

11. Mai 2023

## Übungsprojekt 02: Satellitensichtbarkeiten

Hinweis: Dieses Übungsprogramm ist in **Einzelarbeit** und nicht in den bisherigen Gruppen durchzuführen!

Bei der Satellitenmission GRACE wird die Position der beiden Satelliten zu jedem Zeitpunkt über einen GPS-Empfänger bestimmt, der sich an Bord der Satelliten befindet. Für die Kommandozentrale der Satellitenmission GRACE sollen auf einem Monitor die Sichtbarkeiten von einem der GRACE-Satelliten zu den GPS-Satelliten dargestellt werden.

### Funktionalität

Das Programm muss zumindest folgende Funktionalität bieten:

```
$ python pro02SatelliteVisibility.py --help

usage: pro02SatelliteVisibility.py [-h] [-n] [-o OUTFILE] [-t TIME TIME] date

Generates an animated visualization of GRACE satellite visibility to the GPS
constellation for a single day.

positional arguments:
  date                  date to visualize (format: YYYY-MM-DD)

optional arguments:
  -h, --help            show this help message and exit
  -n, --novisibility    do not show visibility lines
  -o OUTFILE, --outfile OUTFILE  save animation to this file
  -t TIME TIME, --time TIME TIME  start and end time in hours to limit animation
```

Beispielaufruf:

```
$ python pro02SatelliteVisibility.py 2008-07-13 -t 14 15.5 -n -o example.mp4
```

Obiger Aufruf erzeugt eine Animation für den 13. Juli 2008 von 14:00 bis 15:30 Uhr, bei der die Sichtbarkeitslinien nicht angezeigt werden. Die Animation wird angezeigt und zusätzlich in die Datei `example.mp4` als Video exportiert.

## Daten herunterladen

Die Orbits sind pro Satellit in **täglichen Dateien** für das Jahr 2008 auf dem FTP-Server `ftp.tugraz.at` gespeichert. Sie befinden sich im Ordner

`/outgoing/ITSG/teaching/2023SS_Informatik2/orbit/YYYY-MM-DD/`, wobei `YYYY-MM-DD` für das jeweilige Datum steht.

Die Dateinamen haben das Format `orbit_YYYY-MM-DD.SATNAME.txt.gz`, wobei `YYYY-MM-DD` für das Datum (z.B. 2008-06-13) und `SATNAME` für den Satellitennamen (`graceA` für GRACE bzw. `G01` bis `G32` für GPS) steht. Beachten Sie, dass nicht an jedem Tag alle 32 GPS-Satelliten vorhanden sind.

Als **Hintergrundbild** für die animierte Karte soll die zum angezeigten Datum passende NASA Blue Marble verwendet werden. Die zwölf Bilder für die jeweiligen Monate sind ebenfalls am selben FTP-Server im Ordner `/outgoing/ITSG/teaching/2023SS_Informatik2/bluemarble/` gespeichert. Die Dateinamen haben das Format `bluemarbleMM.jpg`, wobei `MM` für den Monat (`01` bis `12`) steht. Die Bilder decken die **gesamte** Erdoberfläche ab.

Die heruntergeladenen Dateien sollen **lokal** in einer passenden Ordnerstruktur (z.B. ähnlich wie auf dem FTP-Server) unterhalb des aktuellen Arbeitsverzeichnisses gespeichert werden. Die lokale Ordnerstruktur soll **automatisch angelegt** werden. Existieren die für das Datum notwendigen Dateien bereits lokal, so sollen sie **nicht** erneut heruntergeladen werden.

## Daten einlesen

Die Orbitdateien wurden zur Verringerung des Speicherplatzbedarfs **komprimiert** und sollen auch komprimiert eingelesen werden (Hinweis: `numpy.loadtxt()` kann `.gz` Dateien direkt einlesen, ansonsten kann das Modul `gzip` verwendet werden).

Der Inhalt der Dateien hat folgendes Format: `mjd [days] x [m] y [m] z [m]`

Das Dateiformat enthält zwei **Header-Zeilen**. Der Zeitpunkt ist als MJD (Modified Julian Date, Dezimaltage seit 1858-11-17) gegeben. X, Y und Z sind bereits in einem erdfesten System gegeben. Das Umrechnen der kartesischen Koordinaten in Breite und Länge erfolgt über:

$$\phi = \arcsin \frac{z}{r} \quad \lambda = \arctan \frac{y}{x}$$

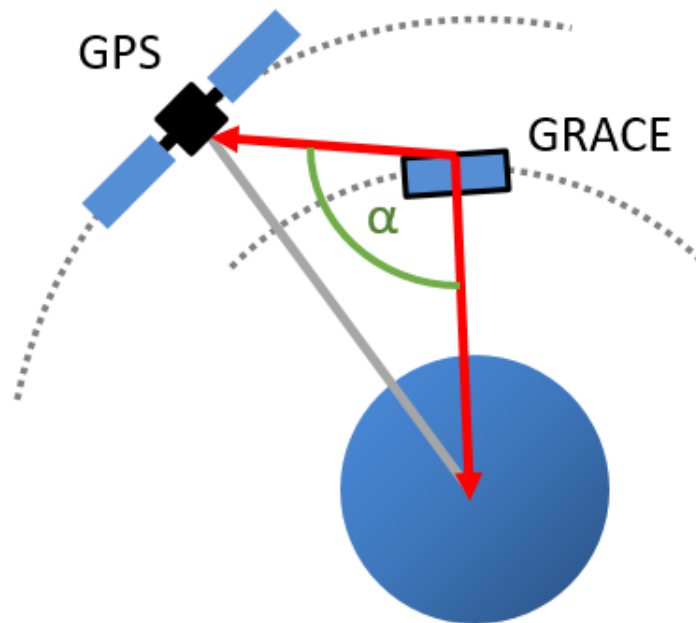
Hinweis: `arctan2` verwenden!  $r$  ist die Länge/Norm des Vektors  $(x,y,z)$ .

## Visualisierung

Die Satelliten sollen z.B. als Punkte mit einem „Schweif“ dargestellt werden, der bei der Interpretation der Bewegungsrichtung hilft. Die **Länge** des Schweifs soll von der **Umlaufgeschwindigkeit** des jeweiligen Satelliten abhängen (GRACE ist deutlich schneller unterwegs als

die GPS-Satelliten). Weiters sind die **Namen** der GPS-Satelliten neben dem jeweiligen Punkt darzustellen.

Sofern das Flag `--novisibility` nicht gesetzt ist, sollen die für den GRACE-Satelliten zu jedem Zeitpunkt sichtbaren GPS-Satelliten mit einer **Sichtbarkeitslinie** verbunden werden. Die GPS-Antenne von GRACE ist auf der Oberseite des Satelliten montiert, wodurch die Sichtbarkeit auf eine Halbkugel auf der Oberseite des Satelliten eingeschränkt ist.



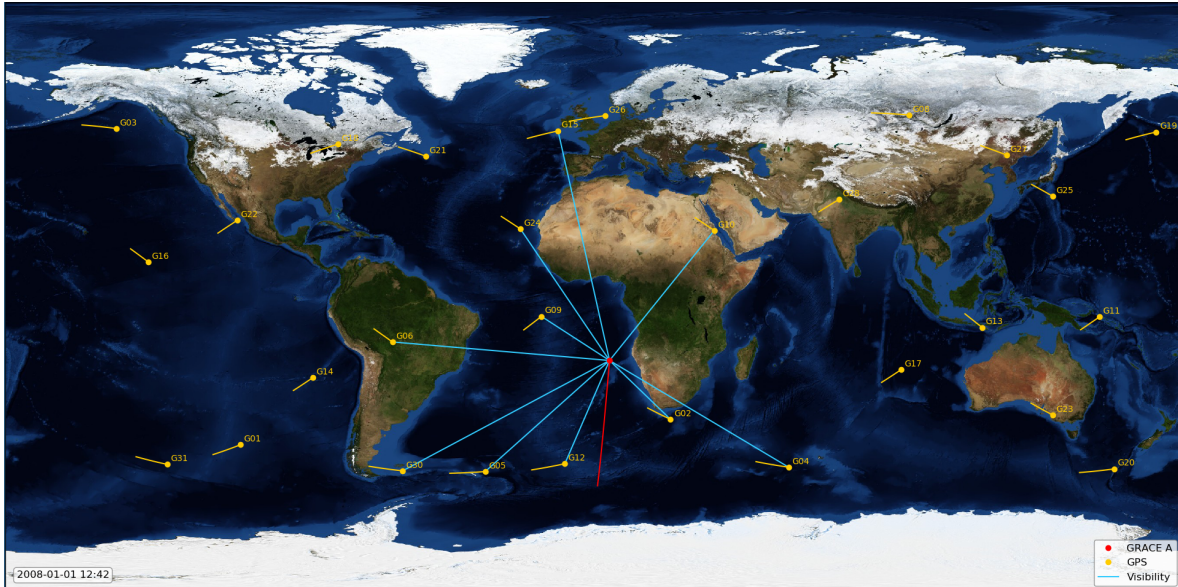
Der Winkel  $\alpha$  zwischen den beiden rot eingezeichneten Vektoren in der obigen Abbildung kann aus dem Inprodukt der **Einheitsvektoren** berechnet werden:

$$\alpha = \arccos(\vec{e}_a \cdot \vec{e}_b)$$

Ist  $\alpha$  größer als  $90^\circ$ , so ist der GPS-Satellit für GRACE sichtbar. Zur Vereinfachung können direkt die Koordinaten aus den Orbitdateien für die Berechnung der Sichtbarkeit verwendet werden.

Das **Datum und die Uhrzeit** sollen an einer passenden Stelle auf der Karte als Text angezeigt werden. Ebenso darf eine **Legende** mit den dargestellten Elementen nicht fehlen.

Die folgende Abbildung kann als Anhaltspunkt für die geforderte Visualisierung verwendet werden. Im TeachCenter ist ebenfalls ein kurzer Ausschnitt einer Beispielanimation (Datei `proj02_example.mp4`) zu finden.



## Tipps/Empfehlungen

- Überlegen Sie sich eine sinnvolle Klassenstruktur mit passenden Methoden. Eine Klasse `Satellite` mit Attributen zum Speichern aller benötigten Daten sowie Methoden zum Einlesen der jeweiligen Daten und zum Zeichnen des Satelliten in der Grafik wäre eine Option. Dadurch wird der Code kürzer, einfacher und übersichtlicher.
- Bevor Sie mit dem Animieren beginnen, versuchen Sie erstmal ein statisches Bild zu erzeugen, das die gewünschten Informationen enthält.
- Testen Sie anfangs hauptsächlich mit kurzen Zeiträumen (z.B. 1 h) und/oder wenigen Satelliten, um Wartezeiten beim Animieren zu verringern (speziell bei älteren Rechnern/Notebooks).
- Überlegen Sie, was in Ausnahmefällen (z.B. keine Daten vorhanden) passieren soll und ob/wie dies dem Benutzer/der Benutzerin mitgeteilt wird.

## Abgabe

Die Abgabe besteht nur aus dem Quellcode, es ist kein technischer Bericht zu verfassen. Der Quellcode (aber nicht die gegebenen Daten oder erzeugte Visualisierungen) muss in einem Archiv `proj02SatellitenSichtbarkeiten.zip` bis 19. Juni 2023, 23:59 Uhr im [TeachCenter](#) abgegeben werden. Die Anmeldung zu den Abgabegesprächen, die am 28. Juni stattfinden werden, erfolgt im [TeachCenter](#).

## Hinweis: Quellcode

Die Formatierung und Präsentation des Quellcodes soll dem Verständnis förderlich sein, achten Sie deshalb auf folgende Gesichtspunkte:

- Dateikopf mit Namen der Ersteller, der Gruppe und des Projekts in jeder Datei.
- Verwendung aussagekräftiger Variablennamen, z. B. `satelliteEpoch` anstatt `se`.
- Erklärende Kommentare. Wieso ist dieser Quellcode so aufgebaut wie er ist? Welche Formel wird hier implementiert? Welches Ergebnis steht am Ende des Quellcodeabschnittes?

Weiters soll der Quellcode auch inhaltlich einer erkennbaren Struktur folgen:

- Sinnvoller Einsatz von Funktionen z. B. zur Vermeidung von Wiederholungen. Klar definierte Schnittstellen auch und speziell für Klassen (Mit Beschreibung von Parametern und zu erwartenden Rückgabewerten).
- Strukturierte Abfolge der Anweisungen, z. B. zuerst Abarbeitung eines Aufgabenteils gefolgt von Anweisungen zum nächsten Aufgabenteil.
- Variablen so lokal wie möglich deklarieren, im Idealfall erst bei der ersten Verwendung.