Computationally-Efficient Collision-Free Trajectory Planning of Satellite Swarms under Modeling Uncertainties and Perturbations

Himadri Basu *, Yasaman Pedari †, Mads Almassalkhi ‡, Hamid R. Ossareh §
University of Vermont, Burlington, VT 05405, USA.

This paper studies the problem of collision-free optimal trajectory planning (TP) for reconfiguration of a satellite swarm in Low Earth Orbit (LEO) under perturbations and modeling uncertainties. By quantifying the modeling errors associated with various relative dynamical models of satellites, we identify a suitable predictive model for TP. To address the computational challenges in selecting the optimal terminal satellite configuration in TP, we decouple the path assignment (PA) problem from the trajectory optimization (TO) algorithm. For the decoupled PA problem, we estimate fuel consumption for each satellite-target pairing, which informs a fuel-optimal PA formulation. To determine collision-free, fuel-optimal satellite maneuvers, we present two novel TP approaches: (1) a distributed TO formulation that iteratively incorporates convexified collision avoidance (CA) constraints; and (2) a decentralized TO-based approach, where convexified CA constraints are now incorporated into the PA subproblem. We find that the first approach scales well while the second approach has low communication requirements for swarms of up to size 50. Finally, to mitigate the effects of modeling uncertainties, we employ feedback in the form of shrinking-horizon model predictive control to update control commands based on satellite measurements.

Nomenclature

R_e	=	6378km, earth's mean equatorial radius
J_2	=	1.082×10^{-3}, second zonal harmonic constant
K_{J2}	=	$1.5\mu J_2 R_e^2 = 2.633 \times 10^{10}$ km^2 s^{-2}
μ	=	398600 km s^{-2}
N	=	number of spacecrafts
$\mathcal{N}_j$	=	neighborhood set of satellite j

*Postdoctoral Associate, Department of Electrical and Biomedical Engineering, himadri.basu@uvm.edu
†Graduate Research Assistant, Department of Electrical and Biomedical Engineering, yasaman.pedari@uvm.edu
‡Associate Professor, Department of Electrical and Biomedical Engineering, malmassa@uvm.edu
§Associate Professor, Department of Electrical and Biomedical Engineering, hossareh@uvm.edu

T	=	one orbital period
R_{safe}	=	minimum allowed inter-satellite distance to avoid collision risk
R_{comm}	=	communication distance between two neighboring spacecrafts
a	=	semimajor axis
e	=	eccentricity
h	=	angular momentum
i	=	inclination angle
Ω	=	right ascension of the ascending node (RAAN)
θ	=	geodesic latitude
(X, Y, Z)	=	Earth-centered inertial (ECI) coordinate system
r	=	orbital position in ECI coordinates
v_x	=	orbital velocity in ECI coordinates
(x, y, z)	=	unit vectors of local-vertical/local-horizontal coordinate frame
$\bar{\boldsymbol{r}}_j$	=	$\left[x_j, y_j, z_j\right]^{\mathrm{T}}$ is relative position vector of j^{th} satellite in LVLH frame
$\dot{\bar{\boldsymbol{r}}}_j$	=	$\left[\dot{x}_j, \dot{y}_j, \dot{z}_j\right]^{\mathrm{T}}$ is relative velocity vector of j^{th} satellite in LVLH frame
$(\omega_x, \omega_y, \omega_z)$	=	angular velocity velocity in LVLH frame
$(\alpha_x, \alpha_y, \alpha_z)$	=	linear acceleration vector of j^{th} satellite in LVLH frame
(a_{jx}, a_{jy}, a_{jz})	=	angular acceleration vector of j^{th} satellite in LVLH frame
$\boldsymbol{V}_a$	=	orbital velocity vector relative to atmospheric drag
ω_e	=	7.9921×10^{-5} rad s^{-1}
$\boldsymbol{V}_{aj}$	=	relative velocity vector of j^{th} satellite relative to atmospheric drag
C_d	=	air drag coefficient
m	=	dry mass of satellite
A/m	=	cross-section of spacecraft per unit mass
ρ	=	air density
t_f	=	terminal time of the optimization problem
k	=	sampling step size
K	=	terminal time-step
$u_i(k)$	=	$\left[u_{ix}^{\mathrm{T}}(k), u_{iy}^{\mathrm{T}}(k), u_{iz}^{\mathrm{T}}(k)\right]^{\mathrm{T}}$ control vector for i^{th} satellite in LVLH frame at time-step k
$u_{m,max}$	=	maximum allowable actuation magnitude for m^{th} thruster
$u_{m,min}$	=	minimum allowable actuation magnitude for m^{th} thruster
$u_{m,max}^{r}$	=	maximum allowable actuation rate for m^{th} thruster

$u^r_{m,min}$ = minimum allowable actuation rate for m^{th} thruster

$u_{\max}$ = $\left[u^{\mathrm{T}}_{1,\max}, u^{\mathrm{T}}_{2,\max}, u^{\mathrm{T}}_{3,\max}\right]^{\mathrm{T}}$

$u^r_{\max}$ = $\left[u^{r^{\mathrm{T}}}_{1,\max}, u^{r^{\mathrm{T}}}_{2,\max}, u^{r^{\mathrm{T}}}_{3,\max}\right]^{\mathrm{T}}$

$\boldsymbol{x}_i(k)$ = $\left[\bar{\boldsymbol{r}}^{\mathrm{T}}_i(k), \dot{\bar{\boldsymbol{r}}}^{\mathrm{T}}_i(k)\right]^{\mathrm{T}}$ is state vector of the optimization problem at time-step k for satellite i

$\bar{\boldsymbol{r}}_i(k)$ = $C_1\boldsymbol{x}_i(k)$ is the position vector of satellite i at time-step k of the optimization problem

$\bar{\boldsymbol{r}}^{\mathrm{NL}}_i(k)$ = relative position vector of satellite i at time-step k under nonlinear dynamics

$\bar{\boldsymbol{r}}^{\mathrm{L}}_i(k)$ = relative position vector of satellite i at time-step k under linearized J2 dynamics

$\bar{\boldsymbol{r}}^{\mathrm{H}}_i(k)$ = relative position vector of satellite i at time-step k under Hill's equation

$\bar{\boldsymbol{r}}^{\mathrm{H2}}_i(k)$ = relative position vector of satellite i at time-step k under modified Hill's equation

$\boldsymbol{e}^{\mathrm{L}}_i(k)$ = $\|\bar{\boldsymbol{r}}^{\mathrm{NL}}_i(k) - \bar{\boldsymbol{r}}^{\mathrm{L}}_i(k)\|_2$ is the relative position error of linearized J2 model

$\boldsymbol{e}^{\mathrm{H}}_i(k)$ = $\|\bar{\boldsymbol{r}}^{\mathrm{NL}}_i(k) - \bar{\boldsymbol{r}}^{\mathrm{H}}_i(k)\|_2$ is the relative position error of Hill's model

$\boldsymbol{e}^{\mathrm{H2}}_i(k)$ = $\|\bar{\boldsymbol{r}}^{\mathrm{NL}}_i(k) - \bar{\boldsymbol{r}}^{\mathrm{H2}}_i(k)\|_2$ is the relative position error of modified Hill's model

$\boldsymbol{e}^{\mathrm{L}}_i$ = $\max\limits_{\forall k \in [0,K]} \{\boldsymbol{e}^{\mathrm{L}}_i(k)\}$

$\boldsymbol{e}^{\mathrm{H}}_i$ = $\max\limits_{\forall k \in [0,K]} \{\boldsymbol{e}^{\mathrm{H}}_i(k)\}$

$\boldsymbol{e}^{\mathrm{H2}}_i$ = $\max\limits_{\forall k \in [0,K]} \{\boldsymbol{e}^{\mathrm{H2}}_i(k)\}$

$\boldsymbol{x}^{\boldsymbol{m}}_i(k)$ = the solution to sequential convex programming (SCP) based optimization problem at $\boldsymbol{m}^{\text{th}}$ iteration

$\bar{\boldsymbol{x}}_i(k)$ = nominal state vector of satellite i used in the optimization problem

$\boldsymbol{x}_{iS}$ = i^{th} state vector in initial formation

$\boldsymbol{x}_{iD}$ = i^{th} state vector in final formation

$\boldsymbol{x}_S$ = $\left[\boldsymbol{x}^{\mathrm{T}}_{1S}, \boldsymbol{x}^{\mathrm{T}}_{2S}, \cdots, \boldsymbol{x}^{\mathrm{T}}_{NS}\right]^{\mathrm{T}}$ is initial state vector of the swarm of spacecrafts

$\boldsymbol{x}_D$ = $\left[\boldsymbol{x}^{\mathrm{T}}_{1D}, \boldsymbol{x}^{\mathrm{T}}_{2D}, \cdots, \boldsymbol{x}^{\mathrm{T}}_{ND}\right]^{\mathrm{T}}$ is final state vector of the swarm of spacecrafts

d_{imqn} = $\min\limits_{\forall k \in \{1,K\}} \|C_1\left(\boldsymbol{x}_i(k) - \boldsymbol{x}_q(k)\right)\|$ with $C_1\boldsymbol{x}_i(k) = \bar{\boldsymbol{r}}_i(k),\ \boldsymbol{x}_i(K) = \boldsymbol{x}_{mD}, \boldsymbol{x}_q(K) = \boldsymbol{x}_{nD},\ m \neq n$

$\|.\|_p$ = l_p norm of a vector, $p \in [1, \infty)$

I_p = identity matrix of order p

0_p = zero vector of dimension $p \times 1$

$O(f(N))$ = computation-time complexity as a function $f(N)$ of varying swarm size N

f_{ij} = fuel cost for satellite i to move to location j in the final formation

F = $[f_{ij}]_{N\times N}$ fuel cost matrix

$d^{\min}_{i,j}$ = minimum euclidean distance between location i and j in initial and final formation respectively

Shorthand Notations

s_θ (or c_θ) = $\sin(\theta)$ (or $\cos(\theta)$)

Subscripts

S = initial condition ($t = 0$)

D = final condition ($t = t_f$)

i (or j) = spacecraft i (or j)

Superscripts

m = iteration count of SCP

I. Introduction

In this paper, we study satellite formation flying (SFF) in LEO with potential applications in earth-observation missions, such as distributed imaging of earth's surface [1, 2], atmospheric sampling [3], Ionosphere measurements [4–7], magnetosphere studies [8], and interferometry [9]. SFF has been a topic of significant research interest in recent years as it offers reduced mission costs, increased flexibility, reconfigurability, and performance compared to a single large spacecraft flying for the same mission [10, 11]. The primary task in an earth-observation-based SFF mission is to place multiple satellites into nearby orbits forming a satellite cluster to cooperatively achieve a group objective [12, 13].

The group objective of a traditional SFF mission either requires the fleet of spacecraft to undergo a formation reconfiguration or maintain a desired formation [14]. In [15, 16], authors proposed a set of J2-invariant initial position and velocity conditions for each spacecraft in a swarm, based on nonlinear energy matching conditions, to perform formation keeping while ensuring collision free trajectories under J2 gravity and air drag for up to 100 orbital periods. These J2-invariant orbits, also called parking orbits, are very effective at swarm keeping once the swarm is in a desired formation. In this work, we specifically look at a formation reconfiguration problem [17] where, starting from an initial formation, the satellite swarm is required to perform a series of complex maneuvers such as moving, reorienting, and rotating to arrive at the desired final formation. In formation reconfiguration problems, it is important to determine such fuel-optimal maneuvers satisfying all physical and mission-specific constraints, that establishes the desired formation for the satellite swarm at the end of these maneuvers.

Our objective is thus to develop fuel-optimal, computationally efficient and scalable trajectory planning (TP) algorithms for formation reconfiguration problem of LEO satellite swarm. We would like our algorithms to run onboard the satellite computers without the need for ground communication, in order to enable missions with sparse ground communication or distant missions such as those in Mars or Jupiter orbits. We also would like our algorithms to be scalable (to up to few hundred satellites) to enable more complex missions than what is possible today. One of the

key challenges in solving such cooperative TP for a large swarm is the computational difficulties associated with the large information flow and the amount of processing required [18]. In a trajectory optimization (TO) framework (*i.e.*, an optimization problem whose solution provides thruster commands and the resulting fuel-optimal, collision-free trajectories), these difficulties are primarily caused by (1) the path assignment (PA) sub-problem (*i.e.*, the assignment of satellites in a fuel-optimal configuration on the final formation [18]), referred to as coupled PATO, which leads to mixed-integer constraints, (2) the non-convexity of the collision avoidance (CA) constraints, and (3) the nonlinear nature of the satellite orbital dynamics [12]. Nonlinear, non-convex coupled PATO formulations are not suitable for large-scale TP even with the modern computational capabilities.

In general, there are three approaches in the literature for solving TP problems: heuristic (e.g., [17, 19]), direct (e.g., [12]) and indirect methods (e.g., [20, 21]): (1) Heuristic TP algorithms such as Rapidly-exploring Random Trees (RRT) and Probabilistic Roadmaps (PRM) were offered in the works of [17, 19]. However, these algorithms require to learn the configuration space through extensive experimental training, which in turn causes a significant memory allocation and processing. Hence, the application of such algorithms remains limited to a small SFF mission. (2) Indirect methods attempt to find solutions satisfying the first-order optimality conditions based on Pontryagin's maximum principle or Euler-Lagrange equations. Indirect methods translate the TO algorithm into a boundary value problem, solving of which is complicated by the nonlinear satellite dynamics [15]. (3) Direct methods on the other hand discretize the control and state space [12], transcribe an infinite-dimensional TO problem into a finite-dimensional nonlinear programming (NLP) problem with discretized control and state variables as optimization parameters. However, these methods solve a centralized problem which scales poorly with the swarm size as the spacecraft state variables are coupled through non-convex CA constraints.

In the works of [15, 16, 22], authors presented a TP problem, considering convexified CA constraints. While CA constraints in [15, 16] take an affine linear form, they were formulated as mixed-integer linear (MIL) constraints in [22]. As noted in [23], the resulting TP algorithm, formulated as mixed-integer linear program (MILP) in [22] has an exponential time-complexity with respect to swarm size and, therefore, these TP algorithms also scale poorly as the swarm size increases even though they use convexified constraints.

In addition to the CA constraints, another key challenge to TP is the PA problem. Given a set of positions and velocities on a final formation of the satellite swarm, the objective of this PA problem is to place each satellite to a specific final destination in a way that is fuel-optimal for the entire fleet (of course, assuming that the satellites are equipped with identical hardware, which is our assumption). However, such assignment problems are also formulated with MIL constraints. With increasing swarm size and integer decision variables, the TP problem consisting of TO algorithm with PA, also referred to as coupled PATO, is not scalable to a large swarm size for the same reason noted above for MIL CA constraints. Therefore, to make our proposed TP algorithm computationally efficient and scalable, in this work we (1) first select a *linear* model of satellite dynamics considering all relevant perturbations in LEO that is

sufficiently accurate for TO, (2) decouple the PA and TO problems from the coupled PATO algorithm, and (3) present two novel TP approaches for formulating convexified CA constraints. More detail on each of these is provided next.

To determine an appropriate model for TO, we present here, for the first time to our knowledge, a comparative analysis of various linear satellite dynamical models under perturbations and then by quantifying modeling errors we identify a TO-suitable model. Specifically, we revisit four dynamical models - namely: (i) high fidelity nonlinear model in [10] that considers both the effects of J2 and air drag, which are known to be the dominant orbital perturbation in LEO [24–27], (ii) linearized J2 model in [14], (iii) Hill's equations in [28], (iv) Modified Hill's equations with J2 in [29]. High fidelity nonlinear model (i) serves as a baseline model with respect to which the modeling accuracy of other dynamical models is evaluated. By quantifying the modeling errors under perturbations for various initial conditions and arbitrary actuations, we identified linearized J2 as the most suitable model for TO. This analysis has been conducted in a cohesive manner with formal quantification of modeling errors in the conference version of our paper in [23]. We include these results here for the sake of completeness.

To make our TP approach computationally efficient and scalable, as discussed above, we decouple the PA problem from the coupled PATO algorithm at the expense of a small loss of optimality (less than 2%). Even after that, TP algorithm still faces computational challenges due to non-convex CA formulations. To overcome this, we propose two solutions in this work, which we refer to as TP1 and TP2, as described next.

In TP1, given the initial satellite positions and velocities and all destination locations on the final formation, satellites independently evaluate the fuel cost to reach each of the possible destinations. This is achieved by each satellite solving a decentralized linear program in parallel [18]. Using these individual fuel costs, a centralized mixed-integer linear program (MILP) is proposed to solve the PA problem and the solution to the PA problem determines the final satellite assignment in the desired formation and also the corresponding optimal trajectories of the swarm towards the target formation. This approach alleviates the large communication overhead in the iterative optimization schemes such as the distributed auction based assignment algorithm in [30]. Once the final assignments are known, the satellites are then required to navigate safely to the target formation using minimum fuel. We achieve this by adopting [15] and presenting a sequential convex programming (SCP)-based approach to satellite swarm with affine CA constraints. This allows the satellites to cooperatively determine collision-free fuel-optimal trajectories from initial to the target formation. In summary, TP1 consists of two sub-problems - (i) decentralized fuel calculations followed by a centralized PA, and (ii) distributed SCP-based TO with CA constraints, which we refer to as the TOCA problem. Since in this case, all the participating satellites share their recent location information with the other neighboring satellites, this TO algorithm may lead to a large communication overhead for a swarm size in the order of hundreds.

To mitigate the communication requirements, we propose an alternative solution, namely TP2, which rather determines a near-optimal final configuration, going towards which renders safe satellite maneuvers. Therefore, CA constraints are now embedded within the PA problem (instead of the TO problem) to obtain a collision-safe final

configuration and the resulting formulation is referred to as PACA problem. This alternate formulation thus consists of two major steps - (i) centralized PACA and (ii) a decentralized TO problem. With the help of numerous illustrative examples, we demonstrate the performance, computational advantage, relative loss of fuel-optimality and communication complexity of both approaches.

Finally, another important aspect that emerges in any practical TP problems, as noted in [18], is the effect of uncertainties (e.g., modeling and uncertain initial positions) and ignored nonlinear effects and disturbances (e.g., solar radiation pressure). To robustify TP against these uncertainties, a robust TP extension is proposed using shrinking horizon model predictive control (SHMPC) that mitigates the accumulation of modeling errors in every prediction horizon. SHMPC can be viewed as a variant of traditional MPC [16, 31], where the prediction horizon shrinks in every iteration unlike receding-horizon MPC. SHMPC computes the control input by optimizing over a finite-horizon subject to control and state constraints with the current state as the initial state of the optimization and terminal state as the fixed final state. Then, the control input is applied to the system until a new computation is completed giving an updated control input.

Contributions: We now summarize our main contributions:

- We quantify uncertainties around different relative dynamical models of satellites under LEO perturbations, and select linearized J2 model as suitable candidate for TP;
- decouple the PA problem from TO algorithm, and to render collision-safe satellite maneuvers we propose two algorithms, namely TP1 and TP2 which efficiently compute near-optimal satellite assignments in the final formation and collision-free trajectories under given mission specifications;
- adapt SHMPC method to multi-satellite formulation to iteratively mitigate the effects of any modeling uncertainties and sampling errors;
- compare performance, computation, and communication complexity, and fuel-optimality between both approaches on a realistic simulation environment in Matlab for validation.

The organization of the paper is as follows. In Section II, we review some preliminaries of swarm reconfiguration problem, present the problem statement, review various relative dynamical models of satellites along with the mission specifications. In Section III, we evaluate modeling errors and consequently quantify uncertainties to identify a suitable model for TP. Before we present our TP algorithms, as preliminary reviews, we first present all the sub-components of the studied problem such as decentralized fuel cost calculation, TO problem and basic SHMPC algorithm in IV along with the disadvantages of the coupled PATOCA (*i.e.* PA, TO and CA sub-problems are considered together in a single optimization algorithm) implementation. This section is followed by presenting two novel approaches such as TP1 and TP2 to solve a TP problem with CA constraints along with suitable numerical examples for validation. We compare the performance, computation-time and communication complexity and fuel-optimality of both approaches in Section VII. Finally, some concluding remarks and future research directions are given in Section VIII.

II. Preliminaries and Problem Statement

In this section, we review the preliminaries of swarm reconfiguration problem and define the problem objective. Given desired initial and final formation of N identical satellites in close proximity in LEO, subject to J2 gravity and air drag, the objective of this manuscript is to present a computationally efficient, scalable TP framework satisfying all physical and mission specific constraints discussed below. These satellites under actuation from the TP algorithm will make optimal maneuvers to reach to the final formation after a specified time duration with minimal fuel consumption. The results presented here are relevant for practical space missions related to SAR interferometry and imaging of earth's surface (and, in the future, other planets). Inspired by the interferometry missions [15, 32], we specifically focus on swarm reconfiguration problem that involves transfer of more than 100 satellites from one J2 invariant passive circular orbits (PCO) to the other with minimum fuel consumption while avoiding any inter-satellite collisions. In the works of [15, 16], J2-invariant PCOs have been found to provide collision free navigation for large satellite swarms ($N \geq 500$ each) for more than 100 orbital periods and thus serves as an excellent configuration for swarm keeping.

In the traditional texts on dynamical motion of satellites [14], an unactuated satellite or a fictitious moving point is usually taken as a "leader" (or reference) and other participating satellites in formation as "followers". Given the initial orbital parameters of the leader, the position and velocity vectors of the leader over time define a target orbit with respect to which the dynamics of all follower satellites are described. This target orbit is generally expressed in Earth-centered inertial (ECI) coordinate frame which has its origin located at the center of the earth, X axis aligned with earth's mean equator and passes through vernal equinox, Z axis along the celestial north pole while the Y axis completing the right hand orthogonal frame with the other two [29]. The relative dynamical motion of all follower satellites on the other hand are described in local-vertical-local-horizontal (LVLH) frame [33], with its origin located on the target orbit. In this local coordinate frame, the relative position vector of j^{th} spacecraft is usually expressed as $\bar{\mathbf{r}}_j = \begin{bmatrix} x_j & y_j & z_j \end{bmatrix}^{\mathrm{T}}$ where the unit vectors associated with x_j, y_j and z_j point in the radial, along-track and across-track directions respectively. LVLH frame is a rotating coordinate frame which rotates at ω_x (rad s^{-1}) in radial direction and ω_z (rad s^{-1}) in across-track direction. The relative motion of satellites in LVLH frame serves as a state-space constraint in the TP problem.

A. Nonlinear relative motion of satellites

A high-fidelity nonlinear relative motion of satellites was presented in [10] considering the effects of both J2 and air drag. As described in [10], the equations of motion for j^{th} spacecraft under J2 perturbation, atmospheric drag and acceleration a_{jx}, a_{jy} and a_{jz} caused by thruster actuation in three directions (we denote thruster commands by

u_{jx}, u_{jy}, u_{jz}) of LVLH coordinate frame, relative to the target orbit are given as follows.

$$\ddot{x}_j = 2\dot{y}_j\omega_z - x_j(n_j^2 - \omega_z^2) + y_j\alpha_z - z_j\omega_x\omega_z + a_{jx} - \bar{\zeta}\mathrm{s}_i\mathrm{s}_\theta - r(n_j^2 - n^2) - l_{1j}(\dot{x}_j - y_j\omega_z) - l_{2j}v_x, \tag{1}$$

$$\ddot{y}_j = -2\dot{x}_j\omega_z + 2\dot{z}_j\omega_x - x_j\alpha_z - y_j(n_j^2 - \omega_z^2 - \omega_x^2) + z_j\alpha_x - \bar{\zeta}\mathrm{s}_i\mathrm{c}_\theta + a_{jy}, \tag{2}$$

$$\ddot{z}_j = -2\dot{y}_j\omega_x - x_j\omega_x\omega_z - y_j\alpha_x - z_j(n_j^2 - \omega_x^2) - \bar{\zeta}\mathrm{c}_i + a_{jz}, \tag{3}$$

where C is air-drag coefficient, $l_{1j} = C\|\mathbf{V}_{aj}\|, l_{2j} = C(\|\mathbf{V}_{aj}\| - \|\mathbf{V}_a\|)$, $\bar{\zeta} = \zeta_j - \zeta$ with

$$\begin{aligned}
\zeta &= \frac{2K_{J2}\mathrm{s}_i\mathrm{s}_\theta}{r^4}, \zeta_j = \frac{2K_{J2}r_{jZ}}{r^5},\ n^2 = \frac{\mu}{r^3} + \frac{K_{J2}}{r^5} - \frac{5K_{J2}\mathrm{s}_i^2\mathrm{s}_\theta^2}{r^5},\\
n_j^2 &= \frac{\mu}{r_j^3} + \frac{K_{J2}}{r_j^5} - \frac{5K_{J2}r_{jZ}^2}{r_j^7},\ r_j = \sqrt{(r+x_j)^2 + y_j^2 + z_j^2},\\
r_{jZ} &= (r+x_j)\mathrm{s}_i\mathrm{s}_\theta + y_j\mathrm{s}_i\mathrm{c}_\theta + z_j\mathrm{c}_i,\ \alpha_z = -\frac{2hv_x}{r^3} - \frac{K_{J2}}{r^5}\mathrm{s}_i^2\mathrm{s}_{2\theta},\\
\alpha_x &= f_x - \frac{K_{J2}\mathrm{s}_{2i}\mathrm{c}_\theta}{r^5} + \frac{3v_xK_{J2}\mathrm{s}_{2i}\mathrm{s}_\theta}{r^4h} - \frac{8K_{J2}^2\mathrm{s}_i^3\mathrm{c}_i\mathrm{s}_\theta^2\mathrm{c}_\theta}{r^6h^2},\\
f_x &= -C\|\mathbf{V}_a\|\omega_e\left(\frac{2rv_x\mathrm{c}_\theta\mathrm{s}_i}{h} - \left(\frac{r}{h}\right)^2\mathrm{c}_\theta\mathrm{s}_i\dot{h} + g_x\right),\ g_x = -\frac{r^2\mathrm{s}_\theta\mathrm{s}_i\dot{\theta}}{h} + \frac{r^2\mathrm{c}_\theta\mathrm{c}_i\dot{i}}{h}\\
\boldsymbol{V}_{aj} &= \boldsymbol{V}_a - \boldsymbol{V}_j,\ \mathbf{V}_a = \begin{bmatrix} v_x & (\frac{h}{r} - \omega_e r\mathrm{c}_i) & \omega_e r\mathrm{c}_\theta\mathrm{s}_i \end{bmatrix}^{\mathrm{T}},\\
\mathbf{V}_j &= \left(v_x + \dot{x}_j - y_j\omega_z\right)\hat{x} + \left(\frac{h}{r} + \dot{y}_j + x_j\omega_z - z_j\omega_x\right)\hat{y} + \left(\dot{z}_j + y_j\omega_x\right)\hat{z}
\end{aligned} \tag{4}$$

where $v_x, \dot{h}, \dot{i}$ and $\dot{\theta}$ are target orbital parameters, given next.

B. Nonlinear target orbit dynamics

Considering the two main disturbances, namely J2 and atmospheric drag, the differential equations governing the motion [10] of the target orbit are

$$\dot{r} = v_x, \tag{5}$$

$$\dot{v}_x = -\frac{\mu}{r^2} + \frac{h^2}{r^3} - \frac{K_{J2}}{r^4}(1 - 3\mathrm{s}_i^2\mathrm{s}_\theta^2) - C\|\mathbf{V}_a\|v_x, \tag{6}$$

$$\dot{h} = -\frac{K_{J2}}{r^3}\mathrm{s}_i^2\mathrm{s}_{2\theta} - C\|\mathbf{V}_a\|(h - \omega_e r^2\mathrm{c}_i), \tag{7}$$

$$\dot{\Omega} = -\frac{2K_{J2}}{hr^3}\mathrm{c}_i\mathrm{s}_\theta^2 - \frac{C\|\mathbf{V}_a\|\omega_e r^2\mathrm{s}_{2\theta}}{2h}, \tag{8}$$

$$\dot{i} = -\frac{K_{J2}}{2hr^3}\mathrm{s}_{2i}\mathrm{s}_{2\theta} - \frac{C\|\mathbf{V}_a\|\omega_e r^2\mathrm{s}_i\mathrm{c}_\theta^2}{h}, \tag{9}$$

$$\dot{\theta} = \frac{h}{r^2} + \frac{2K_{J2}\,\mathrm{c}_i^2\mathrm{s}_\theta^2}{hr^3} + \frac{C\|\mathbf{V}_a\|\omega_e r^2\mathrm{c}_i\mathrm{s}_{2\theta}}{2h}, \tag{10}$$

Nonlinear relative motion of satellites (1) – (3) turn the TO algorithm into an NLP, which is not computationally feasible for a considerable swarm size with modern optimization solvers such as IPOPT. Therefore, it is necessary to find an appropriate linear dynamical model which captures the effects of relevant perturbing forces and can be used as a suitable alternative to this nonlinear model for TO problem. This nonlinear satellite relative model in (1) – (3) then serves as a benchmark model for evaluating accuracy of other models and identifying a suitable model for TP. Next, we revisit three linear relative dynamical models of satellites, capturing the effects of LEO perturbations. As noted in [14], air-drag effect is negligible at the intended altitude of our mission. Thus, none of our candidate linear models consider air-drag effect on satellite relative motion.

C. Linearized J2 model

In the high fidelity nonlinear model (1) – (4), there are nonlinear terms n_j^2, ζ_j which include polynomials of the reciprocal of r_j and consequently x_j, y_j, z_j. By using Gegenbauer polynomials, the terms n_j^2 and ζ_j were shown to bear a linear relationship in [14] with the decision variables x_j, y_j and z_j as follows

$$\zeta_j = \zeta - \frac{8K_{J2}x_j \mathrm{s}_i \mathrm{s}_\theta}{r^5} + \frac{2K_{J2}y_j \mathrm{s}_i \mathrm{c}_\theta}{r^5} + \frac{2K_{J2}z_j \mathrm{c}_i}{r^5},$$
$$n_j^2 = n^2 - \frac{3\mu x_j}{r^4} - \frac{5K_{J2}}{r^6}\left[x_j(1-5\mathrm{s}_i^2\mathrm{s}_\theta^2) + y_j\mathrm{s}_i^2\mathrm{s}_{2\theta} + z_j\mathrm{s}_{2i}\mathrm{s}_\theta\right].$$

With the above substitution, the first order linear J2 model yields the following time-varying dynamics

$$\ddot{x}_j = \frac{2\dot{y}_j h}{r^2} + x_j\left(\frac{2\mu}{r^3} + \frac{h^2}{r^4} + \frac{4K_{J2}(1-3\mathrm{s}_i^2\mathrm{s}_\theta^2)}{r^5}\right) + a_{jx} - y_j\left(\frac{2v_x h}{r^3} - \frac{3K_{J2}\mathrm{s}_i^2\mathrm{s}_{2\theta}}{r^5}\right) + \frac{5K_{J2}\mathrm{s}_{2i}\mathrm{s}_\theta}{r^5}, \tag{11}$$

$$\ddot{y}_j = -\frac{2\dot{x}_j h}{r^2} - \frac{2K_{J2}\dot{z}_j \mathrm{s}_{2i}\mathrm{s}_\theta}{r^3 h} + x_j\left(\frac{2v_x h}{r^3} + \frac{5K_{J2}\mathrm{s}_i^2\mathrm{s}_{2\theta}}{r^5}\right) - y_j\left(\frac{\mu}{r^3} - \frac{h^2}{r^4} + \frac{K_{J2}(1+2\mathrm{s}_i^2 - 7\mathrm{s}_i^2\mathrm{s}_\theta^2)}{r^5}\right)$$
$$+ z_j\left(\frac{3K_{J2}v_x\mathrm{s}_{2i}\mathrm{s}_\theta}{r^4 h} - \frac{2K_{J2}\mathrm{s}_{2i}\mathrm{c}_\theta}{r^5}\right) + a_{jy}, \tag{12}$$

$$\ddot{z}_j = \frac{2K_{J2}\dot{y}_j\mathrm{s}_{2i}\mathrm{s}_\theta}{r^3 h} + \frac{5K_{J2}x_j\mathrm{s}_{2i}\mathrm{s}_\theta}{r^5} - \frac{3K_{J2}y_j v_x \mathrm{s}_{2i}\mathrm{s}_\theta}{r^4 h} - z_j\left(\frac{\mu}{r^3} + \frac{K_{J2}(3-2\mathrm{s}_i^2 - 5\mathrm{s}_i^2\mathrm{s}_\theta^2)}{r^5}\right) + a_{jz}, \tag{13}$$

where r, v_x, h, θ, i are time-varying forcing terms obtained by solving Eqs. (5)–(10).

D. Hill-Clohessy-Wiltshire (HCW) Model with J2

A modified HCW model considering earth's J2 non-spherical effect [29] is given as follows

$$\ddot{x}_j = a_{jx} + 2nc\dot{y}_j + (5k_1^2 - 2)n^2 x_j, \tag{14}$$

$$\ddot{y}_j = a_{jy} - 2nc\dot{x}_j, \tag{15}$$

$$\ddot{z}_j = a_{jz} - k_2^2 z_j, \tag{16}$$

where $k_2 = nk_1 + 1.5J_2\left(\frac{R_e c_i}{r}\right)^2$, $k_1 = \sqrt{1+k_3}$, $k_3 = 0.375J_2\left(\frac{R_e}{r}\right)^2(1+3c_{2i})$. In the absence of J2 perturbations, equations (14) – (16) reduce to original HCW model or popularly known as Hill's equation [34]. In Section III, we study the modeling accuracy of these models, namely Linearized J2, HCW with J2, and HCW without J2. We now summarize below the mission specifications for illustrative numerical simulations presented in this paper.

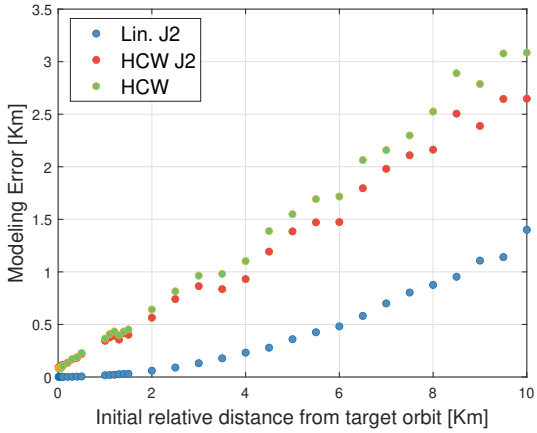
E. Mission specifications for our numerical simulations

Inspired by a formation flying space mission TechSat-21 [35] by U.S. Air Force Research Laboratory (AFRL) in 2006 and their mission specifications, our intended altitude for SFF mission is $420-500$ km. For swarm reconfiguration problem with applications in distributed imaging of earth's surface [32], we consider that both the initial and the final formations, centered at the target low earth sun-synchronous orbit (SSO), are concentric PCOs with satellites maintaining at least 50 m safety distance from each other. For such swarm reconfiguration problem, allotted delta-v budget is 500 m/s and total impulsive actuation of 100 N s. Additionally, we assume that the initial target orbit is circular and all the satellites are initially located within 15 km from this target orbit. The position tolerance of the desired relative states in this work are chosen to be 5 m in each direction of LVLH frame. Each satellite is equipped with three thrusters of 3 mN actuation limits in each direction of LVLH frame. We assume in this work that the thrusters are aligned perfectly with body axes and that they do not induce any torque on the body. Of course, this may not be a realistic assumption and, in practice, reaction wheels and low-level controllers may be employed to counteract any induced rotation (this is outside the scope of this paper). The robust optimal actuation force, as predicted by the TO here is applicable to other thruster configurations as well, namely- coupling thrusters and reaction wheels/magnetometers through a lower-level transformation controller. The results presented here can be directly adopted for practical SFF missions with a swarm size in the order of hundreds.

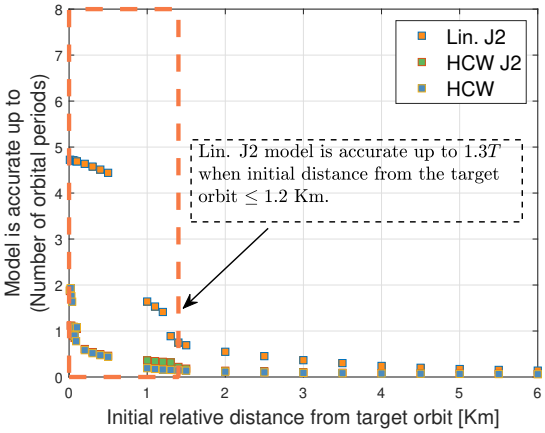
III. Uncertainty Quantification and Modeling Errors

In this section, we compare various linear dynamical models presented above and evaluate the modeling errors with varying initial conditions. Let us denote the relative position vector of a satellite j by $\bar{\boldsymbol{r}}_j^{\mathrm{NL}}$, $\bar{\boldsymbol{r}}_j^{\mathrm{L}}$, $\bar{\boldsymbol{r}}_j^{\mathrm{H}}$, $\bar{\boldsymbol{r}}_j^{\mathrm{H2}}$ respectively for nonlinear dynamics (1)–(3), linearized J2 model (11)–(13), HCW and modified HCW model (14)–(16). We are interested to determine the time instant when the modeling errors go beyond a realistic position tolerance of 5 m [18]. For a given initial condition, we consider a model to be accurate till the time it is within this threshold value. This analysis helps us to determine which model should be used in solving TP problem for a given time window and initial relative position of satellites.

By solving (5)–(10), we obtain the timeseries position, velocity and target orbit elements under J2 perturbation and air drag. Using this target orbit data, we simulate the satellite trajectories in local frame for linearized J2 in (11) – (13) and HCW model (with and without J2) in (14) – (16) respectively. For a given initial position and simulation



(a) Maximum modeling error of the three models - Linearized J2, HCW with and without J2 relative to high fidelity nonlinear model (1) – (3) for initial relative distance (LVLH frame) $[0.01, 10]$ km and simulation time length $5T$



(b) Accuracy of the three models - Linearized J2, HCW with and without J2 relative to the high fidelity nonlinear model (1) – (3) in terms of the length of time elapsed (time duration in which modeling error ≤ 5 m) for initial relative distance (LVLH frame) $[0.01, 6]$ km

Fig. 1 Quantification of modeling errors for various initial distances and identification of accurate model

time window, the modeling error of the linearized J2 model $\boldsymbol{e}_i^{\mathrm{L}} = \max_{\forall k \in [0,K]} \|\bar{\boldsymbol{r}}_i^{\mathrm{NL}}(k) - \bar{\boldsymbol{r}}_i^{\mathrm{L}}(k)\|_2$ is evaluated as the maximum relative distance between the trajectories generated by linearized J2 model (11) – (13) and the nonlinear model (1) – (3) for the entire time. Modeling error for HCW with ($\boldsymbol{e}_i^{\mathrm{H2}} = \max_{\forall k \in [0,K]} \|\bar{\boldsymbol{r}}_i^{\mathrm{NL}}(k) - \bar{\boldsymbol{r}}_i^{\mathrm{H2}}(k)\|_2$) or without J2 ($\boldsymbol{e}_i^{\mathrm{H}} = \max_{\forall k \in [0,K]} \|\bar{\boldsymbol{r}}_i^{\mathrm{NL}}(k) - \bar{\boldsymbol{r}}_i^{\mathrm{H}}(k)\|_2$) are analogously computed with respect to the nonlinear model. For this simulation based study, we consider four different actuation profiles with amplitudes between $[-3, 3]$ mN such as (1) periodic (sinusoidal) actuation with period $0.5T$, (2) arbitrary pulse actuation with overall impulse limited to 100 N s, (3) aperiodic actuation and (4) zero actuation. These actuation profiles cover a wide range of traditional control inputs applicable for LEO SFF missions [36, 37].

In Fig. 1(a), we present the maximum modeling errors of the three linear models under four different actuation profiles, noted above, for a simulation time-length of $5T$ and initial positions drawn randomly from a uniform distribution. Based on the SFF mission requirements, we consider that these initial points are located at a distance $[0.01, 15]$ km from the origin in LVLH frame. Corresponding to a specific initial distance from the origin, we randomly pick 150 initial positions, evaluate the maximum modeling errors $\boldsymbol{e}_{i,\max}^{\mathrm{L}}$ over all these initial positions and replicate this approach for other initial relative distances and models. From Fig. 1(a), we observe that the linearized J2 model is more accurate as compared to the other models under considered actuation scenarios.

Next, we evaluate the duration in which all three models yield less than 5 m modeling error under aforementioned actuation signals. In addition to the modeling errors, we also evaluate the corresponding time instant when this model error exceeds beyond 5 m. The simulation results showing the accuracy of all three models are presented in Fig. 1(b) from which we observe again that, the linearized J2 model is more accurate than the rest. More precisely, when the

initial distance from the target orbit is within 1.5 km, linearized J2 model error is within 5 m for more than one orbital period. Therefore, by considering both the modeling error and accuracy time horizon, linearized J2 model is the most appropriate candidate to formulate the relative dynamics of satellite swarms in the TP problem. For practical mission purposes, where the terminal time is more than one orbital period T, we would have to restart the TO algorithm in every T duration interval and without any loss of generality, we thus consider $t_f = T$ through the rest of the discussions. In the upcoming section, we present all the basic building blocks of a TP problem.

IV. Formulation of the trajectory planning (TP) problem

In this section, we first present the formulation of a centralized, fully-coupled TP problem that simultaneously includes a TO algorithm, PA, CA constraints in addition to the satellite dynamics and actuation constraints. This formulation and its accompanying numerical simulation will serve as a motivating example and a baseline for comparison. As we show, this optimization problem is not scalable to large swarms. We then present our proposed formulations, namely TP1 and TP2.

As noted earlier, for the SFF mission under study, we specifically focus on moving the satellite swarm from one PCO configuration to another with the formation center defining the target orbit, as shown in Fig. 2. The objective of this formation reconfiguration problem is to determine the fuel-optimal collision-free maneuvers of N participating satellites as the swarm moves from one formation to another in a specified duration. The core of this optimization problem is to select state variables $\boldsymbol{x}_i(k) = \left[\bar{\boldsymbol{r}}_i^{\mathrm{T}}(k), \dot{\bar{\boldsymbol{r}}}_i^{\mathrm{T}}(k)\right]^{\mathrm{T}}$, corresponding control $u_i(k) = \left[u_{ix}^{\mathrm{T}}(k), u_{iy}^{\mathrm{T}}(k), u_{iz}^{\mathrm{T}}(k)\right]^{\mathrm{T}}$ for each spacecraft $i = \{1, 2, \cdots, N\}$ satisfying Euler-discretized version of linearized J2 dynamics (11) – (13) with the form

$$\textbf{state space constraints: } \boldsymbol{x}_i(k+1) = A(k)\boldsymbol{x}_i(k) + Bu_i(k), k = 0, 1, \cdots, K-1, \tag{17}$$

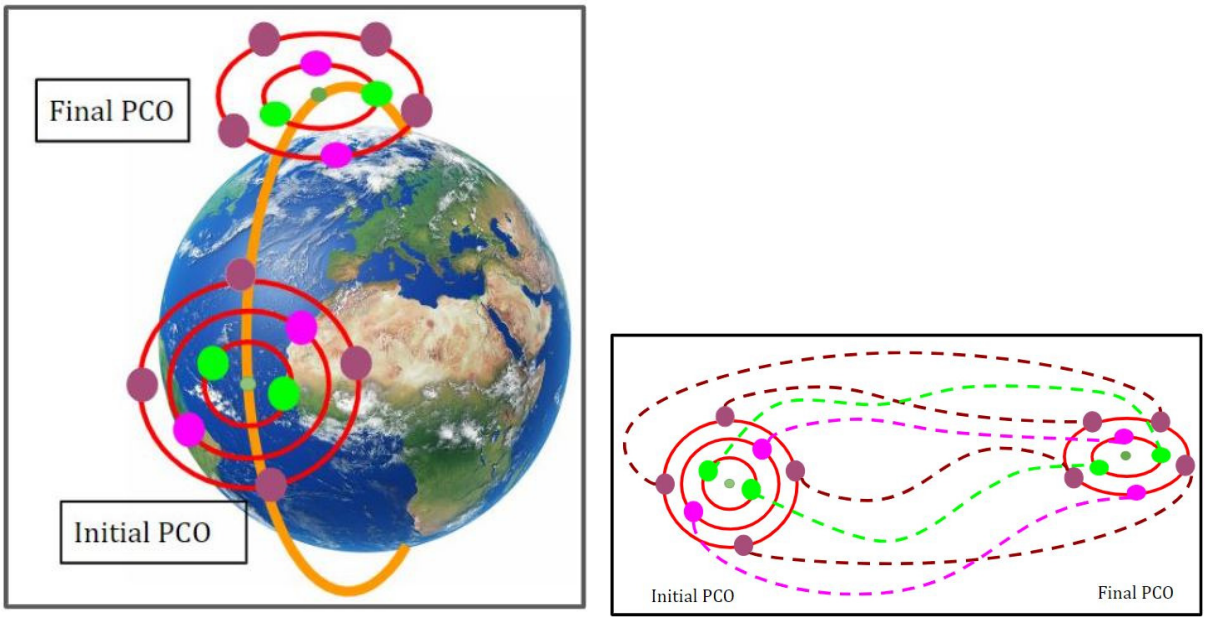
$$\textbf{boundary constraints: } \boldsymbol{x}_i(0) = \boldsymbol{x}_{iS}, \boldsymbol{x}_i(K) = \boldsymbol{x}_{iT}, \tag{18}$$

where $\boldsymbol{x}_{iS}$ and $\boldsymbol{x}_{iT}$ are respectively the initial and terminal state vectors for i^{th} spacecraft with terminal time-step K. Similar to [18, 22], we consider that inputs vectors with their respective slew rate limits,

$$\textbf{actuation constraints: } -u_{m,max} \leq u_{im}(k) \leq u_{m,max}, \tag{19}$$

$$\textbf{slew rate constraints: } u^r_{m,min} \leq u_{im}(k+1) - u_{im}(k) \leq u^r_{m,max}, \tag{20}$$

for $m = 1, 2, 3$, where $u_{im}(k)$ denotes m^{th} component of $u_i(k)$ with its rate limit being bounded within $[u^r_{m,min}, u^r_{m,max}]$. In contrast to some traditional TP problems with well known final configuration (e.g., [12]), we consider that the satellites carry identical hardware and, as such, the assignment of the final configuration to each satellite is not known *a priori*. Assignment of the final configuration (also called the Path Assignment, or PA problem) is formulated using



(a) PCO formation reconfiguration of satellite swarm

(b) Satellite Paths

Fig. 2 A visualization of swarm reconfiguration problem

mixed-integer linear constraints, given as follows

$$\textbf{PA constraints: } \boldsymbol{x}_{iT} = \sum_{j=1}^{N} b_{ij}\boldsymbol{x}_{jD}, \sum_{i=1}^{N} b_{ij} = 1, \sum_{j=1}^{N} b_{ij} = 1, b_{ij} \in \{0,1\}, \tag{21}$$

where $\boldsymbol{x}_{iD},\ i = 1, 2, \cdots, N$ denotes all possible terminal state vectors for a satellite to occupy in the final formation, the unity row sum and column sum of the assignment matrix $\mathbf{B} = \left[b_{ij}\right]_{N\times N}$ ensures that each of these terminal state vectors $\boldsymbol{x}_{iD},\ i = 1, 2, \cdots, N$ is assigned to only one of the satellites. Furthermore, at all times, during the maneuver, the satellites should also respect inter-satellite safety distance R_{safe} to avoid collisions with each other, *i.e.*

$$\textbf{CA constraints: } \|C_1(\boldsymbol{x}_i(k) - \boldsymbol{x}_j(k))\|_2 \geq R_{\text{safe}}, C_1 = \begin{bmatrix} I_3 & 0_{3\times 1} \end{bmatrix},\ j \neq i,\ i = 1, 2, \cdots, N. \tag{22}$$

A. PATOCA problem:

The objective of the coupled PATOCA problem is to minimize the total fuel consumption of all N satellites over the entire time horizon $[0, T]$ discretized into $K + 1$ time-steps as follows:

$$\min_{u,\boldsymbol{x},\mathbf{B}} \sum_{i=1}^{N}\sum_{k=0}^{K-1}\sum_{m=1}^{3} |u_{im}(k)| \text{ subject to (17)} - \text{(22)}.$$

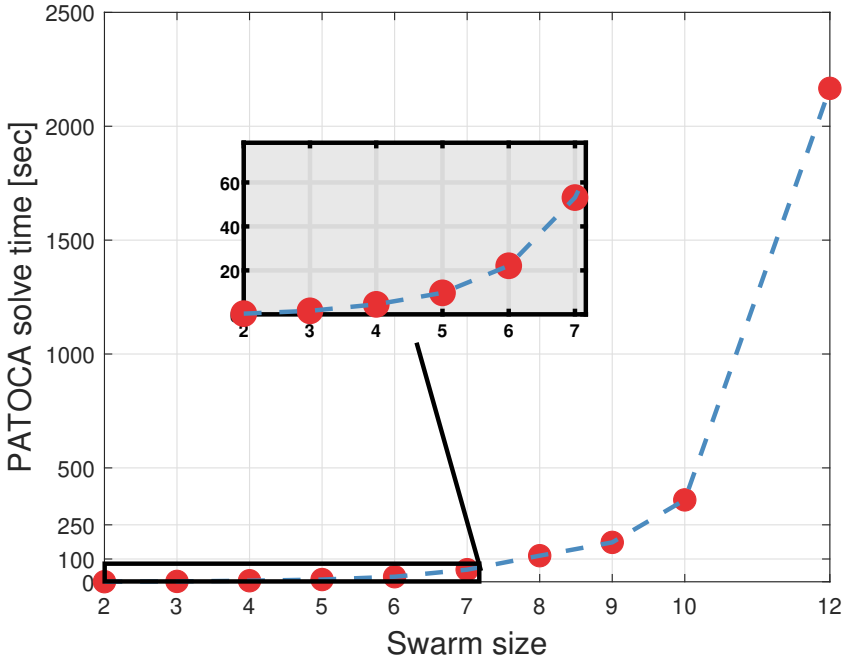


Fig. 3 Computation-time complexity of PATOCA with MILP implementation.

Since the consumption of fuel is directly correlated with the actuation effort of thrusters, minimizing the fuel consumption is therefore equivalent to minimizing the total control input by the thrusters.

As noted in [12, 15], since the CA constraints in (22) are in concave form, they cannot be immediately adopted for an efficient computation of satellite trajectories. Instead, in [38], a Mixed Integer Linear Program (MILP) formulation was proposed as an effective alternative to the Nonlinear Program (NLP), as it reduces the computation solve time compared to NLP with no apparent loss of fuel-optimality. Let $\bar{\boldsymbol{r}}_i$, $\bar{\boldsymbol{r}}_j$ denote the position vector of spacecrafts i and j with $\bar{\boldsymbol{r}}_i = \left[x_i, y_i, z_i\right]^{\mathrm{T}}$ and the safety distances between them in (x, y, z) directions of LVLH frame be respectively $R_{x,\text{safe}}, R_{y,\text{safe}}$ and $R_{z,\text{safe}}$. The objective of the TP problem with MILP-based PATOCA formulation then takes the form

$$\min_{u,\boldsymbol{x},\mathbf{B}} \sum_{i=1}^{N} \sum_{k=0}^{K-1} \sum_{m=1}^{3} |u_{im}(k)| \text{ subject to (17) – (21) and}$$

$$\textbf{MILP CA constraints:} \begin{cases} x_i(k) - x_j(k) \;\geq\; R_{x,\text{safe}} - Mb_1,\; x_j(k) - x_i(k) \geq\; R_{x,\text{safe}} - Mb_2, \\ y_i(k) - y_j(k) \;\geq\; R_{y,\text{safe}} - Mb_3,\; y_j(k) - y_i(k) \geq\; R_{y,\text{safe}} - Mb_4, \\ z_i(k) - z_j(k) \;\geq\; R_{z,\text{safe}} - Mb_5,\; z_j(k) - z_i(k) \geq\; R_{z,\text{safe}} - Mb_6, \\ \sum_{i=1}^{6} b_i \;\leq 5,\; b_i = \{0, 1\}, \end{cases} \tag{23}$$

where $M > 0$ is a sufficiently large number. For an increasing swarm size N, such MILP formulation [22, 38] yields

$O(e^N)$ computation-time complexity, as shown in Figure 3 for a PATOCA problem. Solving this swarm reconfiguration problem for 10 satellites in a standard windows computer (3.8 GHz CPU, 16 GB memory) takes nearly 5 minutes using a mixed-integer solver such as Gurobi [39]. Using a state-of-the-art onboard satellite computer such as a CPU or FPGA (which would not be dedicated to only the PATOCA computations), the computational complexity in solving this problem is prohibitive [40]. Due to this, the usefulness of this MILP formulation thus remains limited to a small swarm size (< 5 satellites, for example).

To reduce the computational complexities of the PATOCA problem, we decouple the TP problem into sub-components: (1) decoupled PA problem and (2) decoupled TO problem. In this work, we present two different approaches to formulate CA constraints and solve the TP problem: one approach, which is inspired by [15, 16], embeds the CA constraints into the decoupled TO problem; we call this formulation "TP1". In the second approach, we embed the CA constraints into the decoupled PA problem; we call this formulation "TP2". We now present a brief overview of two proposed approaches in the upcoming Section.

B. Overview of the proposed TP1 and TP2 solutions

As mentioned above, the fundamental difference between the two approaches lie on how CA constraints are formulated into the problem. In TP1, satellites cooperatively determine collision-free paths to reach to a fuel-optimal final configuration. This is done by embedding CA constraints within the TO problem (thus the TO is actually a "TOCA"). On the other hand, in TP2, satellites select a final configuration based on collision-free trajectories. In this method, CA constraints are embedded within PA problem (thus, "PACA"). A schematic block-diagram illustrating the proposed TP approaches is presented in Figure 4 with step 1 and 4 being common to both of these approaches. In step 1, the satellites compute, in a fully decentralized way (*i.e.*, using local computing resources), the minimum fuel required to go from their current location to each of the locations in the final configuration.

In TP1, the satellites then solve, in a centralized manner (e.g., one satellite performing the computation), the fuel-optimal final assignment of satellites by solving a PA problem, to be discussed in Section V.A. Using these terminal states as input parameters, we then solve a distributed TOCA problem with each satellite sharing its updated location information with other neighboring satellites to cooperatively determine collision-free feasible paths. As shown in Figure 4, TP1 thus consists of a centralized PA and distributed TOCA problem. Although TP1 distributes the processing and computation load across the swarm, it suffers from extensive communication overhead. To avoid the problems associated with such communication requirements in TP1, we present an alternative TP algorithm (TP2) for swarm reconfiguration problem in which CA constraints are augmented with PA problem (PACA).

The objective of the PACA problem in TP2 is to find a final configuration that does not require a satellite to intrude into a sphere of safety radius R_{safe} of the neighboring satellites as they steer towards the desired configuration. Once such terminal states are obtained, each satellite then solves a TO algorithm as in Section IV.C in its local computer to

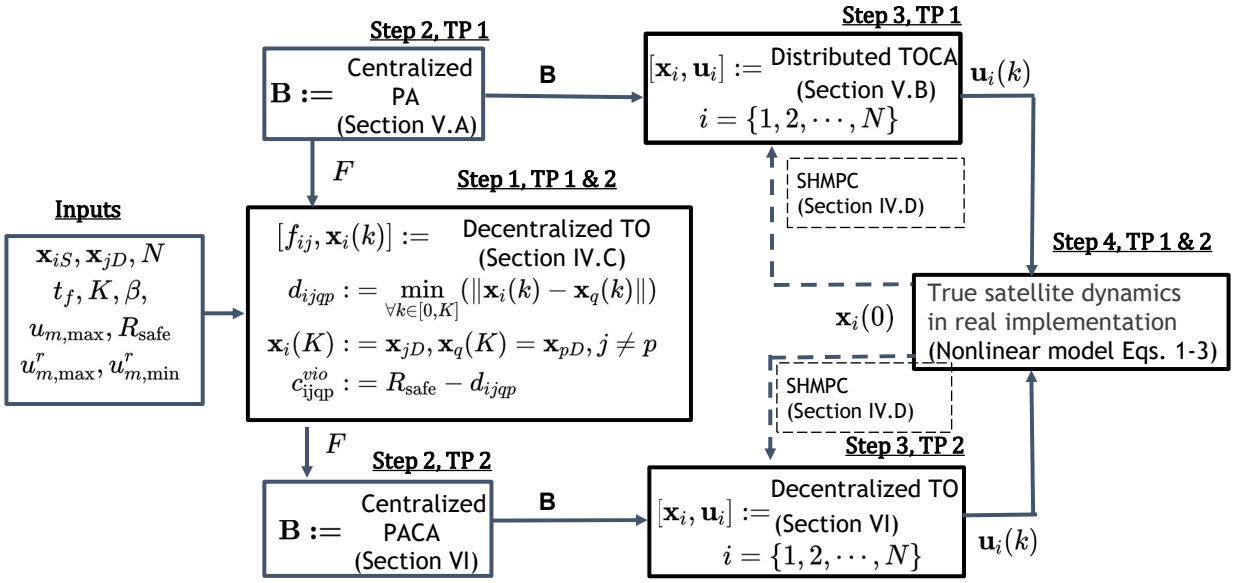


Fig. 4 Schematic diagram illustrating both TP approaches. Steps 1 and 4 are common between the two; steps 2 and 3 are different. TO, TP, CA, and SHMPC stand for Trajectory Optimization, Trajectory Planning, Collision Avoidance, and Shrinking Horizon Model Predictive Control, respectively.

compute the trajectories. Therefore, TP2 is required to solve a centralized PACA and N decentralized TO problems. Since each agent runs a decentralized TO independently to compute optimal satellite paths, TP2 approach overcomes the expensive communication requirement of TP1.

Finally, to robustify against any modeling uncertainties and disturbances, the satellites redo the computations repeatedly throughout the reconfiguration task in a shrinking horizon model predictive control (SHMPC) fashion, as presented in section IV.D.

We now present steps 1 and 4, which are identical in both TP approaches.

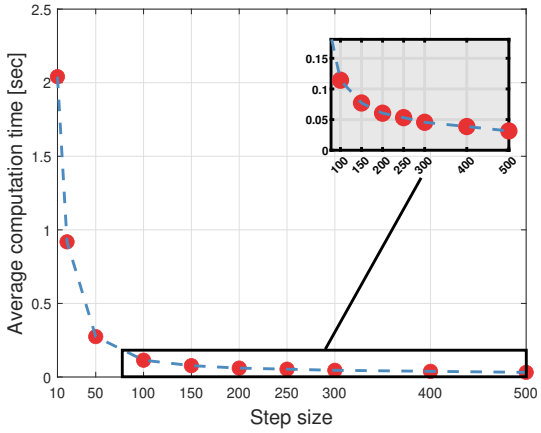
C. Fuel matrix calculation: Step 1

To derive the fuel matrix (*i.e.*, F shown as the output of Step 1 in Fig. 4) for a given initial and final formation pair, we solve a TO problem. Specifically, let f_{ij} denote the fuel cost of satellite i to move from location i in initial formation to j in the final formation:

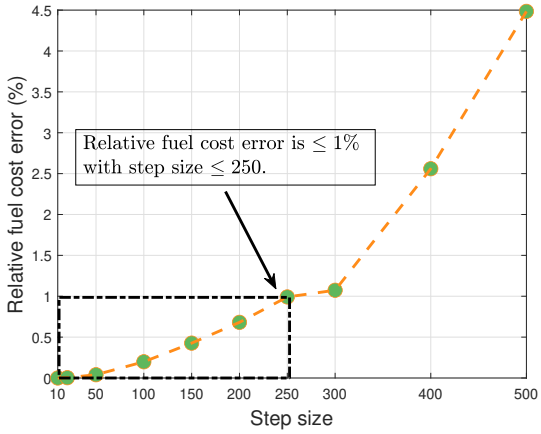
$$\boldsymbol{x}_i(0) = \boldsymbol{x}_{iS},\ \boldsymbol{x}_i(K) = \boldsymbol{x}_{jD}. \tag{24}$$

Each satellite i, in a decentralized manner, evaluates the fuel cost to move from its current location to each of the final locations in the formation by solving the TO problem:

$$\textbf{TO problem to find } F: f_{ij} = \min_{u} \sum_{k=0}^{K-1} \sum_{m=1}^{3} |u_{im}(k)| \text{ subject to (17),(19),(20),(24),} \tag{25}$$



(a) Time complexity of evaluating f_{ij}, $j = 1, 2, \cdots, N$ with swarm size $N = 100$ by satellite i for varying step sizes $\in [1, 500]$ second. Satellite i solves N decentralized LP problem (25) in parallel and computation time is the maximum solve time for evaluating f_{ij} over the entire swarm size N.



(b) From 1000 numerical simulations, we determine the loss of fuel optimality of an assignment corresponding to a higher step size $in [10, 500]$ second compared to that with step size 1 second

Fig. 5 Computation-time complexity of finding fuel cost matrix F and loss of fuel-optimality in selecting an assignment matrix with higher step size

for $j = 1, \ldots, N$. Thus, together, the satellites create the fuel matrix: $F = [f_{ij}]_{N\times N}, i = 1, 2, \cdots, N$ and $j = 1, 2, \cdots, N$. More than the absolute numerical value of f_{ij}, to pick an optimal assignment **B** in step 2 (see Fig. 4), what is more important is the relative comparison between f_{ij} and other entries in F (f_{kl} for example). Through extensive numerical simulations, we have observed that the correlation between the entries in F does not change much over the step size variation in TO (25) and thus the assignment matrix **B** remains identical in most cases. In cases when they are not identical, the resulting fuel-cost to arrive at a new terminal assignment causes a negligible optimality loss (less then 1% for simulation step size up to 250 s). With no loss of generality, to calculate fuel cost matrix F (by solving (25)) in a computationally efficient manner, we thus select a time-discretization step size at least 10 times higher (*e.g.*, 250 s step size) than that of other sub-components in TP. Using a Gurobi solver in a standard Windows computer (3.8 GHz CPU, 16 GB memory), evaluating such fuel cost matrix F with a step size of 250 s has a solve time of nearly 50 milliseconds as shown in Figure 5 while keeping the loss of fuel optimality limited to 1% relative to that with step size 1 second. Note that, in a real SFF mission, the satellites would perform this computation in parallel, so the solve time would be faster and. Thus, this computation is numerically tractable.

Steps 2 and 3 of TP1 and TP2 (Fig. 4) take F as an input to compute optimal actuation commands for the entire swarm. We will present the details of these steps in Sections V and VI. For now, we skip to step 4, which is common between the two formulations.

D. Shrinking Horizon Model Predictive Control (SHMPC): Step 4

The effects of unmodeled disturbances and modeling uncertainities need to be taken into account. If left unaccounted, such disturbances and ignored nonlinear effects will cause large trajectory errors and suboptimal actuation. In this work, we present basic SHMPC formulation to iteratively mitigate the effects of such uncertainties by evaluating a new control input over a prediction horizon based on the current state information. For a fixed final formation and time, SHMPC successively computes an optimal actuation command which can steer the actual satellite from its current location to its final location at t_f. With the update of new state information, the prediction horizon window shrinks as the start time of the TP algorithm gradually approaches to t_f. This method of successive prediction and reoptimizing the trajectory planner with more accurate initial positions thus yields a more robust control law that eventually renders a small trajectory prediction error in the face of modeling uncertainties and ignored perturbations.

Algorithm 1: Finding robust control solution to trajectory planning algorithm

Result: Robust control sequence $u_R^*(k),\ k \in [0, K-1]$

Input: $\boldsymbol{x}_S, \boldsymbol{x}_D, u_{\max}, u_{min}, u^r_{\min}, u^r_{\max}, t_f = T$ with the interval $[0, t_f]$ discretized into $K+1$ time-steps in $[0, K]$, number of prediction windows $\beta > 0$, prediction interval $K_1 = \dfrac{K+1}{\beta}$.

Initialize $c = 1$, $\alpha = 0$, $\boldsymbol{x}(0) = \boldsymbol{x}_S, \boldsymbol{x}(K) = \boldsymbol{x}_D$.

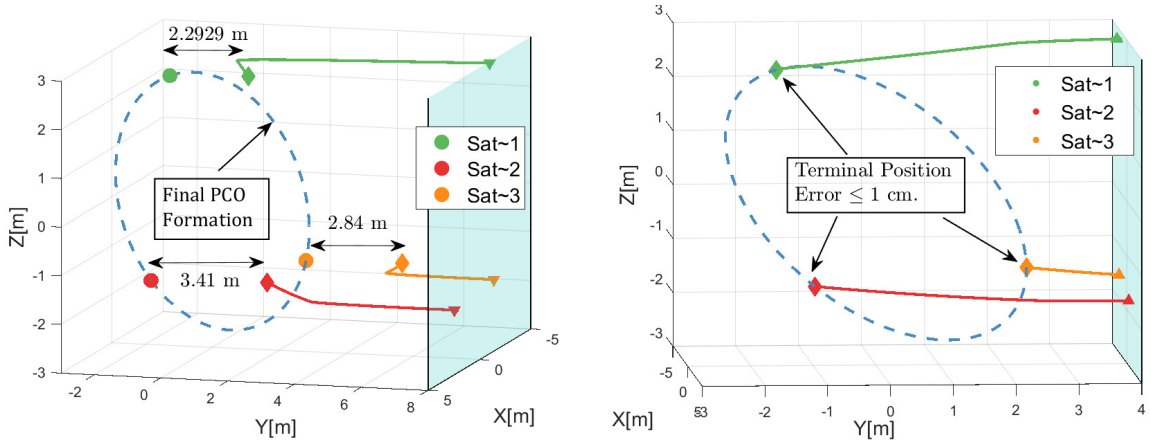
do

- $[\boldsymbol{x}^*, u^*]$ = solution to TO algorithm in step 3 of TP1/TP2 for the prediction horizon $[\alpha, K]$.
- $[\boldsymbol{x}]$=Apply u^* on the actual satellite model in (1)-(3) with $\boldsymbol{x}(0) = \boldsymbol{x}_S$ in interval $[\alpha, cK_1 - 1]$.
- $\boldsymbol{x}_S = \boldsymbol{x}(cK_1 - 1)$.
- $u_R^*(k) = u^*(k),\ k \in [\alpha, cK_1 - 2]$.
- $\alpha = cK_1 - 1$.
- $c = c + 1$

while $c \leq \beta$;

return $u_R^*(k),\ k \in [0, K-1]$.

For numerical simulations, we assume that the actual LEO satellite motion under J2 perturbation and air-drag is mimicked by the high-fidelity nonlinear dynamics in (1) – (3). Using this nonlinear model, we determine the actual position and velocity vector of a satellite under predicted control law u^* at the end of a prediction horizon. Let us assume that the interval $[0, t_f]$ is discretized into $K+1$ samples between $[0, K]$ with $\boldsymbol{x}(0) = \boldsymbol{x}_S, \boldsymbol{x}(K) = \boldsymbol{x}_D$. We choose an integer β representing the number of times the optimization is to be re-run during the maneuver, which gives the prediction interval $K_1 = \dfrac{K+1}{\beta}$. Consider also positive scalar α, initialized to 0, and integer c, initialized to 1. From the TO part in step 3 of TP2 (or TOCA in TP1), in the first round of prediction ($c = 1$) we compute an optimal control sequence $u^*(k),\ k \in [\alpha, K-1]$ with initial and terminal state vectors being respectively $\boldsymbol{x}(\alpha)$ and $\boldsymbol{x}(K)$. Then, similar to a traditional MPC technique [41], by applying the predicted control sequence $u^*(k)$ for a small time window with interval $k \in [\alpha, cK_1 - 2]$, we evaluate the actual state measurements of the satellite from Eqs. (1) – (3) at discrete time instant $cK_1 - 1$. The scalar α is now reinitialized at $cK_1 - 1$. Using the updated state vector $\boldsymbol{x}(cK_1 - 1)$ as the initial state vector for the second round of prediction ($c = 2$) in the interval $[\alpha, K]$, we determine a new optimal control



(a) Trajectory prediction error Without SHMPC

(b) Trajectory prediction error with SHMPC

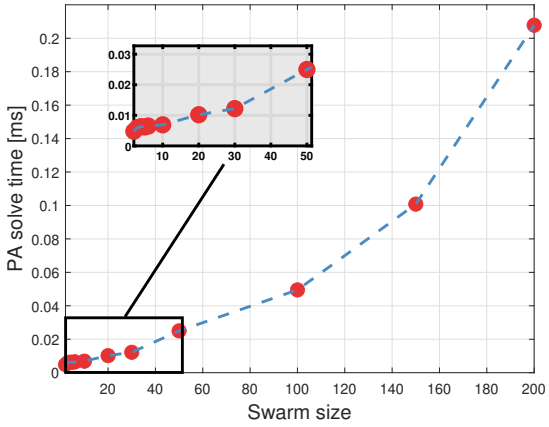
Fig. 6 Trajectory prediction errors with and without SHMPC (desired terminal locations of the three satellites are marked in circles while the actual locations are marked in diamonds. For zooming in, actual satellite trajectories in both the plots are truncated. Under SHMPC, trajectory prediction errors reduce to a cm precision level.)

sequence $u^*(k),\ k \in [\alpha, cK_1 - 2]$ which is then applied in open-loop to the actual satellite motion in (1) – (3) to find a new actual state measurement and this procedure is repeated β times. The more frequent the optimization happens (*i.e.* higher the β), the smaller would be the trajectory prediction error. This algorithm is summarized in Algorithm 1.

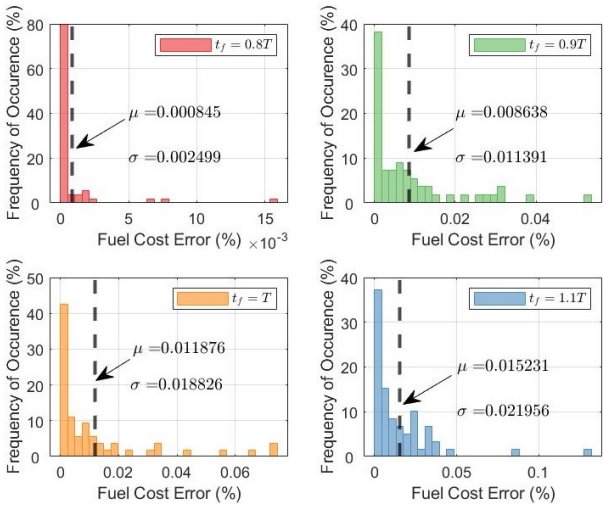
The effectiveness of the SHMPC algorithm is now illustrated with a numerical example. Let us consider a swarm of three satellites, located within 1.5 km initial distance from the origin in LVLH frame, move to a PCO of radius 3 m over one orbital period (*i.e.* $t_f = T$) under maximum actuation of 3 mN and slew rate 1 mN s^{-1}. As shown in Figure 6(a), the predicted control inputs when applied in open-loop to the actual satellite dynamics (1) – (3), yields trajectory prediction error within $[2.92, 3.41]$ m for three satellites. With the help of SHMPC implementation in Algorithm 1, considering 10 prediction windows with window length 10 min, terminal positions of the satellites coincide with the desired locations on final PCO formation shown in Figure 6(b), and the trajectory prediction error is bounded within 1 cm. In the upcoming sections V and VI, we present the remaining steps of both TP approaches.

V. PA+TOCA: Steps 2 and 3 of TP1

In this section, we formally present TP1 which is decoupled into a PA problem and a TOCA problem, which are discussed below.



(a) Time complexity of minimum-fuel PA problem



(b) From 2000 randomized simulations, we determine the histograms of the relative fuel cost error between coupled and decoupled PATO under minimum-fuel PA formulation with mean $\mu \leq 0.025$ and standard deviation $\sigma \leq 0.025$ for $t_f \in \left[0.8T, 1.1T\right]$ and swarm size $\in \left[1, 100\right]$

Fig. 7 Computation-time complexity of minimum-fuel PA and loss of fuel-optimality of decoupled PATO relative to coupled PATO problem with minimum-fuel PA formulation

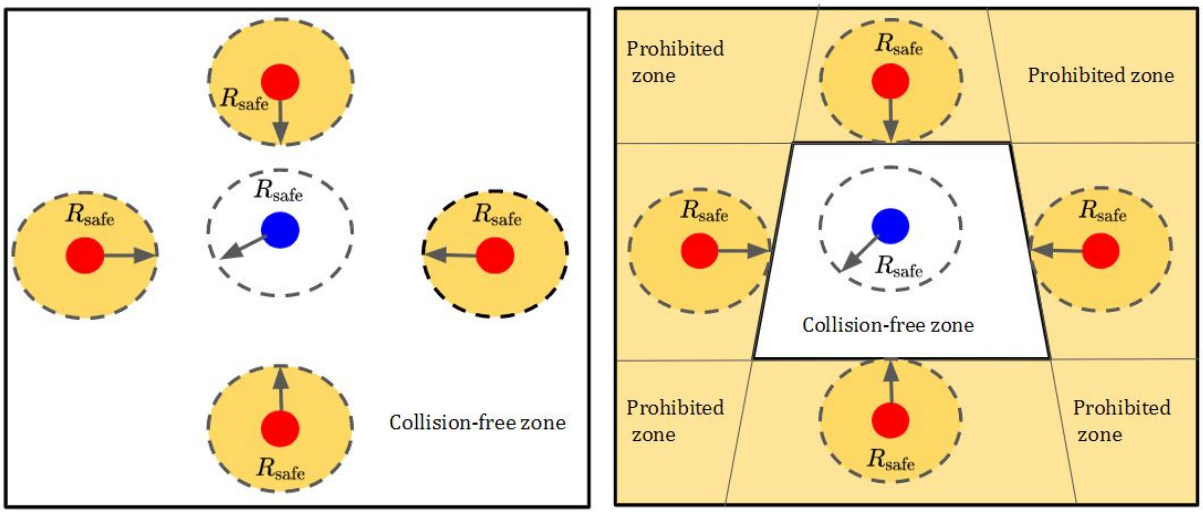
A. PA problem formulation

To solve the PA problem in TP1, we determine a final configuration of satellites that consumes the least amount of fuel compared to the other possible configurations.

$$\textbf{PA problem :} \min_{\mathbf{B}} \sum_{i=1}^{N}\sum_{j=1}^{N} f_{ij}b_{ij} \;\text{ subject to PA constraints in (21),} \tag{26}$$

where f_{ij} are input parameters in this optimization that are calculated by the satellites in a parallel fashion, as described previously (Eq. (25)). The expression $\sum_{i=1}^{N}\sum_{j=1}^{N} f_{ij}b_{ij}$ denotes the fuel cost associated with each configuration. The above optimization problem is solved by a centralized coordinator, for example one of the satellites, which obtains f_{ij} from the other satellites, solves the above optimization problem, and disseminates the terminal location information back to the individual satellites. The resulting configuration is fuel-optimal, though it may not necessarily satisfy collision avoidance constraints, which is treated in the TOCA layer, discussed in the next subsection.

In contrast with the works of [30] where a distributed auction algorithm was presented to solve PA, the minimum-fuel PA formulation in (26) replaces the communication requirement of [30] between all local neighbors with a minimal spanning tree assumption. This centralized PA formulation provides a simple and computationally efficient method of finding an optimal assignment of the satellites in the final formation with the computation time-complexity shown in Figure 7(a). Furthermore, in comparison to the estimated fuel cost in coupled PATO problem, as seen from Figure 7(b), there is no apparent loss of fuel-optimality by the decoupled PATO problem with minimum-fuel PA formulation in (26).



(a) Non-convex CA region

(b) Convexified CA region with affine constraints in (27)

Fig. 8 Collision safe and prohibited zone for a satellite (in blue) surrounded by four satellites in neighborhood (in red) are depicted by non-convex (22) **and convexified CA constraints** (27)**. Prohibited zone for blue satellite at the current instant is marked in light yellow and the safe region in white.**

The results presented here can be extended to other constellation SFF missions with multiple target orbits and a cluster of satellites around each of them. In that case, one of the satellites corresponding to each target orbits serve as a central processor for each local clusters.

B. TOCA problem formulation

A TO problem with non-convex CA constraints (22) is NP-hard, and thus, computationally challenging [17]. Using such non-convex CA formulation, collision-safe zone is presented in Figure 8(a) for a satellite (in blue) when it's surrounded by four neighboring satellites (in red). As noted earlier, even when the non-convex CA constraints are reformulated as an MILP, it cannot readily be scaled up for large SFF problems. Thus, the non-convex CA constraints in (22) are reformulated into a set of affine constraints, shown in Figure 8(b), that provide a conservative approximation as

$$\textbf{Affine CA constraints:} \quad \left(\bar{\boldsymbol{x}}_i(k)-\bar{\boldsymbol{x}}_j(k)\right)^{\mathrm{T}} C_1^{\mathrm{T}} C_1 \left(\boldsymbol{x}_i(k)-\boldsymbol{x}_j(k)\right) \geq R_{\text{safe}} \| C_1 \left(\bar{\boldsymbol{x}}_i(k)-\bar{\boldsymbol{x}}_j(k)\right) \|_2, \;\; j > i, \tag{27}$$

where the condition $j > i$ ensures that only one of the spacecraft will try to avoid the other one, $\bar{\boldsymbol{x}}_i, \; i = 1, 2, \cdots, N$ is an initial guess of the optimal trajectory of i^{th} spacecraft.

The closer the selection of this initial guess to the actual trajectory, the more accurate will be the solutions to the convexified TOCA. An initial guess is generated by first solving a convex TO problem without CA constraints and then this initial guess is passed on as nominal trajectory $\bar{\boldsymbol{x}}_i$ for subsequent iterations of convexified TOCA algorithm. In iteration $\mathbf{m}$ of this TOCA algorithm, the solution to the previous iteration is used as a nominal trajectory $\bar{\boldsymbol{x}}_i(k) = \boldsymbol{x}_{i,\mathbf{m}-1}(k)$ for

formulating CA constraints in (27). These iterations will continue to give an updated nominal trajectory until it eventually converges to actual collision-free trajectories of each satellite, *i.e.* $\|\boldsymbol{x}_{i,\boldsymbol{m}}(k) - \boldsymbol{x}_{i,\boldsymbol{m}-1}(k)\|_2 < \epsilon, \forall i = 1, 2, \cdots, N$. This TOCA problem is solved in a distributed manner where nominal trajectories are communicated by spacecraft i to other neighboring satellites $j \in \mathcal{N}_i$ with $\|C(\boldsymbol{x}_i(k) - \boldsymbol{x}_j(k))\|_2 \leq R_{\text{comm}}$ where R_{comm} is the minimum communication distance between the satellites to exchange information. For convenience, let us take $R_{\text{comm}} = 1.2\ R_{\text{safe}}$. The TOCA problem is formulated as follows:

$$\textbf{TOCA problem: } \min_{u} \sum_{i=1}^{N} \sum_{k=0}^{K-1} \sum_{m=1}^{3} |u_{im}(k)| \text{ subject to (17)-(20), (27).} \tag{28}$$

Algorithm 2: Finding robust control solution to obtain collision-free fuel-optimal trajectories-TP1

Result: Robust fuel-optimal control sequence $u_i^*(k),\ k \in [0, K-1]$ and collision-free predicted trajectories $\boldsymbol{x}_i^*(k), i \in \{1, 2, \cdots, N\}, k \in [0, K]$

Input: $\boldsymbol{x}_{iS}, \boldsymbol{x}_{iD}$, $u_{i,\max}$,$u_{i,\min}$ $u^r_{i,\min}, u^r_{i,\max}$, t_f, β, K, R_{safe}, R_{comm}, ϵ.

f_{ij} := solve decentralized TO problem (25) in Section IV.C with $\boldsymbol{x}_i(0) = \boldsymbol{x}_{iS},\ \boldsymbol{x}_i(K) = \boldsymbol{x}_{jD},\ \forall j \in 1, 2, \cdots, N$.

$F := [f_{ij}]_{N\times N}$.

$\mathbf{B}$:= solve centralized PA problem (26) in Section V.

$\boldsymbol{x}_{iD} = b_{ij}\ \boldsymbol{x}_{jD},$ if $b_{ij} \neq 0$.

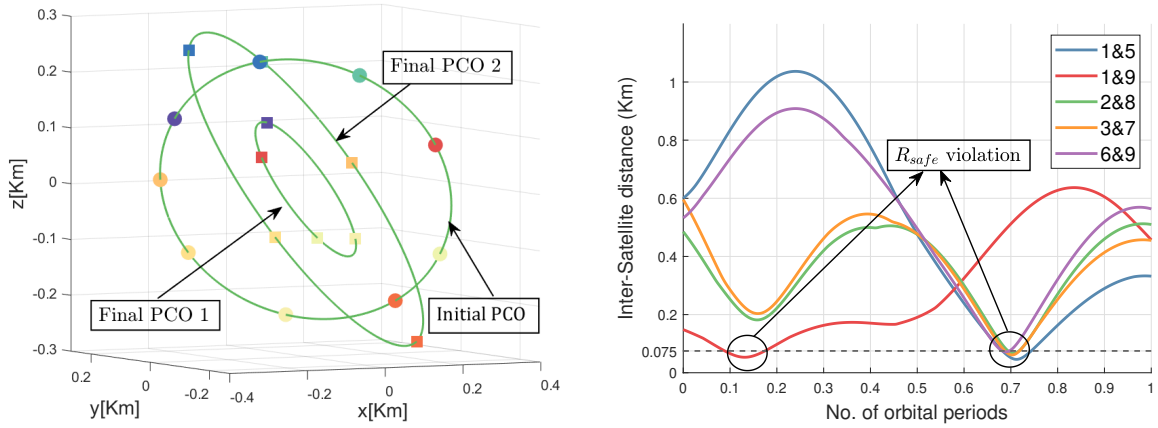
$[u_i^*, \boldsymbol{x}_i^*]$:= solve SHMPC in Algorithm 1 with SCP based distributed TOCA formulation, given $\boldsymbol{x}_{iS},\ \boldsymbol{x}_{iD}$, R_{safe}, R_{comm}, ϵ, and number of SHMPC prediction windows β.

return $u_i^*, \boldsymbol{x}_i^*, i \in \{1, 2, \cdots, N\}$.

A sequential convex programming (SCP) method for solving this distributed TOCA algorithm is given in Method 1 of [42]. The TP algorithm consisting of a centralized PA, distributed TOCA and SHMPC for robustification is summarized in Algorithm 2.

The effectiveness of the proposed TP algorithm is illustrated now with the help of a numerical example. The reference orbit is an SSO with Keplerian parameters $a = 6800$ km, $e = 0, i = 97°, \Omega = 10°, \theta = 0°$. The swarm reconfiguration problem requires a group of 9 satellites move from one PCO of radius 250 m in LVLH frame to another in $t_f = 1.07T$ under 3 mN maximum actuation in each direction while maintaining $R_{\text{safe}} = 75$ m. Out of 9 follower satellites, a set of 4 satellites move to a PCO with radius 150 m and the other 5 satellites to a PCO with radius 350 m. For this swarm reconfiguration problem, we consider the discretization time-step to be 20 s.

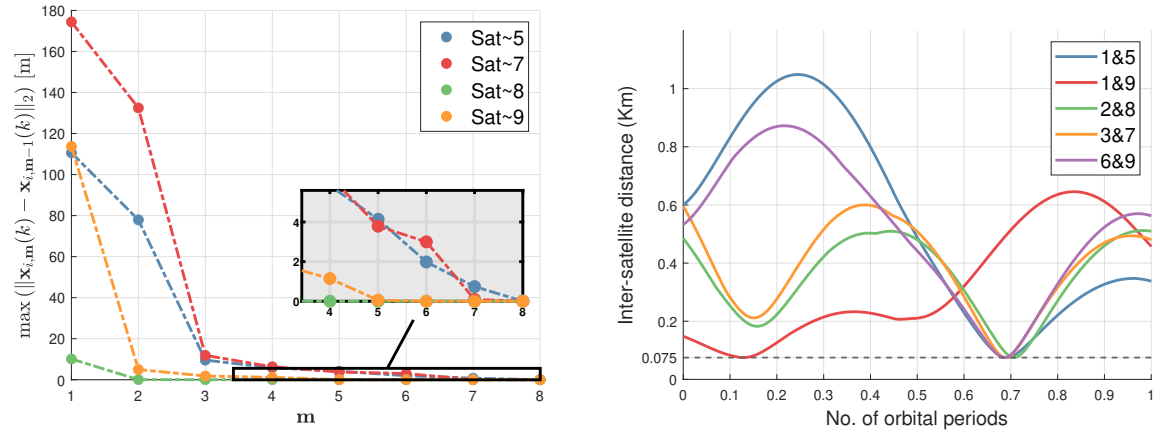
Given the initial and final locations of 9 follower satellites, by solving a centralized PA problem (26) we obtain fuel-optimal final configuration of satellite shown in Figure 9(a) where initial (marked in circles) and assigned final location (marked in squares) pairs are marked in same colors. Next, by solving a decentralized TO problem (without considering CA constraints), we observe that there are 10 instances where 5 pairs violate R_{safe} distance. By solving the presented TOCA problem with SCP, satellites 5, 7, 8 and 9 find collision-free paths from the feasible space with 1.1%, 1.4%, 0.01% and 8.1% additional consumption of fuel. The total number of SCP iterations required for TOCA solution



(a) Final assignment of satellites

(b) Inter-satellite distance without TOCA

Fig. 9 Swarm reconfiguration problem of 9 satellites



(a) $\max\left(\|\mathbf{x}_{i,\mathbf{m}}(k) - \mathbf{x}_{i,\mathbf{m}-1}(k)\|_2\right),\ k \in [0, K]$

(b) Inter-satellite distance with TOCA algorithm of TP1

Fig. 10 Convergence of SCP TOCA algorithm and inter-satellite distance under TOCA

to converge is 8 and the convergence of this SCP algorithm for satellites 5, 7, 8 and 9 with each passing iteration count $\mathbf{m} \in [1, 8]$ is shown in Figure 10(a). under the TOCA algorithm, these satellites now maintain required R_{safe} distance from one another, as seen from the Figure 10(b). Furthermore, the effects of unmodeled perturbations are mitigated with our SHMPC in Agorithm 2, the predicted trajectory error which was in the order of few cm in this specific example problem, now improves to a mm precision level. The approximate run-time of this SHMPC-SCP algorithm in Gurobi solver with 12 SHMPC prediction windows (8.33 minutes) is nearly 9.44 seconds (the distributed and decentralized calculations were performed in series on the computer). Therefore, this distributed TO approach efficiently solves the TP problem.

VI. PACA+TO: Steps 2 and 3 of TP2

In this section, we formally present TP2. Similar to the previous approach, we first solve decentralized TO problem in Section IV.C to evaluate fuel cost f_{ij}, fuel-optimal trajectories $\boldsymbol{x}_i(k)$, actuation $u_i(k)$ with $d_{ijqp} = \min\left(\|C_1\left(\boldsymbol{x}_i(k) - \boldsymbol{x}_q(k)\right)\|, \forall k \in [0, K]\right), \forall q > i, \forall j \neq p,\ i = 1, 2, \cdots, N,\ j = 1, 2, \cdots, N,\ \boldsymbol{x}_i(K) = \boldsymbol{x}_{jD},\ \boldsymbol{x}_q(K) = \boldsymbol{x}_{pD}$ as the satellites i and q occupy the final locations j and p. By using $d_{ijqp} > R_{\text{safe}}, \forall i, j$, PACA problem then finds a *collision-free assignment* $i \to j$ and $q \to p$; in other words, assignments with smallest fuel cost that simultaneously avoid collisions for the duration of the maneuver. The PACA problem can thus be formulated as:

$$\textbf{PACA problem:} \begin{cases} \min\limits_{\mathbf{B}} \sum\limits_{i=1}^{N}\sum\limits_{j=1}^{N} f_{ij} b_{ij} \\ \text{s.t. } \textbf{PA constraints: } \sum\limits_{i=1}^{N} b_{ij} = \sum\limits_{j=1}^{N} b_{ij} = 1, b_{ij} = \{0, 1\}, \\ \textbf{CA constraints: } \quad b_{ij} + b_{qp} \leq 1, \text{ if } d_{ijqp} < R_{\text{safe}}, \end{cases} \tag{29}$$

where CA constraints based on separating hyperplanes transform the prohibited assignment with suitable affine formulation. However, this PACA problem becomes infeasible if there exists no assignment that satisfies the CA constraints (29). To avoid this and also know accurately which satellite pair violates the inter-satellite distance R_{safe} during the maneuvers that attributes to this infeasibility, we soften the CA constraint and include an additional term in the objective function to penalize any large violations from the specified safety margin. Knowing the set of satellites that violate the R_{safe} distance, trajectory planner informs a lower-level controller to prioritize/focus on these unsafe pairs and apply appropriate control inputs to restore safety between them (as will be discussed later). Let us now introduce two four dimensional matrices $\mathbf{H}, \mathbf{C}^{\text{vio}}$ where $h_{ijqp} = \mathbf{H}[i, j, q, p], c^{\text{vio}}_{ijqp} = \mathbf{C}^{\text{vio}}[i, j, q, p]$ are entries at index (i, j, q, p).

The modified PACA problem with softened CA constraints can thus be rewritten as

$$\textbf{Modified PACA problem:} \begin{cases} \min\limits_{\mathbf{B},\mathbf{H}} \sum\limits_{i=1}^{N}\sum\limits_{j=1}^{N} f_{ij} b_{ij} + c^{\text{vio}}_{ijqp}\, h_{ijqp}, \forall q, p \in \{1, 2, \cdots, N\},\ q \neq i,\ p \neq j \\ \text{s.t. PA constraint in(29)}, \\ \textbf{CA constraints: } b_{ij} + b_{qp} \leq 1 + h_{ijqp}, \text{ if } c^{\text{vio}}_{ijqp} \geq 0, \\ h_{ijqp} \in \{0, 1\}, \end{cases} \tag{30}$$

where $c^{\text{vio}}_{ijqp} = R_{\text{safe}} - d_{ijqp}$ and h_{ijqp} is a binary variable that selects an assignment based on minimal violation from R_{safe}. With the resulting assignment $\mathbf{B} = [b_{ij}]_{N\times N}$, we finally solve the decentralized TO problem (now with a smaller discretization step):

$$\min_{\boldsymbol{u},\boldsymbol{x}} \sum_{k=0}^{K-1}\sum_{m=1}^{3} |u_{im}(k)| \text{ subject to (17) – (20)} \tag{31}$$

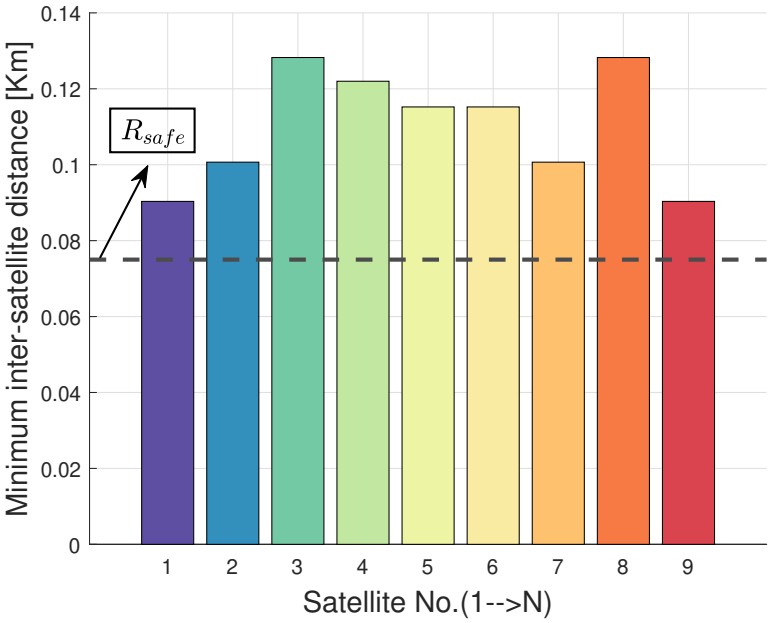


Fig. 11 Inter-satellite distance with PACA algorithm of TP2

to compute satellite trajectories $\boldsymbol{x}_i(k)$ and actuation $u_i(k)$, $\forall k$. Finally, with SHMPC, we robustify the solution to TP problem against modeling uncertainties, similar to the approach in Section V. However, SHMPC in TP2 iteratively runs TO problem, as opposed to TOCA problem in TP1. We now summarize TP2 approach in Algorithm 3.

To analyse the effectiveness of TP2, we now consider the same numerical example from Section V. Solving modified PACA problem yields a collision-free assignment and allows the satellites to always maintain inter-satellite safety distance $R_{\text{safe}} = 0.075$ km throughout the reconfiguration maneuver, as shown in Figure in 11. The computation-time for evaluating satellite trajectories is 2.187 s with Gurobi solver, *i.e.*, nearly $\frac{1}{5}$ times to the computation-time required by the TP1. This is expected because the slow convergence of SCP to arrive at the fuel-optimal solution in TP1 is replaced with the selection of feasible satellite paths that are guaranteed to be collision-free, thanks to an efficient PACA formulation of TP2. Finally, by using SHMPC, the trajectory prediction error is in the order of few mm.

It is to note that, if R_{safe} is now increased to 100 m, there is no feasible assignment to PACA problem (29). However, with modified PACA formulation (30), the TP problem can be turned feasible by changing the inter-satellite safety requirement of either satellite 1 or 9 from R_{safe} to $R'_{\text{safe}} = 0.0903\ R_{\text{safe}}$. However, for a large fleet of satellites in close proximity, maintaining specific R_{safe} inter-satellite distance is a hard constraint that is inviolable. The violation of CA constraints (27) also causes infeasibility of TOCA problem in TP1, which was addressed in [42] by placing additional constraints on satellite's maximum velocities or terminal time and locations. However, these limitations often lead to overwriting the mission specifications. To avoid this, therefore, we need some additional cooperative control efforts from the fleet to refine the baseline trajectories obtained from TP algorithms, presented in Sections V, VI, while adhering to

Algorithm 3: Finding robust control solution to guarantee collision-free near optimal satellite trajectories-TP2

Result: Robust fuel-optimal control sequence $u_i^*(k),\ k \in [0, K-1]$ and predicted trajectories $\boldsymbol{x}_i^*(k), i \in \{1, 2, \cdots, N\}, k \in [0, K]$.

Input: $\boldsymbol{x}_{iS}, \boldsymbol{x}_{iD}, u_{i,\min}, u_{i,\max}, u^r_{i,\min}, u^r_{i,\max}, t_f, K, \beta, R_{\text{safe}}$.

$[f_{ij}, \boldsymbol{x}_i]$:= solve decentralized TO in Section IV.C with $\boldsymbol{x}_i(0) = \boldsymbol{x}_{iS},\ \boldsymbol{x}_i(K) = \boldsymbol{x}_{jD},\ \forall j \in 1, 2, \cdots, N$.

$F := [f_{ij}]_{N\times N},\ d_{ijqp} := \min\left(\|C_1\left(\boldsymbol{x}_i(k) - \boldsymbol{x}_q(k)\right)\|, \forall k \in\ [0, K], q \neq i, p \neq j\right)$ with $\boldsymbol{x}_i(K) = \boldsymbol{x}_{jD},\ \boldsymbol{x}_q(K) = \boldsymbol{x}_{pD}, i = 1, 2, \cdots, N,\ j = 1, 2, \cdots, N$,

$C_{\text{vio}}[i, j, q, p] := R_{\text{safe}} - d_{ijqp}$.

B := solve centralized PACA problem (30) in Section VI.

$\boldsymbol{x}_{iD} = b_{ij}\,\boldsymbol{x}_{jD},$ if $b_{ij} \neq 0$.

$[u_i^*, \boldsymbol{x}_i^*]$:= solve SHMPC in Algorithm 1 on decentralized TO formulation (31) in Section IV.C, given $\boldsymbol{x}_{iS},\ \boldsymbol{x}_{iD}$, and number of SHMPC prediction windows β.

return $u_i^*, \boldsymbol{x}_i^*$.

the given mission specifications. We discuss this in the following remark.

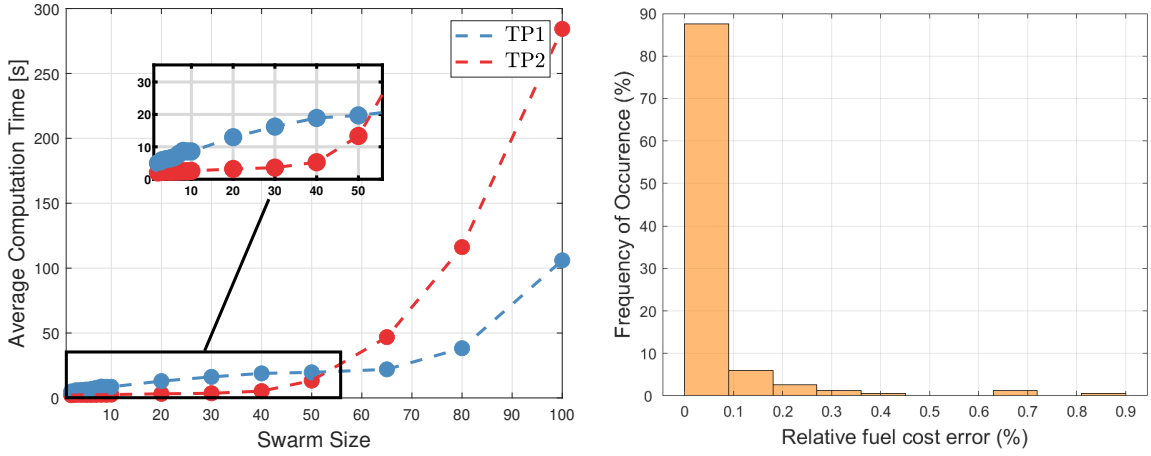
Remark 1 *The infeasibilities of TP*1 *and TP*2 *are mitigated by reducing R_{safe} of violating satellite pair to $\alpha_0 R_{safe},\ 0 < \alpha_0 \leq 1$ where α_0 selects the maximum feasible inter-satellite distance. However, in close proximity SFF missions, a small α_0 will give rise to a significant collision risk unless some corrective control actions are put in place by the satellites to fine tune its predicted baseline trajectories. An artificial potential function (APF) based adaptive control algorithm [43–45] can now be adopted to track the optimal baseline trajectory $\mathbf{x}_i^*(k)$, predicted by TP algorithms and provide additional CA guidance for cooperative satellites in maintaining original R_{safe} distance from its neighbors. Such cooperative control methods, when augmented with the TP algorithms in Section V and VI will allow each satellites to maintain the baseline trajectory through an attractive potential field when there are no neighboring satellites within the safety sphere and applies appropriate actuation through a repulsive potential field when there are. From the works of [46], the repulsive potential to avoid such intrusion for satellite i is given in this form*

$$V_{ij} = \left[\min\left(0, \frac{\|C(\boldsymbol{x}_i^* - \boldsymbol{x}_j^*)\|^2 - R_{comm}^2}{\|C(\boldsymbol{x}_i^* - \boldsymbol{x}_j^*)\|^2 - R_{safe}^2}\right)\right]^2, j \in \mathcal{N}_i.$$

This APF based cooperative control in conjunction with the baseline trajectory prediction will thus guarantee collision-free maneuvers for the satellite swarm as they navigate between initial and final formation without having to alter any given mission specifications. This is an ongoing research endeavor whose results will be reported elsewhere.

VII. Comparative Analysis of Both Approaches on Performance & Computational Complexity

In this section, we present a trade-off analysis between computation-time, loss of fuel-optimality and communication complexity between both approaches with increasing swarm size. Under identical mission specifications as in the previous illustrative examples, for $R_{\text{safe}} = 50$ m, the time-complexity of both approaches, evaluated with Gurobi solver are now presented in Figure 12(a), from which we observe that the computation-time for TP2 is smaller than that of



(a) Computation-time complexity (calculated over 1000 numerical simulations) with increasing swarm size N

(b) Relative fuel-cost error (%) between both approaches with mean μ = 0.0636 and standard deviation σ = 0.1318

Fig. 12 Comparison of computation-time complexity and fuel consumption between both approaches

TP1 for swarm size $N \leq 50$. Since the centralized PACA problem which is the major component of the TP2 and is formulated with MILP, the computation complexity is $O(exp(N))$. On the other hand, the computation-time of TP1 is dominated by the distributed TOCA problem and due to the computation of the trajectories now being distributed, the computation-time complexity increases linearly, *i.e.* $O(N)$. The computation-time of both approaches become nearly the same for $N = 50$, and beyond that the computation-time of TP2 approach is more than that of TP1. However, the relative fuel cost error between both of these approaches is less than 1%, as shown in Figure 12(b), and hence there is no apparent advantage of one approach over the other based on the fuel-optimality.

However, based on both the computation-time and negligible loss of fuel-optimality, from the numerical results in Figure 12 we obtain that TP2 approach serves as a suitable TP algorithm for $N \in [1, 50]$, while TP1 is preferred for an even larger swarm size. Furthermore, both of these proposed approaches reduce the communication requirements of distributed auction based PA algorithm in [15, 42, 47] with communication complexity $O(N^2)$ to $O(N)$ under minimal spanning tree assumption. The total number of inter-satellite connections required during the safe navigation of satellite swarm to the target formation under our proposed approaches is shown in Figure 13 from which we observe that TP2 requires minimal communication. Hence, the selection of a suitable approach for a swarm size beyond $N > 50$ is governed by the trade-off analysis between computation and communication complexity, presented respectively in Figs. 12(a) and 13.

VIII. Conclusions

In this paper, we studied the trajectory planning (TP) problem of a large LEO satellite swarm in cluster-formation under orbital perturbations and modeling uncertainties. We reviewed several relative dynamical models of the satellites,

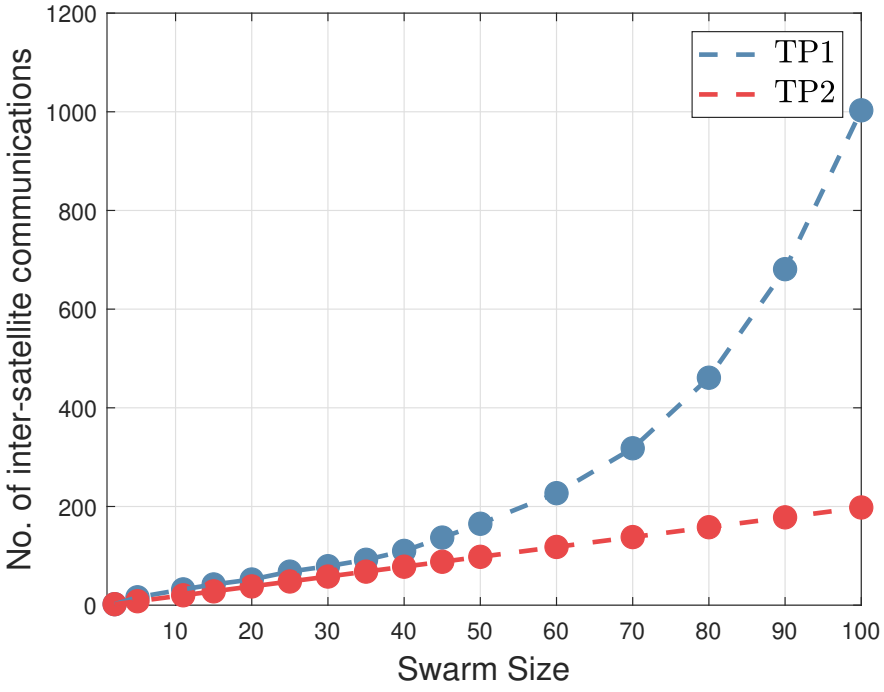


Fig. 13 Communication complexity of proposed approaches

and, for various initial conditions and bounded actuation, analyzed the modeling errors. The analysis informed model selection and resulted in the linearized J2 model as the most appropriate candidate for the TO algorithm. To reduce the computation-time, we decoupled the TP problem into various sub-components such as PA and TO problems with convexified CA constraints. We presented two different formulations of the TP problem - (*i*) TP1: distributed TO-based approach computes the satellite trajectories to the fuel-optimal assignment and (*ii*) TP2: decentralized TO based approach selects an assignment which makes the nominal satellite paths to be collision-free. To robustify this TP problem against uncertainties in sensing and actuation, ignored nonlinear effects, and perturbations, a predictive control framework was proposed that re-computes the optimal trajectories iteratively taking into account the satellite's updated, actual position and velocity.

With the help of numerical simulations, we compared the effectiveness of both approaches in solving TP problems of a large swarm. It turned out that, TP2 is preferred for moderate swarm size (up to 50), while TP1 for a considerably large swarm (in the order of 100s). Both of these approaches reduce the real-time communication requirements in other existing works on path planning problems.

The infeasibility due to CA constraints in both approaches is resolved in this work by reducing the inter-satellite safety distance for a violating pair, which however may not be ideal for an SFF mission in close proximity. To maintain the desired inter-satellite safe distance, our proposed future work includes an APF based adaptive control algorithm to correct the trajectories over baseline prediction. Furthermore, we will also develop scalable, real-time feasible software tools and estimation algorithms to control the satellites, measure its own local states as well its neighbors while

maneuvering through the predicted satellite trajectories.

Acknowledgments

This work was supported by NASA under cooperative agreement VT-80NSSC20M0213. The authors would like to thank Dr. Ryan McDevitt from Benchmark Space Systems and Dr. Amir Rahmani from NASA's Jet Propulsion Laboratory (JPL) for insightful discussions.

References

[1] Lowe, C., Macdonald, M., Greenland, S., and Mckee, D., "'Charybdis'–The Next Generation in Ocean Colour and Biogeochemical Remote Sensing," *in proc of the* 26^{th} *Annual AIAA/USU Conference on Small Satellites*, 2012, pp. 1–13.

[2] Boshuizen, C., Mason, J., Klupar, P., and Spanhake, S., "Results from the planet labs flock constellation," *in Proc. of the* 28^{th} *Annual AIAA/USU Conference on Small Satellites*, 2014, pp. 1–8.

[3] Cutler, J., Ridley, A., and Nicholas, A., "Cubesat investigating atmospheric density response to extreme driving (CADRE)," *in proc. of the* 25^{th} *Annual AIAA/USU Conference on Small Satellites*, 2011, pp. 1–12. URL `https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=1131&context=smallsat`.

[4] Fish, C. S., et al., "Design, development, implementation, and on-orbit performance of the dynamic ionosphere CubeSat experiment mission," *Space Sci. Rev.*, Vol. 181, No. 1, 2014, pp. 61–120.

[5] Bracikowski, P., Lynch, K. A., and Gayetsky, L., "Low-resource cubesat-scale sensorcraft for auroral and ionospheric plasma studies," *in proc. of the* 24^{th} *Annual AIAA/USU Conference on Small Satellites*, Logan,UT, 2010, pp. 1–6. URL `https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=1191&context=smallsat`.

[6] Lyon, R., Sellers, J., and Underwood, C., "Small satellite thermal modeling and design at USAFA: FalconSat-2 applications," *in proc. of the IEEE Aerospace Conference*, Vol. 7, 2002, pp. 3391–3399.

[7] Redd, F., and OLSEN, T., "Microspacecraft and earth observation-The electrical field (ELF) measurement project," *in proc. of the* 6^{th} *Annual Summer Conference: NASA(USRA University Advanced Design Program*, 1990, pp. 217–219.

[8] Cutler, J., and Hutchins, G., "OPAL: Smaller, simpler, and just plain luckier," *in proc. of the* 14^{th} *AIAA/USU Conference Satellites*, Logan, UT, USA, 2000, pp. 1–12.

[9] Krieger, G., Hajnsek, I., Papathanassiou, K. P., Younis, M., and Moreira, A., "Interferometric synthetic aperture radar (SAR) missions employing formation flying," *Proceedings of the IEEE*, Vol. 98, No. 5, 2010, pp. 816–843.

[10] Morgan, D., Chung, S.-J., Blackmore, L., Acikmese, B., Bayard, D., and Hadaegh, F. Y., "Swarm-Keeping Strategies for Spacecraft under J2 and Atmospheric Drag Perturbations," *Journal of Guidance Control and Dynamics*, Vol. 35, No. 5, 2012, pp. 1492–1506.

[11] Krieger, G., Moreira, A., Fiedler, H., Hajnsek, I., Werner, M., Younis, M., and Zink, M., "TanDEM-X: A Satellite Formation for High-Resolution SAR Interferometry," *IEEE Trans. Geosci. Remote Sens.*, Vol. 45, No. 11, 2007, pp. 3317 – 3341.

[12] Lim, H. C., and Bang, H., "Trajectory Planning of Satellite Formation Flying using Nonlinear Programming and Collocation," *J. Astron. Space Sci.*, Vol. 25, No. 4, 2008, pp. 361–374.

[13] Chung, S. J., Bandopadhyay, S., Foust, R., Subramanian, G. P., and Hadaegh, F. Y., "Review of Formation Flying and Constellation Missions Using Nanosatellites," *Journal of Spacecraft and Rockets*, Vol. 53, No. 3, 2016, pp. 1–12.

[14] Wang, D., Wu, B. L., and Poh, E. K., *Satellite Formation Flying*, Intelligent Systems, Control and Automation: Science and Engineering, Vol. 87, Springer, Singapore, 2017.

[15] Morgan, D., Chung, S.-J., and Hadaegh, F. Y., "Spacecraft Swarm Guidance Using a Sequence of Decentralized Convex Optimizations," *in Proc. of the AIAA/AAS Astrodynamics Specialist Conference*, Minneapolis, MN, 2012, pp. 1–16.

[16] Morgan, D., Chung, S.-J., and Hadaegh, F. Y., "Decentralized Model Predictive Control of Swarms of Spacecraft Using Sequential Convex Programming," *in Proc. of the AIAA/AAS Astrodynamics Specialist Conference*, Kauai, HI,, 2013, pp. 1–20.

[17] Garcia, I., and How, J. P., "Trajectory Optimization for Satellite Reconfiguration Maneuvers with Position and Attitude Constraints," *Proceedings of the 2005, American Control Conference, 2005.*, Vol. 2, Portland, USA, 2005, pp. 889–894.

[18] Tillerson, M., Inalhan, G., and How, J. P., "Co-ordination and control of distributed spacecraft systems using convex optimization techniques," *Int. J. Robust Nonlinear Control*, Vol. 12, 2002, pp. 207–242.

[19] Kavraki, L., Svestka, P., Latombe, J.-C., and Overmars, M., "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 4, 1996, pp. 566–580. https://doi.org/10.1109/70.508439.

[20] Rao, A. V., "A Survey of Numerical Methods for Optimal Control," *Advances in the Astronautical Sciences*, Vol. 135, 2010, pp. 497–528.

[21] Lin, X., Shi, X., and Li, S., "Optimal cooperative control for formation flying spacecraft with collision avoidance," *Sci. Prog.*, Vol. 103, No. 1, 2019, pp. 1–19.

[22] Richards, A., Schouwenaars, T., How, J. P., and Feron, E., "Spacecraft Trajectory Planning with Avoidance Constraints Using Mixed-Integer Linear Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 4, 2002, pp. 755–764.

[23] Basu, H., Pedari, Y., Almassalkhi, M., and Ossareh, H. R., "Fuel-Optimal Trajectory Planning of Satellites Using Minimum Distance Assignment and Comparative Analysis of Satellite Relative Dynamics," *in proc. of the 2022 American Control Conference (ACC)*, Atlanta, USA, 2022, pp. 1–7.

[24] Xu, G., and Wang, D., "Nonlinear Dynamic Equations of Satellite Relative Motion Around an Oblate Earth," *Journal of Guidance, Control and Dynamics*, Vol. 31, No. 5, 2008, pp. 1521–1524.

[25] Roscoe, C. W. T., Griesbach, J. D., Westphal, J. J., Hawes, D. R., and J. P. Carrico, J., "Force modeling and state propagation for navigation and maneuver planning for CubeSat rendezvous, proximity operations, and docking," *Advances in the Astronautical Sciences*, Vol. 150, 2014, pp. 573–590.

[26] Eshagh, M., and Alamdari, M. N., "Perturbations in orbital elements of a low earth orbiting satellite," *Journal of the Earth & Space Physics.*, Vol. 33, No. 1, 2007, pp. 1–12.

[27] Khalil, I., and Samwel, S., "Effect of Air Drag Force on Low Earth Orbit Satellites during Maximum and Minimum Solar Activity," *Space Research Journal*, Vol. 9, 2016, pp. 1–9.

[28] Inalhan, G., and How, J., "Relative dynamics and control of spacecraft formations in eccentric orbits," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 1, 2002, pp. 48–59.

[29] Ma, J., "Formation Flying of Spacecrafts for Monitoring and Inspection," Master's thesis, Lulea University of Technology, Sweden, 2009.

[30] Morgan, D., Subramanian, G. P., Chung, S. J., and Hadaegh, F. Y., "Swarm assignment and trajectory optimization using variable-swarm, distributed auction assignment and sequential convex programming," *Int. J. Rob. Res.*, Vol. 35, No. 10, 2016, pp. 1261 – 1285.

[31] Keviczky, T., Borrelli, F., Fregene, K., Godbole, D., and Balas, G. J., "Decentralized Receding Horizon Control and Coordination of Autonomous Vehicle Formations," *IEEE Transactions on Control Systems Technology*, Vol. 16, No. 1, 2008, pp. 19–33. https://doi.org/10.1109/TCST.2007.903066.

[32] Acikmese, B., Scharf, D. P., Murray, E. A., and Hadaegh, F. Y., "A Convex Guidance Algorithm for Formation Reconfiguration," *in Proc. of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA, Keystone, CO, 2006, pp. 1–17.

[33] Wu, B., Xu, G., and Cao, X., "Relative Dynamics and Control for Satellite Formation: Accommodating J2 Perturbation," *Journal of Aerospace Engineering*, Vol. 29, No. 4, 2016, p. 04016011. https://doi.org/10.1061/(ASCE)AS.1943-5525.0000600.

[34] Sidi, M., *Spacecraft Dynamics & Control: A Practical Engineering Approach*, Cambridge Aerospace Series, Vol. 7, Cambridge University Press, 2006.

[35] Directorate, A. F. R. L. S. V., "TechSat 21 factsheet page," https://www.vs.afrl.af.mil/factsheets/TechSat21.html, 2019. Online; accessed 8 September 2021.

[36] Lovera, M., "Periodic Attitude Control for Satellites with Magnetic Actuators: An Overview," *IFAC Proceedings Volumes*, Vol. 34, 2001, pp. 113–118.

[37] Yoshimura, Y., "Magnetic Attitude Control of Satellites Using Coarse Pulse-Width-Modulation of Magnetorquers," *in proc. of the* 68^{th} *International Astronautical Congress (IAC)*, Adelaide, Australia, 2017, pp. 6840–6844.

[38] Borrelli, F., Subramanian, D., Raghunathan, A. U., and Biegler, L. T., "MILP and NLP Techniques for Centralized Trajectory Planning of Multiple Unmanned Air Vehicles," *in proc. of the* 2006 *American Control Conference*, Minneapolis, USA, 2006, pp. 5763–5768.

[39] Gurobi Optimization, LLC, "Gurobi Optimizer Reference Manual," , 2021. URL https://www.gurobi.com.

[40] Grozea, C., Bankovic, Z., and Laskov, P., *FPGA vs. Multi-core CPUs vs. GPUs: Hands-On Experience with a Sorting Application*, Facing the Multicore-Challenge: Aspects of New Paradigms and Technologies in Parallel Computing, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[41] Camacho, E. F., and Alba, C. B., *Model predictive control*, Springer science & business media, 2013.

[42] Morgan, D., Chung, S.-J., and F.Y.Hadaegh, "Model Predictive Control of Swarms of Spacecraft Using Sequential Convex Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 6, 2014, pp. 1725–1740.

[43] Liu, J., and Li, H., "Artificial Potential Function Safety and Obstacle Avoidance Guidance for Autonomous Rendezvous and Docking with Noncooperative Target," *Mathematical Problems in Engineering*, 2019, pp. 1–17. URL https://downloads.hindawi.com/journals/mpe/2019/3451864.pdf.

[44] S.-Kazerooni, E., Verhaegh, J., Ploeg, J., and Alirezaei, M., "Cooperative Adaptive Cruise Control: An artificial potential field approach," *in proc. of the* 2016 *IEEE Intelligent Vehicles Symposium*, Gothenburg, Sweden, 2016, pp. 361–367.

[45] Sun, J., Tang, J., and Lao, S., "Collision Avoidance for Cooperative UAVs With Optimized Artificial Potential Field Algorithm," *IEEE Access*, Vol. 5, 2017, pp. 18382–18390.

[46] Chen, L., Guo, Y., Li, C., and Huang, J., "Satellite Formation-Containment Flying Control with Collision Avoidance," *Journal of Aerospace Information Systems*, Vol. 15, No. 5, 2018, pp. 253–270.

[47] Zavlanos, M. M., Spesivtsev, L., Leonid, and Pappas, G. J., "A distributed auction algorithm for the assignment problem," *in proc. of the 47th IEEE Conference on Decision and Control*, 2008, pp. 1212–1217. https://doi.org/10.1109/CDC.2008.4739098.