Accepted Manuscript

Optimising filtering of two-line element sets to increase re-entry prediction accuracy for GTO objects

Aleksander A. Lidtke, David J. Gondelach, Roberto Armellin

PII: S0273-1177(18)30787-7
DOI: <https://doi.org/10.1016/j.asr.2018.10.018>
Reference: JASR 13973

To appear in: *Advances in Space Research*

Received Date: 26 March 2018
Accepted Date: 12 October 2018

Please cite this article as: Lidtke, A.A., Gondelach, D.J., Armellin, R., Optimising filtering of two-line element sets to increase re-entry prediction accuracy for GTO objects, *Advances in Space Research* (2018), doi: <https://doi.org/10.1016/j.asr.2018.10.018>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Optimising filtering of two-line element sets to increase re-entry prediction accuracy for GTO objects

Aleksander A. Lidtke*

*Department of Integrated System Engineering, Kyushu Institute of Technology,
Kitakyushu, Japan*

David J. Gondelach*

Surrey Space Centre, University of Surrey, Guildford GU2 7XH, UK

Roberto Armellin*

Surrey Space Centre, University of Surrey, Guildford GU2 7XH, UK

Abstract

Predicting re-entry epoch of space objects enables managing the risk to ground population. Predictions are particularly difficult for objects in highly-elliptical orbits, and important for objects with components that can survive re-entry, e.g. rocket bodies (R/Bs). This paper presents a methodology to filter two-line element sets (TLEs) to facilitate accurate re-entry prediction of such objects. Difficulties in using TLEs for precise analyses are highlighted and a set of filters that identifies erroneous element sets is developed. The filter settings are optimised using an artificially generated TLE time series. Optimisation results are verified on real TLEs by analysing the automatically found outliers for exemplar R/Bs. Based on a study of 96 historical re-entries, it is shown that TLE filtering is necessary on all orbital elements that are being used in a given analysis in order to avoid considerably inaccurate results.

Keywords: TLE, two-line element set, filtering, outliers, optimisation,

*Corresponding author

Email addresses: `lidtke.aleksander-andrzej578@mail.kyutech.jp` (Aleksander A. Lidtke), `d.gondelach@surrey.ac.uk` (David J. Gondelach), `r.armellin@surrey.ac.uk` (Roberto Armellin)

re-entry prediction, GTO, geostationary transfer orbit

1. Introduction

Upper stages of rockets are large objects, which contain components that are known to be able to survive atmospheric re-entry. Such surviving material, for example propellant tanks, will impact Earth's surface and might cause casualties. Being able to predict re-entry in advance enables mitigation measures to be implemented to reduce the risk to the ground population. However, at present, re-entry can be predicted with an accuracy of 2 to 28% of the remaining lifetime in orbit (Pardini & Anselmo, 2012). A “rule of thumb” relative re-entry prediction uncertainty of $\pm 20\%$ is recommended by Pardini & Anselmo (2012), which is too low to enable efficient risk mitigation measures to be implemented because of the large area over which the spacecraft might re-enter.

Generally, re-entry prediction is done by propagating an object until it reaches the altitude where the atmospheric break-up occurs, which is typically between 72 and 84 km (Klinkrad, 2010). The main steps of this approach are determining the object's initial orbit and accurately modelling the forces that act on it. The latter task requires object's physical parameters, e.g. mass, to be known or estimated.

The inaccuracies associated with re-entry prediction have several different origins, which are well reviewed by Pardini & Anselmo (2013) and references therein. The largest source of uncertainty is associated with modelling of the acceleration due to atmospheric drag, a_D , an object of mass m , cross-sectional area A and drag coefficient C_D , flying through a fluid of density ρ at speed v , experiences. This acceleration is given in Eq. 1 (Vallado, 2013). The object's physical parameters in this equation, $C_D A/m$, are often combined into a single “ballistic coefficient”, BC. Note that speed v is relative to the fluid, so that the rotation of the atmosphere and the direction of the fluid w.r.t. the orbital velocity also need to be taken into account.

$$a_D = \frac{1}{2} \rho v^2 C_D \frac{A}{m}. \quad (1)$$

Based on Eq. 1, it can be deduced that the more accurately the object's parameters are known, the more accurate the force modelling and hence the re-entry prediction is. The same holds true for the acceleration due to solar radiation pressure, which depends on object's physical parameters that are

often combined into the solar radiation pressure coefficient, SRPC (Vallado, 2013). However, solar radiation pressure is negligible compared to the drag force close to the re-entry (King-Hele & Walker, 1987), and thus the SRPC estimation is ignored in the re-entry predictions done here (Gondelach et al., 2017). The initial state of the object also has to be known and propagated as accurately as possible in order to, amongst others, correctly predict its position, and hence also the local atmospheric density ρ and orbital velocity v . Lastly, an accurate atmospheric model is required to correctly model ρ .

Currently, two-line element sets (TLEs) are the only publicly available data that can be used for re-entry prediction of a space object. However, there is a number of factors that, if left unaddressed, could reduce the accuracy of re-entry prediction based on TLEs:

1. The quality of TLEs of an object is not homogeneous. Sometimes TLEs of low quality or even belonging to a different object (outliers) are published. Outlying TLEs are believed to arise due to, e.g. modelling simplifications, failures in automated TLE generation process, incorrect observations being included in the orbit fit, or human errors during manual orbit determination. TLE generation process is not made public, and thus it is difficult to capture all of the error sources as well as their possible effects on the element sets.
2. Occasionally, the object's physical properties (BC) or its orbit can be altered by collisions or explosions. Space weather phenomena might also change the interaction between the residual atmosphere and the object, and thus affect its C_D . Such space events render the TLEs of the object from before the event inapplicable to its new, altered state.
3. TLEs do not directly provide information on space object parameters, notably the BC. TLEs only include the B^* parameter that is the BC adjusted using the SGP4 reference atmospheric density, not the BC explicitly. More importantly, however, the B^* term is sometimes erroneous because it is the only way to represent forces not included in the SGP4 force model, e.g. solar radiation pressure (Picone, 2005). Thus, B^* is often said to 'soak up' force modelling errors (Vallado et al., 2006).
4. TLEs can only be propagated using the SGP4/SDP4 propagator (Hoots & Roehrich, 1988). However, this propagator is based on the Brouwer theory and, therefore, only models the largest perturbations affecting a satellite. The many assumptions of the theory and poor modelling

of non-conservative forces such as drag (Easthope, 2014) limit the accuracy of the resulting propagation (Hejduk et al., 2013), and thus of the re-entry prediction (Pardini & Anselmo, 2013).

5. TLEs are not supplied with uncertainty information, e.g. a covariance matrix. It is thus challenging to estimate the accuracy with which the re-entry is predicted based on these ephemerides.

In order to overcome these difficulties in TLE-based re-entry prediction, a multi-step procedure is proposed. The first step consists of analysing TLEs, with the goal of identifying outliers and space events (addresses points 1 and 2 from the above list). The filtered TLEs are then used to estimate the unknown spacecraft BC (point 3) and perform orbit determination (OD). OD allows the object's state and its orbit uncertainty to be propagated until the re-entry with a propagator more accurate than SGP4/SDP4 (points 4 and 5).

This paper presents the approach adopted to process the TLEs to improve the accuracy of re-entry prediction. The main steps of this “TLE filtering” are:

1. Filter out TLEs that were published but subsequently corrected;
2. Find large time gaps between TLEs because they hinder proper verification of TLE consistency;
3. Identify single TLEs with inconsistent mean motion, as well as entire sequences thereof, using a sliding window approach;
4. Filter out TLEs outlying in eccentricity;
5. Filter out TLEs outlying in inclination;
6. Filter out TLEs with negative B^* because they cause incorrect SGP4 propagation.

The remaining steps of the re-entry prediction algorithm are described by Gondelach et al. (2016), while Gondelach et al. (2017) elaborates on the methodology of estimating the ballistic coefficient BC. The re-entry prediction algorithm is tailored to rocket bodies (R/Bs) located in geostationary transfer orbits (GTO). A set of 116 R/Bs will be used as a reference sample of such objects. The space surveillance catalogue numbers (SSCs) of these R/Bs are given in Appendix A. Even though the TLE filtering algorithms have been created for the specific purpose of predicting GTO R/B re-entries, they remain applicable to other analyses that use TLEs as inputs. The applicability of the algorithms to other cases is discussed in greater detail in section 5.

Historically, TLE time series have been analysed mostly to detect space weather events or propulsive manoeuvres (Patera, 2008; Song et al., 2012). The latter is of particular interest from the space situational awareness point of view because it allows the end of life of a satellite to be approximately detected (Li & Chen, 2018). Various methodologies have been developed to detect such changes in TLE time series:

- sliding one (Patera, 2008; Lemmens & Krag, 2014) or two (Kelecy et al., 2007) windows through the TLE time series to detect changes between TLEs and the window or the two windows,
- analysing differences of Keplerian (Song et al., 2012) or equinoctial (Dolado-Perez et al., 2014) orbital elements between consecutive TLEs,
- comparing Cartesian state (Lemmens & Krag, 2014) or Keplerian orbital element (Li et al., 2018) differences between a TLE and preceding TLEs propagated to its epoch.

Some of the more recent methods address the presence of noise and outliers in the TLE time series by using robust regression (Lemmens & Krag, 2014), or smoothing the time series with RLOESS (Li & Chen, 2018) or discrete wavelet transform (Li et al., 2018) before proceeding with further analyses.

This work addresses the presence of outliers in the TLE time series by adapting the previously developed manoeuvre detection algorithms to detect both space events (e.g. collisions or explosions) and outlying TLEs. The development of a set of tools that allow outlying TLEs to be identified is a novel application of manoeuvre detection algorithms, the prototype of which was originally described by Lidtke et al. (2016). This paper elaborates on the details of the algorithms, which enable outlying TLEs to be identified in TLE time series of varying quality and generated in different phases of re-entry in section 2. An improvement w.r.t. the work by Lidtke et al. (2016), which enables fewer inclination and eccentricity outliers to be missed, is also stressed. Settings of the filters are then rigorously tuned using a genetic optimiser on exemplar TLE time series in section 3, which is another novel contribution of this paper. This tuning process enables the filter performance to be quantified, as opposed to the previous work. The optimised TLE filtering settings are analysed in more details in section 4. The improvement of re-entry prediction accuracy with varying filtering levels is discussed on several exemplar objects. The performance of filtering the TLE time series of these

objects is investigated in more detail to understand whether the optimisation provided results usable in real scenarios. Lastly, the effect of filtering is analysed on re-entry predictions made for 96 R/Bs that re-entered already, and conclusions are drawn. TLEs keep being used in applications as varied as attitude determination (Burton et al., 2017), atmospheric density estimation (Mendaza et al., 2017), orbital dynamics (San-Juan et al., 2017) and small satellite operations. Therefore, it is important to analyse the impact of TLE quality on the accuracy of TLE-based analyses, as well as develop means to improve their accuracy, e.g. via filtering.

2. Filter levels

This section gives details of all the steps of the TLE filtering algorithm employed before carrying out re-entry prediction. Finding outliers in mean motion is performed simultaneously with splitting of the TLE time series into internally consistent sequences. This is because this orbital parameter is the most likely to change if the object's physical properties change while its orbit passes through the atmosphere. A sequence is defined as a set of TLEs that does not contain any events, which might change object's physical properties (events are sequence endpoints). This algorithm also detects propulsive manoeuvres but those are not expected to occur for objects nearing the re-entry because they are likely derelict. If a large time gap is present in the history of the TLEs for a given object, it cannot be established whether any events took place during that time because TLEs cannot be reliably propagated from the beginning to the end of the time gap. Thus, large time gaps may also be endpoints of TLE sequences and a process for identifying those is described.

2.1. Corrections

It is not uncommon for a TLE to be released soon after a previous one when the orbital elements in the TLE have been corrected (Lemmens & Krag, 2014). Kelecy et al. (2007) filter TLEs so that only one element set is left in a 24-hour window, whereas Lemmens & Krag (2014) use half an orbital period. The time separations between consecutive TLEs of 116 R/Bs in GTO were calculated and normalised w.r.t. the orbital period, which was computed using the mean motion of the more recent TLE. A subset of these data, where the time separation was at most five orbital periods, is shown in Fig. 1a.

Most of the TLEs are updated at integer multiples of the orbital period. This supports the hypothesis that if a TLE is issued less than half an orbital period after the preceding one, it is a correction. If the update interval is between half of orbital period and one orbital period, the TLE could be an early release of the catalogue update. Therefore, in this paper half an orbital period is taken as the correction threshold. In other words, if two TLEs are separated by less time than half an orbital period, the more recent one is kept in the analysis and the older one is discarded. Should the more recent TLE be an outlier, it will be removed from the time series at a later stage of the filtering process.

The fact that TLE epochs are separated by integer multiples of orbital period is associated with the fact that, in the past, many TLEs were released when the objects were at their ascending nodes (Vallado et al., 2006). However, as can be noted in Fig. 2, this no longer seems to be the case for most of the GTO objects since approximately 2011. Had the objects been located at their ascending nodes at the epochs of their TLEs, their arguments of latitude would be equal to zero. This was the case for most of the TLEs published before 2011, but since then the spread in the position along the orbit at the TLE epoch is larger. This can also be seen in Fig. 1b, which shows that the time spacing of TLEs published after the beginning of 2011 is more dispersed than before. However, there still appears to be a concentration of TLE updates at the integer multiples of orbital period, therefore the suggested threshold remains valid.

2.2. Time gaps

Large time gaps make it impossible to determine whether an event has taken place, i.e. whether object's parameters changed. This is because the state from the beginning of the large time gap cannot be reliably propagated until the end of the gap. Thus, large time gaps are considered as sequence endpoints. Similarly, if a TLE is both preceded and followed by a large time gap, it is treated as an outlier because it cannot be reliably compared to any other TLE in the time series.

It is unclear for how long a TLE can be accurately propagated. A TLE older than five days might make automatic tracking of an object difficult (Hejduk et al., 2013). However, enhanced TLEs (eTLE) for highly-elliptical orbits (period greater than 225 minutes, eccentricity greater than 0.25) do not suffer almost any loss of accuracy c.f. numerically propagated states in the first week since the epoch of the element set (Hejduk et al., 2013). Therefore, no

(a) All TLEs

(b) TLEs published after 1 Jan 2011

Figure 1: Cumulative density function of the TLE update frequency for 116 GTO rocket bodies from Appendix A. Normalised to the orbital period computed using the more recent TLE. Showing a subset of the data of all TLEs with update frequency of at most five orbital periods. Also showing the same data but from the TLEs released after the beginning of 2011.

Figure 2: Sum of the argument of perigee and the true anomaly (i.e. the argument of latitude) of all the TLEs of the 116 GTO rocket bodies from Appendix A at the TLE epochs.

validity period of the TLEs can be definitively established and the duration of the large time gaps has to be estimated.

The time spacing between TLEs of one object will vary, sometimes by tens of orbital revolutions. Such large time gaps may be present in the history of TLEs of an object with otherwise approximately constant TLE update frequency. Moreover, the TLE update frequency will vary from object to object. Thus, the threshold for the duration of large time gaps has to be computed for every object individually. Extraordinarily large time gaps, orders of magnitude longer than the typical TLE update interval for a given object, have been observed. Therefore, it was imperative that robust statistical metrics (e.g. median and median absolute deviation instead of mean and standard deviation) be used to reduce the effect of such outlying data points on the time gap threshold (DeGroot & Schervish, 2014).

It was observed that the TLE update frequencies for most objects follow distributions, on which the extraordinarily large time gaps manifest themselves as tails disjoint from the rest of the distribution. Thus, identifying the TLE time separation corresponding to the beginning of this tail enables all large time gaps to be detected. This is advantageous compared with using

a threshold based on e.g. median plus a number of median absolute deviations (MADs), because it does not *a priori* assume that time gaps will be present. If the TLE time spacing distribution does not have a disjoint tail, it does not contain any unexpectedly large time gaps.

The tail of the distribution is found by first computing the time separations between the consecutive TLEs for a given object, as well as their median and MAD. Time separations in a percentile of choice, P_i , are binned in equally spaced bins. The width of the bins is defined as the median time separation plus a number of MADs, W_{MAD} . The first empty bin corresponds to the beginning of the tail of the distribution of the time separations, i.e. to the time separation at which the distribution stops being continuous. The performance of this algorithm is shown in Fig. 3a, where a large time gap in the TLE time series was identified. The corresponding distribution of the TLE time separations is shown in Fig. 3b.

Ultimately, the settings of the algorithm presented in Table 1 were used throughout this work. They were arrived at by a “trial and error” search, where the settings were changed and the performance on the set of objects from Appendix A was investigated. Because the beginning of a tail of a distribution cannot be unambiguously quantified, this filter was not rigorously tuned.

Table 1: Settings of the time gap filter arrived at by examining the performance of the algorithm on the sample of objects given in Appendix A.

Filter setting	Value
P_i	0.90
W_{MAD}	3

2.3. Outliers in mean motion and sequence endpoints

Events, which affect object’s physical properties, are detected by analysing the mean motion n contained in its TLEs because this parameter is proportional to object’s orbital energy (Vallado, 2013). Mean motion and related semi-major axis were used to detect propulsive manoeuvres by Song et al. (2012) and Kelecy et al. (2007), and manoeuvres in the order of centimetres per second were detected in 95% of the cases (Kelecy et al., 2007). Even though the objects nearing the re-entry are likely derelict, and so do not conduct manoeuvres, they interact with the atmosphere. Thus, if their physical

(a) TLE mean motion time series with sequences identified by time gaps

(b) Distribution of TLE time spacing

Figure 3: Large time gap around year 1993.97 found for object 22906 by analysing data in the 95th percentile by binning the time separations into bins with width equal to median plus one MAD. Showing both the TLE time series and the continuous distribution of the TLE time spacings, on which the beginning of the disjoint tail, i.e. the time gap, was found.

properties change so will the drag force acting on them. Consequently, the rate of change of mean motion, $\dot{n}$ will change abruptly and, before the next TLE is published, also their orbital energies and n . Therefore, space events will manifest themselves as discontinuities in the time series of n .

Single outlying TLEs and events are detected by sliding a window, containing a fixed number of TLEs (N_{TLE}), through the TLE time series. A polynomial of the chosen order is regressed through the TLEs contained in the window and the TLE following the window is compared to this polynomial ($i + 1$, where i is the last TLE in the window). Theil-Sen-Seigel robust linear regression¹ or bisquare weighted least-squares regression² are used in order to reduce the impact of the outliers, which may be present in the window, on the regressing function.

The threshold for identifying outliers and events is recomputed throughout the analysis (Lemmens & Krag, 2014; Dolado-Perez et al., 2014) rather than using e.g. a fixed number of standard deviations above the mean (Patera, 2008). Recomputing the threshold is advantageous because the orbit will naturally evolve and the same difference between a TLE and the regressing function at one instant in time might signify an outlier, but at other times it might correspond to a fast, natural evolution of the orbit. Also, the time spacing between the TLEs is not constant, as shown in section 2.1, and thus the differences between the regressing function and individual TLEs will vary throughout the lifetime of an object.

The tolerance for identifying outliers is set on a quantity called the relative threshold, T_R . T_R is the ratio between the difference in n of the $i + 1$ TLE and the regressing function (this difference is called the residual, Δ_A), and the change predicted between the i and $i + 1$ TLE, Δ_P . T_R is given in Eq. 2 and the quantities used to define it are shown in Fig. 4a.

$$T_R = \frac{\Delta_A}{\Delta_P}. \quad (2)$$

The predicted change, Δ_P , will be close to zero when the mean motion does not vary significantly between the first and the last TLEs in the sliding window. This will cause T_R to be amplified and, occasionally, fall above

¹ www.scikit-learn.org/stable/modules/generated/sklearn.linear_model.TheilSenRegressor.html

² www.statsmodels.org/dev/generated/statsmodels.robust.norms.TukeyBiweight.html

(a) Quantities used to compute T_R

(b) Quantities used to compute T_A

Figure 4: Definition of the residual Δ_A and the predicted change Δ_P between the last TLE in the sliding window (i) and the following TLE ($i+1$). The differences are computed using the value of the regressing function at the epoch of the last TLE in the window, $n_{REG}(t_i)$, not the value of that TLE. Used to define the relative threshold from Eq. 2. Also showing the value of the regressing function at the epoch of the last TLE in the window, $n_{REG}(t_i)$, used to define the absolute threshold from Eq. 3.

the set tolerance, even if the TLE following the window is not an outlier or a beginning of an event. In order to avoid such false positives, i.e. incorrectly marking TLEs as outliers, an absolute threshold, T_A , is used. The tolerances on both T_R and T_A need to be exceeded in order for a TLE to be classified as an outlier or a sequence endpoint. T_A is the previously defined residual, Δ_A , normalised by the regression mean motion at the epoch of the last TLE in the window, $n_{REG}(t_i)$, as per Eq. 3. The quantities used to define T_A are shown in Fig. 4b. Due to the normalisation, T_A is in fact a relative quantity. However, it is used to set a threshold on the absolute value of n , while the relative threshold refers to difference in changes from the last TLE in the window to the next.

$$T_A = \frac{\Delta_A}{n_{REG}(t_i)}. \quad (3)$$

As the window is slid through the TLE time series (i.e. the index of the last TLE in the window, i , is incremented), T_R and T_A are computed at every $i + 1$ TLE. If both T_R and T_A are above the set tolerances, $i + 2$ TLE is investigated. If the difference between the mean motions n_{i+1} and n_{i+2} of the $i + 1$ and $i + 2$ TLEs (Δ_{i+2}), given in Eq. 4, is less than the absolute tolerance, then those two TLEs are considered consistent and a start of a new sequence is declared. This situation is shown in Fig. 5. Note that due to how the absolute tolerance is defined, Δ_{i+2} has to be normalised by the regression mean motion at the $i + 1$ TLE, $n_{REG}(t_{i+1})$, so that the actual threshold is set on $T_{A,i+1}$.

$$T_{A,i+1} = \frac{n_{i+2} - n_{i+1}}{n_{REG}(t_{i+1})} = \frac{\Delta_{i+2}}{n_{REG}(t_{i+1})}. \quad (4)$$

2.4. Outliers in eccentricity and inclination

Once the corrected TLEs are filtered out, and the element sets with outlying mean motion and time separation are removed from the sample, some of the remaining TLEs may still have outlying orbital elements.

As far as the ballistic coefficient estimation is concerned, eccentricity e is particularly important because it affects the perigee altitude and thus also the drag force acting on the object. If the eccentricity in the TLE is incorrect, the estimated BC will be inaccurate, thus reducing the accuracy of re-entry prediction. Once the object parameters are estimated, orbit determination is performed. Individual TLEs (or orbital elements derived from them) are

Figure 5: Start of a new sequence after an event that affected the mean motion of the object. The $i + 1$ and $i + 2$ TLEs remain consistent, i.e. have Δ_{i+2} less than the absolute tolerance.

used as pseudo-observations in the OD process. It was found that TLEs outlying in inclination, i , may lead to poor estimates of the state, which is propagated until the re-entry, and consequently inaccurate re-entry predictions. Therefore, after filtering out outliers in n , TLEs with aberrant e and i also need to be removed from the sample before subsequent steps of re-entry prediction are performed.

Instead of filtering TLEs based on eccentricity directly, the perigee radius, r_P , is used. The two quantities are equivalent from the point of view of filtering because $r_P = (1 - e)a$, where a is the semi-major axis (Vallado, 2013). If the outliers in mean motion n are already filtered out, there should be no outliers in a present in the sample because $n = (\mu/a^3)^{1/2}$ (Vallado, 2013). Thus, any outliers in r_P will be present only due to outliers in e . It was found that filtering outliers in r_P is easier than using e directly due to scaling. Inclination extracted from the TLEs is used directly to filter out the outliers in this orbital element.

Outliers in inclination and eccentricity are identified within each sequence found according to the algorithm described in section 2.3, i.e. within sequences terminated by events and large time gaps. Because all the events have already been identified at this stage, a simpler filtering technique is used for i and r_P than for n . A window of a fixed length (N_{TLE}) is slid through the time series of the given orbital element in every consistent sequence, and the median value in the window is computed. This median value is subtracted

from the orbital element value of the TLE in the middle of the window, thus converting the time series of TLE values into a time series of differences (data are detrended). Another window of length N_{MAD} may be slid through the time series of differences, and dispersion of the differences may be computed for the TLEs in this sliding window. Alternatively, the dispersion may be computed from all the differences for a given object. Lastly, TLEs that have a given orbital element further than a fixed number of dispersion measures, MAD_{TH} , away from the median are marked as outliers. In the previous work by Lidtke et al. (2016), *dispersion* was quantified using the statistically robust median absolute deviation. However, during the rigorous filter tuning presented in section 3.4, it was discovered that the *median* absolute deviation might be 0 even if an outlier is present in the sliding window, provided that the other TLEs in the window are consistent. This would cause the outlier not to be filtered out. To avoid such false negatives, *mean* absolute deviation was used to quantify dispersion instead, which constitutes an improvement on the previous TLE filtering methodology.

The above filtering process may be repeated a number of times with different settings to filter more outliers in every orbital element. This filtering is done twice for eccentricity ('local' and 'global' filters) and once for inclination. This configuration of filter passes was devised by filtering the TLEs of the exemplar objects from Appendix A and manually seeking settings that filter the most outliers.

2.5. Outliers in B^*

TLEs contain a B^* parameter that is equivalent to the ballistic coefficient BC ($BC = 12.741621B^*$ (Vallado, 2013)). It was observed that even 30 to 60 days before the actual re-entry, when the perigee altitude is low and the atmospheric drag is the most prevalent force perturbing the satellite, TLEs may contain negative B^* values. Negative B^* , corresponding to negative ballistic coefficient, signifies that the orbital energy of the object is increasing secularly. A situation like this is possible if the solar radiation pressure force acting on the object is strong. However, close to re-entry the atmospheric drag is stronger than the solar radiation pressure and a secular increase in the orbital energy is impossible (drag reduces the orbital energy) (King-Hele & Walker, 1987). Therefore, negative B^* is most likely the result of soaking up modelling errors during TLE fitting (Vallado et al., 2006). Because of that, TLEs with negative B^* values are filtered out because they correspond to poor-quality TLEs that cause inaccurate re-entry predictions.

This filtering technique was found sufficient for the purpose of predicting re-entry in the scope of this work. However, such approach to filtering is ‘one-sided’, i.e. does not remove TLEs with erroneously high B^* . Depending on the analysis that is being conducted, it might be necessary to remove such TLEs as well. If that is the case, employing a relatively simple sliding window filter similar to what was presented for inclination and eccentricity in section 2.4 is recommended.

3. Filter tuning

The filtering algorithms described in section 2 contain a number of numerical parameters, the values of which are not *a priori* known. This section presents the methodology employed to arrive at the optimum values for the numerical parameters of the filters, i.e. the filter settings. The settings for the correction and time filters, which are presented in sections 2.1 and 2.2, respectively, are the parameters that were found best in this study. Arriving at the chosen settings for the correction filter is described in section 2.1, whereas the values for the time filter cited in Table 1 were chosen based on a “trial-and-error” search. The settings of the correction filter cannot be optimised because the TLE validity period is unknown. The time gap filter settings cannot be optimised because it is impossible to unambiguously determine the beginning of a tail of a distribution. The n , e and i filters were tuned rigorously because identification of outliers in these orbital elements is crucial for the accuracy of re-entry prediction. Moreover, careful tuning of the n filter was necessary in order to ensure that outliers are not misidentified as events and *vice versa*.

Manually identifying TLEs as outlying or correct is subject to a number of factors, which make it unreliable. Firstly, it is affected by operator fatigue and inattention, which makes the results unrepeatable in certain ambiguous cases. What is more, while generally well-suited for pattern recognition, humans are also prone to generate biases, e.g. identifying outliers to support the conclusions they wish to draw. Therefore, such outliers cannot be used as *the only* reference to quantify the performance of a filtering algorithm. In order to provide time series of n , e and i with unambiguous outliers, thus enabling the performance of the filtering algorithms to be quantitatively assessed, synthesised TLE time series were used. Nevertheless, the performance of the filters is manually verified in section 5 by means of deciding whether the automatically identified outliers look plausible to the human eye.

The process of arriving at the optimal settings for the n , e and i filters follows a similar logic in all cases. First, a time series without outliers is devised for the orbital element by smoothing and resampling a time series of an exemplar object. Then, outliers of varying magnitudes are simulated at known locations in this time series to form a ‘training time series’. The time series with simulated outliers is filtered using various settings of the filtering algorithms described in section 2. The missed outliers (false negatives) and false outliers (false positives) are counted to quantify the performance of the given permutation of filter settings, which defines a fitness function for the filtering algorithm. Multi-objective genetic optimisation is employed to identify the set of Pareto-optimal filter settings to enable a trade-off between the number of false positives and false negatives (false and missed outliers, respectively). The process of generating the ‘training time series’ is described in section 3.1, while section 3.2 presents the generic fitness function that was used to optimise all filters. Then, the optimisation of mean motion, eccentricity and inclination filters is described in sections 3.3, 3.4 and 3.5, respectively.

3.1. Training time series

Time series of orbital elements of object 13025 (Ariane 1 R/B) was used to generate the ‘training time series’ because it was long (spanned 6.9 years) and contained a number of phases of high drag (rapidly increasing n) separated by phases of low drag (slowly varying n). These properties made it possible to test the n filter for different values of n and $\dot{n}$. $\dot{n}$ may affect the absolute value of the residual Δ_A and consequently also the filter performance. Therefore, testing the algorithm for a range of values of n and $\dot{n}$ was needed to find robust settings for the n filter. Because of its long orbital lifetime compared to other investigated R/Bs, 13025 experienced a number of long-period oscillations of inclination, which made it for a challenging object for the i filter as well. Regarding the eccentricity time series of 13025, it was typical of all examined GTO objects and thus made for a representative test case in this orbital element.

Corrected TLEs were removed from the time series and the resulting time series of n , e and i were smoothed using locally weighted scatterplot smoothing (LOWESS) (Perktold et al., 2017). LOWESS settings, namely the fraction of 13025’s TLEs used to smoothen the value of every TLE and the number of re-weightings, were first optimised to minimise the difference between the smoothed and original time series. This optimisation of LOWESS

settings was performed independently for each orbital element, and is presented in the linked Jupyter notebooks for mean motion³ eccentricity⁴ and inclination⁵. These notebooks can be readily used to reproduce the results presented in this paper because all the necessary input files have been provided as well. Only the proprietary TLE filtering code is not included in the notebooks. However, it follows the algorithms described in section 2 and can, therefore, be reproduced as well.

The smoothed time series were then resampled on a uniform 1 day time grid. This enabled the filter tuning to focus on identifying the best settings to find outliers in n , e and i , without emphasising certain TLEs because of their different time spacing. As discussed in section 2.2, in practice the time spacing between element sets of every object will vary. To assess that the resampling the ‘training time series’ did not produce results that are unusable in practice, the performance of the algorithm with the optimised settings was tested on real objects, which will be described later in this paper. The process of smoothing and resampling the time series of the orbital elements is presented in detail in a companion Jupyter notebook⁶.

After removing the corrected TLEs, and smoothing and resampling the time series, different outlier combinations were added into the ‘training time series’ at known locations. These outlier combinations included single, two or three consecutive outlying TLEs. Multiple outlying TLEs of different magnitudes and directions (e.g. orbital element increasing in one and decreasing in the following TLE), were found to be particularly prone to cause false positives and negatives. Therefore, such combinations of outliers were included in the data. The outliers were located in consecutive phases of varying or steady n , e and i , which dictated their total number.

A number of outlier magnitudes, OM , was used to scale the given orbital element of the TLEs at the simulated outlier locations. To simulate an outlier in any orbital element z , the value from the TLE was scaled to an outlying z_o as $z_o = z + OM \times z$ for each simulated outlier magnitude. For each orbital element, the range of used OM s was chosen to span from outliers several times smaller to several times larger than the outliers observed in the actual

³[www.github.com/\(\dots\)/findGoodLOWESSSettings_n.ipynb](https://www.github.com/(\dots)/findGoodLOWESSSettings_n.ipynb)

⁴[www.github.com/\(\dots\)/findGoodLOWESSSettings_e.ipynb](https://www.github.com/(\dots)/findGoodLOWESSSettings_e.ipynb)

⁵[www.github.com/\(\dots\)/findGoodLOWESSSettings_i.ipynb](https://www.github.com/(\dots)/findGoodLOWESSSettings_i.ipynb)

⁶ www.github.com/AleksanderLidtker/TLEFilterTuning/blob/master/generatePristineTimeSeries.ipynb

TLE time series.

In addition to outliers, sequences were included in the ‘training time series’ of mean motion. These were meant to replicate the effects of sudden changes in the orbit of the object caused by events that were observed in n time series of actual objects (see work by Lidtke et al. (2016) for examples). The sequences were included in various phases of orbital decay, i.e. in periods of different n and $\dot{n}$. The magnitude of the sequences was kept twice as large as the maximum simulated outlier magnitude, and the used sequence lengths were 30, 40, 50 and 60 TLEs. The effect of these assumptions on the filter optimisation results will be analysed by testing the n filter tuning on actual objects. No outlying sequences were simulated for eccentricity or inclination because these filters were only designed to identify outliers, not sequences. The process of simulating outliers in the previously smoothed and resampled time series can be reproduced by using a companion Jupyter notebook⁷, which also compares the simulated outlier magnitudes to the ones observed for 13025.

3.2. Fitness function

To test the performance of one set of settings, the outlier detection algorithm was run on the ‘training time series’ with outliers for all outlier magnitudes. The identified outliers were compared to the known outlier locations. In cases where the algorithm detected an outlier, which was not present in the data a false positive (a false outlier) was recorded. Conversely, when the algorithm did not identify a simulated outlier, a false negative (a missed outlier) was recorded. The numbers of false positives and negatives for all outlier magnitudes were summed and normalised to a $[0, 1]$ interval to produce two metrics for every filtering algorithm setting. The number of false positives was normalised with the number of non-outlying TLEs in the ‘training time series’, and the number of false negatives with the number of outlying TLEs. This normalisation ensured that the two components of the fitness function, no. false positives fp and no. false negatives fn , were always in the $[0, 1]$ interval. Thus, the fitness function given by Eq. 5 was constructed, where the number of false and missed outliers was a function of the filtering algorithm settings, $\vec{x}$. This fitness function is generic and was used for mean motion,

⁷ www.github.com/AleksanderLidtke/TLEFilterTuning/blob/master/compareRealAndSimulatedOutliers.ipynb

eccentricity and inclination filters but with different control vectors $\vec{x}$, which contained the settings of the given filter.

$$\vec{f} = \begin{pmatrix} f^{(1)} \\ f^{(2)} \end{pmatrix} = \begin{pmatrix} fp \\ fn \end{pmatrix} = f(\vec{x}). \quad (5)$$

3.3. Mean motion filter tuning

3.3.1. Problem setup

The generic process of obtaining a ‘training time series’ was described in section 3.1. The outlier magnitudes used for mean motion were $\pm \{0.005, 0.01, 0.05, 0.1\}$. Using more *OMs* was found to affect the value of the fitness function by at most 3% while increasing the optimisation wall time by over 30%. Also, the largest outlier identified in the time series of 13025 had magnitude of 1.5% w.r.t. smoothed time series⁷. Therefore, the investigated values of *OMs* covered what can be expected for actual time series. The time series of 13925 had only two outliers larger than 1%, while 81 outliers were simulated. This made distinguishing between outliers and correct TLEs following outliers more difficult than in reality. The resulting n time series, with the largest *OM*, as well as sequences is shown in Fig. 6.

The vector of algorithm settings, which was optimised for the mean motion filter based on the fitness function described in section 3.2, $\vec{x}_n$, is given by Eq. 6. Therein, N_{TLE} is the number of TLEs in the sliding window, while R_{TOL} and A_{TOL} are tolerances on T_R and T_A , respectively. The regressing polynomial order was not included in $\vec{x}_n$. Instead, the optimisation was carried out for different polynomial orders and the best solution was chosen.

$$\vec{x}_n = \begin{pmatrix} N_{TLE} \\ R_{TOL} \\ A_{TOL} \end{pmatrix} \quad (6)$$

3.3.2. Filter tuning results

The fitness function from Eq. 5 with the control vector from Eq. 6 was optimised using the NSGA-II algorithm (Deb et al., 2002) implemented in an island model (Izzo & Biscani, 2015). Previous work has shown that the optimal filter performance is achieved using low-order regressing polynomials (Lidtke et al., 2016). Therefore, only 1st (Theil-Sen-Seigel) and 3rd order regressing polynomials were investigated. The settings of the optimiser and the investigated ranges of each component of $\vec{x}_n$ for both polynomial orders

Figure 6: ‘Training time series’ of the mean motion of object 13025 used to tune filtering algorithm. Locations of the simulated outliers and their largest magnitudes are marked with dots. Also showing the simulated sequences. Details of outlier simulation are given in text.

are shown in Table 2. Note that for higher polynomial orders more TLEs are needed in the sliding window to define the polynomial. Also, higher order polynomials can accurately regress the evolution of mean motion over a longer period of time. This allows more correct TLEs to be placed in the moving window and reduces the effect of the outliers that might be present in it (Lidtke et al., 2016).

Table 2: Settings of the optimiser (no. islands, no. individuals per island, total number of iterations) used to find the optimum settings of the mean motion filter for two regressing polynomial orders. Also showing the investigated ranges of the components of the control vector $\vec{x}_n$.

		1 st order	3 rd order
Optimiser setting	No. islands	12	16
	No. individuals	8	8
	No. iterations	300	255
Search range	N_{TLE}	[2, 50]	[5, 100]
	R_{TOL}	$[1.0 \times 10^{-5}, 2.0]$	
	A_{TOL}	$[1.0 \times 10^{-10}, 4.0 \times 10^{-1}]$	

Figures 7 and 8 show the Pareto front found by the optimiser for 1st and 3rd order regressing polynomials, respectively. Table 3 presents the optimal (closest to the origin, i.e. minimising both the number of false and missed outliers) mean motion filter settings. The settings that minimise the number of missed outliers, which deteriorate the accuracy of the TLE-based analyses because the orbital elements in an outlying TLE are incorrect, as well as the fitness corresponding to each $\vec{x}_n$, are also presented.

3.3.3. Discussion

It was not possible to find mean motion filter settings that reduce either or both the number of false and missed outliers to zero. Figure 9 shows the number of false and missed outliers for 64 filter setting permutations, uniformly spaced in every dimension of $\vec{x}_n$ for the Theil-Sen-Seigel regression. The larger the absolute and relative tolerances are, the fewer false positives, or false outliers, are identified. This is because the algorithm tolerates larger deviations from the regressing polynomial. However, large tolerances also give rise to many false negatives, i.e. missed outliers. The same behaviour is observed for all regressing polynomials and window lengths. A situation like

Figure 7: Performance for various permutations of the mean motion filter settings found by the genetic optimiser for 1st order polynomial regression. Each dot represents one combination of the settings for the given regression order, i.e. a different $\vec{x}_n$. Point closest to the origin is indicated with a red dot, and the point that minimises the number of missed outliers with a yellow dot.

Figure 8: Performance for various permutations of the mean motion filter settings found by the genetic optimiser for 3rd order polynomial regression. Each dot represents one combination of the settings for the given regression order, i.e. a different $\vec{x}_n$. Point closest to the origin is indicated with a red dot, and the point that minimises the number of missed outliers with a yellow dot.

Table 3: Optimised settings of the mean motion, n , filter for the two investigated orders of polynomial regression. Showing the settings that correspond to the non-dominated point closest to the origin and the point that minimises the number of missed outliers (false negatives). Also showing the normalised value of the fitness function from Eq. 5, and the actual number of the simulated outliers that the filter did not identify out of the simulated 648.

1st order polynomial		
Filter setting	Non-dominated point	
	Closest to origin	Fewest missed outliers
Window length, N_{TLE} (no. TLEs)	2	3
T_A tolerance, A_{TOL}	1.47×10^{-5}	1.51×10^{-5}
T_R tolerance, R_{TOL}	3.63×10^{-1}	6.93×10^{-2}
No. false outliers, fp	0.0437	0.2382
No. missed outliers, fn	0.0293	0.0123
Actual missed outliers	19	9
3rd order polynomial		
Filter setting	Non-dominated point	
	Closest to origin	Fewest missed outliers
Window length, N_{TLE} (no. TLEs)	15	5
T_A tolerance, A_{TOL}	1.39×10^{-5}	1.41×10^{-5}
T_R tolerance, R_{TOL}	1.28×10^0	9.77×10^{-2}
No. false outliers, fp	0.0622	0.2093
No. missed outliers, fn	0.1188	0.0123
Actual missed outliers	77	8

this necessitates a trade-off between the number of false positives and negatives, i.e. choosing the settings that correspond to one of the non-dominated points in Fig. 7 or Fig. 8.

In the context of re-entry prediction, false positives (false outliers) will reduce the number of TLEs that can be used in the analysis, whereas false negatives (missed outliers) will deteriorate the accuracy of the results because subsequent analyses will use incorrect TLEs. Reducing the number of only false or missed outliers to the minimum can reduce the re-entry prediction accuracy because of small TLE sample sizes or TLEs with incorrect orbital elements being used for the analysis (Lidtke et al., 2016; Gondelach et al., 2016). Therefore, it is recommended to use the settings from Table 3 that

(a) False positives (false outliers)

(b) False negatives (missed outliers)

Figure 9: Performance of the mean motion filter using 1st order polynomial regression. Showing the variation of each component of the fitness function $\vec{f}$ from Eq. 5 with varying $\vec{x}_n$ for several N_{TLE} . Shown permutations of $\vec{x}_n$ were obtained from a full-factorial grid search with 64 points, evenly spaced in every dimension of $\vec{x}_n$.

are the closest to the origin, i.e. do not favour either extreme ($N_{TLE} = 15$, $A_{TOL} = 1.39 \times 10^{-5}$, $R_{TOL} = 1.28 \times 10^0$, 3^{rd} order regression). 3^{rd} order regressing polynomial was preferred over Theil-Sen-Seigel because it uses a longer sliding window. Longer windows in the algorithm from section 2.3 reduce the fluctuations of the regressing polynomial due to the outliers that are kept in the analysis while the window is slid and, therefore, also reduce the number of false positives. When tested on the real objects from section 5, 3^{rd} order regression marked 3 to 8% of TLEs as outliers, depending on the object, while 1^{st} order between 6 and 26%. Only a fraction of these TLEs were indeed outlying, which shows that the 1^{st} order polynomial identified more false outliers and that 3^{rd} order regression performed better on actual objects. During the optimisation, however, 1^{st} order regression outperformed 3^{rd} order. This signifies that tuning the filter exclusively based on the ‘training time series’ might be insufficient to always achieve the best possible results when filtering TLEs of a real object. This will be discussed in more detail in section 5.

Even with the best settings that minimise the number of missed outliers, the algorithm did not identify 8 outliers. The total number of outliers simulated in the study was 648, meaning that the algorithm correctly identified 98.8% of the simulated outliers, or at worst 88.1%. It is unlikely that this performance could be improved further because the largest variation in optimal settings $\vec{x}_n$ was less than 1.24% after 220 iterations as shown in Fig. 10a. Figure 10b demonstrates that the average value of the fitness of the non-dominated points did not vary by more than 1.29% after 220 iterations for 3^{rd} order regressing polynomial. This means that the set of found non-dominated points likely corresponds to Pareto solutions.

3.4. Eccentricity filter tuning

3.4.1. Problem setup

The ‘training time series’ of eccentricity was obtained according to the methodology presented in section 3.1 and is shown in Fig. 11. In case of eccentricity, the used outlier magnitudes were $\pm \{0.005, 0.01, 0.05, 0.1, 0.2, 0.25\}$. Larger maximum outlier magnitude was used for eccentricity than for mean motion because the largest difference between the ‘training time series’ and e extracted from 13025’s TLEs was 10.2%⁷, as opposed to 1.5% for mean motion. Also, the eccentricity time series had six outliers $\geq 1\%$.

The fitness function from Eq. 5 with e filter settings as $\vec{x}$ and e ‘training time series’ were used to tune the eccentricity filter. Recall from section 2.4

(a) $\vec{x}_n$ closest to origin

(b) Components of f

Figure 10: Convergence of the optimisation of the mean motion filter with 3^{rd} order polynomial regression. Showing the value of the settings corresponding to the non-dominated point located closest to the origin, as well as the mean value of both components of the fitness function from Eq. 5 computed for all non-dominated points.

that the eccentricity is filtered twice using the same process. Thus, $\vec{x}_e$ given in Eq. 7 is a vector of length six, even though the filter only requires three inputs. The first pass of the ‘global’ e filter removes the largest e outliers, which are present in time series of certain objects. The second, ‘local’ pass filters relatively small outliers, which can affect the accuracy of re-entry prediction in spite of their smaller magnitude.

$$\vec{x}_e = \begin{pmatrix} N_{TLE}|^{GLOB} \\ N_{MAD}|^{GLOB} \\ MAD_{TH}|^{GLOB} \\ N_{TLE}|^{LOC} \\ N_{MAD}|^{LOC} \\ MAD_{TH}|^{LOC} \end{pmatrix} \quad (7)$$

Figure 11: “Training time series” of eccentricity of object 13025 used to tune the filtering algorithm. Locations of the simulated outliers and their largest magnitudes are marked with dots. Details of outlier simulation are given in text.

3.4.2. Filter tuning results and discussion

NSGA-II optimiser was run on the eccentricity tuning problem given in section 3.4.1 with the settings presented in Table 4. Unlike in the case of

mean motion, it was possible to reach perfect performance of the filter, i.e. to identify all outlying TLEs without erroneously rejecting any good TLEs. The optimiser converged to this solution after 11 iterations. The perfect performance of the filter can be achieved with more than one setting value. However, the region of the setting parameter space where perfect performance can be achieved is reasonably well defined, i.e. the variation in each setting was at most 21.3% w.r.t. the mean after 400 iterations as shown in Fig. 12a. The final settings found by the optimiser as well as the mean settings after 400 iterations are given in Table 5.

When using real TLE data, false positives and negatives were observed when automatically filtering eccentricity as well as mean motion (Lidtke et al., 2016), whereas the eccentricity filter optimisation yielded no false positives or negatives. This discrepancy will be investigated by filtering TLE time series of real objects in the section 5.

Table 4: Settings of the optimiser (no. islands, no. individuals per island, total number of iterations) used to find the optimum settings of the eccentricity filter. Also showing the investigated ranges for the components of the control vector $\vec{x}_e$. 2527 corresponds to using the entire time series of de-trended TLE values to compute the sample dispersion, against which individual TLEs are compared.

		Value
Optimiser setting	No. islands	12
	No. individuals	20
	No. iterations	500
Search range	$N_{TLE} ^{GLOB}$	[3, 100]
	$N_{MAD} ^{GLOB}$	[3, 2527]
	$MAD_{TH} ^{GLOB}$	[1, 60]
	$N_{TLE} ^{LOC}$	[3, 50]
	$N_{MAD} ^{LOC}$	[3, 2527]
	$MAD_{TH} ^{LOC}$	[1, 30]

3.5. Inclination filter tuning

3.5.1. Problem setup

The same outlier magnitudes as for eccentricity, $\pm \{0.005, 0.01, 0.05, 0.1, 0.2, 0.25\}$, were used for i to generate the ‘training time series’ according to the methodology presented in section 3.1. In case of inclination, the largest difference

(a) Eccentricity e

(b) Inclination i

Figure 12: Convergence of the optimisation of the eccentricity and inclination filters. Showing the value of the settings corresponding to the point located closest to the origin. This point also minimises the number of missed outliers because the optimiser converged to perfect solution.

Table 5: Optimised settings of the eccentricity, e , filter. Showing the settings that correspond to the non-dominated point closest to the origin at the end of the optimisation, as well as the mean of each setting between 400 and 500 iterations. Also showing the normalised value of the fitness function from Eq. 5, and the actual number of the simulated outliers that the filter did not identify out of the simulated 972.

Filter setting	Value	
	Closest to origin	Mean after 400 iterations
$N_{TLE} ^{GLOB}$	5	5
$N_{MAD} ^{GLOB}$	1799	1792
$MAD_{TH} ^{GLOB}$	2.98	3.17
$N_{TLE} ^{LOC}$	10	9
$N_{MAD} ^{LOC}$	2139	2042
$MAD_{TH} ^{LOC}$	20.79	24.59
Number of filter passes	2	2
No. false outliers, fp	0.0	0.0
No. missed outliers, fn	0.0	0.0
Actual missed outliers	0	0

between 13025's TLEs and smoothed time series was 2.0%⁷, but large maximum OM was used to keep the filtering robust. Inclination time series of this object had the most outliers $\geq 1\%$ (nine) and the largest 95th percentile of the difference between the TLEs and the smoothed time series (0.43%). The resulting 'training time series' is shown in Fig. 13.

The fitness function from Eq. 5 with i filter settings as $\vec{x}$ and i 'training time series' were used to tune the inclination filter. The i filter only uses one pass of the algorithm described in section 2.4. Therefore, $\vec{x}_i$ was a vector of length three as given in Eq. 8.

$$\vec{x}_i = \begin{pmatrix} N_{TLE} \\ N_{MAD} \\ MAD_{TH} \end{pmatrix} \quad (8)$$

3.5.2. Filter tuning results

Settings of the NSGA-II used to optimise the fitness function described in section 3.5.1 are given in Table 6. As in the case of eccentricity, the optimiser converged to a perfect solution after 5 iterations. Convergence was quicker in this case because of the shorter control vector x_i . The variability

Figure 13: ‘Training time series’ of inclination of object 13025 used to tune the filtering algorithm. Locations of the simulated outliers and their largest magnitudes are marked with dots. Details of outlier simulation are given in text.

in the settings that corresponded to the perfect performance was at most 15.0% after 400 iterations, as demonstrated in Fig. 12b. The final optimiser solution as well as the mean value of each setting after 400 iterations are given in Table 7. Next section investigates whether the fitness functions of e and i filters were appropriate to produce results that are usable in case of real TLE data.

4. Filter effects on re-entry prediction

Previous work has shown that for certain objects, TLE filtering is necessary to reduce the error in re-entry prediction to less than 50% and to achieve consistent prediction accuracy in the order of 10% (Lidtke et al., 2016). Instead of repeating the analysis that supports this conclusion, this section will focus on asserting that the filter settings optimised based on an artificial TLE time series are usable on real objects. To this end, TLE time series of six objects that have been found to contain various outlier types were analysed in detail. The space surveillance catalogue numbers of these objects were: 10983, 14287, 21057, 21766, 22906 and 22932. At the end of this section, the

Table 6: Settings of the optimiser (no. islands, no. individuals per island, total number of iterations) used to find the optimum settings of the inclination filter. Also showing the investigated ranges for the components of the control vector $\vec{x}_i$. 2527 corresponds to using the entire time series of de-trended TLE values to compute the sample dispersion, against which individual TLEs are compared.

		Value
Optimiser setting	No. islands	12
	No. individuals	20
	No. iterations	500
Search range	N_{TLE}	[3, 100]
	N_{MAD}	[3, 2527]
	MAD_{TH}	[1, 30]

re-entry prediction accuracy for all the objects from Appendix A with and without filtering will be analysed for the sake of completeness.

The objects from Appendix A have decayed already and, therefore, their re-entry epoch was made publicly available via www.space-track.org (these reference epochs are rounded to one day). This enables the discrepancy between the actual time that the object spent on-orbit since the prediction epoch, τ_A , and the predicted time on-orbit, τ_P , to be compared. The percent prediction error, $\delta\tau$, of τ_P relative to τ_A was computed as per Eq. (9) for different filtering levels ($|\dots|$ is the absolute value operator). The investigated filtering levels and their acronyms are given in Table 8.

$$\delta\tau = \left| \frac{\tau_P - \tau_A}{\tau_A} \right| \times 100. \quad (9)$$

Figure 14 shows the relative errors in predicting the re-entry epoch using orbit determination for the six R/Bs with different filtering levels. These objects exhibit typical types of behaviour that have been observed amongst the investigated sample. In all cases, the TLEs were filtered with 3rd order regressing polynomial settings closest to the origin from Table 3 because these settings were deemed optimal as per the discussion in section 3.3.3. The predictions were made by using the most-recent TLE that was older than 30 days from the re-entry epoch, and the preceding TLEs to complete the orbit determination sample. Because of differences in the availability of TLEs for different objects, τ_A in Eq. (9) varied depending on the object. The maximum possible error this could have caused will be discussed later in

Table 7: Optimised settings of the inclination, i , filter. Showing the settings that correspond to the non-dominated point closest to the origin at the end of the optimisation, as well as the mean of each setting between 400 and 500 iterations. Also showing the normalised value of the fitness function from Eq. 5, and the actual number of the simulated outliers that the filter did not identify out of the simulated 972.

Filter setting	Value	
	Closest to origin	Mean after 400 iterations
N_{TLE}	7	8
N_{MAD}	2100	2076
MAD_{TH}	5.97	5.93
Number of filter passes	1	1
No. false outliers, fp	0.0	0.0
No. missed outliers, fn	0.0	0.0
Actual missed outliers	0	0

this section when analysing the magnitude of the effects of filtering. Firstly, the effects of various filtering stages will be examined on a per-object basis, where this variability does not play a role because the same TLEs are always used for a given object.

Different filtering stages may improve the prediction accuracy by removing outliers. For example, filtering outliers in eccentricity of 22932 improved the accuracy from over 14.40% to 1.47%. In certain cases, however, filtering TLEs may decrease the number of usable element sets and reduce the prediction accuracy, which was the case for 21057. For this object, $\delta\tau$ increased from 11.99 to 15.56% when introducing eccentricity filtering because the number of available TLEs reduced from three to one, as shown in Fig. 19. The prediction accuracy for 21057 was low regardless of the filtering level due to the fact that only three TLEs were available for this object.

The accuracy improvements brought by filtering may not always be clearly visible because significant outliers are not always present in the used TLE set. For example, $\delta\tau$ for 10983 improved from 1.75 to 1.67% (4.79% change) with no and full filtering, respectively. This was because only one outlier was present in the 14 available TLEs, as shown in Fig. 18. For object 14287, the prediction accuracy reduced from 7.28 to 7.41% (1.79% change) because only one outlier and two corrected TLEs were present amongst its 24 TLEs, as shown in Fig. 20. The accuracy reduced with more filtering because an inclination outlier (green cross on Fig. 20a close to the beginning of the OD

Table 8: Investigated TLE filtering levels and the acronyms by which they are referred to.

Filter acronym	Included filters
n, e, i, B^*	Correction, time, mean motion, perigee radius, inclination, B^*
n, e, i	Correction, time, mean motion, perigee radius, inclination
n, e	Correction, time, mean motion, perigee radius
n	Correction, time, mean motion
<i>NONE</i>	No filters

window, i.e. 19 April) was removed from the analysis based on its outlying i even though it had an accurate perigee radius, as can be seen in Fig. 20b (also a green cross). Filtering this TLE with correct r_P caused the OD and re-entry prediction to be less accurate because other TLEs present in the window, with less accurate r_P , had a greater impact on the prediction result. This signifies that incorrectly filtering out too many TLEs (too many false positives) will deteriorate the analysis accuracy. However, in the presence of outliers, in particular when an outlying TLE is being used for BC estimation, e.g. for 22932, filtering outliers plays a key role. The summary of $\delta\tau$ without and with optimised filtering is presented in Table 9.

Table 9: Relative re-entry prediction error, $\delta\tau$, for the six objects shown in Fig. 14 for no filtering and full filtering with the optimised filters. Difference is computed between the filtered and unfiltered $\delta\tau$, such that negative differences correspond to improved predictions.

SSC	$\delta\tau$ (%)		
	Unfiltered	Optimised filters	Difference
10983	1.75	1.67	-0.08
14287	7.28	7.41	0.13
21057	12.03	15.56	3.53
21766	4.31	2.98	-1.33
22906	0.89	2.09	1.20
22932	14.40	1.47	-12.93

Another noteworthy case is object 21766, for which two levels of filtering

improved the re-entry prediction accuracy. First, introducing eccentricity filtering reduced the error from 4.31% to 4.09%. Adding inclination filtering further lowered $\delta\tau$ to 2.98%. This shows that every orbital element used in the analysis should be filtered because TLEs outlying in one element might not be outlying in others.

Object 21766 re-entered on 20 Oct 2014 and its re-entry prediction was performed on 9 Sep 2014. Thus, its TLEs that were used for the prediction were likely published after the change in the TLE generation process discussed by Hejduk et al. (2013). Figure 15 shows the argument of latitude of 21766 at the TLE epochs. The observed dispersion along the orbit at the TLE epoch changes similarly to what was observed in Fig. 2, which suggests that the TLE generation process for this object changed around 1 Jan 2013. Figure 16 shows that the time series of perigee radius of this object became smoother around 1 Jan 2013, which further corroborates the hypothesis about the change in the TLE generation process around that epoch. Still, 27166 r_P time series contained outliers both before and after 1 Jan 2013. These outliers were correctly identified by the filters in all cases, meaning that the presented set of filters is applicable to both historical and recent TLE time series in the GTO regime. The filter applicability to other orbital regimes or after future changes in the TLE generation process is discussed in section 5.

Average position residual after orbit determination, which forms part of the re-entry prediction algorithm, is another metric besides $\delta\tau$ that can be used to gauge the effect of TLE filtering. As shown in Fig. 17, OD fits improved with more filtering for all cases with high residuals. Only for the cases with low residuals before filtering, namely objects 10983, 14287 and 21766, the position residual increased slightly. However, it only increased from 20.3 to 20.8 km, 39.9 to 40.9 km and 22.9 to 30.7 km for these respective objects. Interestingly, for objects 21057 and 22906 the orbit fit improved after filtering but the prediction accuracy did not. For object 21766, on the other hand, the residual increased but the prediction improved. This is because an accurate OD fit does not guarantee accurate re-entry prediction, which was also concluded by Gondelach et al. (2017).

TLE filtering does not, therefore, guarantee accurate re-entry predictions or improved predictions. This can be seen when comparing the effects of TLE filtering on the accuracy of re-entry prediction using only BC estimation (i.e. without OD) for all the objects from Appendix A shown in Fig. 21. Note that the actual time left on-orbit since the prediction epoch, τ_A in Eq. 9, is known with uncertainty of one day (reference re-entry epochs are all at

midnight), which gives rise to the uncertainty on $\delta\tau$. All the predictions were performed at least 30 days before the actual re-entry. Therefore, the worst-case uncertainty on $\delta\tau$ is $\frac{1}{30} \approx 3.3\%$. This uncertainty applies to all the predictions in Fig. 21 and its significance will be discussed later in this section.

Up to $\delta\tau = 16\%$, no filter outperforms the unfiltered time series as far as the re-entry prediction accuracy is concerned. This is caused by the fact that for most (66 out of 96) of the objects, just like for four out of six cases reported in Table 9, filtering does not affect the re-entry prediction accuracy by more than 2.0% if there happens to be no significant outlier being used for the prediction. Beyond $\delta\tau = 16\%$, TLE filtering improves the prediction accuracy. In particular, TLE filtering helps avoid predictions where $\delta\tau \geq 50\%$ for the unfiltered cases. Such inaccurate predictions occurred twice from amongst the 96 unfiltered cases shown in Fig. 21. The TLE filters tuned based on a manually identified outliers from Lidtke et al. (2016) improved the accuracy of both of these predictions, while the optimised filters presented here improved the accuracy in only one case. The accuracy for object 18923 remained poor when filtering the TLEs using the optimised filters because, as shown in Fig. 22, only one n outlier was filtered out. Because of the noise in the r_P of the remaining TLEs, the BC estimation was inaccurate and the resulting re-entry prediction accuracy was also poor. The ‘manual filters’ removed the pivot TLE used for BC estimation for 18923 based on the mean motion filter. It cannot be asserted whether the TLEs in the BC estimation window are indeed outlying in n or not because they are scarce and noisy. Therefore, it cannot be asserted whether the ‘manual filters’ performed better than the ‘optimised filters’ in this case.

The difference between $\delta\tau$ of the filtered and unfiltered TLE time series, defined in Eq. 10, was computed for 96 objects and is shown in Fig. 23 for the cases where this difference was non-zero (33 objects).

$$\Delta\delta\tau = \delta\tau_{\text{filtered}} - \delta\tau_{\text{unfiltered}}. \quad (10)$$

The $\delta\tau$ for different filtering methods for the 20 objects with the largest $\delta\tau$ when using the unfiltered TLEs is shown in Fig. 24. This Figure also shows the maximum possible uncertainty on $\delta\tau$, which is less than the difference between the different filtering methods for the objects with the largest prediction errors. On average, optimised filters improved the predictions by 1.1 and manually-tuned filters by 2.3% but the accuracy reduced due to filtering for certain objects.

In three cases (18923, 22932 and 23916), manual filtering improved the prediction (reduced the prediction error, which corresponds to negative $\Delta\delta\tau$) more than optimised filtering. However, optimised filtering never made the prediction worse by more than 8.4%, while the manual filtering once made the prediction worse by 15.2%. Still, both sets of filters performed similarly for all analysed objects, meaning that the optimisation using a ‘training time series’ produced results usable in real cases and is thus considered successful. Section 5 analyses the filtered TLE time series in greater detail and examines whether the automatically identified outliers are not contrived.

Note that the settings of the manually tuned filters were obtained based on the performance of the algorithms on the same sample of 96 objects. Therefore, manual tuning results might be biased towards removing the most significant outliers, which deteriorate the predictions for these objects the most.

Overall, filtering only significantly affected the predictions for four out of the 96 analysed objects (4.2%, 18923, 39499, 37257 and 23916), when outliers were being used for the predictions. In those rare cases, however, filtering improved the prediction by up to 42.6%, which signifies that filtering is necessary to prevent considerably inaccurate predictions. Optimised filters did not deteriorate the results by more than 8.4%. Therefore, it is recommended to always filter the time series to avoid considerably inaccurate results, at the expense of slightly deteriorated predictions in some cases.

5. Discussion

Previous work has shown that when testing the automated filtering on real TLEs, it often cannot be asserted whether a TLE is outlying or not. Therefore, the algorithm performance cannot be confidently quantified because the reference data are uncertain. For this reason, artificially simulated outliers were used in this work as explained in section 3.1.

Using an artificial time series for algorithm tuning poses a risk that the tuning will only be applicable to the artificial training series, not real objects. This was mentioned in section 3.3.3, where a different set of optimised settings was chosen because it performed better on real objects than the settings, which performed better on the artificial time series. Such heuristic approach to interpreting optimisation results can be argued for because, after all, optimisation is meant to support decision making (Keane & Nair, 2005). However, in order to make sure that the optimised filter settings

Figure 14: Relative errors in predicting the re-entry epoch (defined in Eq. 9) with different TLE filtering levels from Table 8 for six objects 30 days before the actual re-entry.

Figure 15: Argument of latitude of 21766 computed at the epoch of the TLEs. Also showing the re-entry prediction epoch for this object (9 Sep 2014).

Figure 16: Time series of perigee radius of 21766 in the beginning of 2013, when its TLE generation process is believed to have changed based on the change in the behaviour of argument of latitude at TLE epoch shown in Fig. 15. This change in the behaviour of argument of latitude is accompanied by a smoother r_P time series, where outliers are present nonetheless. Eccentricity (perigee radius) outliers are indicated with red crosses. Mean motion and inclination outliers are indicated with purple crosses.

Figure 17: Average position residuals after orbit determination for different TLE filtering levels from Table 8 for six objects 30 days before the actual re-entry.

Figure 18: Time series of perigee radius of object 10983 used for re-entry prediction. Showing outliers identified by the filters as well as the extents of the window within which orbit determination was performed.

Figure 19: Time series of perigee radius of object 21057 used for re-entry prediction. Showing outliers identified by the filters as well as the extents of the window within which orbit determination was performed.

were not contrived, the filtered time series of the six objects from section 4 were investigated in detail. Examples of the automated filtering identifying credible outliers in mean motion, eccentricity and inclination for object 21057 are shown in Fig. 26, Fig. 27 and Fig. 28. The already mentioned Fig. 16, Fig. 18 and Fig. 19 also present outlier identification that is consistent with a manual assessment. Complete figures for these three objects as well as the remaining three test cases are available in an interactive format at www.plot.ly/~AleksanderLidtke. The outliers shown in these figures stand in agreement with what a human eye would identify as outliers. This increases confidence that the used ‘training time series’ did not cause the tuning results to be contrived.

Figure 29 shows the same data as Fig. 26 but filtered with 1st order regressing polynomial settings closest to the origin from Table 3. It can be seen that, as the sliding window of two TLEs is moved across an outlier, the regression inside the window temporarily stops representing the actual evolution of the mean motion of the object. This, in turn, causes false outliers to be identified (in certain cases, it could also cause missed outliers). The same behaviour can be observed in the ‘training time series’ filtered with the 1st

(a) Inclination outliers

(b) Perigee radius outliers

Figure 20: Time series of inclination and perigee radius of object 14287 used for re-entry prediction. Showing outliers identified by the filters as well as the extents of the window within which orbit determination was performed.

Figure 21: Cumulative histogram of the re-entry prediction error, $\delta\tau$, for all the objects from Appendix A computed 30 days before the actual re-entry and using only BC estimation for predictions. Comparing the results obtained when filtering the TLEs with the filter settings optimised here, the filters tuned based on manually identifying outliers (Lidtkje et al., 2016) and no filtering. For $\delta\tau \geq 50\%$, the error was limited to 50%.

Figure 22: Time series of mean motion and perigee radius of 18923 used for re-entry prediction showing the identified n and e (filtered on r_P) outliers together with the extents of the window which BC estimation was performed.

Figure 23: Change in the re-entry prediction error (defined in Eq. 10) when filtering the TLEs with manually tuned filters (see Lidtke et al. (2016)) and the setting optimisation performed here. Note that negative $\Delta\delta\tau$ corresponds to a prediction improved due to filtering, i.e. reduced error in the re-entry prediction after filtering.

Figure 24: Comparison of $\delta\tau$ without, and with manual or optimised filtering for the 20 objects with the largest $\delta\tau$ without filtering. Showing the worst possible uncertainty in $\delta\tau$ (3.3%) associated with the fact that τ_A is rounded to one day.

order polynomial shown in Fig. 25a. During the optimisation, the 3rd order polynomial shown in Fig. 25b identified more false outliers at the beginning of the simulated sequences, which caused its fitness to be lower than for the 1st order polynomial. However, in real TLE time series sequences are not as common and, therefore, the 3rd order polynomial was chosen instead based on experience because its longer window does not result in false positives at actual outlying TLEs. This could signify that the ‘training time series’ was not completely representative of all actual objects. However, in numerous examples given here it performed correctly, i.e. it was consistent with what an operator would mark as outliers. The optimisation could be repeated to find different solutions to the filtering problem. In fact, the re-entry prediction algorithms could be included in the fitness function and the errors of the entire set of 96 R/Bs minimised. However, this would greatly increase the computational cost of the optimisation without making the results any more applicable to other sets of objects.

Overall, the optimised filter settings seem to perform correctly, including the eccentricity and inclination filters that achieved perfect performance during the optimisation. It cannot be quantified how many of the identi-

(a) 1st order (Theil-Sen-Seigel) regression

(b) 3rd order regression

Figure 25: Part of the ‘training time series’ of mean motion with the smallest used OM , filtered with both 1st and 3rd order polynomials. Using the optimised settings closest to the origin from Table 3. Showing false positives at the beginning of a simulated sequence found by the 3rd order regression, and false positives after outliers found by the 1st order regression.

fied outliers are false or how many are missed, because it is impossible to obtain unambiguous reference data. This is because of the uncertain nature of the TLE generation process, and incomplete list of possible TLE error sources and their effects on the actual time series. Therefore, manual inspection of the TLEs used for analysis is recommended as a complement to the automated filtering. Neither automated nor manual outlier identification is believed to be capable of correctly identifying all outliers in all cases, and the two should be used to reinforce each other. In practice, the automated filtering can identify most outliers and highlight ambiguous cases to an operator.

Caution needs to be taken when transferring the results obtained here to other use cases. The filters were tuned on a specific set of objects and so might not perform equally well in other cases. The outliers in the TLEs that were detected here were in most cases caused by errors related to the GTO orbital regime, such as poor observation accuracy, and limited accuracy of the SGP4 force model because of the truncation of the higher-order terms for eccentricity, lack of solar radiation pressure modelling, and inaccurate atmospheric model (Picone, 2005). When applying the set of filters to objects with e.g. different eccentricity or higher area-to-mass ratios (investigated R/Bs had a relatively large mass and low area), other sources of error might become more prevalent. Thus, the outliers present in the TLE time series might be of a slightly different nature and the chosen settings might no longer be optimal. Because some of the outliers in the investigated time series were also caused by atmospheric drag mis-modelling, the optimum filter settings are a function of the solar activity, which is expected to cause larger outliers during more active periods. When applying the described filters to different cases, the filter performance should be manually analysed on a set of representative objects to confirm whether the algorithms still appear to perform correctly. The same verification process might even need to be repeated for GTO orbits, should the TLE generation process change again as it did around 1 Jan 2013 for the previously discussed object 21766. Because of the use of a ‘training time series’ that included a more exhaustive set of outliers than any single object, the filter tuning is believed to be robust. However, this cannot be ascertained with absolute certainty. Should filters need to be re-tuned or even the underlying algorithms altered, the presented filter tuning methodology is superior to the previously employed ‘manual tuning’ because it only requires one ‘training time series’ to be generated and used to optimise the settings, rather than manually analysing dozens of

objects. This paper has shown that this tuning methodology is capable of achieving similar performance to manually tuned filters.

What concerns the developed filters, they have been shown to identify credible outliers in mean motion, inclination, eccentricity and B^* . As a result, they have improved re-entry predictions w.r.t. unfiltered TLE time series, particularly in cases where the prediction error was high without filtering. The filters treat all four parameters independently, even though a certain degree of correlation might be expected between e.g. n and e (various orbits can be fitted to the same perigee or apogee radius), e and B^* (both are associated with perigee altitude and atmospheric drag for highly-eccentric orbits), and i and Ω (associated with the orientation of the orbital plane). Indeed, such correlated errors have been observed in certain cases, e.g. Figure 1 in Gondelach et al. (2017) shows perigee radius outliers correlated with B^* outliers. However, this error correlation does not occur for all outliers, which is why all of the filters here were applied in sequence, and sometimes more than one filter improved the prediction for a particular object. Had the TLE errors always been correlated, a single filter should have sufficed. Indeed, TLE filtering could be performed on completely different orbital elements (Lemmens & Krag, 2014; Dolado-Perez et al., 2014) or variables that include more than one orbital element in outlier identification. In principle, any set of filters that identifies outliers in every used orbital element, employs robust statistics, does not rely on SGP4 propagation and is rigorously tuned should identify most of the outlying TLEs.

6. Conclusions

This work has presented a set of algorithms that filter outlying two-line element sets in three orbital elements, B^* parameter and epoch. Improvements to the already existing filtering algorithms were made, which resulted in a final set of filters.

The numerical parameters of the filters were optimised and rigorously quantified using an artificial TLE time series to provide unambiguous reference data. Based on this study, it was shown that the filters can identify 88.1 to 100% of outlying TLEs. The performance of the optimised filters was also qualitatively assessed on six real objects and it was verified that the filters improve the accuracy of TLE-based analyses. Quantitative assessment on real objects was not possible because it is impossible to rigorously assert which TLEs are outliers.

Figure 26: Mean motion outliers identified for object 21057 using the optimised filter settings for 3^{rd} order polynomial regression closest to the origin from Table 3. Also showing outliers identified at other stages of filtering. Full figure for this object is available at www.plot.ly/~AleksanderLidtkke/67/.

Figure 27: Eccentricity outliers identified for object 21057 in the perigee radius time series using the optimised filter settings from Table 5. Also showing outliers identified at other stages of filtering. Full figure for this object is available at www.plot.ly/~AleksanderLidtke/69/.

Figure 28: Inclination outliers identified for object 21057 using the optimised filter settings from Table 7. Also showing outliers identified at other stages of filtering. Full figure for this object is available at www.plot.ly/~AleksanderLidtke/71/.

Figure 29: Mean motion outliers identified for object 21057 using the optimised filter settings for 1st order polynomial regression. Also showing outliers identified at other stages of filtering.

The optimised filters performed similarly to the ones that were tuned manually. This was likely caused by the fact that the manually tuned filter settings were chosen so as to filter all the significant outliers in the training set of 96 R/Bs. Because these outliers were in most cases also filtered out using the optimised filters, the filter optimisation methodology based on an artificial time series is considered verified. It could be readily applied to tune e.g. a modified filtering algorithm or to tailor the existing filters to a different, non-GTO orbital regime.

Even though this paper only investigated TLE-based orbit determination and ballistic coefficient estimation, which form key steps in the re-entry prediction algorithm, TLE filtering is recommended prior to any analyses that use these element sets. Filtering is necessary on any used orbital element because TLEs outlying in one element might not be outlying in others. This was shown by analysing changes in re-entry prediction accuracy for six rocket bodies when adding consecutive filtering stages. Each of the filters improved the accuracy for some of the investigated objects. The exact improvement of accuracy brought by filtering TLEs in a given orbital element depends on the details of the analysis that is being conducted, and so cannot be quantified in a generic fashion.

In four extreme cases out of the analysed sample of 96 R/Bs, filtering improved the re-entry prediction accuracy by up to 43.6%. This signifies that TLE filtering is necessary to avoid considerably inaccurate re-entry predictions. However, it does not guarantee accuracy in all cases because filtering may reduce the amount of usable data and outlying TLEs are not the only source of inaccuracy in re-entry prediction.

7. Acknowledgements

This work was partly performed within the scope the European Space Agency project ITT AO/1-8155/15/D/SR, titled “Technology for Improving Re-Entry Predictions of European Upper Stages through Dedicated Observations”.

The authors acknowledge the use of the IRIDIS High Performance Computing Facility at the University of Southampton in the completion of this work. Additionally, the Joint Space Operations Center is acknowledged for providing Vector Covariance Messages for 53 objects used for testing of the algorithms.

The authors acknowledge Dr Hugh G. Lewis and Dr Artur Lidtke (University of Southampton), Carlos Losada de la Lastra (BMT SMART), Dr Camilla Colombo (Politecnico di Milano), and Dr Tim Flohrer and Quirin Funke (European Space Agency) for their valuable contributions.

Aleksander A. Lidtke would like to acknowledge the funding he received from the Ministry of Education, Culture, Sports, Science and Technology of Japan. R. Armellin acknowledges the support received by the Marie Skłodowska-Curie grant 627111 (HOPT - Merging Lie perturbation theory and Taylor Differential algebra to address space debris challenges). D.J. Gondelach was funded by the Surrey Space Centre of the University of Surrey.

Appendix A. Test object catalogue numbers

00625, 02609, 03030, 07252, 07468, 07794, 08479, 08598, 08702, 08793, 08810, 08910, 09017, 09542, 09787, 09859, 10857, 10983, 11072, 11117, 11718, 11719, 12562, 12810, 13025, 13087, 13098, 13136, 13294, 13447, 13599, 13639, 13684, 13940, 14130, 14168, 14287, 14332, 14369, 14423, 14787, 14989, 15157, 15165, 15679, 16600, 18352, 18923, 19218, 19332, 19877, 20042, 20123, 20254, 20778, 20920, 21057, 21141, 21654, 21766, 21874, 21895, 21990, 22118, 22254, 22906, 22928, 22932, 22997, 23315, 23416, 23572, 23797, 23916, 24314, 24666, 24770, 24799, 24821, 24847, 24933, 25051, 25129, 25154, 25240, 25313, 25372, 25496, 25776, 26560, 26576, 26579, 26641, 27514, 27719, 27808, 28185, 28239, 28253, 28418, 28452, 28623, 28703, 29497, 32764, 36829, 37211, 37239, 37257, 37482, 37764, 37805, 37837, 37949, 39499, 40142.

Burton, R., Rock, S., Springmann, J., & Cutler, J., 2017, Online attitude determination of a passively magnetically stabilized spacecraft, *Acta Astronautica*, vol 133, pp. 269–281.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T.A.M.T., 2002, A fast and elitist multi-objective genetic algorithm: NSGA-II, *IEEE transactions on evolutionary computation*, vol. 6(2), pp. 182–197.

DeGroot, M.H., & Schervish, M.J., *Probability and Statistics*, Fourth Edition, Pearson Education Limited, Harlow, UK, 2014.

Dolado-Perez, J.D., Garcia, L.A., Agueda Mate, A., & Llamas de la Sierra, I., 2014, Opera: A tool for lifetime prediction based on orbit determination from TLE data, 24th International Symposium on Space Flight Dynamics, Laurel, MD, USA.

Easthope, P.F., 2014, Examination of SGP4 along-track errors for initially circular orbits, *IMA Journal of Applied Mathematics*, on line, pp. 1–15.

Gondelach, D.J., Lidtke, A.A., Armellin, R., Colombo, C., Lewis, H.G., Funke, Q., & Flohrer, T., 2016, Re-entry prediction of spent rocket bodies in GTO, 26th AAS/AIAA Space Flight Mechanics Meeting, Napa, CA, USA.

- Gondelach, D.J., Armellin, R., & Lidtke, A.A., 2017, Ballistic coefficient estimation for re-entry prediction of rocket bodies in eccentric orbits based on TLE data, *Mathematical Problems in Engineering*, Volume 2017, Article ID 7309637.
- Hejduk, M.D., Casali, S.L., Cappellucci, D.A., Ericson, N.L., & Snow, D.E., 2013, A catalogue-wide implementation of general perturbations orbit determination extrapolated from higher order orbital theory solutions, *Spaceflight Mechanics Conference*, Kauai, HI, USA.
- Hoots, F.R., & Roehrich, R.L., 1988, SPACETRACK REPORT NO. 3 Models for Propagation of NORAD Element Sets.
- Izzo, D., & Biscani, F., 2015, Welcome to PyGMO, <http://esa.github.io/pygmo/>, Accessed: 28 Nov 2017.
- Keane, A.J. and Nair, P.B., 2005, *Computational Approaches for Aerospace Design: The Pursuit of Excellence*, John Wiley & Sons, Ltd. doi:10.1002/0470855487.
- Kelec, T., Hall, D., Hamada, K., & Stocker, D., 2007, Satellite maneuver detection using two-line element (TLE) data, *Advanced Maui Optical and Space Surveillance Technologies Conference*, Wailea, Maui, HI, USA.
- King-Hele D.G., & Walker, D.M.C., 1987, The Contraction of Satellite Orbits Under the Influence of Air Drag. VII. Orbits of High Eccentricity, with Scale Height Dependent on Altitude, *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 411 (1840), pp. 1–17.
- Klinkrad, H., 2010, *Space Debris: Models and Risk Analysis*, Wiley Online Library. doi:10.1007/3-540-37674-7.
- Lemmens, S., & Krag, H., 2014, Two-line-elements-based maneuver detection methods for satellites in low earth orbit, *Journal of Guidance, Control, and Dynamics*, vol. 37, pp. 861–868.
- Li, T., & Chen, L., 2018, Historical-orbital-data-based method for monitoring the operational status of satellites in low Earth orbit, *Acta Astronautica*, vol. 151, pp. 88–94.
- Li, T., Li, K., & Chen, L., 2018, New manoeuvre detection method based on historical orbital data for low Earth orbit satellites, *Advances in Space Research*, vol. 62(3), pp. 554–567.
- Lidtke, A.A., Gondelach, D., Armellin, R., Colombo, C., Lewis, H.G., Funke, Q., & Flohrer, T., 2016, Processing two line element sets to facilitate re-entry prediction of spent rocket bodies from geostationary transfer orbit, *International Conference on Astrodynamics Tools and Techniques*, Darmstadt, Germany.

- Mendoza, T., Blanco-Avalos, J.J., & Martin-Torres, J., 2017, Interplanetary Coronal Mass Ejection effects on thermospheric density as inferred from International Space Station orbital data, *Advances in Space Research*, vol. 60(10), pp. 2233–2251.
- Pardini, C., & Anselmo, L., 2012, Reentry predictions of three massive uncontrolled spacecraft, 23rd International Symposium on Space Flight Dynamics, Pasadena, CA, USA.
- Pardini, C., & Anselmo, L., 2013, Re-entry predictions for uncontrolled satellites: results and challenges, 6th IAASS Conference Safety is Not an Option, Montreal, Canada.
- Patera, R., 2008, Space event detection method, *Journal of Spacecraft and Rockets*, vol. 45, pp. 554–559.
- Perktold, J., Seabold, S., & Taylor, J., 2017, Statsmodels: LOWESS, http://www.statsmodels.org/dev/generated/statsmodels.nonparametric.smoothers_lowess.lowess.html, Accessed: 19 Dec 2017.
- Picone, J.M., Emmert, J.T., & Lean, J.M., 2005, Thermospheric densities derived from spacecraft orbits: Accurate processing of two-line element sets, *Journal of Geophysical Research*, vol. 110.
- San-Juan, J.F., Perez, I., San-Martin, M., & Vergara, E.P., 2017, Hybrid SGP4 orbit propagator, *Acta Astronautica*, vol. 137, pp. 254–260.
- Song, W.D., Wang, R.L., & Wang, J., 2012, A simple and valid analysis method for orbit anomaly detection, *Advances in Space Research*, vol. 49, pp. 386–391.
- Vallado, D.A., Crawford, P., Hujsak, R., & Kelso, T.S., 2006, Revisiting spacetrack report# 3: Rev 2, AIAA/AAS Astrodynamics Specialist Conference, Keystone, CO, USA.
- Vallado, D., *Fundamentals of Astrodynamics and Applications*, 4th edition, Microcosm Press, 2013.