Orbital lifetime estimation using ESA's OSCAR tool

V. Braun⋆*, S. Flegel*⋆*, J. Gelhaus*⋆*, M. Möckel*⋆*, C. Kebschull*⋆*, J. Radtke*⋆*, C. Wiedemann*⋆*, H. Krag*⋆⋆*,*
P. Vörsmann⋆
⋆*Institute of Aerospace Systems, Technische Universität Braunschweig,*
Hermann-Blenk-Str. 23, 38108 Braunschweig, Germany
⋆⋆*ESA/ESOC Space Debris Office,*
Robert-Bosch-Str. 5, 64293 Darmstadt, Germany

Abstract

The UN Space Debris Mitigation Guidelines (SDMG) recommend a disposal manoeuvre for spacecraft in low Earth orbit (LEO) after the end of the operational phase. A typical disposal manoeuvre results in the perigee altitude being decreased so that the spacecraft will enter Earth's atmosphere within 25 years at maximum. However, the definition of such an orbit is non-trivial. The OSCAR (Orbital Spacecraft Active Removal) tool allows for the investigation of such scenarios and uses an iteration algorithm to find the appropriate disposal orbit. In this paper, different iteration methods are compared for the orbital lifetime estimation problem.

1. Introduction

The increasing number of objects in the frequently used LEO (low Earth orbit) and GEO (geostationary Earth orbit) regions has led to the implementation of mitigation guidelines in a step towards the control of the evolution of the space debris environment. The UN Space Debris Mitigation Guidelines (SDMG),[7] which were endorsed by the UN General Assembly in 2008, define the protected regions in LEO (from Earth's surface up to an altitude of 2,000 km) and in GEO (geostationary altitude $\pm 200 km$, $-15° <$ latitude $< +15°$). All spacecraft residing in or crossing those regions are supposed to perform a disposal manoeuvre in order to prevent a long-term stay in the protected regions after the end of the operational phase. While for GEO spacecraft operators have to make sure that their spacecraft does not interfere with that region for at least 100 years, for spacecraft in LEO the 25-year rule was implemented, recommending to manoeuvre the spacecraft on a disposal trajectory which leads to an atmospheric re-entry within the next 25 years. The ESA (European Space Agency) software DRAMA (Debris Risk Assessment and Mitigation Analysis) consists of five tools, with OSCAR (Orbital Spacecraft Active Removal) being the tool designed specifically for the analysis of different disposal strategies as the ones described above. In Section 2, more details on this tool will be given, which has been upgraded recently.[1]

In this paper, the estimation of the appropriate disposal orbit, with respect to an user-defined orbital lifetime and the selected disposal strategy, shall be shown for different root finding algorithms which were investigated during the development of the OSCAR tool. The iteration methods are described in Section 3 with their properties regarding the orbital lifetime function. They have been primarily applied to different LEO orbits (Section 4), however, an outlook is also given for high-eccentricity orbits (HEO), where orbital lifetime is also a function of perigee altitude, but can be even more affected by the variation of other orbital parameters. Therefore, the search algorithm applied to the perigee altitude only does not lead to an optimized solution for HEO objects. The *orbital lifetime function*, the terminology used in this paper, is given via the semi-analytical orbit propagation tool FOCUS-1A (Fast Orbit Computation Utility Software), which is also used within OSCAR.

2. The OSCAR tool

OSCAR is one of five tools within the DRAMA software and addresses the disposal phase of a satellite mission. It allows for the analysis of different disposal strategies considering standardized future solar and geomagnetic activity and assessing the compliance of the disposal manoeuvre with the SDMG.

As is described in more detail in Section 3, the modeling of the future solar and geomagnetic activity is the main driver in the estimation of the residual lifetime for a specific disposal orbit. Several different methods exist to predict future solar activity. However, due to a poor understanding of the physical process associated with that activity, as well as only limited historical data available for statistical analysis, the prediction accuracy of all of the available methods declines relatively fast the longer the prediction interval is. Therefore, it is required to have a concensus approach which can be used to verify the compliance of a selected disposal orbit with e.g. the 25-year rule. The ISO 27852:2011[5] (Space systems - Estimation of orbit lifetime) provides two approaches for the activity predictions, which both were implemented in OSCAR. Furthermore, OSCAR also allows to model future solar and geomagnetic activity by applying the ECSS[3] standard cycle, and a standard equivalent solar cycle.[4]

For an user-defined orbit, OSCAR computes the orbital lifetime and checks whether an action is required to be compliant with the SDMG. If a manoeuvre needs to be performed, the user may select between different disposal systems:

- Chemical propulsion (direct or delayed de-orbit, re-orbit)
- Electric propulsion (delayed de-orbit, re-orbit)
- Electrodynamic tethers (delayed de-orbit, re-orbit)
- Drag augmentation system (delayed de-orbit)

All of the user-defined scenarios are evaluated with respect to the SDMG, which is done via a set of pre-defined non-compliance criteria:

1. Lifetime of LEO crossing spacecraft > 25 years.
2. LEO protected region crossing within 100 years.
3. GEO protected region crossing within 100 years.

Besides the computation of the user-defined scenario, OSCAR also estimates key system requirements, being the Δv and fuel mass, for a disposal manoeuvre which would be compliant with the SDMG.

3. Searching disposal orbits with specified lifetime

The OSCAR tool allows to specify the required lifetime of the disposal orbit, which is then estimated during the execution of the program, taking into account the initial orbit of the spacecraft and the system properties of the selected disposal system. The main problem is that the orbital lifetime can only be estimated by propagating it forward in time until the atmospheric re-entry of the spacecraft. Depending on the method used, this can be a quite tedious task, as e.g. high precision numerical propagation for some decades already takes minutes of computation time, which would correspond to only one iteration step. The ISO 27852:2011[5] gives three different methods for the estimation of orbit lifetime:

- High-precision numerical integration
- Rapid semi-analytical orbit propagation
- Numerical table look-up, analysis and fit equations

While the numerical integration is supposed to provide the most accurate results, it consumes a lot of time and the results may be questionable anyway, as very detailed input data would be required for accurate trajectories, which is not available in general for long-term propagations, and the inherent inaccuracies of the atmosphere models used.

For OSCAR, the second ISO method was selected, using the fast semi-analytical orbit propagator FOCUS-1A. It accounts for geopotential coefficients J_2 through J_5, including J_2 short-periodic variations, drag perturbations as well as third body perturbations (Moon and Sun) and solar radiation pressure. The latter two perturbations become significant for sun-synchronous and high-eccentricity orbits.

The justification for using that method is that there are many parameters which may have completely different sensitivities associated with the estimation of orbital lifetime. This is shown in Figure 1 for different disposal orbits. The orbit lifetime as a function of perigee altitude was computed for different scenarios. First of all, the two red curves in the center of Figure 1 show the lifetime for a spacecraft with an area-to-mass ratio of 0.01 m^2/kg, which was initially positioned at a 800 km altitude mid-inclination orbit and then transferred to a disposal orbit with the given perigee altitude. It can be seen that compliance with the 25-year rule could be achieved for perigee altitudes below 570 km. There are, however, variations depending on when the disposal manoeuvre is performed with respect to the solar cycle. The thick red curve corresponds to a disposal manoeuvre performed at May 1, 1976, which is near the solar minimum

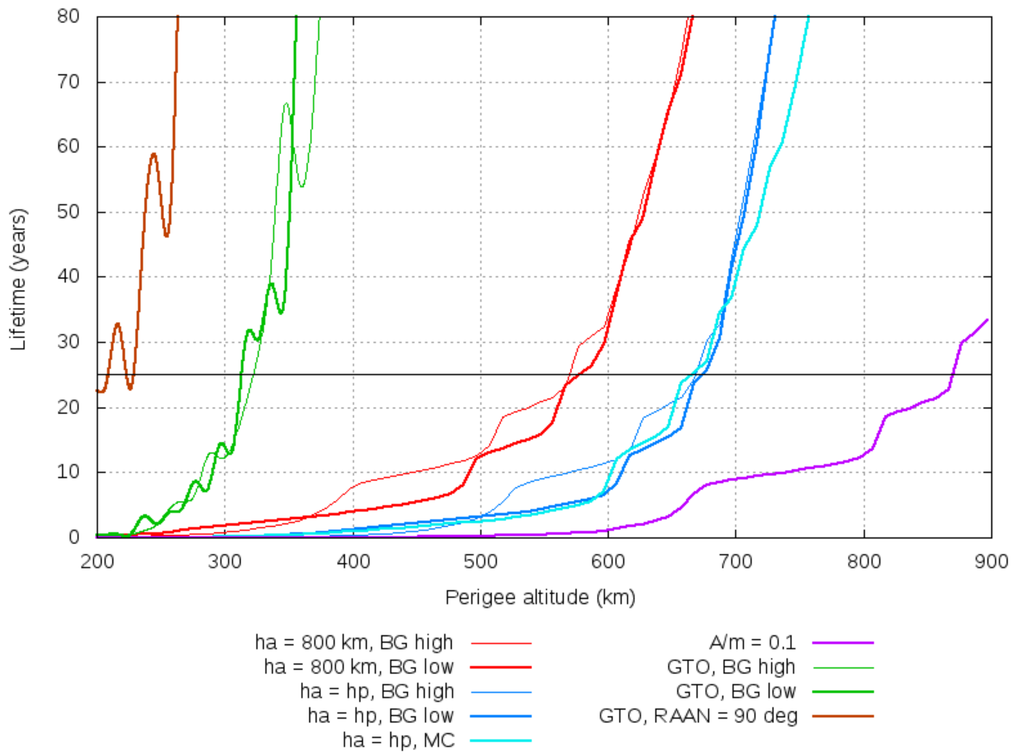

Figure 1: Orbital lifetime as a function of perigee altitude for different disposal orbits as well as different begin dates ('low' corresponds to May 1976, 'high' to May 1970) and future solar activity modeling ('BG' = best-guess (latest prediction), 'MC' = Monte-Carlo)

of solar cycle 20, while the thin red curve shows the orbit lifetime for a manoeuvre performed at May 1, 1970, which is near the maximum of the same cycle.

For circular orbits (blue curves), which e.g. would result from using an electric propulsion system, the shape of the lifetime function looks similar but is shifted towards higher perigee altitudes, as now the apogee altitude is significantly lower and drag is acting along the complete orbital arc with the same magnitude. The electric propulsion system would have to transfer the same spacecraft to an altitude below 680 km in order to be compliant with the 25-year rule, again with similar variations due to different manoeuvre dates. A third light blue curve is shown, which was computed for a Monte Carlo generated solar and geomagnetic activity cycle. It can be seen that differences between a best-guess (latest prediction) and a Monte Carlo approach become significant for higher altitudes, but there are also differences in this example near the 25-year line (perigee altitude ca. 650 km), where the Monte Carlo approach results in an orbital lifetime estimate which is six years higher than the latest prediction estimate.

For the same scenario as represented by the red curves, the area-to-mass ratio was increased by a factor of ten to result in the purple curve shown in Figure 1. As the spacecraft is on a 800 km altitude orbit, it would not be required to perform a disposal manoeuvre, as the residual lifetime is well below 25 years. This could be, e.g., the motivation to use a drag augmentation system.

As spacecraft on high-eccentricity orbits may also be crossing the LEO region and thus have to perform a disposal manoeuvre, the lifetime estimation has to be performed for those objects, too. In Figure 1 this is shown by the green curve for an object on a low-inclination GTO (geosynchronous transfer orbit), varying the perigee altitude, while the apogee is at geosynchronous altitude. Here, not only the levels of solar and geomagnetic activity at the start of the simulation affect the results, but due to the complex interaction of the geopotential, Earth's atmosphere and luni-solar perturbations, there are also significant oscillations with respect to the initial perigee altitude. Even more important is the initial orientation of the orbit $\Omega + \omega$, as a RAAN (right ascension of ascending node) difference of 90° does result in a completely different behaviour as shown by the dark red curve for the same GTO.

All the effects and parameters with their associated sensitivities, as described above, have to be taken into account, when estimating a disposal orbit which provides the residual orbital lifetime according to the user-defined requirements. From a mathematical point of view this problem corresponds to finding a root for a given function. The *root* is the perigee altitude for the specified lifetime, e.g. 25 years for the SDMG, while the *function* would be the orbit propagation tool. In the following, the root finding methods are described, which were then applied for the orbital lifetime estimation process.

3.1 Bisection method

The bisection method is one of the simplest and most robust methods to search for roots of a given function. However, it is relatively slow. In general, for all of the presented root finding methods to work, two function values have to be known, bracketing the root. As OSCAR performs a first propagation in any case to check whether the lifetime of the initial orbit is lower than the specified limit, this result will always yield the first function value. If it is below the limit, the iteration does not need to be performed, as no manoeuvre is required. Therefore, when the iteration starts, OSCAR always will have a positive (above the specified lifetime, as this is assumed to be the root, see above) function value available for the perigee altitude $h_{p,i}$ of the initial orbit. In order to complete the bracketing, one only needs a negative value, thus being below the specified limit. This is achieved by setting the perigee altitude $h_{p,100}$ to 100 km in OSCAR, which will always result in a quick re-entry. Having the two values available, the bisection method now computes the function value at the mid-altitude $h_{p,new}$:

$$h_{p,new} = \frac{h_{p,i} + h_{p,100}}{2} \tag{1}$$

If the lifetime for $h_{p,new}$ is below the specified limit, $h_{p,new}$ becomes the new lower boundary, while $h_{p,i}$ remains the upper boundary. If it is above the limit, then $h_{p,100}$ is the lower and $h_{p,new}$ the upper boundary in the next step. An example for a typical 25-year orbit computation using the bisection method for an initial orbit at 700 km altitude, reducing only the perigee altitude, while the apogee altitude remains constant, is shown in Table 1. The example in

Table 1: Bisection method example for an initial orbit at 700 km altitude, searching for the required perigee altitude to be compliant with the 25-year rule.

Step	$h_{p,100}$ in km	$h_{p,i}$ in km	$h_{p,new}$ in km	$f(h_{p,new})$ in y
1	100.00	700.00	400.00	0.4545
2	400.00	700.00	550.00	9.5443
3	550.00	700.00	625.00	19.0367
4	625.00	700.00	662.50	28.4578
5	625.00	662.50	643.75	20.9258
:	:	:	:	:
16	659.72	659.79	659.75	24.9615

Table 1 shows, that 16 iterations are required, meaning that the propagation is performed 16 times, which is quite time consuming. The stop criterion was applied to the lifetime so that it is definitely below the specified value of 25 years and is above 24.95 years, the tolerance thus being 0.05 years or about 18 days. The same tolerance was applied to the other methods.

3.2 Secant method

The secant method uses secant lines connecting $f(h_{p,i})$ and $f(h_{p,100})$ to obtain the next perigee altitude for the evaluation of the lifetime function. With $x_0 = h_{p,a}$ and $x_1 = h_{p,b}$, the secant line has the following equation:

$$y(x) = f(x_1) + (x - x_1) \cdot \frac{f(x_1) - f(x_0)}{x_1 - x_0} \tag{2}$$

As for the root of this line, it is required to set $y(x) = 0$, so that the equation can be solved for x, which is the next iterate:

$$x = x_1 - f(x_1) \cdot \frac{x_1 - x_0}{f(x_1) - f(x_0)} \tag{3}$$

The lifetime function is evaluated at x and for the next iteration step, x_1 becomes x_0 and x is set to x_1. An example how the secant method behaves for the orbit lifetime estimation problem is shown in Figure 2. The first secant connects the two initial points at 100 km and 900 km altitude. The root of the resulting secant is approximately at 200 km, which again results in a very low orbital lifetime. Here, a problem inherent to the secant method can be observed: In the next iteration, the function values at 100 and 200 km will be used, resulting in a secant with a very flat slope, which intersects the 25-year line at about 15.150 km (not shown in the plot). As can be seen from the red curve, OSCAR has a cutoff for orbital lifetimes above 200 years, i.e. the propagation time is limited to this value. This is a severe

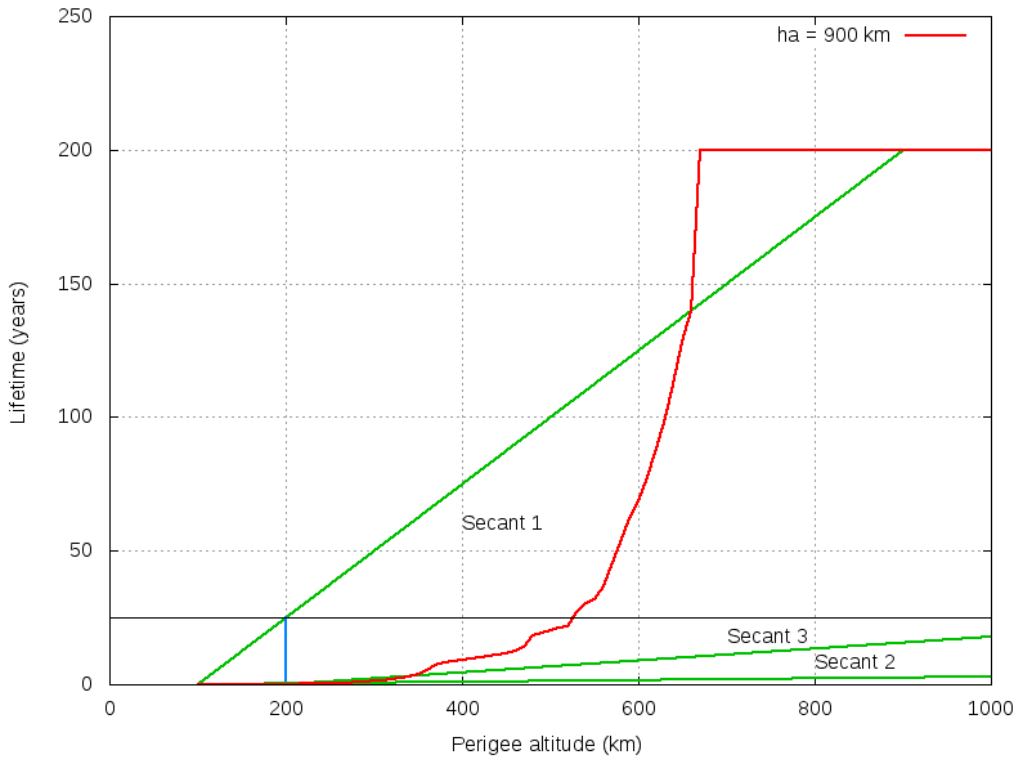

Figure 2: Secant method example for an initial orbit at 900 km altitude. The iteration searches the perigee altitude which would be required for a 25-year residual lifetime orbit.

problem for the secant method, as the third secant will connect the point at 200 km altitude and the point (15.150 km, 200 years), which is again a flat secant, in the end providing the next iterate which will again be above 700 km and thus have a residual lifetime of more than 200 years. Having two function values at 200 years now means that the secant will be horizontal and the secant method diverges as the next iterate can not be computed.

This example clearly shows, that the secant method is not applicable for the orbital lifetime estimation problem, when a cutoff exists and the initial estimates x_0 and x_1 are not close to the root. Therefore, in order to get the method working, one would have to provide those two values close to the required root. However, as was already shown in Fig. 1 this is not an easy task, as the searched-for perigee altitude is highly sensitive to many parameters. The secant method was thus not selected for the detailed analysis in Section 4.

3.3 Brent method

In 1973, Richard P. Brent proposed a method[2] which combines the bisection and the secant method as well as an inverse quadratic interpolation to find a root for a given function. If there are three iterates $x_k = c$, $x_{k-1} = b$ and $x_{k-2} = a$, the next iterate x is computed via:

$$x = \begin{cases} \frac{c \cdot f(b) \cdot f(a)}{(f(c)-f(b)) \cdot (f(c)-f(a))} + \frac{b \cdot f(c) \cdot f(a)}{(f(b)-f(c)) \cdot (f(b)-f(a))} + \frac{c \cdot f(a) \cdot f(b)}{(f(c)-f(a)) \cdot (f(c)-f(b))} & \text{if} \quad f(c) \neq f(a) \quad \text{and} \quad f(b) \neq f(a) \\ & \quad \text{(inverse quadratic interpolation)} \\ c - f(c) \cdot \frac{c-b}{f(c)-f(b)} & \text{else (secant method)} \end{cases} \tag{4}$$

For the first iteration step, x_{k-1} would correspond to $h_{p,i}$ (perigee altitude of initial orbit) and x_{k-2} to $h_{p,100}$ (100 km). As no third value is available, the Brent method would always start with a secant method, as x_k (= c) is set to x_{k-2} (= a) initially. But as was shown for the Secant method example in Section 3.2, the lifetime function would provide a value which is very close to $h_{p,b}$ (or a). Now comes what is special about Brent's method: The computed value for x is only accepted, if certain specific algorithm conditions are met, otherwise the bisection method is used. In this example, the condition

$$||x - b|| < \frac{1}{2} \cdot ||b - a|| \tag{5}$$

is not fulfilled, as $x-b$ spans almost the complete interval and is thus larger than half the interval $b-a$. So the first step is actually a bisection step and the next iterates are then computed via an inverse quadratic interpolation or the secant method and switched to the bisection method if convergence would be too slow otherwise.

An improvement was proposed by Zhang in 2011,[8] which simplifies many of the implementation issues in the Brent algorithm and also increases the algorithm efficiency, as fewer logical evaluations have to be performed. However, it requires an additional function evaluation per iteration step. As an orbit propagator is used for the function evaluations, and almost all of the computation time is spent within the propagation, the efficiency of the root finding algorithm can be considered negligible. Only the minimization of the number of function calls leads to a significant performance improvement. Therefore, Zhang's method was not considered for further analysis.

3.4 Regula Falsi method

The *regula falsi* or *false position* method combines the bisection and the secant method. Using Equation 2, the next iterate is found via the secant method. But while the secant method always uses the current and the last iterate in each step, the false position method decides, which two points are used for the secant in the next step. As the first two values $a = h_{p,i}$ and $b = h_{p,100}$ are bracketing the root, the secant method will definitely provide a value c between these two points. If now $f(a)$ has the same sign as $f(c)$, then c becomes a in the next step, otherwise b is set to c. This is actually the behaviour, which a bisection method shows. This way, the false position method can always ensure that the root is bracketed, while the secant method alone is not able to do so, which led to the divergent behaviour in the example in Section 3.2.

3.5 Ridder method

Another root finding algorithm is Ridder's method.[6] It is similar to the Regula Falsi method and uses an exponential function for the estimation of the next iterate. However, it requires two function evaluations per step and was thus not considered to be further analysed, as was also the case for Zhang's method (Section 3.3).

4. Results

In the following, the three root-finding algorithms Bisection, Brent's method and Regula Falsi, which were presented in Section 3, were applied to LEO and HEO disposal trajectories. The goal was to identify the method which performs best in finding the disposal orbit for a given initial orbit and a pre-defined disposal strategy. The number of lifetime function evaluations was considered to be the quality criterion. As methods which contain multiple function evaluations per iteration step were not considered in this analysis, the number of iterations of an algorithm was always equal to the number of function evaluations. While the root finding algorithms normally assume convergence if the difference between subsequent iterates is less than a given tolerance, for the problem of finding an orbit with a specified lifetime, the stop criterion can also be with respect to the difference in the computed orbital lifetime for two iteration steps. A tolerance of 0.05 years, which corresponds to about 18 days, was defined, so that the search algorithm was stopped, as soon as the orbital lifetime was between 24.95 and 25 years. In general, all of the following results are based on finding the orbit for a 25-year lifetime.

4.1 Low Earth orbits

The type of disposal orbit a LEO spacecraft typically transfers to depends on the disposal system used. For example, a spacecraft on an initially circular orbit at 800 km, would adapt its perigee altitude within one impulsive engine burn using a chemical propulsion system. With the apogee still at its initial altitude, the satellite will now experience increased drag perturbations near its perigee which leads to the required orbital decay.

If, however, electric propulsion or electrodynamic tethers are used, the spacecraft would be transferred from an initially circular orbit to an orbit with lower altitude (but still circular) by means of a low-thrust manoeuvre.

Therefore, the behaviour of the selected methods was tested for circular orbits (Section 4.1.1) at different altitudes, as well as orbits with a moderate eccentricity, resulting from an impulsive engine burn at the apogee of the initial orbit (Section 4.1.2).

4.1.1 Circular orbits in LEO

In Figure 3 the convergence of each method is shown for an initial orbit at 800 km, where the algorithm searches for a circular disposal orbit with 25 years lifetime. It can be seen that Brent and Bisection behave in a similar way in

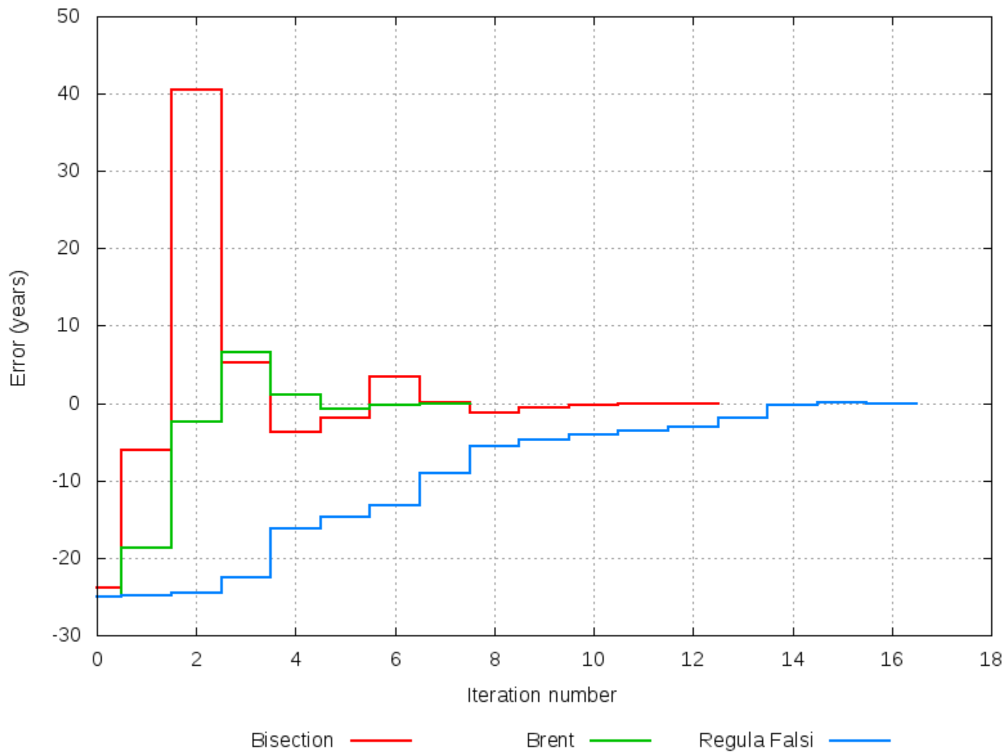

Figure 3: Iteration properties for the Bisection, Brent and Regula Falsi methods for an 800 km orbit and a searched-for disposal manoeuvre which would transfer the spacecraft to a 25-year lifetime trajectory, which is circular.

terms of having both positive and negative errors. But the Brent method converges faster. The Regula Falsi method, however, starts with an initial error of about -25 years (which is an orbit of nearly zero lifetime) and slowly increases towards zero, leading to a high number of sixteen required iterations, while Brent converged within seven iterations. The explanation for the behaviour the Regula Falsi method shows is the combination of using the Secant method only and the properties of the lifetime function which were shown in Figure 2. As Regula Falsi, in contrast to the Secant method presented in Section 3.2, always takes care about the root being bracketed, the resulting secants will maintain its upper interval value at the initial orbit altitude, as is shown in Figure 4. Due to the shallow increase of the orbit lifetime in lower altitudes, the gradient of the secant is only slightly changed within each iteration, leading only to a very slow convergence rate. This, however, is a function of the initial orbit altitude, which results in the upper interval value being shifted along the 200 years cutoff line and thus resulting in the secant being less inclined for higher altitudes. A simulation for different initial altitudes up to 2,000 km was performed to demonstrate this effect. The results are shown in Figure 5. It can be seen that the Regula Falsi method requires significantly more iterations at altitudes near 800 km compared to the Brent and Bisection methods. As soon as the initial altitude gets above 1,000 km, however, the Secant method will always provide a first guess (the root of the secant) which is already close to the searched-for 25 years or at least in an area, where the lifetime function already shows a significant increase with increasing altitude. While only looking at the number of iterations so far, a very important point regarding the lifetime function properties has not been discussed: In terms of run time performance, the number of iterations is not necessarily the most important criterion when using the orbital lifetime function, as the function evaluation is comparatively short when evaluated for lower altitudes in contrast to the evaluations at higher altitudes. Therefore, even if the Regula Falsi method requires more iterations than the Brent or Bisection methods, it only evaluates the lifetime function at lower altitudes in those cases, leading to a maximum of 25 years propagation per evaluation step. A comparison for three different altitudes, based on the scenario in Figure 5, is shown in Table 2. While for the 775 km case the 29 iterations of the Regula Falsi method, resulting in 425 propagated years, are significantly higher than the about 200 years by the Brent method, the situation changes for an orbit at 800 km. Here, the Regula Falsi method still requires two times the iterations as the Brent method and still 50 % more iterations than the Bisection method, but the number of propagated years shows that the Brent method (with about 160 years) and Regula Falsi (219 years) are getting close even with the high difference in iterations. What is even more interesting is that the Bisection method results in about 50 % increased propagation duration while having 25 % less iterations. This is just due to the effect described above - while the Brent and the

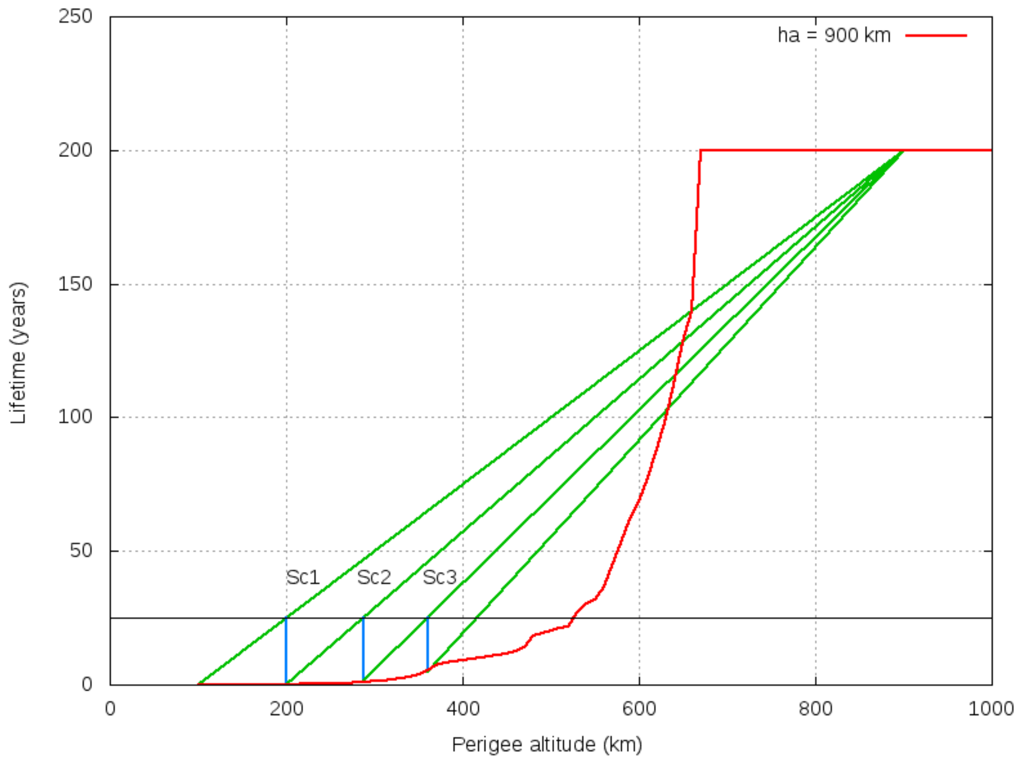

Figure 4: Properties of the Secant method in the Regula Falsi method for a 900 km initial orbit.

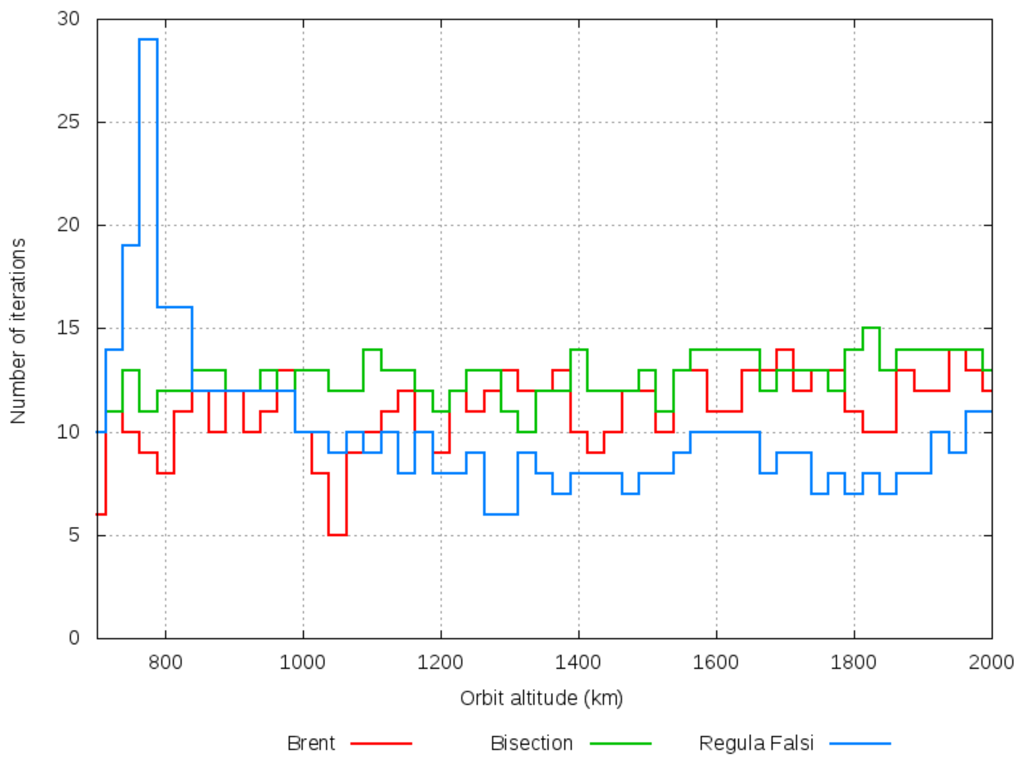

Figure 5: Number of iterations required to find a 25-year orbit for spacecraft with circular initial and disposal orbits for a given initial altitude.

Table 2: Number of propagated years for three distinct initial altitudes to find the circular 25-year disposal orbit.

	h = 775 km		h = 800 km		h = 1,400 km	
	# of iter.	# of years	# of iter.	# of years	# of iter.	# of years
Bisection	11	266	12	315	14	416
Brent	9	199	8	160	10	322
Regula Falsi	29	425	16	219	8	156

Bisection method both also compute iterations for orbit altitudes with high orbital lifetimes, the Regula Falsi method approaches the solution with function evaluations containing propagation durations of less than 25 years. For the 1,400 km case the Regula Falsi method requires only half of the propagation time compared to the Brent method, while the number of iterations is quite comparable (ten for Brent, eight for Regula Falsi).

In general the Brent method requires less iterations than the Bisection method, where in a few cases the number of iterations is even below ten for Brent. The Bisection method always needs at least ten iterations and is better than Brent only for a few altitude bins. The Regula Falsi method performs very well for orbits above 1,000 km and is better than Brent due to the already described properties of the lifetime function. But also for orbits below 1,000 km, the Regula Falsi method may provide better results, as different orbits and disposal strategies may result also in significantly less iterations compared to the example in Figure 5, as is shown for eccentric cases in the next section.

4.1.2 Eccentric orbits in LEO

For the estimation of a delayed de-orbit in LEO, OSCAR computes the required perigee altitude for a given disposal orbit lifetime, which would result after one perigee lowering manoeuvre at the apogee of the initial orbit. In OSCAR, this can only be achieved by the chemical propulsion system. In Figure 6 a similar behaviour of the three methods under consideration to the case of circular orbits, as shown in Figure 3, can be seen. The Brent method requires nine,

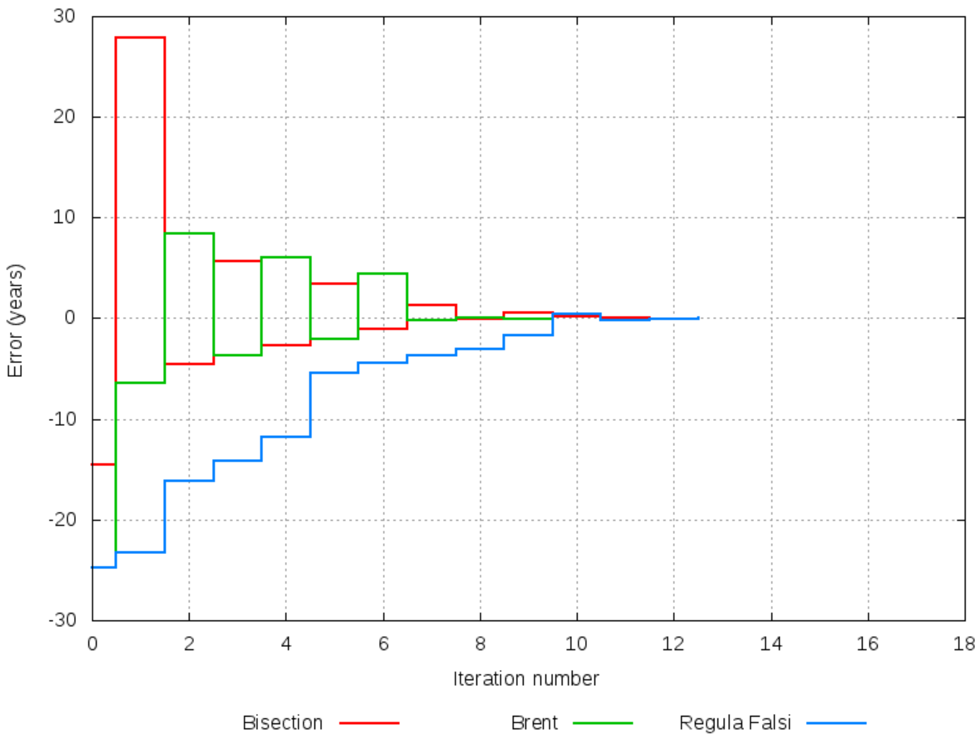

Figure 6: Iteration properties for the Bisection, Brent and Regula Falsi methods for an 800 km orbit and a searched-for disposal manoeuvre which would result in the perigee being decreased so that the resulting orbit provides 25 years lifetime at maximum.

the Bisection eleven and the Regula Falsi method twelve iterations to find the 25-year disposal orbit. The Regula Falsi method once again approaches the final result in smaller steps from the negative error values, while both, Brent and Bisection show higher oscillations compared to the scenario in Section 4.1.1. A similar behaviour as in Section 4.1.1

can again be observed when varying the initial apogee altitude in Figure 7. While the Regula Falsi method once again

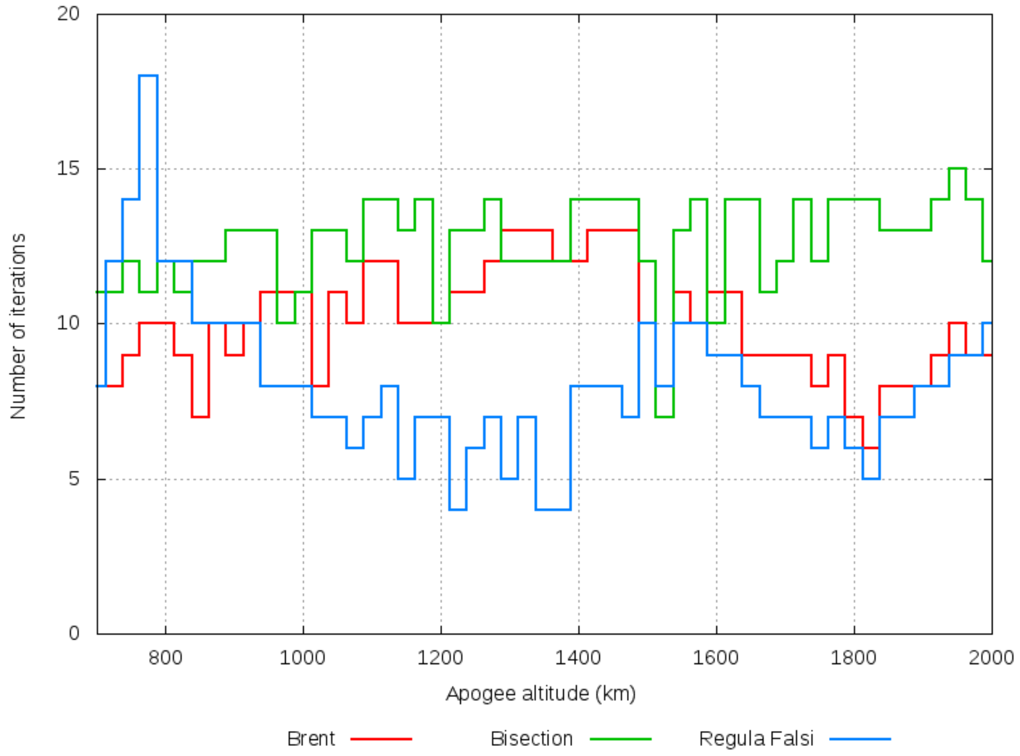

Figure 7: Number of iterations required to find a 25-year orbit for spacecraft at initially circular orbits with given apogee altitude.

shows that a high number of iterations is required for altitudes at about 800 km, that number decreases below ten for almost any altitude higher than 1,000 km. However, as already stated in Section 4.1.1, the optimization criterion should be the number of propagated years due to the properties of the lifetime function. In Table 3 the results for the propagation duration are shown for three different initial apogee altitudes. The Regula Falsi method requires 17

Table 3: Number of propagated years for three distinct initial apogee altitudes to find the elliptic 25-year disposal orbit.

	h = 775 km		h = 800 km		h = 1,400 km	
	# of iter.	# of years	# of iter.	# of years	# of iter.	# of years
Bisection	11	285	12	342	14	675
Brent	10	229	10	232	12	535
Regula Falsi	17	324	12	215	8	209

iterations for the 775 km orbit, which is twelve iterations less than for the circular case in Section 4.1.1. The value of 324 propagated years already comes close to the results of the Bisection and Brent methods. For an orbit at 800 km, the situation changes completely. Having the same number of iterations as the Bisection method and only two additional iterations compared to the Brent method, the Regula Falsi method results in requiring the least propagation years for that estimate and is thus the most efficient method here. This is even more pronounced for the 1,400 km orbit, where the Regula Falsi method requires only 209 years and eight iterations, while the Brent method requires 50 % of additional iterations and more than 100 % of additional propagation time.

4.2 High eccentricity orbits

While the estimation of the orbital decay for objects moving in LEO is mainly a function of perigee altitude, solar and geomagnetic activity, as well as the ballistic parameter, for orbits with high eccentricity, other effects may have a significant impact, too. Due to the interaction of Earth's oblateness and third body perturbations mainly due to the

Sun, the initial orientation of the orbit determines the evolution of the eccentricity. For a given orientation, eccentricity may increase with time, leading to a further decrease of the perigee altitude and thus higher drag, which also decreases the perigee altitude. However, if the orientation of the orbit changes, it may also happen, that eccentricity decreases, leading to an increase in perigee altitude. This effect was shown in Figure 1 for the two different GTO cases, where a difference in RAAN of 90° was applied. For the GTO case with RAAN at 0° (Argument of perigee also being at 0°) the three methods under consideration were applied for different initial perigee altitudes. Once again, the disposal orbit was searched, which results in a residual lifetime of about 25 years. The results are shown in Figure 8. It can

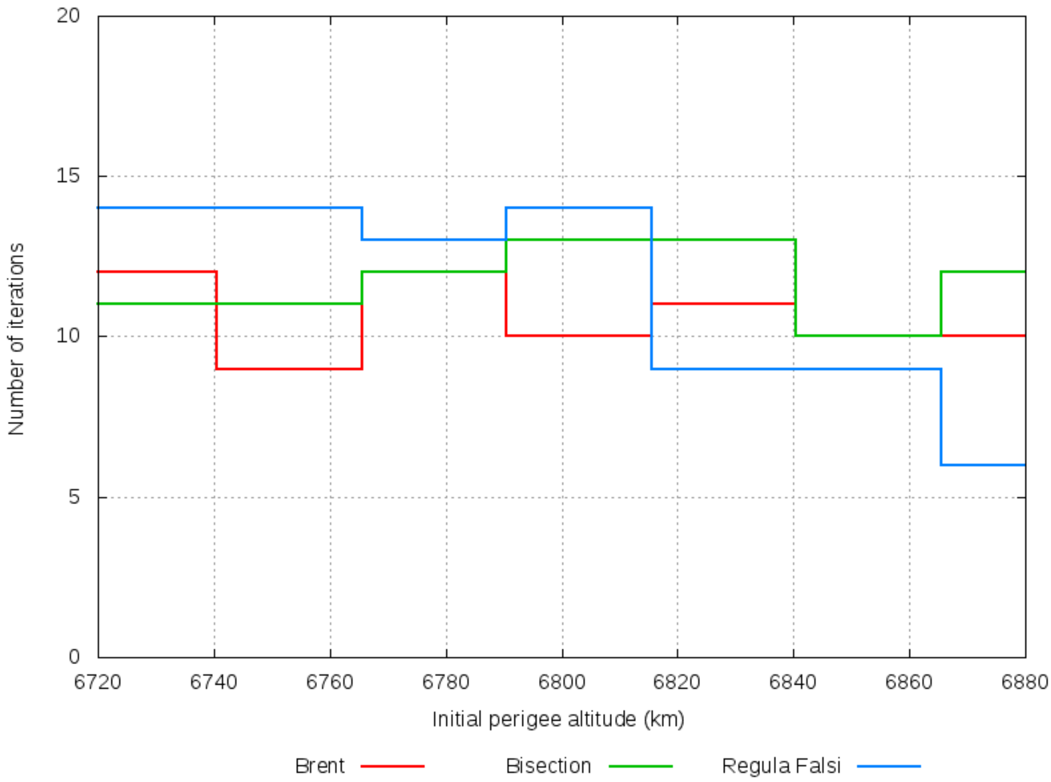

Figure 8: Number of iterations required to find a 25-year orbit for spacecraft in GTO as a function of initial perigee altitude.

be seen that the general behaviour of the three methods, which was already described in Section 4.1, shows up for high-eccentricity orbits too. The Brent method requires less iterations with only a few exceptions when compared to the Bisection method. For orbits with a low perigee altitude, the Regula Falsi requires more iterations than for those with higher perigee altitudes, which was also the case for LEO. However, the difference in number of iterations is not that high, as was for example the case for some orbits near 800 km in the LEO case. Taking a closer look at a GTO with a perigee altitude of 400 km, the Regula Falsi method results in a total propagation time of 218 years in 13 iterations, while the Brent emthod with 296 years (12 iterations) and Bisection method with 363 years (12 iterations) do not perform that well, although the number of iterations is lower.

The convergence rate of the three methods for a GTO is shown in Figure 9. The initial convergence is quite fast in terms of all methods being close to the root after about five iterations. However, the final convergence takes up from seven to eight additional iterations. Similar to the LEO examples, the Regula Falsi method again approaches the result from negative errors, while the Brent and Bisection method both show oscillations.

5. Conclusion

The definition of a disposal orbit for spacecraft residing in or crossing the LEO protected region, which is in accordance with the space debris mitigation guidelines, presupposes the search for a trajectory which would result in the spacecraft burning up in the Earth's atmosphere within the next 25 years. The tool OSCAR provides the means to analyse different disposal strategies, evaluate the disposal strategy selected by the user, as well as search for an orbit with a defined orbital lifetime. For the latter case, assumptions regarding the solar and geomagnetic activity have to be made,

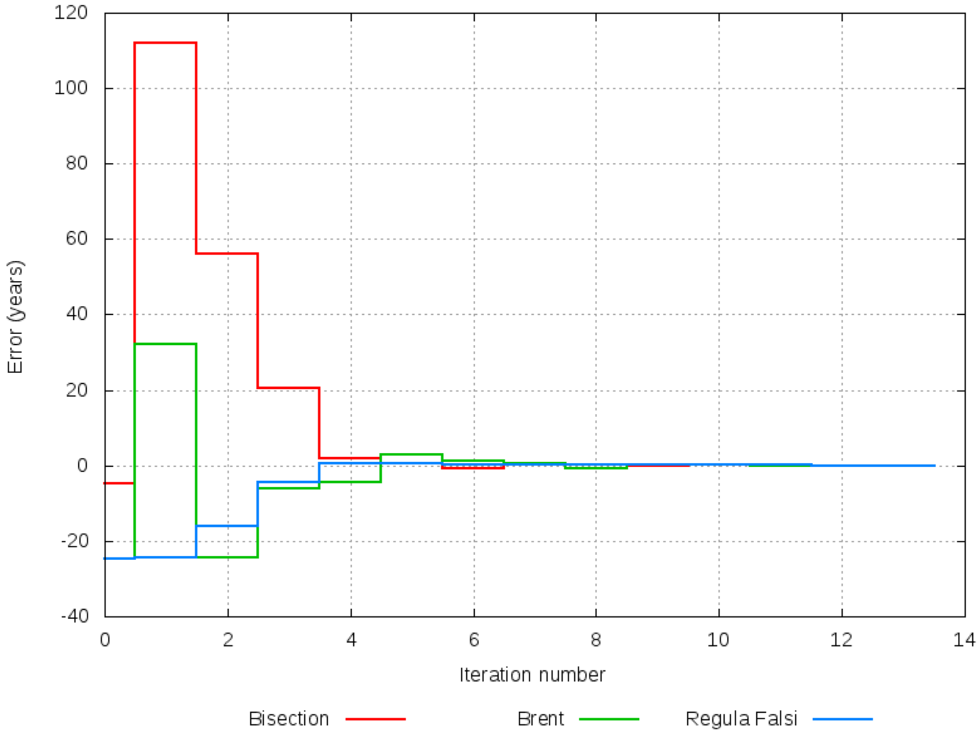

Figure 9: Iteration properties for the Bisection, Brent and Regula Falsi methods for a GTO with an initial perigee altitude at 400 km and a searched-for disposal manoeuvre which would result in the perigee being decreased so that the resulting orbit provides 25 years lifetime at maximum.

as they are the drivers in the orbital decay process. In order to have a standardized process which allows operators to verify their selected disposal orbit towards space agencies, standardized methods have to be used. Several standards have been published in the recent time to provide the methods to use in that process. In OSCAR, the different methods as recommended by different standards have been implemented, so as to provide a tool for operators combining the recommendations from the SDMG as well as the ISO and ECSS standards.

In this paper, root finding algorithms have been identified and analysed for the application in the problem of finding an orbit with a specified lifetime, e.g. 25 years. Due to the properties of the lifetime function, which has a very gentle slope for lower altitudes, followed by a steep increase and finally a cutoff, the Secant method was the first method identified to be not suitable. The cutoff is required in order to reduce computation time on the one side, but also to limit the size of the arrays containing daily kepler elements for the orbit state and daily as well as mean solar and geomagnetic activity data. The cutoff is arbitrary though, but for OSCAR it was decided to provide 200 years of propagation time at maximum. The Secant method is also applied by the Regula Falsi method, however, the latter method has an additional logic which takes care of the root always being bracketed between the interpolation values. Besides the Regula Falsi, also the Brent and Bisection method were investigated. Other methods, such as Ridder or Zhang required two function evaluations per iteration. As the optimization of the search algorithm consists in reducing the number of total lifetime function evaluations because the propagation run time is significantly higher than the algorithm logic, those methods were not selected for a more detailed analysis.

In the exemplary results as shown in Section 4, there are quite significant differences between the three methods Brent, Bisection and Regula Falsi. In general, the Brent method showed less required iterations than the Bisection method, however, there were some orbits where this was not the case. The Regula Falsi method was strongly affected by the lifetime function cutoff but could make use of the fact that the function evaluates significantly faster for lower altitudes. The method showed very good results especially for LEO cases above 1,000 km altitude, while being still comparable to the other methods below 1,000 km altitude with only a few exceptions.

For high eccentricity orbits, where it has to be pointed out that orbital lifetime is sensitive not only to perigee altitude but also to the orientation of the orbit, a similar behaviour of the three methods as compared to the LEO cases could be observed for a given example. The Regula Falsi method required less iterations for those orbits, where the initial perigee altitude was not close to the searched solution. The Brent method computed the disposal orbit in about ten iterations in the perigee altitude variation scenario, while the bisection method was between ten and fifteen iterations. When searching for an optimized solution, however, one would have to perform also variations of the other orbital elements - while for the LEO case it was fine to look at the perigee altitude, this does not have to be true for HEO.

For the application of the root finding algorithm in OSCAR, the Regula Falsi method proved to be the best option, which was basically due to the fact that the difference in processor run time does not only depend on the difference in the number of iterations. It is strongly affected by the altitude at which the evaluations take place. Therefore, the performance of each algorithm has to be evaluated by integrating the number of propagated years, which was done in a few examples and showed that in general the Regula Falsi method is quite suitable for the root finding problem under consideration. The Regula Falsi method is also very easy to implement and thus allows for easier maintenance if the root finding method in OSCAR should be adapted in some way, e.g. by changing the two initial guesses.

The root finding algorithm is one of the key elements in OSCAR when a disposal orbit has to be searched for a given residual lifetime. In order to evaluate the user-defined strategy, but also to provide the system requirements to be compliant with the SDMG, OSCAR applies the root finding algorithm two times per run. As it is likely, e.g. for high eccentricity orbits, that OSCAR is used for sensitivity studies and is called many times, an efficient root finding algorithm is necessary and results in minutes to hours of saved computation time.

References

[1] Braun, V., Gelhaus, J., Flegel, S., Möckel, M., Kebschull, C., Wiedemann, C., Krag, H., Vörsmann, P., Upgrade of the ESA DRAMA OSCAR tool: Analysis of disposal strategies considering current standards for future solar and geomagnetic activity, *Sixth European Conference on Space Debris*, Paper 6c.O-3, Darmstadt, 2013

[2] Brent, R. P., Algorithms for Minimization without Derivatives, *Dover Publications*, 1973

[3] European Cooperation for Space Standardization (ECSS), Space engineering (Space environment), ECSS-E-ST-10-04C, 2008

[4] Fraysse H., Morand V., Le Fevre C., et al., Long term orbit propagation techniques developed in the frame of the French Space Act, 22nd International Symposium on Space Flight Dynamics, 2011

[5] International Organization for Standardization, Space systems - Estimation of orbit lifetime, ISO 27852:2011, 2011

[6] Press W. H., Teukolsky S. A., Vetterling, W. T., Flannery, B. P., Numerical Recipes in Fortran 90 - The Art of Parallel Scientific Computing, Volume 2 of Fortran Numerical Recipes, Second Edition, Cambridge University Press, 1996

[7] UN Office for Outer Space Affairs, Space Debris Mitigation Guidelines of the Committee on the Peaceful Uses of Outer Space, Vienna, 2010.

[8] Zhang, Z., An Improvement to the Brent's Method, *International Journal of Experimental Algorithms* 2 (1):21-26, 2011