Effect of perturbations on debris-to-debris orbital transfers: a quantitative analysis

Kartik Kumar[a,b,*], Enne Hekma[b], Abhishek Agrawal[b], Francesco Topputo[c]

[a] *Dinamica Srl, Via R. Morghen, 13 20158 Milan, Italy*
[b] *Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands*
[c] *Politecnico di Milano, Via G. La Masa, 34, 20156 Milan, Italy*

Abstract

We investigated the applicability of the Lambert solver (Izzo, 2014) for preliminary design of Multi-Target Active Debris Removal missions. Firstly, we computed ≈25 million debris-to-debris transfers using the Lambert solver for selected sets of debris objects in Low Earth Orbit, Geostationary Transfer Orbit, and Geosynchronous Orbit. Subsequently, we propagated the departure states of the Lambert transfers below selected ΔV cut-offs using the SGP4/SDP4 propagator (Vallado et al., 2006). We recorded the arrival postion and velocity error vectors incurred by neglecting perturbations and analyzed the results for each orbital regime. Our results indicate that perturbations can play a significant role in determining the feasibility of debris-to-debris transfers. By using the Lambert solver and neglecting perturbations, the errors in the arrival position and velocity for individual legs can be large. The largest errors were obtained for transfers between debris objects in Sun-Synchronous Orbit ($\mathcal{O}(100)$ km error in magnitude of position vector and $\mathcal{O}(0.1)$ km/s error in magnitude of velocity vector). Hence, solely employing the Lambert solver to rank transfer legs could lead to incorrect choices for sequencing of multi-target trajectories. This is particularly relevant for transfers in Low Earth Orbit, where the effects of perturbations are the strongest.

Keywords: space debris; active debris removal; mission design; orbital perturbations; lambert solver; sgp4

*Corresponding author
Email address: kumar@dinamicatech.com (Kartik Kumar)

1. Introduction

Active Debris Removal (ADR) missions such as e.Deorbit (Biesbroek, 2012; Biesbroek et al., 2013), RemoveDEBRIS (Lappas et al., 2014; Forshaw et al., 2015), and CleanSpace One (Esmiller and Jacquelard, 2011; Richard et al., 2013) have gained prominence in recent years due to mounting risk of on-orbit collisions and the prospect of further growth of the debris population (Liou et al., 2010; Liou, 2011; Bonnal et al., 2013). Recent disruption events, such as the 2009 Iridium-Cosmos collision (Kelso, 2009; Tan et al., 2013), have exacerbated concerns about the imminent threat posed to valuable space-based assets. The development of ADR mission concepts is necessary to fully assess the capability to mitigate the threat posed by large, non-cooperative Resident Space Objects (RSOs) in near-Earth space (Pelton, 2015). Mitigating the threat posed by multiple RSOs in a single mission can lead to significant cost-savings. Hence, numerous studies have been published on the topic of planning Multi-Target Active Debris Removal (MTADR) missions to remove multiple hazardous RSOs (Barbee et al., 2011; Zuiani and Vasile, 2012; Izzo et al., 2015; Peters et al., 2015).

Preliminary mission design for MTADR missions is difficult for a variety of reasons. Computing optimal sequences of RSOs is a complex combinatorial optimization problem, akin to the Travelling Salesman Problem (TSP) (Laporte, 1992), which is known to be NP-hard (non-deterministic polynomial), meaning that in the worst-case, the run-time needed to obtain a solution can increase exponentially. The TSP has been studied for a slew of applications within the field of astrodynamics, including finding optimal sequences for on-orbit servicing missions (Alfriend et al., 2006; Holzinger and Scheeres, 2011), asteroid rendezvous and interception missions (Conway et al., 2007; Wall and Conway, 2009), and MTADR missions (Barbee et al., 2011; Izzo et al., 2015). Research into these applications has led to the development of new tools and techniques to support rapid generation of near-optimal, multi-target sequences.

Brute-force search (exhaustive search) to compute and rank multi-target sequences quickly becomes numerically intractable as the set of possible targets grows. For astrodynamics applications, a few different methods are typically employed to reduce numerical complexity and rapidly generate candidate sequences. As a first step, static filters are usually applied to pre-prune transfer legs on the basis of energy and angular momentum considerations, by filtering out large changes in semi-major axis, inclination and right ascen-

sion of the ascending node (RAAN). Often, transfers between target pairs are modeled as bi-impulsive (high-thrust) for preliminary design. The Lambert solver is used, as it can rapidly provide solutions for such transfers. A number of variants of the Lambert solver can be found in the literature (Lancaster and Blanchard, 1969; Gooding, 1990; Izzo, 2014): for this study, we employed the "Izzo variant" (Izzo, 2014). In combination with optimization algorithms, the Lambert solver has proven to be an essential tool for preliminary mission planning. Given mission constraints, such as departure window, total mission duration, total fuel, and stay time, the Lambert solver can be combined with algorithms that are able to efficiently process the tree of multi-target sequences, such as branch-and-bound and greedy algorithms (Wall and Conway, 2009; Leite Pinto Secretin, 2012; Madakat et al., 2013). Using these tools, candidate multi-target sequences can be ranked for further analysis.

This paper summarizes an investigation into orbital transfers between pairs of RSOs, which we refer to as debris-to-debris (D2D) transfers. Specifically, we analyze whether neglecting orbital perturbations by employing the Lambert solver to compute transfers can lead to significant errors in spacecraft state. D2D transfers are important to understand, as they are basic building blocks for MTADR mission planning. We selected sets of RSOs in Low-Earth Orbit (LEO), Geostationary Transfer Orbit (GTO), and Geosynchronous Orbit (GSO) and employed the multi-revolution Lambert solver developed by Izzo (2014) to compute transfer legs. The rationale for selecting these three orbital classes is given in Section 3.1. Using the Lambert solver, we generated a ranking of the legs based on transfer ΔV.

We selected a set of transfers for each orbital regime and propagated them using the SGP4/SDP4 propagator[1] (hereinafter the SGP4 propagator). The SGP4 propagator belongs to the collection of simplified perturbations models (Hoots et al., 2004; Vallado et al., 2006). We employed the SGP4 propagator, since it is computationally-efficient and includes an accurate representation of major perturbations in the Earth environment. By propagating the departure states for the Lambert transfers using the SGP4 propagator, we quantified the effect of perturbations in terms of position and velocity errors with respect to the target arrival state. Cataloguing these errors provided a means to assess the impact of perturbations on transfers across the different orbital regimes.

[1] https://github.com/openastro/sgp4

This paper is organized as follows. Section 2 provides an overview of MTADR mission design. We review the tools and algorithms employed for previous studies to generate candidate multi-target sequences. In Section 3, we outline the setup adopted for the simulations using the Lambert solver and summarize the approach to assessing the position and velocity errors due to perturbations. Subsequently, the results for all three orbital regimes are presented in detail in Section 4. Finally, we provide concluding remarks in Section 5 and outline steps for future work.

2. Multi-Target Active Debris Removal missions

The primary aim of preliminary mission design for MTADR missions is to determine feasibility by assessing the ΔV required. Given a large pool of RSOs, this can be achieved by rapid generation and evaluation of candidate target sequences, which is related to efficiently solving variants of the TSP (Laporte, 1992). In general, it is infeasible to rank all possible sequences through brute-force methods, due to the large dimensionality and size of the search-space. For astrodynamics applications, multi-target sequencing problems are often tackled by computing pairwise legs and enumerating candidate sequences by using combinatorial optimization algorithms (Alfriend et al., 2006; Wall and Conway, 2009; Leite Pinto Secretin, 2012; Izzo et al., 2015). In this Section, we provide a synopsis of MTADR mission studies and the extent to which orbital perturbations have been included in modeling transfer trajectories.

Zuiani and Vasile (2012) investigated a mission to de-orbit five debris objects using low-thrust propulsion. They generated multi-target, multi-revolution trajectories, however orbital perturbations were omitted in their calculations. Barbee et al. (2011) generated a slew of trajectories using the "Series Method". Although they employed Two-Line Elements (TLEs) to initiate the states of the RSOs in their simulations, they also neglected orbital perturbations when computing transfers. Izzo et al. (2015) developed advanced algorithms to tackle the TSP-variant that describes the MTADR scenario. They also employed TLEs to model the states of the debris objects and used the SGP4 propagator to allow their orbits to evolve in time. However, they utilized a three-impulse model for the transfer trajectories and neglected the effects of perturbations. Alfriend et al. (2006) presented an on-orbit servicing mission that shares similarities with MTADR scenarios. They studied the TSP to generate an optimal path between a sequence of

GEO satellites. They made several simplifying assumptions in their orbit model however and neglected orbital perturbations.

Some MTADR missions studies have been published that included first-order effects of perturbations on transfer trajectories. For instance, Bonnal et al. (2013) considered different mission architectures and utilized natural drift of the RAAN due to J_2 to minimize out-of-plane trajectory maneuvers. Similarly, Van Der Pas et al. (2014) studied different mission scenarios for ADR and permitted drift of the RAAN within their simplified dynamics model. Quinlan et al. (2011) considered random sequences of five RSOs in LEO and generated transfer ΔV estimates using a simplified orbital model including Hohmann transfers, impulsive inclination changes and phasing due to J_2. Braun et al. (2013) studied different mission scenarios, and when pruning the debris catalog, they accounted for the effect of change of inclination and RAAN on transfer ΔV. Their simulations revealed that although RAAN drift can help to reduce the mission ΔV, it comes at the cost of increased transfer time due to phasing.

Mission planning for multi-target scenarios typically involves optimization of discrete- and continuous-valued variables (Wall and Conway, 2009). The continuous-valued variables are associated with the continuous-time dynamics that describe the one-to-one transfers between pairs of targets. The discrete-valued variables are associated with sequencing in the outer-loop. The research presented in this paper pertains to the continuous-time dynamics model within the inner loop. We investigated whether modeling the continuous-time dynamics as a ballistic, Kepler arc (part of a conic section) is sufficiently accurate for MTADR missions. Our conjecture was that the ranking of the most fuel-efficient transfer arcs using the Lambert solver, which employs Kepler dynamics, would be distorted by the fact that perturbations in the near-Earth environment cannot always be neglected to first-order, as can be assumed for other multi-target missions e.g., asteroid missions, multi-moon missions, etc. We analyzed this hypothesis by studying the effect of perturbations on transfer legs computed using the "Izzo variant" of the Lambert solver (Izzo, 2014). We propagated the departure state for the Lambert transfers using the SGP4 propagator to obtain insight into the errors introduced by neglecting orbital perturbations.

3. Debris-to-debris transfer simulations

We provide an overview of the salient aspects of the simulation setup employed for this study. In Section 3.1, a short summary is included of the pruned TLE catalogs generated for the three orbital regimes investigated. Subsequently, in Section 3.2, we briefly describe the procedure adopted to compute transfers using the Lambert solver for the RSOs listed in the pruned TLE catalogs. In Section 3.3, an overview is provided of the steps taken to assess the impact of orbital perturbations on the transfers computed. We also briefly summarize the use of the SGP4 propagator and describe the concept of a "virtual" TLE.

3.1. Two-Line Element catalog

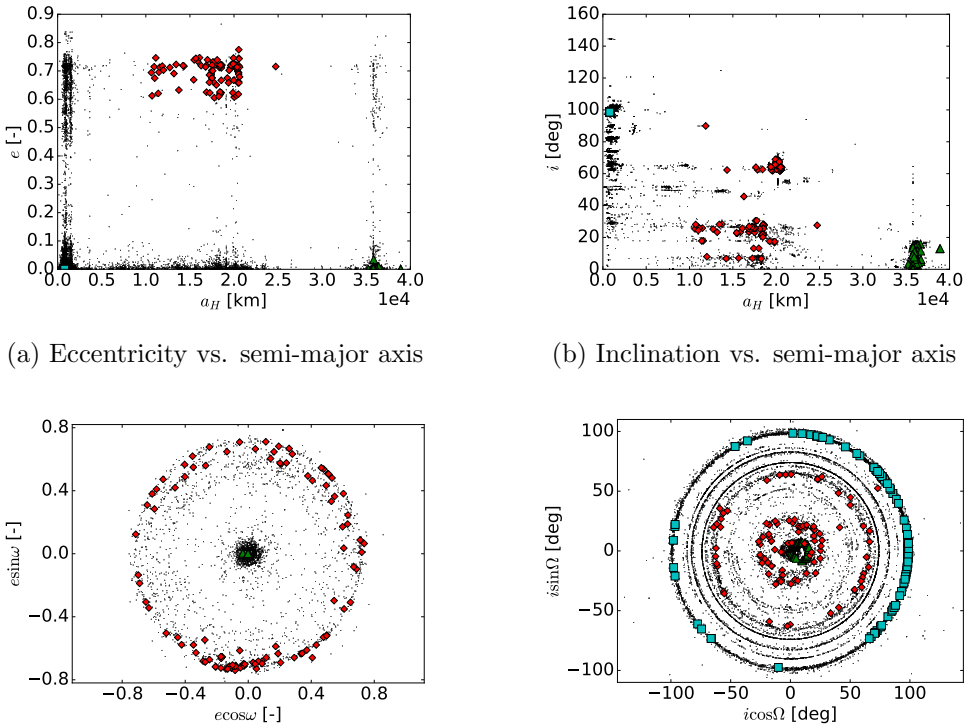
The TLE catalog, developed and maintained by NORAD (North American Aerospace Defense Command)[2], provides a listing of all RSOs currently tracked. Rapid growth of the catalog over the last decade has been precipitated by collision and fragmentation events. The catalog contains in excess of 15,000 objects, including operational satellites, Rocket Bodies (R/Bs) and debris fragments. We used the full, three-line TLE catalog[3] to generate subsets of RSOs for our study. The subsets were chosen across different orbital regimes: LEO, GTO and GSO. For each of these sets, we picked a hundred objects. Figure 1 provides an illustration of the distribution of the pruned catalog subsets in orbital-element space. In the following, we provide a brief description of these subsets.

3.1.1. Low-Earth Orbit: debris in Sun-Synchronous Orbit

D2D transfers in LEO are important to understand, since RSOs in this orbital regime carry the greatest risk of collision. Specifically, the Sun-Sychronous Orbit band is a "high-traffic zone" that poses signficant danger to operational satellites. Hence, we selected a hundred R/Bs and debris fragments from this region for our study. We pruned the full catalog using the filters given in Table 1. The "semi-major axis altitude" is the semi-major axis minus the Earth's mean radius. The "name" filter was used to filter out TLEs based on whether Line 0 contained "R/B" or "DEB". The cut-off filter was used to restrict the catalog subset to a hundred objects.

[2]Catalog available through https://space-track.org

[3]Catalog fetched on 10th January, 2016



(a) Eccentricity vs. semi-major axis

(b) Inclination vs. semi-major axis

(c) Components of eccentricity vector

(d) Components of inclination vector

Figure 1: Distribution of selected LEO (cyan, square), GTO (red, tilted square) and GSO (green, triangles) catalog objects in orbital-element space, against the backdrop of the entire TLE catalog (a_H = semi-major axis (altitude above the Earth's surface), e = eccentricity, ω = argument of perigee, i = inclination, Ω = right ascension of ascending node).

3.1.2. Geostationary Transfer Orbit: Rocket Bodies

The large population of R/Bs in GTO poses a sizable risk to operational satellites for a number of reasons. Firstly, collision risk is elevated due to the large size and shape of the R/Bs. Secondly, since they are in highly-elliptical orbits, they sweep through the altitude range from LEO to GSO and the relative velocity in collision events can be on the order of km/s; hence they carry significant energy. Lastly, it is likely that they carry residual fuel, which means that there is tangible risk of explosions that can lead to large debris clouds. We selected a subset of R/Bs in GTO using the pruning filters given in Table 2.

Table 1: Pruning filter parameters for LEO TLE catalog subset.

Parameter	Value
Semi-major axis altitude	[750, 850] km
Eccentricity	[0.0, 0.1]
Inclination	[98, 99] deg
Name	(R/B | DEB)
Size cut-off	100

Table 2: Pruning filter parameters for GTO TLE catalog subset.

Parameter	Value
Semi-major axis altitude	[2000, 40000] km
Eccentricity	[0.6, 1.0]
Inclination	[0, 180] deg
Name	(R/B)
Size cut-off	100

3.1.3. Geosynchronous Orbit: telecommunication satellites

We selected a subset of telecommunication satellites in GSO for the last case. The satellite telecommunications industry is a multi-billion Euro sector and individual satellites are valuable assets. Orbital slots in GSO are also highly prized; hence developing a robust strategy to handle satellites at End-Of-Life (EOL) is essential. Currently, satellites in GSO are moved to supersynchronous graveyard orbits. We selected a subset of Intelsat and Galaxy satellites, owned and operated by Intelsat[4]. Studying transfers within this subset is not only interesting from the perspective of developing active removal missions, but also in the context of on-orbit servicing, life-extension missions, etc. The pruning filters used to generate the GSO TLE catalog subset are summarized in Table 3.

3.2. Debris-to-debris transfers using the Lambert solver

Using the TLE catalog subsets presented in Section 3.1, we generated a scan of all 9,900 D2D transfers within each subset. Each transfer was described by a departure object ID and an arrival object ID. We generated a

[4]Intelsat, Ltd., http://www.intelsat.com

Table 3: Pruning filter parameters for GSO TLE catalog subset.

Parameter	Value
Semi-major axis altitude	[30000, 40000] km
Eccentricity	[0.0, 0.1]
Inclination	[0, 1] deg
Name	(INTELSAT | GALAXY)
Size cut-off	100

Table 4: Summary of the departure epoch and time-of-flight bounds for each orbital regime.

Orbital regime	Departure epoch bounds [MJD]	Time-of-flight bounds [s]
LEO	[57399.0, 57399.125]	[100, 32400]
GTO	[57399.0, 57399.833]	[100, 216000]
GSO	[57399.0, 57401.000]	[100, 518400]

pork-chop plot for each transfer, specified by a grid within departure epoch and time-of-flight (t_{TOF}) bounds, as given in Table 4.

The transfer orbits were computed using the "Izzo variant" of the Lambert solver (Izzo, 2014), which solves Lambert's problem. This problem has been studied extensively over the last few centuries and has found important applications in orbit determination, mission design and spacecraft rendezvous. Lambert's problem states the following: given a particle in an inverse-square, central force field, located at positions $\bar{r}_1$ (3×1 vector) and $\bar{r}_2$ (3×1 vector) at epochs t_1 and t_2 respectively, find a Kepler orbit that connects the two positions, with the time-of-flight $t_{TOF} = t_2 - t_1$ for the orbital arc from $\bar{r}_1$ to $\bar{r}_2$. Computing the corresponding, unique Keplerian orbit that satisfies the given t_{TOF} involves working through a set of equations that tie the geometry of the problem to time through Kepler's equation and related trigonometric relationships. We refer the reader to any classical text on astrodynamics for further information, e.g., Volk (1980); Conway and Prussing (1993); Battin (1999). We utilized the "Izzo variant" of Lambert solver implemented in PyKEP[5] to compute transfers.

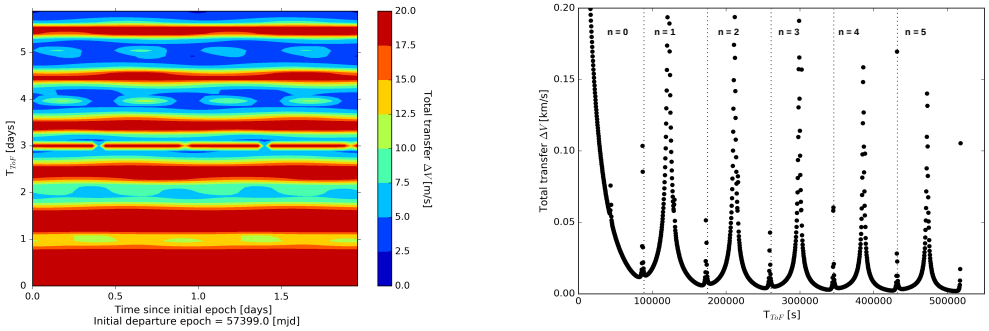
We used a 50 × 50 (equidistant) grid within the bounds given in Table 4 to generate the pork-chop plots for each transfer. We set the maximum

[5]PyKEP, European Space Agency, https://github.com/esa/pykep

number of revolutions to five. Initial numerical experiments indicated that five would enable us to generate a large data set, while keeping the computation time reasonable. For each combination of departure object ID, arrival object ID, departure epoch and time-of-flight, we selected the (multi-revolution) transfer with the lowest ΔV, meaning that we generated a set of $100 \times 99 \times 50 \times 50 = 2.475 \times 10^7$ transfers for each subset. The positions of the departure and arrival objects at the desired epochs were computed by propagating their TLEs to those epochs using the SGP4 propagator (Section 3.3.1). Our approach to determining the departure and arrival positions necessary to execute the Lambert solver parallels the method used by Izzo et al. (2015).

Figure 2 provides an illustration of a pork-chop plot generated for a D2D transfer in GSO from object 24812 to object 27445, using the departure epoch and time-of-flight bounds given in Table 4. The results presented in this figure illustrate the complex underlying dynamics that manifest as multiple local maxima and minima. The corresponding ΔVs range from a few m/s to several tens of m/s. The repeated horizontal banding in Figure 2a is a signature for each additional transfer revolution. Figure 2b illustrates a vertical slice of the pork-chop plot at 57400.0 MJD (one day after the initial departure epoch given in Table 4), highlighting the total transfer ΔV, as a function of time-of-flight, for a fixed departure epoch. For each time-of-flight grid point, we executed the Lambert solver for a number of revolutions and plotted the lowest transfer ΔV. The figure demonstrates the progression of the number of complete transfer revolutions (n), as a function of time-of-flight. The peaks occur at the known singularities for the Lambert solver ($\theta = k \times \pi$ rad, where $k = 0, 1, 2 \ldots$ and θ is the angle between $\bar{r}_1$ and $\bar{r}_2$). Note that Figure 2b is based on a denser grid of 500 data points for illustration purposes.

We aggregated all of the results obtained by running the Lambert solver for a given TLE catalog subset. Figure 3 illustrates an example of the histogram of ΔVs obtained for all of the GSO transfers. We investigated correlations with the underlying dynamics using the data set for all transfers. Figure 4 is a scan map that illustrates the lowest transfer ΔV for each pair of departure and arrival objects in the GSO catalog (note that the values along the diagonal are zero). The departure and arrival objects in this case are sorted in terms of increasing inclination of the departure object's orbit, revealing interesting correlations between the transfer ΔV and the underlying dynamics. For instance, the inclination plot (upper) illustrates families of objects



(a) Example pork-chop plot

(b) Example pork-chop slice

Figure 2: Example of a pork-chop plot (Figure 2a) generated for a single D2D transfer in GSO (departure object ID: 24812, arrival object ID: 27445), using a 50 × 50 grid for the departure epoch and time-of-flight windows. All ΔVs exceeding 20 m/s are given in red. Figure 2b illustrates a slice of the pork-chop plot for departure epoch 57400.0 MJD (resampled on a denser grid of 500 data points to highlight all singularities).

on (near-)zero and high (≈20 deg) inclination orbits. This results in low ΔVs for transfers between objects within these families (near-diagonal) and high ΔVs for transfers between the families (off-diagonal). This is expected, since plane-change maneuvers are expensive in terms of ΔV.

3.3. Debris-to-debris transfers with perturbations

After generating the data set for each orbital regime using the Lambert solver, as described in Section 3.2, we executed a series of simulations to investigate the effect of orbital perturbations on the computed transfers. In Section 3.3.1, we provide a brief summary of the dynamical model employed to simulate the effect of perturbations. Subsequently, in Section 3.3.2, we introduce an algorithm to convert a Cartesian state to a "virtual" TLE. The Cartesian-to-TLE (Cart2TLE) conversion algorithm was necessary to utilize the dynamical model. Propagation of the transfer using the dynamical model yielded arrival position and velocity errors, which were analyzed to shed light on the effect of perturbations.

3.3.1. Simplified perturbations models

The family of simplified perturbations models includes the Simplified General Perturbations (SGP) and Simplified Deep Space Perturbations (SDP) models, which incorporate accurate representations of the main perturbing

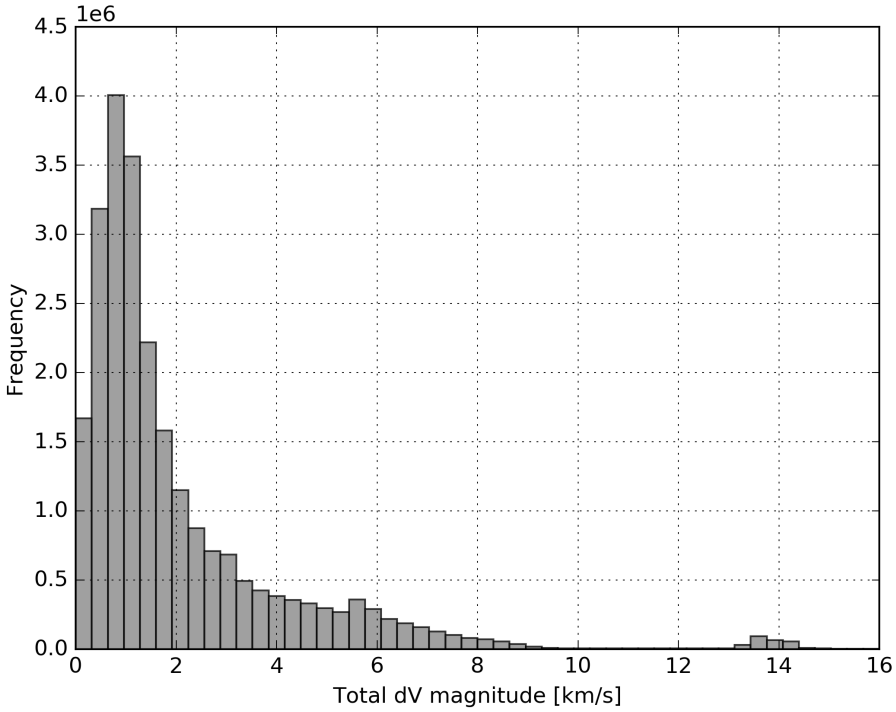


Figure 3: Histogram of transfer ΔV data set obtained from large-scale simulations using the Lambert solver for the GSO TLE catalog subset. This data set contains 2.475×10^7 D2D transfers.

forces that affect objects orbiting the Earth: gravity due to non-sphericity of the Earth, atmospheric drag and third-body gravity due to the Moon and Sun. The development of these models catalogs a long history of research. The SGP4 and SDP4 models are used by NORAD to compute the TLEs for each object tracked in the catalog. The TLEs are generated by fitting ground-based observations using the SGP4 (for low orbits) and SDP4 (for high orbits) models. These models must be used for reconstruction of the orbital state vector (Cartesian state) in the inertial frame at a given epoch. The mathematical models for SGP4 and SDP4, which are too lengthy to reproduce here, can be found in Hoots and Roehrich (1980); Hoots et al. (2004); Vallado et al. (2006).

As previously mentioned, we used the SGP4 and SDP4 dynamical models (hereinafter referred to as the SGP4 propagator) to determine the departure

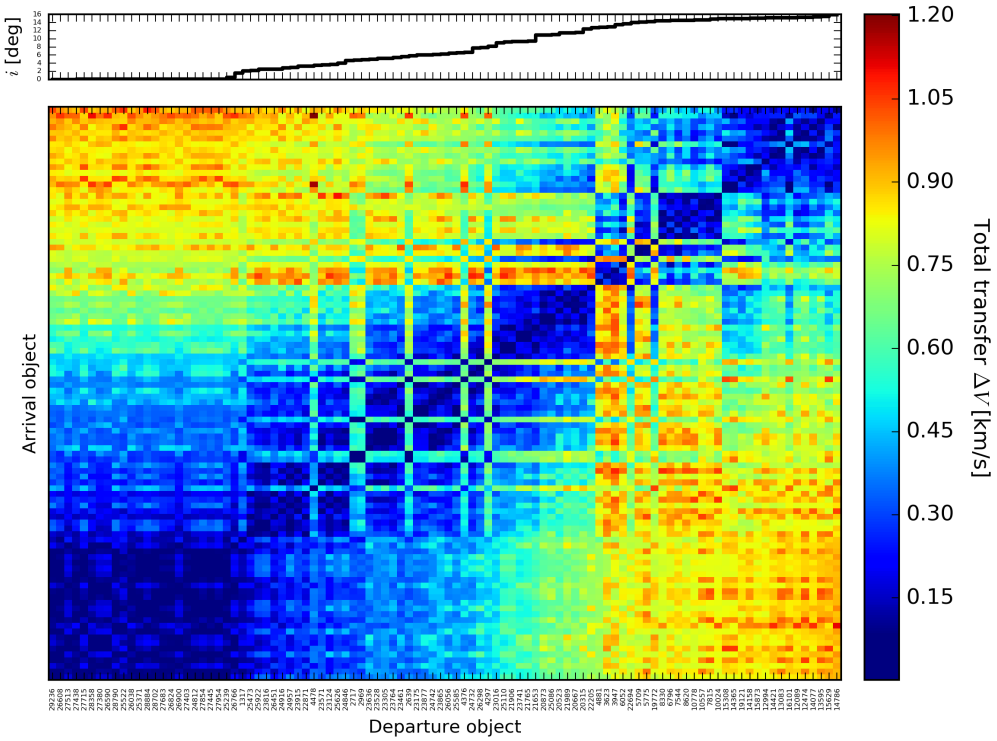


Figure 4: Example of a transfer ΔV scan map obtained from large-scale simulations using the Lambert solver for the GSO TLE catalog subset. The scan map illustrates the lowest transfer ΔV computed for each pair of departure and arrival objects. The list of departure (x-axis) and arrival (y-axis) objects are sorted in terms of increasing inclination of the departure object's orbit (shown in the upper plot).

and arrival positions for the Lambert solver. Subsequently, we employed the SGP4 propagator to propagate the departure state of the computed Lambert transfer. We estimated the effect of perturbations by comparing the target arrival state to the propagated state at $t_1 + t_{\mathrm{TOF}}$, where t_1 is the departure epoch. The reason we opted to use the SGP4 propagator instead of a full-fidelity numerical model to characterize the effect of orbital perturbations on the transfers is because it is of sufficiently high accuracy, whilst remaining computationally efficient, allowing us to analyze a large set of transfers. As noted previously however, in order to utilize the SGP4 propagator, we required a TLE at the departure epoch that described the departure state for the Lambert transfer. In the following section, we outline an algorithm to

generate a "virtual" TLE from a given Cartesian state and associated epoch.

3.3.2. Cartesian-to-Two-Line-Elements conversion

The SGP4 propagator described in the previous section employs dynamical models that are non-autonomous and non-linear. Although the reconstruction of the Cartesian state vector at a specified epoch from a given TLE is straightforward, the inverse process is more difficult due to the complex nature of the models. For the purposes of this study, we implemented a conversion algorithm that takes a Cartesian state and an associated epoch and generates a representative, "virtual" TLE (TLE*). This virtual TLE is generated by inverting Equation 1.

$$\bar{x}_1 = \text{SGP4}(\Delta t = t_1 - t_{\text{TLE}^*} = 0, \text{TLE}^*), \tag{1}$$

where $\bar{x}_1$ is the Cartesian state specified at epoch t_1, SGP4() is the propagator, and t_{TLE^*} is the epoch associated with TLE*. Equation 1 is set up such that $t_1 = t_{\text{TLE}^*}$; hence, we term this a Zero-Time Evaluation (ZTE) of the SGP4 propagator.

The process to compute TLE* is complex due to the fact that the dynamical models cannot be easily inverted. We employed a numerical approach by using a hybrid, derivative-free, multi-dimensional root-finding algorithm (Powell, 1968, 1970), implemented in the GNU Scientfic Library[6], to solve Equation 1 for TLE*. The statement of the root-finding problem is:

$$\begin{aligned} \text{find} \quad & \text{TLE}^* \\ \text{given that} \quad & \bar{x}(t_1) = \bar{x}_1 \\ \text{subject to} \quad & \bar{x} = \text{SGP4}(\Delta t, \text{TLE}^*) \\ \text{and} \quad & t_1 = t_{\text{TLE}^*} \rightarrow \Delta t = 0 \end{aligned} \tag{2}$$

In the following, we present a step-by-step overview of the Cart2TLE conversion algorithm, inspired by the strategies described by Eagle (2001) and Lee (2002). The algorithm is also illustrated in the flowchart given in Figure 5.

1. Given $\bar{x}_1$ at t_1, convert the Cartesian state to an equivalent set of osculating, Keplerian elements (KEP) (Battin, 1999), with the gravitational parameter of the central body given as $\mu_{\text{Earth}} = 398600.8$ km^3 s^{-2}.

[6]GNU Scientific Library, Free Software Foundation, Inc., https://www.gnu.org/software/gsl/manual/html_node/Multidimensional-Root_002dFinding.html

2. Use the Keplerian elements to generate first guesses for the TLE mean elements (ME), which are: mean inclination ($\tilde{i}$) [deg], mean RAAN ($\tilde{\Omega}$) [deg], mean eccentricity ($\tilde{e}$) [-], mean argument of perigee ($\tilde{\omega}$) [deg], mean mean anomaly ($\tilde{M}$) [deg], mean mean motion ($\tilde{n}$) [rev/day]. The algorithm internally converts KEP to the appropriate quantities and units for ME.
3. Based on t_1 and the computed ME, the virtual TLE (TLE_k^*) for the current iteration (k) is constructed. Although the TLE format requires the floating-point representation of the ME to be truncated, TLE_k^* is constructed by directly accessing and updating the internal variables within the TLE class with full-precision, floating-point values.
4. Given TLE_k^*, the ZTE ($\Delta t = 0$) can be executed using the SGP4 propagator. The output of this evaluation yields a Cartesian state vector for the current iteration ($\bar{x}_{1,k}$).
5. The residual vector for the current iteration ($\Delta\bar{x}_{1,k}$) are computed by subtracting $\bar{x}_{1,k}$ from $\bar{x}_1$. The residuals are checked against the user-specified absolute (ϵ_{abs}) and relative (ϵ_{rel})) tolerances, based on the following condition: $|\Delta\bar{x}_{1,k}| < \epsilon_{\text{abs}} + \epsilon_{\text{rel}}|\bar{x}_{1,k}|$. If this inequality is satisfied, the algorithm is stopped, and the last computed TLE_k^* is returned to the user. Else, the root-finding algorithm alters the ME by internally computing finite differences and the next iteration of the algorithm proceeds from Step 3.

The mean elements in the TLE (on Line 2) are defined with respect to an Earth-Centered Inertial reference frame, with coordinates given in terms of the True Equator Mean Equinox (Hoots et al., 2004). Our numerical algorithm uses an iterative process to obtain the values of these mean elements that satisfy Equation 2. The algorithm seeks to solve for the six unknown ME in TLE_k^*, given the six elements of $\bar{x}_1$. It is important to note that for the purposes of this algorithm, we set all elements in the virtual TLE, other than the ME and t_1, to zero. This means that the TLE_k^* that is generated by this process must not be used as a real TLE, and only serves to represent $\bar{x}_1$ at t_1 in an equivalent form that can be used in conjunction with the SGP4 propagator.

We performed numerical experiments indicating that atmospheric drag had negligible effects on the arrival state errors computed for all three orbital regimes. Hence, we set B^* to zero for the Cart2TLE algorithm, which ensures drag is not included when propagating the transfers. The assumption that

drag can be neglected is compatible with the short propagations carried out for this study.

3.3.3. Quantification of effect of perturbations

Finally, we propagated the virtual TLEs generated (Section 3.3.2) using Equation 3.

$$\bar{x}_2^* = \text{SGP4}(\Delta t = t_{TOF} = t_2 - t_1, \text{TLE}^*), \tag{3}$$

where $\bar{x}_2^*$ is the computed arrival state and t_2 is the associated epoch. Using $\bar{x}_2^*$, we computed the arrival position ($\delta\bar{r}_2$) and velocity ($\delta\bar{v}_2$) errors (3×1 vectors), as given by Equations 4 and 5.

$$\delta\bar{r}_2 = \bar{r}_2^* - \bar{r}_2 \tag{4}$$

$$\delta\bar{v}_2 = \bar{v}_2^* - \bar{v}_2 \tag{5}$$

where $\bar{r}_2^*$ and $\bar{v}_2^*$ are the arrival position and velocity vectors obtained after propagating using SGP4, and $\bar{r}_2$ and $\bar{v}_2$ are the target position and velocity vectors associated with the Lambert transfer. We analyzed the computed arrival state errors to quantify the effect of perturbations on the transfers.

Instead of propagating the full data set of ≈ 25 million transfers for each orbital regime, we restricted our analysis to transfers below a specified ΔV cut-off. Table 5 summarizes the selected ΔV cut-off values for each regime. The cut-off values were selected such that the 500,000 Lambert transfers with the lowest total transfer ΔVs were retained. We passed these transfers through a number of filters to ensure that the arrival state errors computed using the SGP4 propagator were valid. For instance, we checked if the perigee of the transfer orbit computed using the Lambert solver was greater than the mean radius of the Earth and removed transfers that passed through the Earth. We also performed a series of convergence tests on the virtual TLE generated for each departure state and omitted transfers that failed. This yielded the number of transfers for each orbital regime listed in Table 5.

4. Debris-to-debris transfer simulation results

We provide an overview of the results obtained by generating a large set of transfers for each orbital regime. In Section 4.1, results are presented for the

Table 5: Summary of the ΔV cut-off employed for each orbital regime and the resulting number of transfers propagated using the SGP4 propagator.

Orbital regime	ΔV cut-off [km/s]	# of transfers
LEO	0.864	472,481
GTO	2.273	370,679
GSO	0.147	461,714

transfers computed using the pruned catalog of RSOs in LEO. Subsequently, in Section 4.2, we present an overview of the transfers computed for the set of R/Bs in GTO. In Section 4.3, we summarize the results obtained from generating a scan of transfers between the set of telecommunication satellites selected in GSO. Finally, in Section 4.4, we present an analysis of the results and comment on their implication for MTADR preliminary mission design.

4.1. Low Earth Orbit

Table 6 provides an overview of the top ten Lambert transfers in terms of lowest transfer ΔV for the LEO TLE catalog. Additionally, the arrival position and velocity errors generated by propagating the transfers using the SGP4 propagator are provided. A number of interesting features can be noted from this data. Firstly, several objects appear multiple times in the top ten list, suggesting that they could lie in the neighborhood of optima for individual transfer legs. Secondly, both the position and velocity errors do not increase monotonically as the transfer ΔV increases. Finally, the number of complete revolutions ranges from three to five. Although the transfer ΔVs are computed using the Lambert solver, since perturbations affect the departure and arrival positions, they could play a role in determining the transfers that require the least ΔV.

Figure 6 illustrates the position and velocity errors generated for the selected transfers in LEO (Table 5). If the effect of orbital perturbations is to fully randomize the arrival position and velocity, which means that the vector components are uncorrelated and drawn from normal distributions with zero mean, it would be expected that the histograms would approximate the Rayleigh distribution (Press et al., 2007). There are however five peaks visible in Figures 6a and 6b. These features correlate to the number of complete revolutions and is a signature for the periodic nature of the perturbations affecting the transfers. Both histograms illustrate that the ranges of position and velocity errors grow as the transfer revolutions increase. Interestingly,

Table 6: Ranking of top ten D2D transfers for the LEO catalog subset, ranked in terms of lowest transfer ΔV. Each transfer is defined by a departure object ($\mathrm{ID_{dep}}$), an arrival object ($\mathrm{ID_{arr}}$), a departure epoch (t_{dep}), given as the time since $t_0 = 57399.0$ MJD, a time-of-flight (t_{TOF}) and the number of complete revolutions (n). The magnitude of the arrival position ($\|\delta\bar{r}_2\|$) and velocity ($\|\delta\bar{v}_2\|$) errors vectors, generated by propagating the departure state using the SGP4 propagator, are also included.

$\mathrm{ID_{dep}}$	$\mathrm{ID_{arr}}$	t_{dep} [s]	t_{TOF} [s]	n	ΔV [km/s]	$\|\delta\bar{r}_2\|$ [km]	$\|\delta\bar{v}_2\|$ [km/s]
18467	13574	10368	27878	4	0.0296	634	0.648
26236	26280	7344	29170	4	0.0435	471	0.481
20884	16615	7344	28524	4	0.0454	333	0.338
17121	16615	3672	31108	5	0.0544	573	0.608
20852	23258	4536	24002	3	0.0567	131	0.127
17121	20884	10584	31754	5	0.0581	501	0.522
13570	13580	5616	31108	5	0.0638	86	0.096
12170	12166	216	31754	5	0.0757	156	0.165
26280	26236	7344	23356	3	0.0767	224	0.226
21911	23259	7776	30462	5	0.0769	240	0.256

the 4-revolution and 5-revolution cases also include transfers with relatively small position and velocity errors. It should be noted that applying the Δ V cut-offs (Table 5) skews the number of data points for each n-revolution set. This means that the stacked histograms presented here neglect high ΔV transfers. Since we are analyzing the data for MTADR mission design, we consider these histograms to represent the error distributions for the subset of "feasible" transfers legs.

Table 7 provides a summary of some of the key statistics resulting from the arrival position and velocity error distributions. For both the arrival position and velocity, the minimum and maximum errors differ by several orders of magnitude, highlighting the broad range of effects of perturbations on the transfers.

As was previously mentioned, we conducted numerical experiments through which we determined that the effect of drag on these error distributions was negligble. This is due to the fact that we simulated short transfers in LEO. Hence, the results presented here are based on omitting drag from the dynamical model.

Table 7: Arrival position and velocity error statistics for the LEO catalog subset.

Parameter	Value
mean $\|\delta\bar{r}_2\|$ [km]	259
median $\|\delta\bar{r}_2\|$ [km]	207
minimum $\|\delta\bar{r}_2\|$ [km]	0.0749
maximum $\|\delta\bar{r}_2\|$ [km]	731
interquartile range $\|\delta\bar{r}_2\|$ [km]	327
mean $\|\delta\bar{v}_2\|$ [km/s]	0.268
median $\|\delta\bar{v}_2\|$ [km/s]	0.214
minimum $\|\delta\bar{v}_2\|$ [km/s]	0.000278
maximum $\|\delta\bar{v}_2\|$ [km/s]	0.775
interquartile range $\|\delta\bar{v}_2\|$ [km/s]	0.337

4.2. Geostationary Transfer Orbit

Similar to the previous section, Table 8 summarizes the best transfers obtained with the Lambert solver for the GTO TLE catalog. The table also includes the arrival position and velocity errors derived by propagating the departure states using the SGP4 propagator. A number of differences can be noted with respect to the LEO results. Firstly, the transfers listed in Table 8 include cases with zero to five revolutions with no obvious pattern. Secondly, the position and velocity errors vary by approximately two orders-of-magnitude whilst the ΔV increases steadily. This can be related to the semi-major axis, eccentricity and inclination, which all directly affect the strength of perturbations acting on the transfers (Battin, 1999). In addition, the argument of perigee and RAAN play a crucial role in randomizing the objective-function space.

Figure 7 illustrates the distribution of the arrival position and velocity errors for the selected transfers in GTO (Table 5). Compared to Figure 6, the histograms in Figure 7 are more regular. Figures 7a and 7b both show one overall peak and a long tail (the tail has been truncated). As for the LEO case, the arrival errors generally increase as the number of revolutions increase. However, in this case it is clear that the distributions for each revolution contain single defined peaks. The distributions are shifted along the x-axis: the greater the number of revolutions, the further the distributions are shifted. In contrast to the LEO case, it seems that the errors are more directly a function of the number of revolutions in GTO. Hence, the 4-

Table 8: Ranking of top ten D2D transfers for the GTO catalog subset, ranked in terms of lowest transfer ΔV. Each transfer is defined by a departure object ($\mathrm{ID_{dep}}$), an arrival object ($\mathrm{ID_{arr}}$), a departure epoch (t_{dep}), given as the time since $t_0 = 57399.0$ MJD, a time-of-flight (t_{TOF}) and the number of complete revolutions (n). The magnitude of the arrival position ($\|\delta\bar{r}_2\|$) and velocity ($\|\delta\bar{v}_2\|$) errors vectors, generated by propagating the departure state using the SGP4 propagator, are also included.

$\mathrm{ID_{dep}}$	$\mathrm{ID_{arr}}$	t_{dep} [s]	t_{TOF} [s]	n	ΔV [km/s]	$\|\delta\bar{r}_2\|$ [km]	$\|\delta\bar{v}_2\|$ [km/s]
9927	9850	20160	185774	3	0.106	718	0.395
9850	9927	36000	194410	4	0.195	349	0.0826
11909	10155	8640	21690	0	0.207	33	0.00333
10155	11909	8640	21690	0	0.213	35	0.00341
6779	16294	54720	194410	4	0.279	181	0.0386
18953	16657	56160	181456	5	0.280	210	0.0738
9980	12159	2880	177138	4	0.280	103	0.00700
13075	8462	41760	155548	3	0.302	119	0.00972
16657	18953	40320	203046	5	0.303	404	0.0997
12159	9980	1440	86460	2	0.306	67	0.00429

revolution and 5-revolution transfers generate relatively large arrival position and velocity errors.

Some statistics for the arrival position and velocity errors are given in Table 9. The mean position and velocity errors are smaller compared to the LEO case, however the maximum values are significantly larger. This can be understood by the fact that the ΔV cut-off adopted for GTO (see Table 5) is a lot higher.

4.3. Geosynchronous Orbit

Here we summarize the results obtained by generating transfers for the GSO TLE catalog subset. Table 10 lists the transfer input parameters, the ΔVs and the arrival position and velocity errors. In all cases, the transfers make use of the maximum allowable number of complete revolutions (five), which is markedly different compared to the other two cases. The transfer list is populated by ten objects that occur multiple times. Although the departure epoch does vary, the time-of-flight values are relatively constant. The position and velocity errors do not change significantly across the list.

The distributions of the arrival position and velocity errors are given in Figure 8 for the selected transfers in GSO (Table 5). Figures 8a and 8b display a number of distinct peaks. As for the GTO case, the distributions for

Table 9: Arrival position and velocity error statistics for the GTO catalog subset.

Parameter	Value
mean $\|\|\delta\bar{r}_2\|\|$ [km]	115
median $\|\|\delta\bar{r}_2\|\|$ [km]	88
minimum $\|\|\delta\bar{r}_2\|\|$ [km]	0.232
maximum $\|\|\delta\bar{r}_2\|\|$ [km]	3064
interquartile range $\|\|\delta\bar{r}_2\|\|$ [km]	100
mean $\|\|\delta\bar{v}_2\|\|$ [km/s]	0.0180
median $\|\|\delta\bar{v}_2\|\|$ [km/s]	0.00906
minimum $\|\|\delta\bar{v}_2\|\|$ [km/s]	6.19×10^{-5}
maximum $\|\|\delta\bar{v}_2\|\|$ [km/s]	2.42
interquartile range $\|\|\delta\bar{v}_2\|\|$ [km/s]	0.0141

Table 10: Ranking of top ten D2D transfers for the GSO catalog subset, ranked in terms of lowest transfer ΔV. Each transfer is defined by a departure object ($\mathrm{ID_{dep}}$), an arrival object ($\mathrm{ID_{arr}}$), a departure epoch (t_{dep}), given as the time since $t_0 = 57399.0$ MJD, a time-of-flight (t_{TOF}) and the number of complete revolutions (n). The magnitude of the arrival position ($\|\delta\bar{r}_2\|$) and velocity ($\|\delta\bar{v}_2\|$) errors vectors, generated by propagating the departure state using the SGP4 propagator, are also included.

$\mathrm{ID_{dep}}$	$\mathrm{ID_{arr}}$	t_{dep} [s]	t_{TOF} [s]	n	ΔV [km/s]	$\|\|\delta\bar{r}_2\|\|$ [km]	$\|\|\delta\bar{v}_2\|\|$ [km/s]
24812	27445	86400	508034	5	0.00210	87	0.00625
26900	27380	117504	508034	5	0.00273	105	0.00762
27445	24812	6912	497668	5	0.00342	87	0.00616
26900	27513	20736	508034	5	0.00342	103	0.00753
27380	26900	20736	435472	5	0.00368	86	0.00630
27954	28790	114048	508034	5	0.00381	106	0.00772
27954	27715	24192	508034	5	0.00383	105	0.00767
27513	26900	17280	435472	5	0.00398	84	0.00614
24812	28702	134784	508034	5	0.00417	87	0.00625
27683	27438	38016	508034	5	0.00418	83	0.00603

Table 11: Arrival position and velocity error statistics for the GSO catalog subset.

Parameter	Value
mean $\|\|\delta\bar{r}_2\|\|$ [km]	57
median $\|\|\delta\bar{r}_2\|\|$ [km]	58
minimum $\|\|\delta\bar{r}_2\|\|$ [km]	0.471
maximum $\|\|\delta\bar{r}_2\|\|$ [km]	121
interquartile range $\|\|\delta\bar{r}_2\|\|$ [km]	32
mean $\|\|\delta\bar{v}_2\|\|$ [km/s]	0.00412
median $\|\|\delta\bar{v}_2\|\|$ [km/s]	0.00421
minimum $\|\|\delta\bar{v}_2\|\|$ [km/s]	3.80×10^{-5}
maximum $\|\|\delta\bar{v}_2\|\|$ [km/s]	0.907
interquartile range $\|\|\delta\bar{v}_2\|\|$ [km/s]	0.00231

each revolution number are distinct: as the number of revolutions increases, the distributions are shifted along the x-axis towards larger arrival errors. These figures suggest that for MTADR mission design in GSO, it might be possible to make use of a rule-of-thumb to determine the average effect of perturbations on the transfer as a direct function of the number of revolutions. This is significantly different to the LEO case, which exhibits a more complicated structure within the stacked histrograms.

Table 11 provides a summary of some key statistics for the arrival errors for this case. Compared to the LEO and GTO cases, the arrival state errors are much smaller for GSO, despite the fact that the time-of-flight is significantly longer.

4.4. Analysis of results

Here we present an analysis of the results obtained through propagation of the D2D transfers using the SGP4 propagator to assess the impact of orbital perturbations. Figure 9 summarizes the outcome of our study using box plots. Figure 9a provides an illustration of the effect of orbital perturbations on the arrival position as a function of the three orbital regimes investigated. The complementary box plots that describe the impact of perturbations on the arrival velocity are given in Figure 9b.

The box plots highlight a number of features. Firstly, for both the arrival position and velocity errors, the median values are the largest for LEO. The median drops off sharply for GTO and GEO (note that the y-axis for both plots is on a log-scale). Secondly, the whiskers illustrate the fact that

the spread of the data is similarly largest for LEO and tapers off for GTO and GEO. This systematic relationship between semi-major axis and arrival state error is best understood as reflecting the effect of non-sphericity on the transfers. In LEO, the perturbative effect of non-sphericity is the greatest and leads to complex error histograms. As the effect tapers off for GTO and GEO, the stacked histograms also show greater direct correlation to the number of revolutions. One noticeable difference for the GTO data set is the band of outliers that lie beyond the upper whisker. These data points are identified as being standard outliers, as they lie beyond the following thresholds[7]: $Q_3 + 1.5 \times \text{IQR}$ and $Q_1 - 1.5 \times \text{IQR}$, where Q_1 is the lower (first) quartile, Q_3 is the upper (third) quartile and IQR is the inter-quartile range. The outliers are associated with large ΔVs; in the GTO case the cut-off was set to a value that is significantly larger than for the other two cases (Table 5). The fact that there is a continuous band of outliers highlights the long tail for the GTO distribution.

The box plots serve as an interesting reference for MTADR mission design, as they provide insight into when it might be important to consider orbital perturbations and the extent to which design margins can be utilized. The results however cannot be considered as global statements, as they are clearly a function of the input parameters for the individual transfers, such as departure epoch, time-of-flight, etc. The degree of variance of the results in LEO is particularly interesting, as it suggests that optimal D2D transfers in this regime are sensitive to the input parameters. Furthermore, the band of outliers in the GTO data set suggest that when dealing with large ΔVs, it is particularly important to assess the impact of orbital perturbations. The arrival error histograms for GSO imply that it might be possible to derive a rule-of-thumb to determine the extent to which perturbations affect transfers, as a function of the time-of-flight or number of revolutions. Further research is necessary to ascertain the whether these quantitative results represent general principles that could serve as guidelines for MTADR mission planning.

5. Conclusions

The study presented in this paper provides a quantitative assessment of the effect of orbital perturbations on D2D transfers within three different

[7] http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.boxplot

orbital regimes. The aim of the paper was to address the assumption that orbital perturbations can be neglected for MTADR mission planning, by quantifying the resulting errors derived from large-scale simulation.

For the purposes of our investigation, we generated pruned TLE catalogs, containing a hundred objects in LEO, GTO and GSO. Using each of these catalog subsets, we computed one-to-one transfers using the Lambert solver, for specified departure epoch and time-of-flight bounds. This yielded large data sets that were used to identify the best legs that would serve as candidate transfers for preliminary sequencing.

To quantify the effect of orbital perturbations on these transfers, we employed the SGP4/SDP4 dynamical models (SGP4 propagator). These models enabled us to rapidly propagate the best transfers generated using the Lambert solver. We introduced the concept of a "virtual" TLE and an algorithm to convert a given Cartesian state and associated epoch to a representative TLE, to allow use of the SGP4 propagator. Virtual TLEs were generated for each departure state and used to propagate the transfers under the effects of orbital perturbations. By propagating the transfers using the specified departure epoch and time-of-flight and comparing the computed arrival state to the target arrival state provided to the Lambert solver, we were able to systematically quantify the position and velocity errors caused by orbital perturbations.

Our results provide insight into the effects of perturbations on D2D transfers in near-Earth space. The most pronounced effect was seen in LEO, where the arrival position error can reach $\mathcal{O}(100)$ km and the arrival velocity error can reach $\mathcal{O}(0.1)$ km/s. The statistical distributions for the arrival position and velocity errors include a number of distinct features. We illustrated the relationship of these features with the number of revolutions. The mean position and velocity errors in LEO are relatively large. Our results indicate that as semi-major axis increases, the absolute arrival state errors decrease. The GTO case highlights the importance of judiciously considering the effect of perturbations for large transfer ΔVs, as the resulting errors can be significant. The smallest errors were witnessed in GSO, despite the fact that the time-of-flight for the transfers was the longest. Additionally, the GSO results suggest that it might be possible to derive an effective rule-of-thumb to assess the impact of perturbations on transfer legs.

Further investigation is required to fully understand the impact of orbital perturbations on transfers legs for MTADR missions, hence the preliminary mission design process. Moreover, our quantitative analysis provides the

starting point for more fundamental research to develop insight into the effects of perturbations as a function of a variety of model and input parameters. The complex nature of the results obtained suggests that ignoring perturbations for mission planning can lead to sub-optimal and possibly erroneous results. Hence, efforts should be geared towards developing a robust framework to include perturbations within the mission planning process from the start.

6. Acknowledgements

The research work here presented is supported by the Marie Curie Initial Training Network Stardust[8], FP7-PEOPLE-2012-ITN, Grant Agreement 317185. The authors would like to acknowledge the European Commission and the Research Executive Agency for their support and funding. Abhishek Agrawal and Enne Hekma would like to thank the Erasmus Programme for providing scholarships for their internships at Dinamica Srl. Finally, the authors would like to thank Roberto Armellin for his contribution towards conception of this research project.

References

D. Izzo, Revisiting Lamberts problem, Celestial Mechanics and Dynamical Astronomy 121 (2014) 1–15.

D. A. Vallado, P. Crawford, R. Hujsak, T. S. Kelso, Revisiting Spacetrack Report 3, AIAA/AAS Astrodynamics Specialist Conference 6753 (2006) 1–88.

R. Biesbroek, The e.Deorbit study in the concurrent design facility, in: Proceedings of the CleanSpace Workshop, Darmstadt, Darmstadt, 2012.

R. Biesbroek, T. Soares, J. Huesing, K. Wormnes, L. Innocenti, The e.deorbit CDF study: a design study for the safe removal of a large space debris, in: 64th International Astronautical Congress (IAC), Beijing, 2013.

[8]Stardust Programme: Advanced Research Network on Asteroid and Space Debris Manipulation, http://www.stardust2013.eu

V. J. Lappas, J. L. Forshaw, L. Visagie, A. Pisseloup, T. Salmon, E. Joffre, T. Chabot, I. Retat, R. Axthelm, S. Barraclough, A. Ratcliff, A. Bradford, H. Kadhem, N. Navarathinam, J. Rotteveel, C. Bernal, F. Chaumette, A. Pollini, W. H. Steyn, Removedebris: an EU low cost demonstration mission to test adr technologies, Proceedings of the 65th International Astronautical Congress (2014).

J. Forshaw, G. Aglietti, N. Navarathinam, H. Kadhem, T. Salmon, E. Joffre, T. Chabot, I. Retat, R. Axthelm, S. Barraclough, J. Forshaw, G. Aglietti, N. Navarathinam, H. Kadhem, T. Salmon, An in-orbit active debris removal mission - REMOVEDEBRIS: Pre-Launch update, Proceedings of the 66th International Astronautical Congress, Jerusalem, Israel (2015) 1–15.

B. Esmiller, C. Jacquelard, CLEANSPACE "small debris removal by laser illumination and complementary technologies", AIP Conference Proceedings 1402 (2011) 347–353.

M. Richard, L. G. Kronig, F. Belloni, V. Gass, O. A. Araromi, H. Shea, C. Paccolat, J.-P. Thiran, Others, Uncooperative rendezvous and docking for microsats, in: 6th International Conference on Recent Advances in Space Technologies, Istanbul, Turkey, 2013. URL: https://infoscience.epfl.ch/record/187720/files/CleanSpaceOneRASTpaperJune2013.pdf.

J.-C. Liou, N. Johnson, N. Hill, Controlling the growth of future LEO debris populations with active debris removal, Acta Astronautica 66 (2010) 648–653.

J.-C. Liou, An active debris removal parametric study for LEO environment remediation, Advances in Space Research 47 (2011) 1865–1876.

C. Bonnal, J.-M. Ruault, M.-C. Desjean, Active debris removal: Recent progress and current trends, Acta Astronautica 85 (2013) 51–60.

T. Kelso, Analysis of the Iridium 33 Cosmos 2251 Collision, Technical Report, 2009. URL: http://www.agi.com/resources/user-resources/downloads/white-paper.aspx?id=18.

A. Tan, T. X. Zhang, M. Dokhanian, Analysis of the iridium 33 and cosmos 2251 collision using velocity perturbations of the fragments, Advances in Aerospace Science and Applications 3 (2013) 13–25.

J. N. Pelton, New Solutions for the Space Debris Problem, SpringerBriefs in Space Development, Springer International Publishing, Cham, 2015. URL: http://link.springer.com/10.1007/978-3-319-17151-7. doi:10.1007/978-3-319-17151-7.

B. W. Barbee, S. Alfano, E. Pinon, K. Gold, D. Gaylor, Design of spacecraft missions to remove multiple orbital debris objects, in: 2011 Aerospace Conference, IEEE, 2011, p. 14. doi:10.1109/AERO.2011.5747303.

F. Zuiani, M. Vasile, Preliminary Design of Debris Removal Missions by Means of Simplified Models for Low-Thrust, Many-Revolution Transfers, International Journal of Aerospace Engineering 2012 (2012) 1–22.

D. Izzo, I. Getzner, D. Hennes, L. F. Simões, Evolving Solutions to TSP Variants for Active Space Debris Removal, Genetic and Evolutionary Computation Conference (GECCO 2015) (2015) 1207–1214.

S. Peters, H. Fiedler, R. Forstner, ADReS-A: Mission Architecture for the Removal of SL-8 Rocket Bodies, in: IEEE Aerospace Conference Proceedings, 2015.

G. Laporte, The traveling salesman problem: An overview of exact and approximate algorithms, European Journal of Operational Research 59 (1992) 231–247.

K. T. Alfriend, D.-J. Lee, N. G. Creamer, Optimal Servicing of Geosynchronous Satellites, Journal of Guidance, Control, and Dynamics 29 (2006) 203–206.

M. Holzinger, D. Scheeres, On-Orbit Range Set Applications, in: Advanced Maui Optical and Space Surveillance Technologies Conference, Maui Economic Development Board Maui, HI, 2011, pp. 61–70.

B. A. Conway, C. M. Chilan, B. J. Wall, Evolutionary principles applied to mission planning problems, Celestial Mechanics and Dynamical Astronomy 97 (2007) 73–86.

B. J. Wall, B. A. Conway, Genetic algorithms applied to the solution of hybrid optimal control problems in astrodynamics, Journal of Global Optimization 44 (2009) 493–508.

E. Lancaster, R. Blanchard, A Unified Form of Lambert's Theorem, Technical Report, NASA, Greenbelt, Maryland, USA, 1969.

R. H. Gooding, A procedure for the solution of Lambert's orbital boundary-value problem, Celestial Mechanics and Dynamical Astronomy 48 (1990) 145–165.

T. A. Leite Pinto Secretin, Design of a Combinatorial Tool for Preliminary Space Mission Analysis, applied to the GTOC2 Problem, Master's thesis, Delft University of Technology, Kluyverweg 1, 2629HS, Delft, The Netherlands, 2012.

D. Madakat, J. Morio, D. Vanderpooten, Biobjective planning of an active debris removal mission, Acta Astronautica 84 (2013) 182–188.

F. R. Hoots, P. W. Schumacher Jr., R. a. Glover, History of Analytical Orbit Modeling in the U. S. Space Surveillance System, Journal of Guidance, Control, and Dynamics 27 (2004) 174–185.

N. Van Der Pas, J. Lousada, C. Terhes, M. Bernabeu, W. Bauer, Target selection and comparison of mission design for space debris removal by DLRs advanced study group, Acta Astronautica 102 (2014) 241–248.

J. R. Quinlan, C. A. Jones, V. Vittaldev, A. Wilhite, On the Design of an Active Debris Removal Architecture for Low Earth Orbit Space Debris Remediation, AIAA SPACE 2011 Conference & Exposition 27 - 29 September 2011, Long Beach, California AIAA (2011) 1–15.

V. Braun, A. Lüpken, S. Flegel, J. Gelhaus, M. Möckel, C. Kebschull, C. Wiedemann, P. Vörsmann, Active debris removal of multiple priority targets, Advances in Space Research 51 (2013) 1638–1648.

O. Volk, Johann Heinrich Lambert and the Determination of Orbits for Planets and Comets, Celestial Mechanics 21 (1980) 237–250.

B. A. Conway, J. E. Prussing, Orbital Mechanics, Oxford University Press, New York, NY, USA, 1993.

R. H. Battin, An Introduction to the Mathematics and Methods of Astrodynamics, 1999. URL: http://books.google.com.hk/books?id=0jH7aVhiGdcC. doi:10.2514/4.861543.

F. R. Hoots, R. L. Roehrich, Spacetrack Report No. 3, Spacetrack Report 3 (1980).

M. J. D. Powell, A FORTRAN subroutine for solving systems of nonlinear algebraic equations., Technical Report, Atomic Energy Research Establishment, Harwell (England), 1968.

M. J. D. Powell, A hybrid method for nonlinear equations, Numerical methods for nonlinear algebraic equations 7 (1970) 87–114.

C. Eagle, NORAD Propagators and Two Line Element Sets, Technical Report, 2001. URL: https://www.dropbox.com/sh/19aym46s6jijtnj/AADKqzXIWRngkG-oGwyKlcVca/Two-Line-Elements.pdf?dl=0http://www.cdeagle.com/pdf/tnote2.pdf.

B.-S. Lee, NORAD TLE Conversion From Osculating Orbital Element, Journal of Astronomy & Space Sciences 19 (2002) 395–402.

W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, Numerical recipes 3rd edition: The art of scientific computing, Cambridge University Press, 2007. URL: http://www.cambridge.org/nl/academic/subjects/mathematics/numerical-recipes/numerical-recipes-art-scientific-computing-3rd-edition.

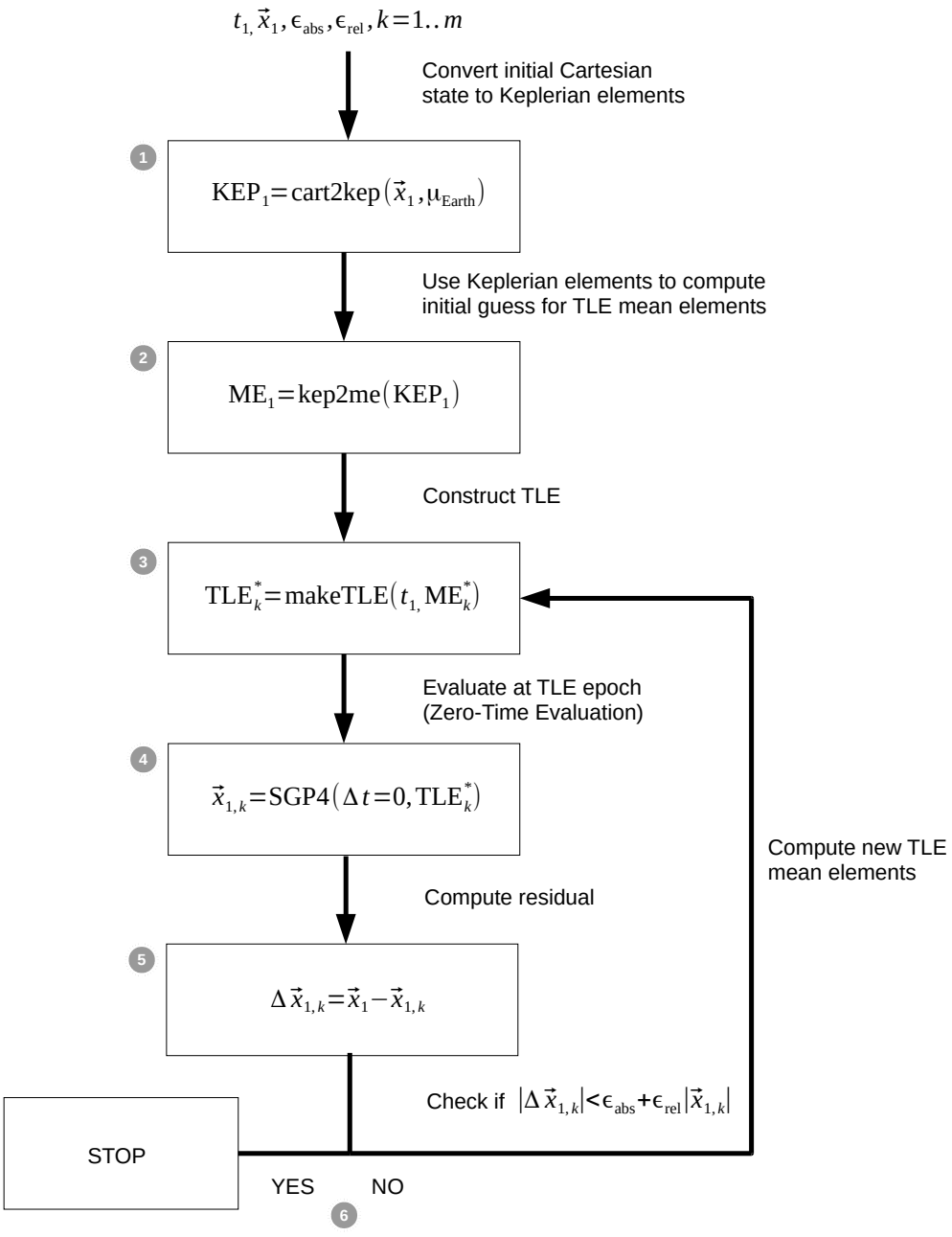
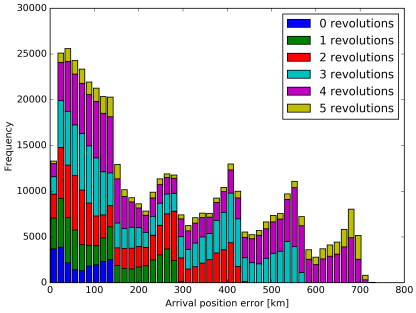
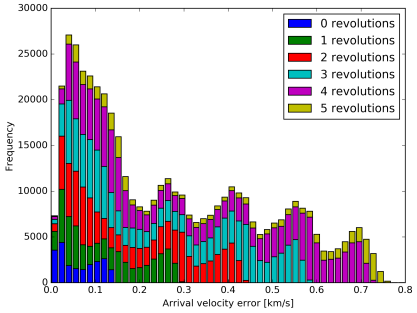


Figure 5: Schematic illustration of the Cartesian-to-TLE conversion algorithm.

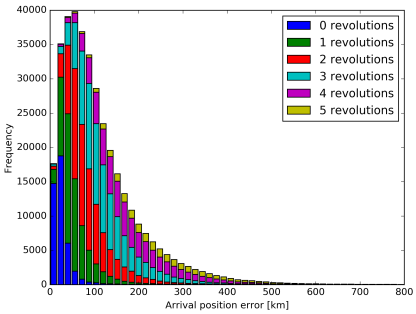


(a) Arrival position error magnitude

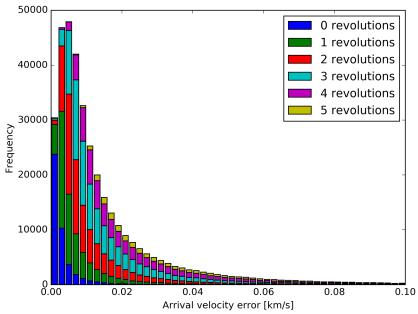


(b) Arrival velocity error magnitude

Figure 6: Stacked histograms of arrival position and velocity errors (magnitude and components) for the LEO TLE catalog, illustrating distribution of errors as a function of number of transfer revolutions.

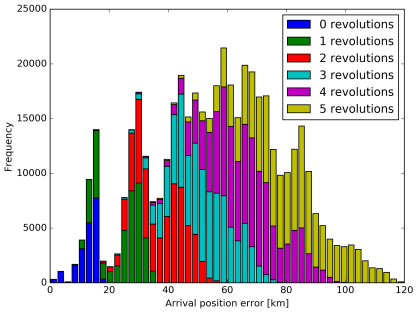


(a) Arrival position error magnitude

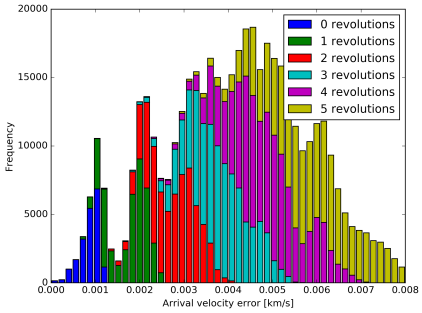


(b) Arrival velocity error magnitude

Figure 7: Stacked histograms of arrival position and velocity errors (magnitude and components) for the GTO TLE catalog, illustrating distribution of errors as a function of number of transfer revolutions.

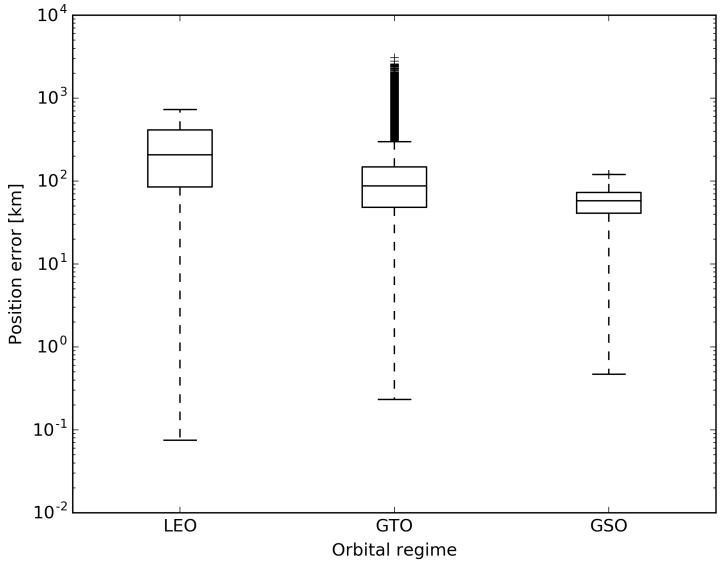


(a) Arrival position error magnitude

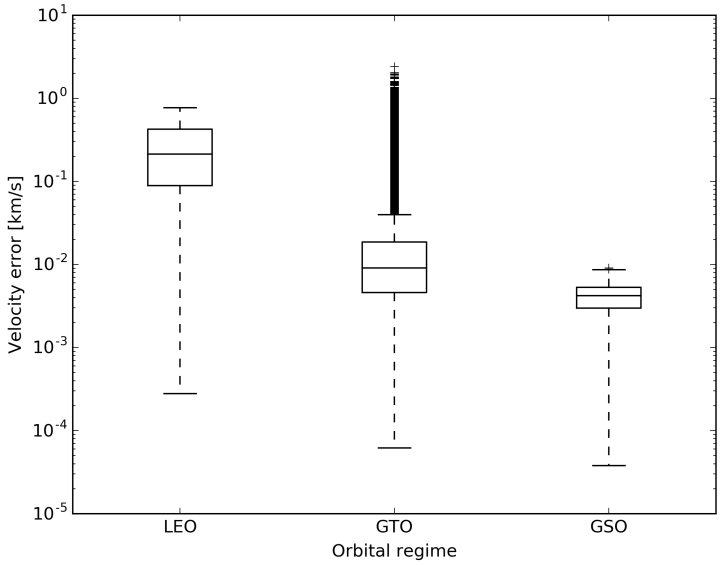


(b) Arrival velocity error magnitude

Figure 8: Stacked histograms of arrival position and velocity errors (magnitude and components) for the GSO TLE catalog, illustrating distribution of errors as a function of number of transfer revolutions.



(a) Arrival position error vs orbital regime



(b) Arrival velocity error vs orbital regime

Figure 9: Box plot representation of arrival position and velocity errors as a function of orbital regime.