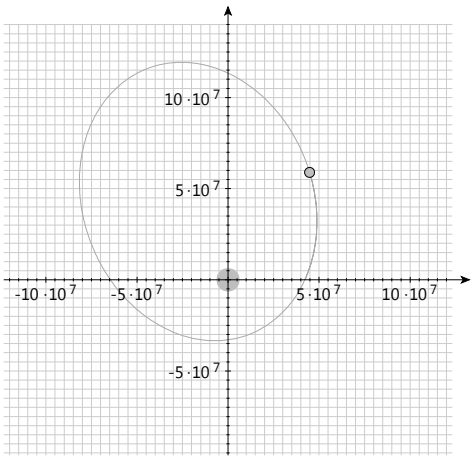
Aufgabenblatt 8

Satellit



Ein Satellit bewegt sich in Erdnähe. Durch die Gravitationskraft wird der Satellit in eine Bahn um die Erde gezwungen. Von Erde und Satellit wird vereinfachend angenommen, dass es sich um punktförmige Massen handelt. Die Erde befinde sich im Ursprung des Koordinatensystems.

Die Masse der Erde ist $m_E = 5.974 \cdot 10^{24} kg$, die Masse des Satelliten sei $m_S = 100\,kg$. Die Gravitation sorgt dafür, dass sich die Massen m_E und m_S mit der Kraft

$$F_G = G\frac{m_E m_S}{r^2}$$

gegenseitig anziehen. Da die Erde im Vergleich zum Satelliten eine sehr große Masse hat, erfährt die Erde durch diese Kraft nur eine sehr, sehr kleine Beschleunigung. Vereinfachend soll angenommen, dass sich die Erde im Ursprung des Koordinatensystems befindet und ihre Position nicht verändert.

Vorbereitung

Zur Visualisierung der Flugbahn des Satelliten soll, wie schon in Aufgabenblatt 6, die Klasse *MechanicsTVG* verwendet werden. Binden Sie, falls noch nicht geschehen, die Library *mechanics.jar* in Ihr Projekt ein. Die Datei *mechanics.jar* finden Sie auf der Webseite im Bereich dieser Aufgabe.

Übernehmen Sie den folgenden Programmcode:

```
import static java.lang.Math.*;
import mechanics.tvg.MechanicsTVG;
import de.physolator.usr.*;
import de.physolator.usr.tvg.Shape;
import de.physolator.usr.util.Colors;

public class Satellit extends PhysicalSystem {

    public void initGraphicsComponents(GraphicsComponents g, Structure s, Recorder r,
                    SimulationParameters sp) {
        MechanicsTVG t = new MechanicsTVG(this, s, r) {
            public void paint() {
                super.paint();
                style.fillColor = Colors.grey;
                drawCircle(0,0,6.370e6, Shape.POLYGON);
            }
        };
        double p = 2e8;
        t.geometry.setUserArea(-p, p, -p, p);
        t.showPaths = true;
        t.showVelocity = false;
        t.showAcceleration = false;
        t.showLabels = false;
        t.addPointMass("x", "y", "vx", "vy", "ax", "ay");
        g.addTVG(t);
    }

    public void initSimulationParameters(SimulationParameters s) {
        s.fastMotionFactor = 20000;
    }
}
```

Erläuterungen zum Programmcode

Dieser Programmcode verwendet die Grafikkomponente *MechanicsTVG*. Mit *MechanicsTVG* soll die Bahn des Satelliten auf dem Bildschirm dargestellt. Analog zu Aufgabenblatt 6 sei *(x,y)* die Position des Satelliten, seine Geschwindigkeit sei *(vx,vy)* und seine Beschleunigung *(ax,ay)*. Der Programmcode enthält noch keine physikalische Variablen und es fehlen auch noch die Formeln, die das Verhalten des Satelliten beschreiben (Methode *f*). Der Methodenaufruf *addPointMass* legt jedoch bereits fest, dass die Grafikkomponente *MechanicsTVG* in dem physikalischen Systeme auf die Variablen *x, y, vx, vy, ax* und *ay* zugreifen soll, um diese punktförmige Masse grafisch darzustellen.

Durch das Überschreiben der Methode *initSimulationParameter* können Parameter verändert werden, die festlegen, wie der Physolator die Simulation durchführt. Satelliten benötigen Stunden, Tage oder gar Wochen um die Erde zum umrunden. Die Zuweisung

```
s.fastMotionFactor = 20000;
```

bewirkt, dass die Simulation im Zeitraffer abläuft – zehntausend mal schneller als in der Realität.

1. Teilaufgabe

Die Anfangsposition des Satelliten sei $\begin{pmatrix} 4\cdot 10^7 \\ 0 \end{pmatrix} m$, seine Anfangsgeschwindigkeit $\begin{pmatrix} 0 \\ 4000 \end{pmatrix} \frac{m}{s}$.

Programmieren Sie dieses physikalische System analog zu Aufgabenblatt 1. Ergänzen Sie dazu den vorgegebenen Programmcode um Deklarationen für die physikalische Variablen, legen Sie die Anfangswerte der physikalischen Variablen fest und definieren Sie die Ableitungsbeziehungen zwischen den physikalischen Variablen. Verwenden Sie die Variablennamen *x, y, vx, vy, ax* und *ay*, wobei *(x,y)* für

die Position des Satelliten, *(vx,vy)* für dessen Geschwindigkeit und *(ax,ay)* für die Beschleunigung steht, die der Satelliten erfährt. Programmieren Sie eine Methode *f* mit den Formeln zur Bestimmung der abhängigen Variablenwerte.

2. Teilaufgabe

Bestimmen Sie, wie lange der Satellit für eine Umrundung benötigt.

Verwenden dazu einen Schwellwerttrigger (Klasse *ThresholdTrigger*). Beachten Sie dazu die Hinweise in Teilaufgabe 3 von Aufgabenblatt 6.

3. Teilaufgabe

Starten Sie den Satellit von unterschiedlichen Positionen aus und mit unterschiedlichen Anfangsgeschwindigkeiten. Welche Formen haben die Flugbahnen des Satelliten? Wie lässt sich die Form variieren?

4. Teilaufgabe

Leiten Sie die zunächst die Anfangsposition und die Anfangsgeschwindigkeit für einen geostationären Satelliten mathematisch her. Verwenden Sie dann diese Anfangsposition und diese Anfangsgeschwindigkeit für eine physikalische Simulation.

Erläuterungen

Geostationäre Satelliten fliegen mit der einer Geschwindigkeit, die gerade so groß ist, dass sie die Erde an einem Sterntag umrunden. Sie umrunden die Erde mit der gleichen Winkelgeschwindigkeit, mit der die Erde rotiert. Aus diesem Grund hat ein Beobachter, der auf der Erdoberfläche am Äquator steht den Eindruck, dass der Satellit unbeweglich über ihm steht.

Unter einem Sterntag versteht man die Zeit, die die Erde benötigt, um sich einmal um die eigene Achse zu drehen. Bezugspunkt ist der Fixsternhimmel, das Weltall. Ein Sterntag dauert 23 Stunden, 56 Minuten, 4,091 Sekunden, also insgesamt 86164,091s. Der Sterntag ist etwas kürzer als ein „gewöhnlicher“ Tag mit 24 Stunden. Der gewöhnliche Tag mit 24 Stunden beschreibt eine Erdrotation bezüglich der Sonne.

Hinweise

Die Winkelgeschwindigkeit, mit der der geostationäre Satellit fliegt, kann man direkt aus der Dauer des Sterntags gewinnen. Um zum richtigen Abstand zwischen Satellit und Erde zu kommen setzt man die Gravitationskraft der Zentrifugalkraft gleich. Mit der Winkelgeschwindigkeit und dem Abstand kommt man zur Bahngeschwindigkeit, die auch gleichzeitig die Anfangsgeschwindigkeit ist.