

# Computerphysik I: Swing-by Manöver

Aurel Müller-Schoenau, Leon Oleschko und Moritz Schröer

30. September 2022

## 1 Einleitung

Beim Swing-By geht es darum, einen Satelliten gezielt an einem Himmelsobjekt vorbeifliegen zu lassen, damit die wirkende Gravitation den Satelliten auf einen gewünschten Orbit bringt. Dadurch lässt sich der sonst zur Beschleunigung benötigte Treibstoff sparen.

In diesem Bericht geht es um eine historische Raumfahrtmission der ESA, es handelt sich um die Ulysses-Raumfahrtmission. Bei dieser wurde nur ein Swing-by, nämlich am Planeten Jupiter, durchgeführt und die Daten zur Flugbahn sind verfügbar<sup>1</sup>. Ziel ist es, mithilfe einer Simulation anhand der Shooting-Methode herauszufinden, wie der Satellit von der Erde aus ins All geschickt werden muss, um einen Swing-By auf den entsprechenden Orbit zu erreichen, und dann den errechneten Orbit mit den historischen Daten zu vergleichen.

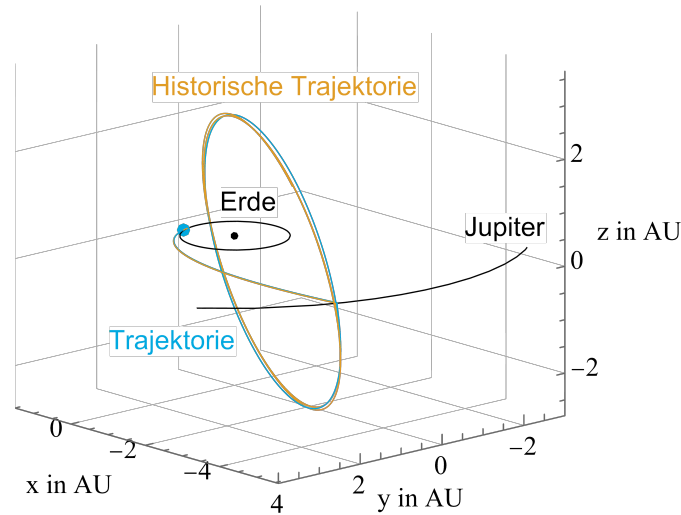
## 2 Trajektorie

Um die Flugbahn zu simulieren, muss die wirkende Gravitationskraft berechnet werden. Zu Vereinfachung werden nur die Sonne, die Erde und der Jupiter berücksichtigt, weil der Satellit an keinem anderen Planeten nahe vorbeiflog. Die Planeten-Daten wurden aus Mathematica<sup>2</sup> in gebräuchlicher Auflösung exportiert, um die Datenmenge zu reduzieren. Da die Koordinaten nur tageweise vorliegen, werden sie in der Simulation linear über

---

<sup>1</sup><https://www.cosmos.esa.int/web/ulysses/mission-trajectory>

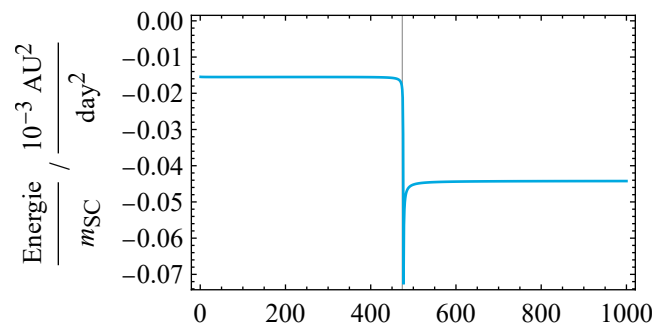
<sup>2</sup>Wolfram Research, Inc., Wolfram|Alpha Notebook Edition, Champaign, IL (2022).



**Abbildung 1:** Abbildung der historischen Trajektorie, der simulierten Flugbahn und der relevanten Himmelskörpern

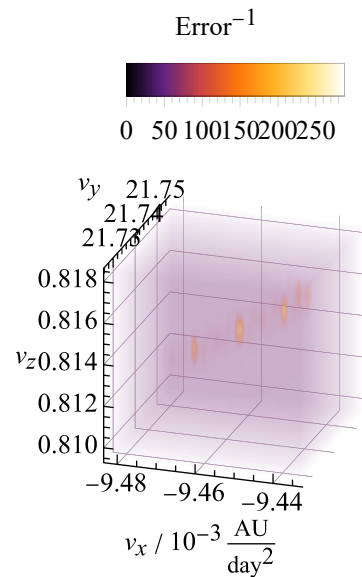
den Tag interpoliert. In Abbildung 1 sind die Positionen der Planeten, die Historische Trajektorie und ein simulierte Flugbahn dargestellt.

Für die Simulation wurde eine Dauer von 2800 Tagen betrachtet, wobei der Satellit historisch ungefähr 474 Tage für den Transfer zum Jupiter benötigte, und in der restlichen Zeit etwa einen vollständigen Orbit nach dem Swing-By durchlief. Jeder dieser Tage wurde in 1000 Zeitschritte aufgeteilt, da bei mehr Schritten sich die Ergebnisse nicht mehr verändert haben. Für jeden dieser Schritte wurde ein Leap-Frog-Schritt durchgeführt, um die Position und Geschwindigkeit des Satelliten zu berechnen. Dabei wurden die Positionen der Himmelskörper als vom Satelliten unbeeinflusst angenommen.



**Abbildung 2:** Energieänderung während des Swing-by Manövers

### 3 Fehlerfunktionen



Damit ein gewünschter Orbit erreicht werden kann, muss zunächst definiert werden, wie dieser aussehen soll. Der Startpunkt des Satelliten liegt natürlich auf der Erde. Den genauen Endzustand kennen wir im Fall der historischen Raumfahrtmission, ansonsten bleibt nur übrig, den gewünschten Zielorbit auf andere Weise zu definieren. In beiden Fällen handelt es sich um ein Randwertproblem, welches sich mithilfe der Shooting-Methode lösen lässt.

Dazu definieren wir eine Fehlerfunktion, die genau dann zu null wird, wenn der Satellit sich am Ende der Simulationszeit auf dem gewünschten Kurs befindet. Um den Satelliten in unserer Simulation auf den finalen Orbit des Ulysses-Satelliten zu schicken, haben wir verschiedene Fehlerfunktionen ausprobiert.

#### 3.1 Vergleich der Endgeschwindigkeit und des Ortes

Um einen Orbit zu erhalten, der dem historischen ähnelt, wurde zunächst *zu einem festen Zeitpunkt* nach dem Swing-By der *Ort und die Geschwindigkeit* des Satelliten mit den historischen Daten verglichen. Da diese keine Geschwindigkeitsinformationen enthalten, wurde dafür die mittlere Geschwindigkeit des Tages angenommen.

Der Ort allein als Vergleichswert reichte nicht aus, da das Programm sonst eine Abkürzungslösung fand, bei der der Satellit an der (im Vergleich zu den historischen Daten) entgegengesetzten Seite des Jupiter vorbeiflog, dadurch jedoch beim Swing-By in die

falsche Richtung, und somit nicht auf den gewünschten Orbit abgelenkt wurde.

### 3.2 Mehre Endpunkte zu festen Zeiten

Eine andere Methode ist der Vergleich der *Ortskoordinaten* nach dem Swing-By *zu mehreren Zeitpunkten*, dafür aber *ohne Vergleich der Geschwindigkeiten*. Der Zielorbit wird nun also durch eine Reihe an zu erreichenden Wegpunkten definiert, die zu bestimmter Zeit zu erreichen sind.

### 3.3 Integral des Ort-Fehler-Quadrates

### 3.4 Vergleich von Zielpunkten bei nächster Zeit

Um die Tatsache, dass in der Realität bei der Planung eines Swing-By-Manövers wahrscheinlich keine passenden Daten vorlägen, nicht völlig außer Acht zu lassen, wurden bei der dritten Fehlerfunktion einige Punkte auf dem Wunsch-Orbit ausgewählt - in unserem Fall genau solche, die in den historischen Daten vorlagen, aber es hätten beliebige Punkte sein können. Und dann wurde beim Berechnen der Trajektorie bei jedem Zeitschritt der aktuelle Abstand zu jedem der ausgewählten Punkte gemessen. Der Wert wurde mit einem gespeicherten Wert verglichen, und dieser überschrieben, falls der aktuelle Wert kleiner war. So wurde zu jedem der ausgewählten Punkte der nächstgelegene Punkt der aktuellen Trajektorie gefunden.

Bei dieser Methode muss also der Zeitpunkt des Manövers nicht im Voraus bekannt sein, jedoch besteht die Möglichkeit, dass ein korrekter Orbit nicht erkannt wird, wenn die ausgewählten Punkte des gewünschten Orbits gerade zwischen denen des berechneten Orbits liegen. Solange die Zeitschritte klein genug sind, wird dann aber der Fehler immernoch sehr klein. Es wurde deshalb *bei jedem Sub-Step*, also pro tausendstel Tag, der Abstand verglichen.

## 4 Optimierung

Die Shooting-Methode besteht nun darin, die Fehlerfunktion für einen gegebenen Orbit auszuwerten, und ausgehend davon den Orbit in irgendeiner Weise anzupassen, um schließlich die Nullstelle der Fehlerfunktion zu finden. Als Optimierungsparameter wurde die Anfangsgeschwindigkeit gewählt, da diese nur 3 Dimensionen hat und somit leichter zu optimieren ist, als z.B. der Transferorbit. Als initialer Wert wurde die echte gemessene Geschwindigkeit verwendet. Wichtig war vor allem, eine Anfangsgeschwindigkeit zu wählen, für welche der Satellit möglichst nahe am Jupiter vorbeifliegt, um die Laufzeit

des Suchalgorithmus zu verringern.

Die Daten liegen aber in Form von Koordinaten in einem sphärischen heliozentrischen Koordinatensystem vor und haben eine Auflösung von  $0.01^\circ$ . Daraus lässt sich eine Unsicherheit der Position von  $4 \text{ AU} \cdot \sin 0.01^\circ \approx 7 \cdot 10^{-4} \text{ AU}$  und daraus eine Unsicherheit der Geschwindigkeit von  $\sqrt{2} \cdot 7 \cdot 10^{-4} \approx 10^{-3} \text{ AU/day}$  abschätzen. Solche Unsicherheit ist viel zu groß für die nötige Genauigkeit von  $10^{-6}$  bis  $10^{-8} \text{ AU/day}$ , denn so viele Stellen muss der Suchalgorithmus optimieren, um die Nullstelle der Fehlerfunktion möglichst genau zu finden. Eine sehr kleine Änderung der Startgeschwindigkeit kann bereits zu einem völlig anderen Orbit führen. Dies dämpft die Erwartungen an die Ergebnisse. Zur Nullstellensuche wurden verschiedene Verfahren ausprobiert.

#### 4.1 Newton-Verfahren

Die Nullstelle einer stetig differenzierbaren Funktion lässt sich mithilfe des Newton-Verfahrens finden. Dabei wird die Funktion lokal linear durch ihre Ableitung angenähert. Ist die Determinante der Ableitung ungleich Null, so lässt sich das entsprechende Gleichungssystem für jedes gewünschte Ergebnis eindeutig lösen. Die Nullstelle der Taylor-Entwicklung bis zur ersten Ordnung an einem beliebigen Startpunkt wird auf diese Weise bestimmt, und die Lösung wird als neuer Startwert für die nächste Iteration desselben Vorgehens verwendet.

Das Newton-Verfahren konvergiert unter bestimmten Bedingungen (die unsere Fehlerfunktionen erfüllen) lokal quadratisch gegen eine Nullstelle der untersuchten Funktion (ohne Beweis).

#### 4.2 Newton-Hesse-Verfahren

#### 4.3 Gradienten-Verfahren

### 5 Fazit