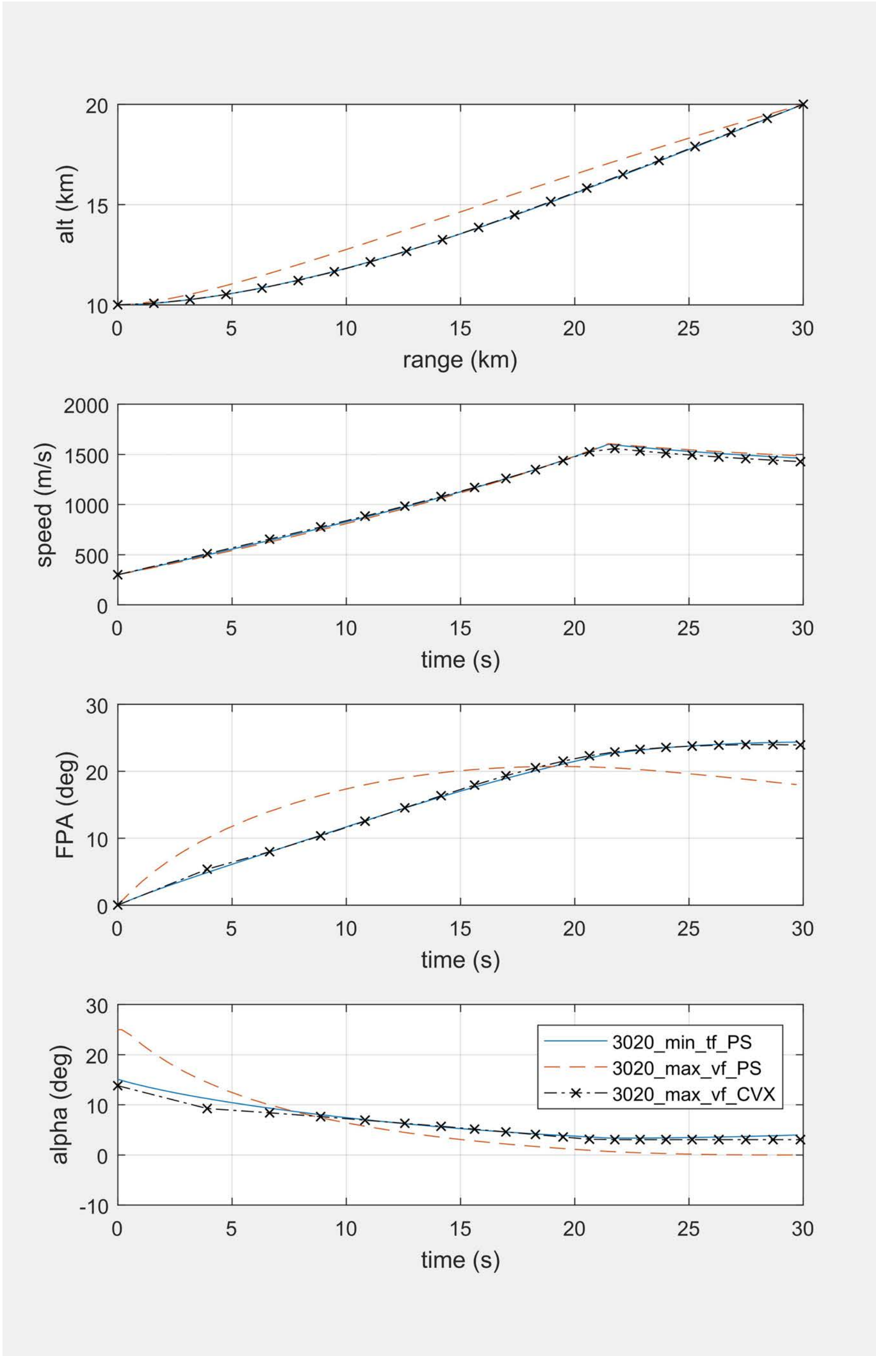

Fig. 2: Optimization results for 3020 scenarios

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

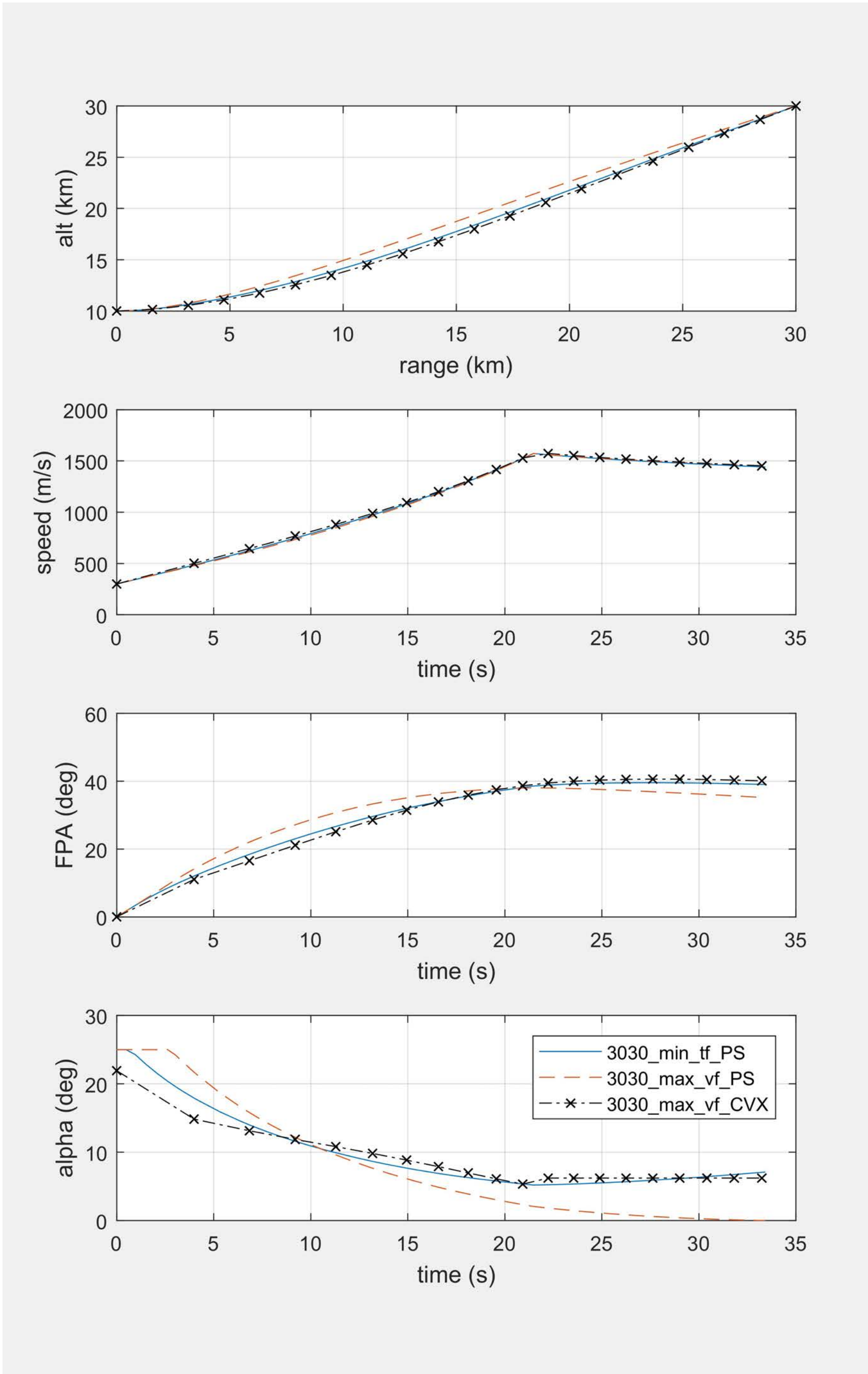

Fig. 3: Optimization results for 3030 scenarios

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

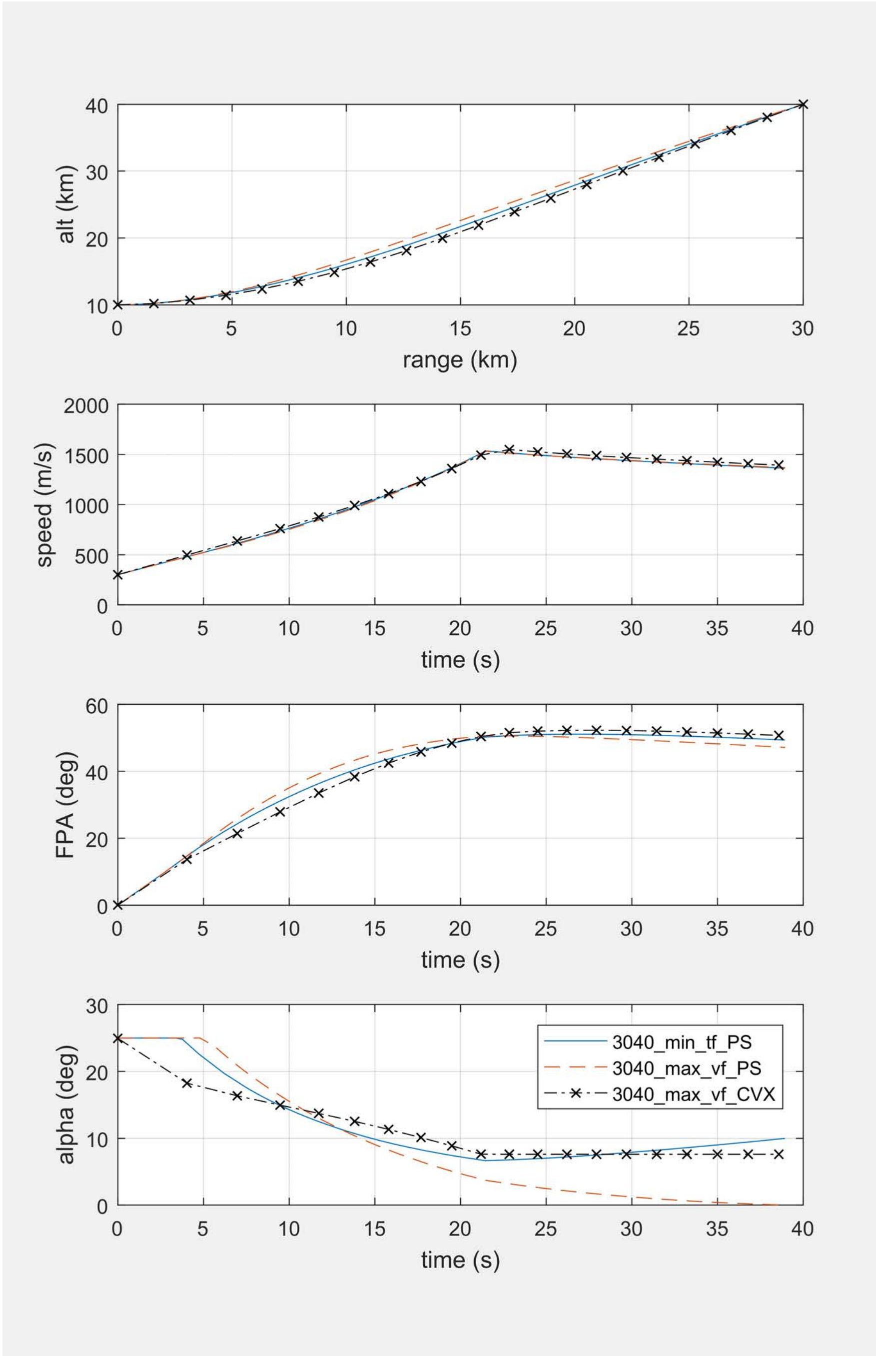

Fig. 4: Optimization results for 3040 scenarios

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

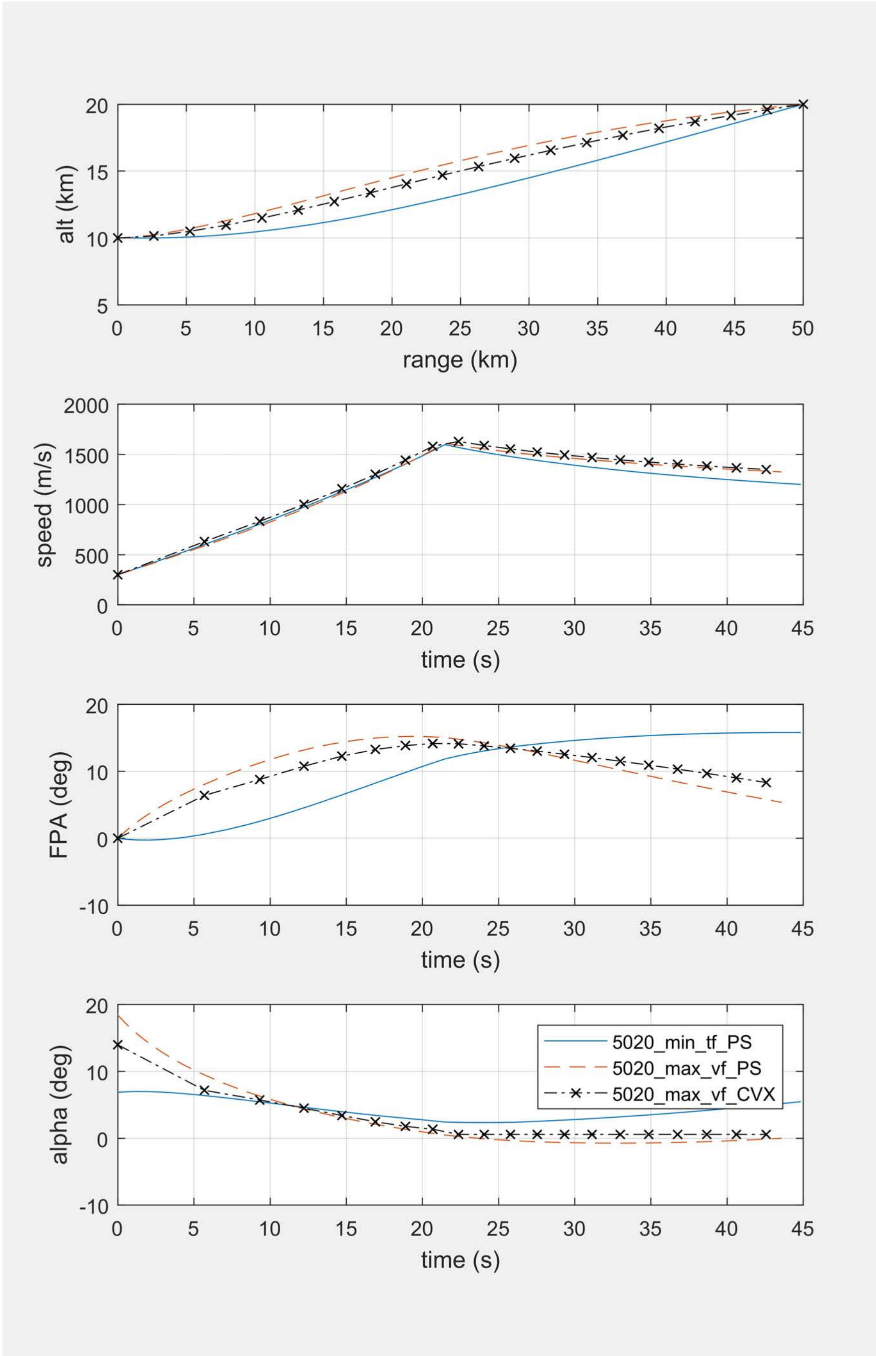

Fig. 5: Optimization results for 5020 scenarios

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

4.2 Analysis on LPSCP Results

Using LPSCP methods, it is possible to obtain optimized trajectories less than 0.5 second, which highly surpasses that of the pseudospectral method. (Reduction by 95%) Furthermore, it is observable that the trajectories generated from LPSCP algorithm successfully reaches one of nominal trajectories, even with local convex approximations. While not included in the paper, simple linearization of dynamics fail to produce results, while suggested control-affine method converges to nominal solutions. It validates the effectiveness of above suggested convexification process.

The results from LPSCP method obeys initial and final state conditions, as well as state and control limits. As the result shows, applicability of several constraints is distinct advantage of numerical guidance algorithms, which was impossible for classic guidance laws.

Due to relaxation, LPSCP solutions are not strictly feasible. However, the effect of relaxation is negligible in the resulting trajectories. We expect that strictly feasbility can be improved by penalizing the relaxation error, which is beyond scope of this initial research.

Furthermore, note that the optimal trajectories automatically emerged from crude initial estimates. When generic SCP algorithm is utilized, it often halts due to infeasibility of poor initial estimate, while LPSCP does not. It shows the versatility of LPSCP algorithm, which improved the sensitivity and stability of SCP algorithms to initial guesses.

Also note that LPSCP method does not require any knowledge on feasible solutions. On the contrary, classic firing table method, or modern machine learning based guidance both require enormous amount of prior study on optimal/suboptimal solutions. This highlights the advantage of LPSCP over other labor-intensive guidance techniques.

5 Conclusion

This paper presents a sequential convex optimization method for missile trajectory optimization. The main contribution of this paper can be summarized as: (1) LP-SCP, a stability-improved version of SCP algorithm using L1-penalty method is proposed. (2) Transformation method of generic missile guidance problem to convex subproblem is developed, which is one of a few kind which incorporates nonlinear thrust component. The suggested algorithm is numerically implemented to an ALTHK/PAC-3 missile operation model, regarding several scenarios. Although certain relaxations are exerted on the part of constraints, numerical simulations show that the solution of convexified problem converges well to the nominal feasible solutions under domain of 0.5 second, regardless of quality of initial guess. The simulation results clearly supports the effectiveness and versatility of the LPSCP algorithm on the optimal missile guidance problem.

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.

Acknowledgements This work was conducted at High-Speed Vehicle Research Center of KAIST with the support of the Defense Acquisition Program Administration and Agency for Defense Development. (Contract number: UD170018CD)

References

1. Acikmese, B., and Ploen, S.R. "Convex Programming Approach to Powered Descent Guidance for Mars Landing", Journal of Guidance, Control, and Dynamics, Vol. 30, No. 5, pp. 1353-1366. (2007)
2. Bhagat, M., "Convex Guidance for Envisat Rendezvous "(Master's thesis), TU Delft, Netherlands. (2016)
3. Blackmore, L. Autonomous Precision Landing of Space Rockets, Winter Bridge on Frontiers of Engineering, vol. 4, no. 46. (2016)
4. Boyd, S., and Vandenberghe, L. Convex Optimization. Cambridge University Press. (2015)
5. Erer, K.S., Tekin, R., and Ozgoren, M.K. "Look Angle Constrained Impact Angle Control Based on Proportional Navigation", AIAA Guidance, Navigation, and Control Conference, AIAA SciTech Forum, (2005)
6. Fletcher. R, An L1 penalty method for nonlinear constraints, Numerical Optimization 1984, pp. 2640, (1985)
7. Kim, T.H., Lee, C.H., Jeon, I.S., and Tahk, M.J., "Augmented Polynomial Guidance With Impact Time and Angle Constraints," in IEEE Transactions on Aerospace and Electronic Systems, vol. 49, no. 4, pp. 2806-2817. (2013)
8. Kim, T.H., Park, B.G., and Tahk, M.J. "Bias-Shaping Method for Biased Proportional Navigation with Terminal-Angle Constraint", Journal of Guidance, Control, and Dynamics, Vol. 36, No. 6, pp. 1810-1816. (2013)
9. Jeon, I.S., Lee, J.I., and Tahk, M.J., "Guidance law to control impact time and angle," 2005 International Conference on Control and Automation, Budapest, , pp. 852-857 Vol. 2. (2005)
10. Liu, X., and Lu, P. "Solving Nonconvex Optimal Control Problems by Convex Optimization", Journal of Guidance, Control, and Dynamics, Vol. 37, No. 3, pp. 750-765. (2014)
11. Liu, X., Shen, Z., and Lu, P. "Exact convex relaxation for optimal flight of aerodynamically controlled missiles," in IEEE Transactions on Aerospace and Electronic Systems, vol. 52, no. 4, pp. 1881-1892, (2016)
12. Liu, X., Shen, Z., and Lu, P.. "Closed-Loop Optimization of Guidance Gain for Constrained Impact", Journal of Guidance, Control, and Dynamics, Vol. 40, No. 2, pp. 453-460. (2017)
13. Lu, P. "Introducing Computational Guidance and Control", Journal of Guidance, Control, and Dynamics, Vol. 40, No. 2, pp. 193-193. (2017)
14. MOSEK ApS, The MOSEK optimization toolbox for MATLAB manual. Version 8.1., 2017.
15. Patterson, M.A., and Rao. A.V. "GPOPS-II: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using hp-Adaptive Gaussian Quadrature Collocation Methods and Sparse Nonlinear Programming", ACM Transactions on Mathematical Software, Vol.41, No.1 (2014)
16. Ryoo, C.K., Cho, H.J., and Tahk, M.J. "Optimal Guidance Laws with Terminal Impact Angle Constraint", Journal of Guidance, Control, and Dynamics, Vol. 28, No. 4, pp. 724-732. (2005)
17. Szmuk, M., Acikmese, B., and Berning, A.W. "Successive Convexification for Fuel-Optimal Powered Landing with Aerodynamic Drag and Non-Convex Constraints", AIAA Guidance, Navigation, and Control Conference, AIAA SciTech Forum, (2016)
18. Zarchan, P., Tactical and Strategic Missile Guidance. American Institute of Aeronautics and Astronautics. (2012)

Copyright by the author(s) and/or respective owners. Published with authorisation by CEAS.