Efficient Input-Constrained Impulsive Optimal Control of Linear Systems with Application to Spacecraft Relative Motion

Ethan Foss[1] and Simone D'Amico[1]

***Abstract*— This work presents a novel algorithm for impulsive optimal control of linear time-varying systems with the inclusion of input magnitude constraints. Impulsive optimal control problems, where the optimal input solution is a sum of delta functions, are typically formulated as an optimization over a normed function space subject to integral equality constraints and can be efficiently solved for linear time-varying systems in their dual formulation. In this dual setting, the problem takes the form of a semi-infinite program which is readily solvable in online scenarios for constructing maneuver plans. This work augments the approach with the inclusion of magnitude constraints on the input over time windows of interest, which is shown to preserve the impulsive nature of the optimal solution and enable efficient solution procedures via semi-infinite programming. The resulting algorithm is demonstrated on the highly relevant problem of relative motion control of spacecraft in Low Earth Orbit (LEO).**

I. Introduction

The problem of minimizing the norm of a control input function space of a linear time-varying system with boundary conditions has received significant attention due to its practical application to many systems of interest. Such optimal control problems arise in many fields and are central to the study of spacecraft optimal control and trajectory optimization. Despite its widespread applicability, impulsive optimal control remains challenging for many applications of interest, which may involve complex state and input constraints and require efficient and reliable solution procedures. For example, in the context of spacecraft relative motion control, for which computational resources are limited, safety is essential, and algorithmic reliability is crucial, few existing approaches are adequate.

Approaches to solving impulsive optimal control problems are numerous, and include direct methods, such as sequential convex programming [1], indirect methods, leveraging moment problems [2], reachable set theory [3], or primer-vector theory [4], and other techniques, such as analytical methods [5]. In direct methods, the dynamics and cost are discretized over a time grid of interest and solved as a finite-dimensional optimization problem. Despite widespread utility, these techniques do not explicitly exploit solution sparsity and therefore tend to introduce many optimization variables or can produce suboptimal solutions in the case of coarse gridding.

Indirect methods, in comparison, involve formulating optimality conditions on the continuous-time problem, and include moment-based formulations [2], reachable set theory [3], and primer-vector theory approaches [6], which provide different perspectives to the problem and yet all arrive at equivalent optimality conditions.

Early algorithmic frameworks for leveraging the optimality conditions derived from these indirect methods are proposed in Gilbert [3], which employed an early numerical algorithm based on quadratic programming [7] to efficiently solve impulsive rendezvous problems. Other approaches involve simplified techniques to solve the necessary conditions to suboptimality [8] or applying mixed iterative schemes to solving the polynomial necessary and sufficient conditions of optimality [9]. In 2013, Arzelier [10] connected the moment problem theorized by Neustadt [2] to a semi-infinite programming convex optimization scheme leveraging exchange methods [11] to produce an efficient, convergent, and reliable strategy to solving the linearized impulsive optimal control problem. This semi-infinite programming approach has served as the foundation for subsequent impulsive optimal control studies [12], [13] and has heritage on CubeSat missions such as VISORS [14].

Despite the practicality of these theoretical and algorithmic frameworks, the applicability of indirect methods to constrained problems remains limited. With respect to spacecraft rendezvous, a major constraint of interest involves bounding the maximum impulse that a spacecraft can execute at a given time. All satellites, such as those of the VISORS mission, have propulsion systems that naturally constrain the amount of ΔV that can be applied at a given time, necessitating the use of maneuver splitting algorithms or other heuristic approaches [14] which lead to suboptimal maneuver plans or increased computational burden. More broadly, it is often practical to distribute ΔV expenditure for spacecraft relative motion irrespective of propulsion constraints to reduce the aggressiveness of maneuver plans.

To address this theoretical and algorithmic gap, this paper introduces impulse magnitude constraints by applying inequality constraints to the integral of the input norm over specified time windows. It is shown that the resulting formulation preserves the impulsive quality of the solution and can be easily extended to semi-infinite programming techniques to yield an efficient and reliable solution procedure. The algorithm is demonstrated for several optimal control problems of interest in spacecraft relative motion control. In contrast to other formulations of input-constrained optimal control [15], [16], which are ubiquitous in aerospace literature, the formulation presented here is unique in that it preserves solution impulsivity, allowing for both improved algorithmic procedures and more desirable maneuver plans.

[1]Department of Aeronautics and Astronautics, Stanford University, 94305, USA, {erfoss,damicos}@stanford.edu

II. Problem Background

This work considers linear, time-varying systems of the form

$$\dot{\boldsymbol{x}}(t) = A(t)\boldsymbol{x}(t) + B(t)\boldsymbol{u}(t), \tag{1}$$

where $\boldsymbol{x} : [t_0, t_f] \to \mathbb{R}^{n_x}, t \to \boldsymbol{x}(t)$ is the state trajectory represents the system state, $\boldsymbol{u} : [t_0, t_f] \to \mathbb{R}^{n_u}, t \to \boldsymbol{u}(t)$ is the input signal, $A : [t_0, t_f] \to \mathbb{R}^{n_x \times n_x}$ is the piecewise-continuous plant matrix, and $B : [t_0, t_f] \to \mathbb{R}^{n_x \times n_u}$ is the input matrix. Boundary conditions are enforced at fixed initial and terminal times such that

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \ \boldsymbol{x}(t_f) = \boldsymbol{x}_f, \tag{2}$$

where $\boldsymbol{x}_0 \in \mathbb{R}^{n_x}$ and $\boldsymbol{x}_f \in \mathbb{R}^{n_x}$ are fixed initial and terminal constraints. The set of input functions that satisfy the system dynamics and boundary conditions can be expressed as an equality constraint in terms of the state transition matrix via

$$\boldsymbol{x}_f = \Phi(t_f, t_0)\boldsymbol{x}_0 + \int_{t_0}^{t_f} \Phi(t_f, \tau)B(\tau)\boldsymbol{u}(\tau)d\tau. \tag{3}$$

Impulsive optimal control problems typically seek to minimize an accumulated norm of the control input, i.e.,

$$J(\boldsymbol{u}(\cdot)) = \int_{t_0}^{t_f} f(\boldsymbol{u}(\tau))d\tau, \tag{4}$$

where $f : \mathbb{R}^{n_u} \to \mathbb{R}^+$ is a norm that satisfies positive definiteness ($f(\boldsymbol{u}) \geq 0 \ \forall \boldsymbol{u} \in \mathbb{R}^{n_u}$, homogeneity ($f(\alpha\boldsymbol{u}) = \alpha f(\boldsymbol{u}) \ \forall \alpha \in \mathbb{R}^+$), and the triangle inequality ($f(\boldsymbol{u}_1) + f(\boldsymbol{u}_2) \geq f(\boldsymbol{u}_1 + \boldsymbol{u}_2) \ \forall \boldsymbol{u}_1, \boldsymbol{u}_2 \in \mathbb{R}^{n_u}$).

Problem 1 (Basic Impulsive Control): Find $\boldsymbol{u}(\cdot)$ to minimize $J(\boldsymbol{u}(\cdot))$ (Equation 4) such that the dynamics (Equation 3) are satisfied.

Problem 1 is a moment problem of the same form given in Neustadt [2], whose optimal solution is a sum of delta functions, i.e. the input is an atomic measure. In its primal form, this problem is difficult to solve due to its infinite dimensional nature. However, in the dual setting, the problem is conveniently reformulated as the following convex semi-infinite program

Problem 2 (Basic Dual Problem):

$$\begin{aligned} \sup_{\boldsymbol{\lambda}_f} \quad & \boldsymbol{\lambda}_f^\intercal(\boldsymbol{x}_f - \Phi(t_f, t_0)\boldsymbol{x}_0) \\ \text{s.t.} \quad & f^\circ(B^\intercal(t)\Phi^\intercal(t_f, t)\boldsymbol{\lambda}_f) \leq 1 \quad \forall t \in [t_0, t_f] \end{aligned} \tag{5}$$

where the optimization variable $\boldsymbol{\lambda}_f$ is a Lagrange multiplier and $f^\circ(\cdot) : \mathbb{R}^{n_u} \to \mathbb{R}^+$ is the dual norm of f, i.e. $f^\circ(\boldsymbol{y}) = \sup_{f(\boldsymbol{x})\leq 1} \boldsymbol{x}^\intercal\boldsymbol{y}$. For many relevant norms, e.g., l_p norms, the dual norm is analytically available ($\|\cdot\|_p^\circ = \|\cdot\|_q, 1/p+1/q = 1$, see also [12]).

In contrast to the infinite dimensional primal problem, the dual problem is an optimization problem over a finite number of variables subject to an infinite constraint space. The dual form of this problem is in fact easier to solve in this form, as Arzelier [10] leveraged, through exchange methods for semi-infinite programs [11], [17].

The moment problem is connected to both primer-vector theory and reachable set theory approaches through its dual form. Indeed, $\boldsymbol{\lambda}_v(t) \triangleq B^\intercal(t)\Phi^\intercal(t_f, t)\boldsymbol{\lambda}_f$ is in fact the classic primer-vector [6], with the Lagrange multiplier $\boldsymbol{\lambda}_f$ being the costate at the terminal time, and the semi-infinite constraint in Problem 2 is equivalent to that derived in Prussing [4]. From the perspective of reachable set theory [3], the dual norm of the costate, $f^\circ(B^\intercal(t)\Phi^\intercal(t_f, t)\boldsymbol{\lambda}_f)$, is simply the support function of the sublevel set of the unit cost at time t, and $\max_{t\in[t_0,t_f]} f^\circ(B^\intercal(t)\Phi^\intercal(t_f, t)\boldsymbol{\lambda}_f)$ is nothing but the support function of the pseudostate reachable set under unit cost [12].

Problem 3 (Basic Maneuver Reconstruction): Let $\boldsymbol{\lambda}_f^*$ be the optimal solution to Problem 2 and $t^* = \{t \in [t_0, t_f] \mid f^\circ(B^\intercal(t)\Phi^\intercal(t_f, t)\boldsymbol{\lambda}_f^*) = 1\}$. The set t^* represents the optimal impulse times to the system. Neustadt [2] showed that the number of impulse times is at most n_x, and that selecting n_x components of t^* recovers the optimal semi-infinite variables. From these times, the optimal control can be constructed as $\boldsymbol{u}(t) = \sum_{t_k\in t^*} \boldsymbol{v}_k\delta(t-t_k)$ where $\{\boldsymbol{v}_k\}_{k=1}^n$ satisfies

$$\boldsymbol{x}_f = \Phi(t_f, t_0)\boldsymbol{x}_0 + \sum_{k=1}^{n} \Phi(t_f, t_k)B(t_k)\boldsymbol{v}_k \tag{6}$$

III. Problem Formulation

The novel contribution of this paper is the inclusion of input magnitude constraints on the control input. The consideration of input magnitude constraints is nontrivial, since bounded control magnitudes, as are typically enforced in aerospace literature for continuous thrust formulations [15], [16], prevents the optimal solution from concentrating mass at discrete time points. As will be shown, the formulation presented here maintains impulsivity of the input profile, which provides both algorithmic benefits and a desirable input structure for spacecraft which are often modeled as impulsively actuated.

The following modification to Problem 1 is proposed:

Problem 4 (Magnitude Constrained Problem): Find $\boldsymbol{u}(\cdot)$ that solves the primal optimization problem $\mathcal{P}(\{T_k\}_{k=1}^N) =$

$$\begin{aligned} \min_{\boldsymbol{u}(\cdot)} \ & \sum_{k=1}^{N} \int_{T_k} f_k(\boldsymbol{u}(t))\mathrm{d}t \\ \text{s.t. } & \boldsymbol{x}_f = \Phi(t_f, t_0)\boldsymbol{x}_0 + \sum_{k=1}^{N} \int_{T_k} \Phi(t_f, t)B(t)\boldsymbol{u}(t)\mathrm{d}t, \\ & \int_{T_k} f_k(\boldsymbol{u}(t))\mathrm{d}t \leq \Delta V_k \quad \forall k \in \mathbb{Z}_1^N, \end{aligned} \tag{7}$$

where $f_k(\cdot)$ are norms, ΔV_k are fixed impulse magnitude limits, and T_k represent ordered control input time windows that satisfy $\bigcup_{k=1}^N T_k \subseteq [t_0, t_f], \ T_i \cap T_j = \emptyset \ \forall i \neq j$.

A. Dual Formulation

To show that Problem 4 is amenable to being solved through semi-infinite programming, the dual form can be derived in a similar manner to that used for Problem 1. The

Lagrangian of Equation 7 is

$$\mathcal{L}(\boldsymbol{u}(\cdot),\boldsymbol{\lambda}_f,\boldsymbol{\sigma}) = \boldsymbol{\lambda}_f^\intercal(\boldsymbol{x}_f - \Phi(t_f,t_0)\boldsymbol{x}_0) - \sum_{k=1}^{N}\Delta V_k\sigma_k + \sum_{k=1}^{N}\int_{T_k}(1+\sigma_k)f_k(\boldsymbol{u}(t)) - \boldsymbol{\lambda}_f^\intercal\Phi(t_f,t)B(t)\boldsymbol{u}(t)\mathrm{d}t, \tag{8}$$

where $\boldsymbol{\lambda}_f$ and $\boldsymbol{\sigma} = (\sigma_k)_{k=1}^N$ are Lagrange multipliers associated with the equality and inequality constraints of the primal in Problem 4 respectively. The associated dual function is given by

$$g(\boldsymbol{\lambda}_f,\{\sigma_k\}_{k=1}^N) = \inf_{\boldsymbol{u}(\cdot)}\mathcal{L}(\boldsymbol{u}(\cdot),\boldsymbol{\lambda}_f,\{\sigma_k\}_{k=1}^N) = \begin{cases}\boldsymbol{\lambda}_f^\intercal(\boldsymbol{x}_f - \Phi(t_f,t_0)\boldsymbol{x}_0) - \sum_{k=1}^N\Delta V_k\sigma_k, \\ \text{if } f_k^\circ(B^\intercal(t)\Phi^\intercal(t_f,t)\boldsymbol{\lambda}_f) \le 1+\sigma_k \\ \forall t\in T_k, \forall k\in\mathbb{Z}_1^N; -\infty, \text{otherwise}\end{cases} \tag{9}$$

From this, the dual form is arrived to with

Problem 5 (Magnitude Constrained Dual Problem): Find $\boldsymbol{\lambda}_f,\boldsymbol{\sigma}$ such that $\mathcal{D}(\{T_k\}_{k=1}^N) =$

$$\begin{aligned}\sup_{\boldsymbol{\lambda}_f,\boldsymbol{\sigma}\ge 0}\quad & \boldsymbol{\lambda}_f^\intercal(\boldsymbol{x}_f - \Phi(t_f,t_0)\boldsymbol{x}_0) - \sum_{k=1}^N\Delta V_k\sigma_k \\ \text{s.t.}\quad & f_k^\circ(B^\intercal(t)\Phi^\intercal(t_f,t)\boldsymbol{\lambda}_f) \le 1+\sigma_k \\ & \forall t\in T_k,\ \forall k\in\mathbb{Z}_1^N\end{aligned} \tag{10}$$

Similar to Problem 2, the problem has been reduced from an optimization over an infinite function space in the primal to a convex semi-infinite optimization problem over n_x+N variables.

Lemma 1: Problem 5 is strongly dual to the primal Problem 4, i.e. $\mathcal{D}=\mathcal{P}$.

Proof: Evidently, Problem 5 satisfies Slater's condition for semi-infinite programs (see Definition 2.1 from [18]) since there is a feasible interior point, i.e. $\exists\boldsymbol{\lambda}_f,\boldsymbol{\sigma} > 0, \text{s.t.} f_k^\circ(B^\intercal(t)\Phi(t_f,t)\boldsymbol{\lambda}_f) < 1+\sigma_k\ \forall t\in T_k,\forall k\in\mathbb{Z}_1^N$. From Theorem 2.3 of [18], this implies that strong duality holds for the semi-infinite program. ∎

Lemma 2: The semi-infinite dual Problem 5 is reducible [18], i.e., there exists a finite subset of times $\{t\} \triangleq \{t_1,\dots,t_M\}\subset\bigcup_{k=1}^N T_k$ such that $\mathcal{D}(\{t\}) = \mathcal{D}(\{T_k\}_{k=1}^N)$.

Proof: Since the problem exhibits strong duality, the problem is reducible so long as an optimal solution exists (the primal is feasible) according to Theorem 3.1 of [18]. ∎

Lemmas 1 and 2 extend several important properties to the proposed constrained problem. Namely, unboundedness of the dual problem implies infeasibility of the primal, the optimal solution retains impulsivity (i.e., $\boldsymbol{u}(t)=0$ a.e.), and exchange methods [11] are valid for solving Problem 5.

B. Input Reconstruction

The technique used to recover the optimal input to the unconstrained problem from the dual formulation can be similarly applied to its constrained version. If $\boldsymbol{\lambda}_f^*,\boldsymbol{\sigma}^*$ are the optimal solution variables to Problem 5, then the optimal impulse times for a given window are recovered from the set

$$\begin{aligned}\{t\}_k^* &= \{t\in T_k \mid f_k^\circ(B^\intercal(t)\Phi^\intercal(t_f,t)\boldsymbol{\lambda}_f^*) = 1+\sigma_k^*\} \\ &= \{t_{k1},t_{k2},\dots,t_{k,m_k}\}.\end{aligned} \tag{11}$$

and the corresponding input is given by

$$\boldsymbol{u}^*(t) = \sum_{k=1}^N\sum_{j=1}^{m_k}\boldsymbol{v}_{kj}\delta(t-t_{kj}). \tag{12}$$

Furthermore, from the complementary slackness condition, it must hold that

$$\sigma_k\left(\int_{T_k}f_k(\boldsymbol{u}(t))\mathrm{d}t - \Delta V_k\right) = 0\ \forall k\in\mathbb{Z}_1^N, \tag{13}$$

which can be leveraged to show the interesting property that

$$\sigma_k > 0 \implies \int_{T_k}f_k(\boldsymbol{u}(t))\mathrm{d}t = \sum_{j=1}^{m_k}f_k(\boldsymbol{v}_{kj}) = \Delta V_k \tag{14}$$

meaning that the non-zero elements of $\boldsymbol{\sigma}$ correspond to time windows in which the input is saturated.

From the optimal impulse times, the optimal input can be recovered from the tractable finite LP/SOCP:

Problem 6 (Constrained Input Reconstruction Problem): Solve the discretized primal Problem 4 from the optimal impulse times recovered from Problem 5 with $\mathcal{P}_d = \mathcal{P}(\{\{t\}_k^*\}_{k=1}^N)$.

Theorem 3.1: A feasible optimal solution to Problem 4 consists of at most n_x+N impulses.

Proof: Since strong duality holds and the dual form is a convex semi-infinite program of n_x+N variables, Theorem 3.2 of [18] states that there exists a discretization S which satisfies $|S|\le n_x+N$ such that $\mathcal{D}(S)=\mathcal{D}$, leading to at most n_x+N points or atoms in time at which $\boldsymbol{u}(t)$ may be nonzero. ∎

Remark 1: The number of impulses can be further reduced when considering the addition of the $\boldsymbol{\sigma}\ge 0$ constraints, which reduces the active support of the semi-infinite constraint, leading to at most n_x+N_a points in time at which the input may be nonzero, where N_a is the number of strictly positive $\boldsymbol{\sigma}$.

IV. Algorithmic Approach

Solving Problem 4 in an algorithmically tractable manner can be done through first solving the dual problem using exchange methods to determine the optimal impulse locations and then reconstructing the input through the discretized primal.

A. Dual Problem Algorithm

This work leverages an exchange method [11] for solving the semi-infinite program, which consists of iteratively solving a finite version of Problem 5 and adding constraints corresponding to the worst semi-infinite constraint violation. To illustrate this iterative technique, define the modified form of Problem 5, given by

Problem 7: Solve the discretized form of Problem 5 $\mathcal{D}(S)$, where $S \triangleq \{s_1, s_2, \dots s_d\} \subset \bigcup_{k=1}^{N} T_k$ is a finite set, with the addition of the constraint:

$$\boldsymbol{\lambda}_f^{\intercal}(\boldsymbol{x}_f - \Phi(t_f, t_0)\boldsymbol{x}_0) - \sum_{k=1}^{N} \Delta V_k \sigma_k \leq \Delta V_{tot}, \tag{15}$$

where $\Delta V_{tot} = \sum_{k=1}^{N} \Delta V_k$ is the total amount of expendable fuel imposed.

Since Problem 5 is reducible according to Lemma 2, then it holds that the finite version of the problem can solve the semi-infinite problem if S is chosen correctly. The additional constraint in Problem 7 prevents artificial infeasibility in the case that the discrete form is under constrained in the exchange method process. This constraint is natural to apply since $\mathcal{D} \leq \mathcal{P} \leq \Delta V_{tot}$ if a feasible point exists, meaning there is no loss in feasibility when this constraint is imposed.

To leverage the reducibility property of the dual, exchange methods gradually add values to the set S such that $\mathcal{D}(S^{(j)}) \to \mathcal{D}$ [11], [17]. Several prominent techniques exist in exchange methods for augmenting the discrete set S, which effect problem size growth and convergence rate. The technique proposed here simply evaluates the semi-infinite constraint over a grid and adds time points corresponding to the largest constraint violations.

$$S^{(j+1)} \leftarrow S^{(j)} \cup \underset{t \in \cup_{k=1}^{N} T_k : f_k^{\circ}(\boldsymbol{\lambda}_v^{(j)}(t)) > 1 + \sigma_k^{(j)}}{\arg\operatorname{loc}\max} f_k^{\circ}(\boldsymbol{\lambda}_v^{(j)}(t)) \tag{16}$$

Here, $\boldsymbol{\lambda}_v^{(j)}(t) = B^{\intercal}(t)\Phi^{\intercal}(t_f, t)\boldsymbol{\lambda}_f^{(j)}$ is the primer vector at iteration j of the algorithm. Since $f_k^{\circ}(\boldsymbol{\lambda}_v^{(j)}(t))$ is trivial to compute in this case, this process does not introduce significant computational burden in the algorithmic framework. The discrete set can be initialized with simply the initial and final times, i.e., $S^{(1)} = \{t_0, t_f\}$ or with the optimal times associated with a two-impulse burn as is done in [10]. The selection of the initial times and the addition of new times are hyperparameters of the algorithmic framework which can be adjusted for specific problems, akin to that done in [12].

These algorithmic modifications retain several important features of exchange methods. Namely, it is evident that $\mathcal{D} \leq \mathcal{D}(S^{(j+1)}) \leq \mathcal{D}(S^{(j)})$ since $S^{(j)} \subseteq S^{(j+1)}$. Moreover, given a tolerance ϵ, an important result is as follows, which is extended from [10].

Lemma 3: Suppose Problem 7 is solved such that $f_k^{\circ}(B^{\intercal}(t)\Phi^{\intercal}(t_f, t)\boldsymbol{\lambda}_f^*) - \sigma_k^* \leq 1 + \epsilon \ \forall t \in T_k, \forall k \in \mathbb{Z}_1^N$. Then it follows that $\mathcal{D}(S^{(j)}) \leq (1+\epsilon)\mathcal{D}$.

Proof: Let $\bar{\boldsymbol{\lambda}}_f = \boldsymbol{\lambda}_f^*/(1+\epsilon)$ and $\bar{\sigma}_k = \sigma_k^*/(1+\epsilon)$. Evidently, $(\bar{\boldsymbol{\lambda}}_f, \bar{\boldsymbol{\sigma}})$ is a feasible point to Problem 5 that satisfies $\mathcal{D}(S^{(j)})/(1+\epsilon) = \bar{\boldsymbol{\lambda}}_f^{\intercal}(\boldsymbol{x}_f - \Phi(t_f, t_0)\boldsymbol{x}_0) - \sum_{k=1}^{N} \Delta V_k \bar{\sigma}_k \leq \mathcal{D} \implies \mathcal{D}(S^{(j)}) \leq (1+\epsilon)\mathcal{D}$. ∎

Lemma 3 can be exploited in an algorithmic framework to ensure that the dual problem is solved to within a specified suboptimality tolerance. Algorithm 1 summarizes the procedure for solving the dual problem.

Algorithm 1 Dual Semi-Infinite Program Algorithm.

Require: LTV Dynamics $\{\Phi(\tau_f, \tau_0), A(\tau), B(\tau)\}$, Time Intervals $\{T_k\}_{k=1}^N$, Norms $\{f_k(\cdot)\}_{k=1}^N$ Boundary Conditions $\{\boldsymbol{x}_0, \boldsymbol{x}_f\}$, Convergence Tolerance ϵ

Ensure: Optimal Cost $\mathcal{D}(S^{(j)})$, Optimal Values $\boldsymbol{\lambda}_f, \boldsymbol{\sigma}$

Discretize $\{T_k\}_{k=1}^N$, $S^{(1)} \leftarrow \{t_0, t_f\}$, $j \leftarrow 1$
while $\exists k \in \mathbb{Z}_1^N, t \in T_k$ s.t. $f_k^{\circ}(B^{\intercal}(t)\Phi^{\intercal}(t_f, t)\boldsymbol{\lambda}_f^{(j)}) - \sigma_k^{(j)} > 1 + \epsilon$ **do**
  $\boldsymbol{\lambda}_f^{(j)}, \boldsymbol{\sigma}^{(j)}, \mathcal{D}(S^{(j)}) \leftarrow$Solve Problem 7 with $S^{(j)}$
  $S^{(j+1)} \leftarrow S^{(j)} \cup \Delta S$ (Equation 16), $j \leftarrow j + 1$
end while
if $\mathcal{D}(S^{(j)}) = \Delta V_{tot}$ **then**
  return Problem Infeasible
end if

B. Input Reconstruction

Reconstructing the input signal from the dual solution requires extracting a discrete set of times where the semi-infinite constraints are active and solving the discretized form of the primal problem. These aspects are summarized in Algorithm 2.

Algorithm 2 Input Reconstruction Algorithm.

Require: LTV Dynamics $\{\Phi(\tau_f, \tau_0), A(\tau), B(\tau)\}$, Time Intervals $\{T_k\}_{k=1}^N$, Norms $\{f_k(\cdot)\}_{k=1}^N$ Boundary Conditions $\{\boldsymbol{x}_0, \boldsymbol{x}_f\}$, Dual Solution $\{\boldsymbol{\lambda}_f^*, \boldsymbol{\sigma}^*\}$

Ensure: Optimal Cost $\mathcal{P}_d$, Input Signal $\boldsymbol{u}(t)$, State $\boldsymbol{x}(t)$

$\{t\}_k^* = \{t \in T_k \mid f_k^{\circ}(B^{\intercal}(t)\Phi^{\intercal}(t_f, t)\boldsymbol{\lambda}_f^*) - (1 + \sigma_k^*) \in [-\epsilon, \epsilon]\}$
$\{\{\boldsymbol{v}_{kj}\}_{j=1}^{m_k}\}_{k=1}^N, \mathcal{P}_d \leftarrow$ Solution to Problem 6
$\boldsymbol{u}(t) \leftarrow \sum_{k=1}^{N} \sum_{j=1}^{m_k} \boldsymbol{v}_{kj} \delta(t - t_{kj})$
$\boldsymbol{x}(t) \leftarrow \Phi(t, t_0)\boldsymbol{x}_0 + \int_{t_0}^{t} \Phi(t, \tau) B(\tau) u(\tau) \mathrm{d}\tau$

Though the above procedure is the most general, it is possible to exploit further algorithmic performance in the case of smooth norm costs and constraints, in which case the direction of the input is known and the ambiguous input magnitudes can be solved through least squares.

Theorem 4.1: Solving the infinite program Problem 4, by means of Algorithm 1 followed by Algorithm 2, is globally convergent to the global minimum to within the imposed tolerance ϵ.

Proof: Since Problem 7 is convex and bounded above, each subproblem of Algorithm 1 will converge. Moreover, Algorithm 1 will always terminate since $|S^{(j)}|$ is bounded. Finally, since $\mathcal{D}(S^*) \leq \mathcal{D}(1+\epsilon)$ (Lemma 3), $\mathcal{D} = \mathcal{P}$ (Lemma 1), and $\mathcal{D}(S^*) = \mathcal{P}(S^*)$ (strong duality of SOCPs/LPs), $\mathcal{P}(S^*) \leq \mathcal{P}(1+\epsilon)$. ∎

V. Performance and Validation

The problem formulation and corresponding algorithmic approach is validated on the problem of impulsive control for proximity operations of satellites.

A. System Definition

Relative motion of satellites in orbits subject to single body gravitational force can be modeled as linear time-varying systems under a variety of perturbations, such as drag

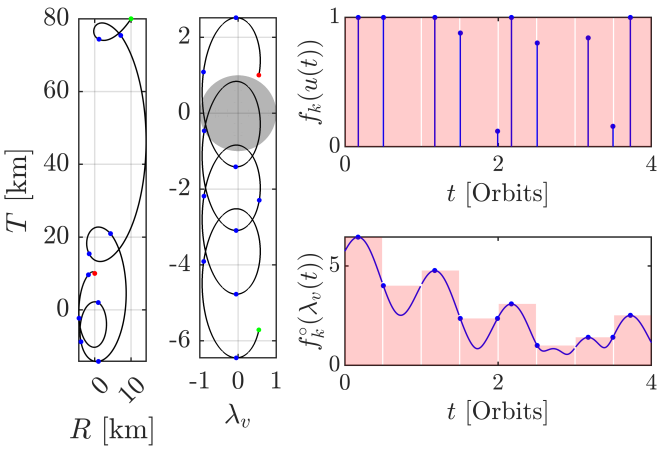

Fig. 1. Planar Reconfiguration Trajectory.

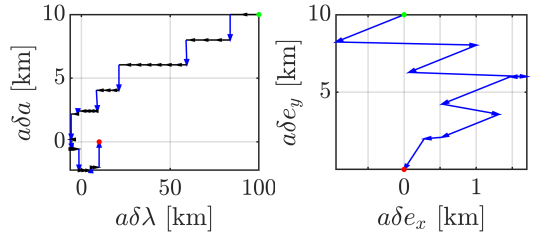

Fig. 2. Planar Reconfiguration Relative Orbital Elements

or spherical harmonics, through a relative orbital element state representation [19], [20], defined by

$$\delta\boldsymbol{\alpha} = \begin{bmatrix}\delta a & \delta\lambda & \delta e_x & \delta e_y & \delta i_x & \delta i_y\end{bmatrix}^\intercal. \tag{17}$$

These states are functions of the classical Keplerian orbital elements of circular two-body motion of the target and chaser spacecraft. The linearized equations of motion under this state are of Jordan form and can be found in [20] along with transformation matrices to RTN frame cartesian state. The ensuing examples use an orbital mean motion of $n = .00113$, consistent with LEO.

B. Planar Reconfiguration Demonstration

The first example considers a planar rendezvous problem consisting of reducing along track separation from a distance of 100 km to 10 km, i.e., $a\delta\boldsymbol{\alpha}_0 = [10, 100, 0, 10, 0, 0]^\intercal$ [km] to $a\delta\boldsymbol{\alpha}_f = [0, 10, 0, 0, 0, 0]^\intercal$ [km]. The transfer occurs over four orbits, with eight equally spaced time windows enforced over the duration of the transfer with a buffer of $1/10$th of an orbit between each window. Each time window is discretized over 1000 time points in the dual optimization scheme to provide sufficient resolution. The input is scaled such that a unitary impulse corresponds to 1.13 [m/s] of ΔV expenditure. Moreover, the norm applied is the l_2 norm, such that $f_k(\cdot) = \|\cdot\|_2\ \forall k$.

Figure 1 displays the RT-plane relative position, primer vector trajectory, input impulse profile, and dual norm trajectory. Figure 2 shows the trajectory from the perspective of the planar relative orbital elements. Blue dots/arrows denote impulse events, green dots denote starting time, and red dots denote terminal time. The computed dual optimal solution for the problem is $\boldsymbol{\lambda}_f^* = [0.7380, 0.0891, -0.2359, -0.7236, 0, 0]^\intercal, \boldsymbol{\sigma}^* = [5.4503, 3.0092, 3.7714, 1.3608, 2.0926, 0, 0.4142, 1.5183]^\intercal$. The total ΔV expenditure is 8.860 [m/s]. Notably, the dual norm exceeds the value of 1 at all but one of the time windows, corresponding exactly to the time windows at which the input constraint is active.

To validate and compare the performance of the proposed algorithm, the problem is also solved under a direct optimization framework whereby the primal problem is discretized with the inputs at each discrete node as solution variables. Under this approach, the number of solution variables increases drastically compared to the dual solution framework, with the total number of solution variables amounting to $n_u NM$, with n_u being the input dimension, N the number of time windows, and M the discretization count per time window. In contrast, the dual SIP involves $n_x + N$ solution variables, with cone count being on the order of $\sim 2N$-$3N$, depending on the problem structure. The poor scalability of direct primal optimization manifests as longer solver times, with this issue being especially apparent for long duration transfers, whereby the discretization induces substantial variable counts.

Under this direct optimization approach, the equivalent problem is solved and produces an equivalent solution. However, the solver runtime is 0.6 seconds, whereas the cumulative solver time for the dual SIP with input reconstruction is 0.1262 seconds[1].

C. VISORS Transfer Demonstration and Comparison

The VISORS mission [14] is a two-nanosatellite formation-flying mission that aims to perform telescopic measurements of the solar corona. Part of the mission operations involves performing a long duration, impulse constrained transfer from a 200 meter radius standby orbit to a 40 meter science orbit, i.e., $a\delta\boldsymbol{\alpha}_0 = [0, 0, 0, 200, 0, 200]^\intercal$ [m] to $a\delta\boldsymbol{\alpha}_f = [-2.62, 45.21, -34.51, 4.78, -18.72, 2.72]^\intercal$ [m]. The challenge of this transfer is compounded by the limitations of the propulsion systems aboard the nanosatellites, which can provide a maximum impulse of just 2 millimeters per second with at least 45 seconds of cool down time between each impulse.

To demonstrate the strength of the proposed algorithm for this difficult problem, the standby to science scenario from [14] is demonstrated with a transfer duration of 10 orbits and 500 equally spaced time windows are enforced with a buffer of 45 seconds and a maximum impulse expenditure of 2 millimeters per second per window. The l_1 norm is applied, $f_k(\cdot) = \|\cdot\|_1\ \forall k$, to be consistent with the fuel metric in [14]. Each time window is discretized over 100 time points.

The computed VISORS transfer trajectory is displayed in Figure 3, including the RTN trajectory, a subset of the impulse profile, and a subset of the dual norm trajectory. Additionally, the relative orbital element trajectory is plotted in Figure 4, along with the RN plane separation, which

[1]Runtime experiments were performed on an Intel i7-10750H CPU in MATLAB R2024a with the MOSEK solver [21] and YALMIP modeler [22]. Solver times are determined from extracting the MOSEK solver time from the YALMIP solution.

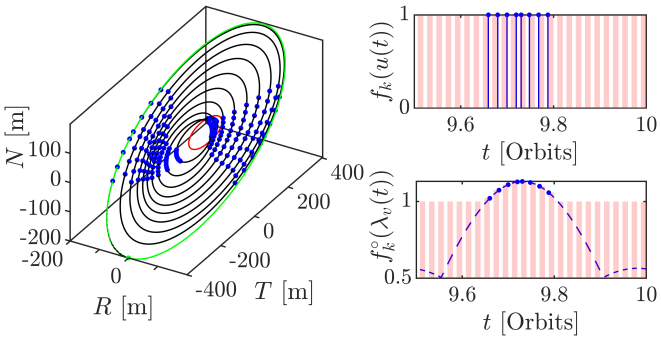

Fig. 3. VISORS Transfer Trajectory.

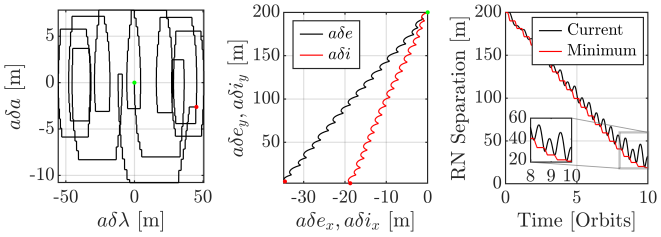

Fig. 4. VISORS Transfer Relative Orbital Elements

TABLE I
VISORS TRANSFER PERFORMANCE OVER 250 MONTE CARLO TRIALS WITH 1σ BOUNDS

Total ΔV	Proposed Solver Time	Direct Solver Time
33.27 ± 0.72 [cm/s]	0.1912 ± 0.0480 [s]	0.4234 ± 0.0263 [s]

validates safe spacecraft separation over the transfer duration [23].

Finally, this scheme is embedded in a high-fidelity simulation, with the inclusion of execution errors, navigation uncertainty, and dynamical effects that are consistent with [14]. The control dynamical model includes drag and J_2 effects, as derived in [20], and the transfer is replanned every orbit for the first 9 orbits and every quarter orbit during the final orbit. Table I summarizes the performance over 250 Monte Carlo trials. Importantly, the proposed approach results in an average of 33.27 [cm/s] of ΔV expenditure for the transfer, which substantially improves on the 49.34 [cm/s] metric reported in [14]. Given that this transfer is performed 20 times over the mission span, this amounts to an estimated total ΔV saving of 3.19 meters per second, which is $\sim$25% of the total ΔV budget of the VISORS CubeSats.

VI. CONCLUSION

This paper demonstrated an extension to a class of moment problems that includes constraints on the integral of the norm of the input signal over time windows of interest. It was shown that the resulting optimal solution remains impulsive and that semi-infinite programming can be applied to obtain optimal solutions with guaranteed convergence. This algorithmic framework was applied to challenging spacecraft relative motion control problems, where it was used to compute maneuver sequences that outperformed established techniques in terms of optimality and runtime.

REFERENCES

[1] Y. Mao, M. Szmuk, X. Xu, and B. Acikmese, "Successive convexification: A superlinearly convergent algorithm for non-convex optimal control problems," 2019.

[2] L. W. Neustadt, "Optimization, a moment problem, and nonlinear programming," *Journal of the Society for Industrial and Applied Mathematics Series A Control*, vol. 2, no. 1, pp. 33–53, 1964.

[3] E. Gilbert and G. Harasty, "A class of fixed-time fuel-optimal impulsive control problems and an efficient algorithm for their solution," *IEEE Transactions on Automatic Control*, vol. 16, no. 1, pp. 1–11, 1971.

[4] J. Prussing, "Optimal impulsive linear systems: Sufficient conditions and maximum number of impulses," *Journal of the Astronautical Sciences*, vol. 43, pp. 195–206, 04 1995.

[5] M. Chernick and S. D'Amico, "New closed-form solutions for optimal impulsive control of spacecraft relative motion," *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 2, pp. 301–319, 2018.

[6] D. F. Lawden, *Optimal Trajectories for Space Navigation*. London: Butterworths, 1963.

[7] R. Barr and E. Gilbert, "Some efficient algorithms for a class of abstract optimization problems arising in optimal control," *IEEE Transactions on Automatic Control*, vol. 14, no. 6, pp. 640–652, 1969.

[8] T. E. Carter and J. Brient, "Linearized impulsive rendezvous problem," *J. Optim. Theory Appl.*, vol. 86, no. 3, pp. 553–584, Sep. 1995.

[9] D. Arzelier, C. Louembet, A. Rondepierre, and M. Kara-Zaitri, "A new mixed iterative algorithm to solve the fuel-optimal linear impulsive rendezvous problem," *J. Optim. Theory Appl.*, vol. 159, no. 1, pp. 210–230, Oct. 2013.

[10] D. Arzelier, F. Bréhard, N. Deak, M. Joldes, C. Louembet, A. Rondepierre, and R. Serra, "Linearized impulsive fixed-time fuel-optimal space rendezvous: A new numerical approach," *IFAC-PapersOnLine*, vol. 49, no. 17, pp. 373–378, 2016.

[11] R. Reemtsen and S. Görner, *Numerical Methods for Semi-Infinite Programming: A Survey*. Boston, MA: Springer US, 1998, pp. 195–275.

[12] A. W. Koenig and S. D'Amico, "Fast algorithm for fuel-optimal impulsive control of linear systems with time-varying cost," *IEEE Transactions on Automatic Control*, vol. 66, no. 9, pp. 4029–4042, 2021.

[13] M. Hunter and S. D'Amico, "Fast fuel-optimal constrained impulsive control with application to distributed spacecraft," in *2025 IEEE Aerospace Conference*, 2025, pp. 1–19.

[14] T. Guffanti, T. Bell, S. Y. W. Low, M. Murray-Cooper, and S. D'Amico, "Autonomous guidance navigation and control of the visors formation-flying mission," in *2023 AAS/AAIA Astrodynamics Specialist Conference*, no. AAS 23-149, 2023.

[15] B. Acikmese and S. R. Ploen, "Convex programming approach to powered descent guidance for mars landing," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 5, pp. 1353–1366, 2007.

[16] E. Taheri, I. Kolmanovsky, and E. Atkins, "Enhanced smoothing technique for indirect optimization of minimum-fuel low-thrust trajectories," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 11, pp. 2500–2511, 2016.

[17] R. Hettich, A. Kaplan, and R. Tichatschke, *Semi-infinite Programming: Numerical Methods*. Boston, MA: Springer US, 2009, pp. 3429–3434.

[18] A. Shapiro, "Semi-infinite programming, duality, discretization and optimality conditions†," *Optimization*, vol. 58, no. 2, pp. 133–161, 2009.

[19] S. D'Amico, "Autonomous formation flying in low earth orbit," Ph.D. dissertation, TU Delft, 2010.

[20] A. W. Koenig, T. Guffanti, and S. D'Amico, "New state transition matrices for spacecraft relative motion in perturbed orbits," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 7, pp. 1749–1768, 2017.

[21] M. ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 9.0.*, 2019.

[22] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in matlab," in *In Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[23] S. D'Amico and O. Montenbruck, "Proximity operations of formation-flying spacecraft using an eccentricity/inclination vector separation," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 3, pp. 554–563, 2006.