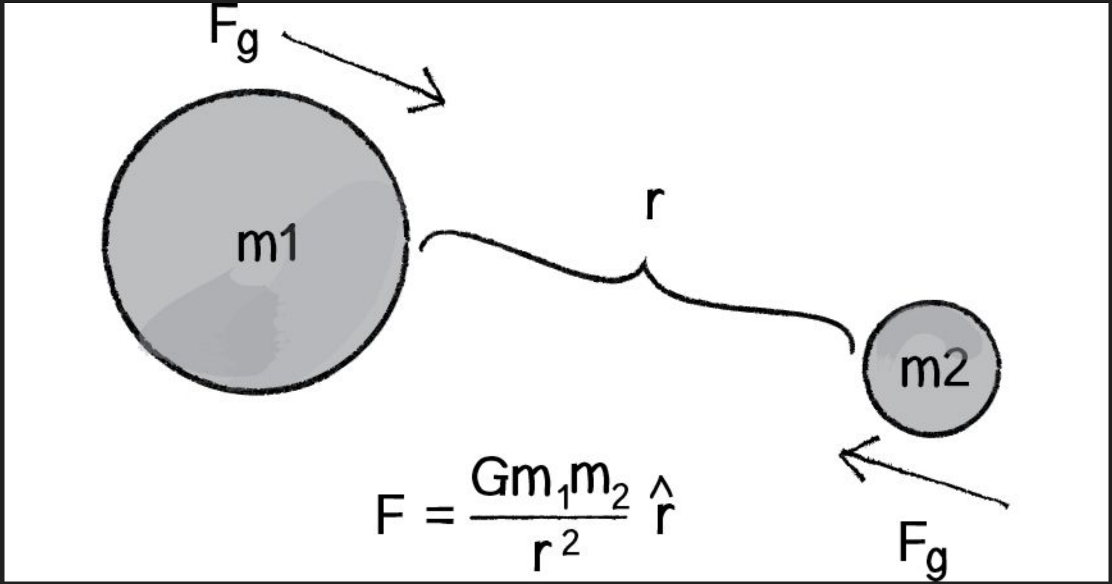
Sun, Earth, Moon System Model

Quentin Adolphe, Christian Bignotti, Brandon Mickelson

Problem definition and Connection to E19



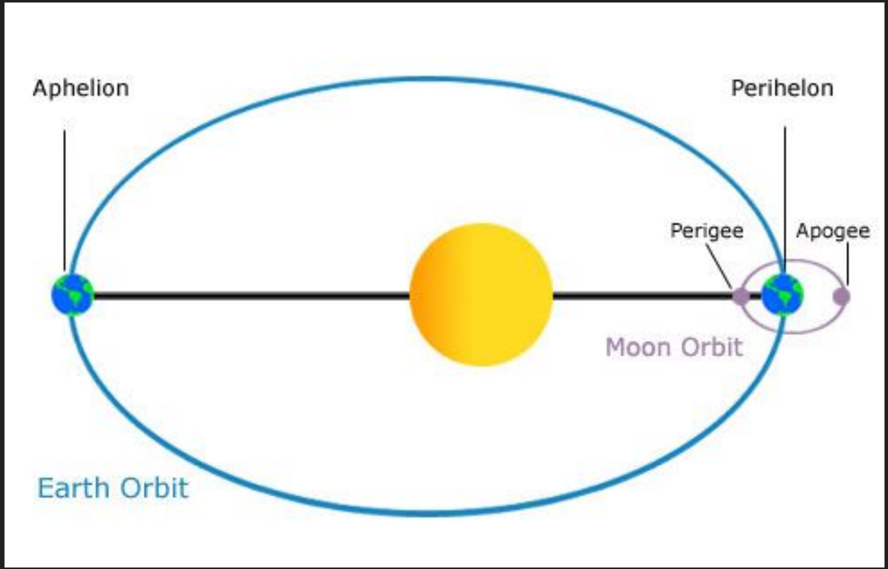
$$F = ma, \quad F_g = G\frac{m\,M}{r^2}$$

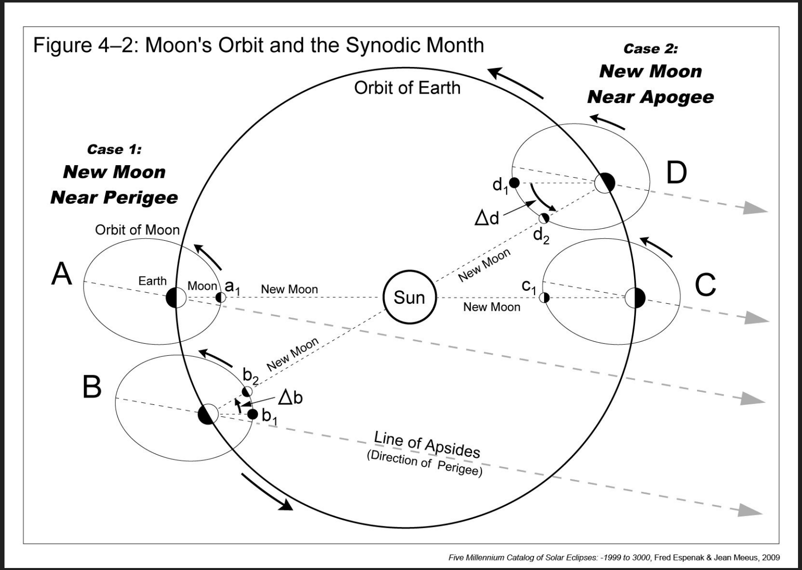
$$ma = G\frac{m\,M}{r^2}$$

$$\frac{d^2x}{dt^2} = G\frac{M}{r^2}$$

ODE!!!!! :)

Initial Conditions: Elliptical Orbits





https://www.nasa.gov/sites/default/files/orbit-3.jpg

https://eclipse.gsfc.nasa.gov/SEhelp/image/Fig4-2a.png

Initial Conditions: Apogee and Perigee

```
               Perigee                                   Apogee
---------------------------------     ---------------------------------
Jan  1 23:01 358036 km     N-   19h   Jan 14  9:29 405804 km     F-3d14h
Jan 30  7:10 362249 km     N-1d22h   Feb 11  2:40 404896 km     F-5d14h
Feb 26 22:19 367785 km     N-3d19h   Mar 10 23:06 404267 km     F-7d 8h
Mar 23 23:29 369762 km     F+5d16h   Apr  7 19:12 404437 km     N+6d12h
Apr 19 15:17 365142 km     F+2d20h   May  5 12:47 405286 km     N+4d16h
May 17 15:24 360297 km     F+1d11h   Jun  2  1:15 406190 km -   N+2d13h
Jun 14 23:23 357433 km +   F+   11h   Jun 29  6:10 406580 km --  N+   3h
Jul 13  9:09 357263 km ++  F-    9h   Jul 26 10:23 406274 km -   N-2d 7h
Aug 10 17:16 359829 km     F-1d 8h   Aug 22 21:54 405418 km     N-4d10h
Sep  7 18:18 364490 km     F-2d15h   Sep 19 14:46 404555 km     N-6d 7h
Oct  4 17:02 369334 km     F-5d 3h   Oct 17 10:22 404329 km     F+7d13h
Oct 29 14:49 368287 km     N+4d 4h   Nov 14  6:42 404923 km     F+5d19h
Nov 26  1:32 362825 km     N+2d 2h   Dec 12  0:31 405868 km     F+3d20h
Dec 24  8:34 358269 km     N+   22h
```

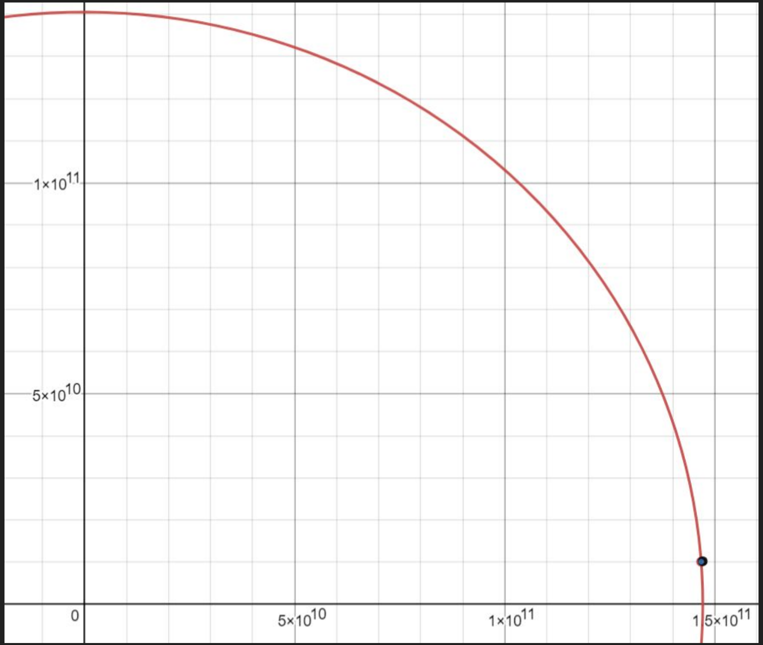
https://www.fourmilab.ch/earthview/pacalc.html

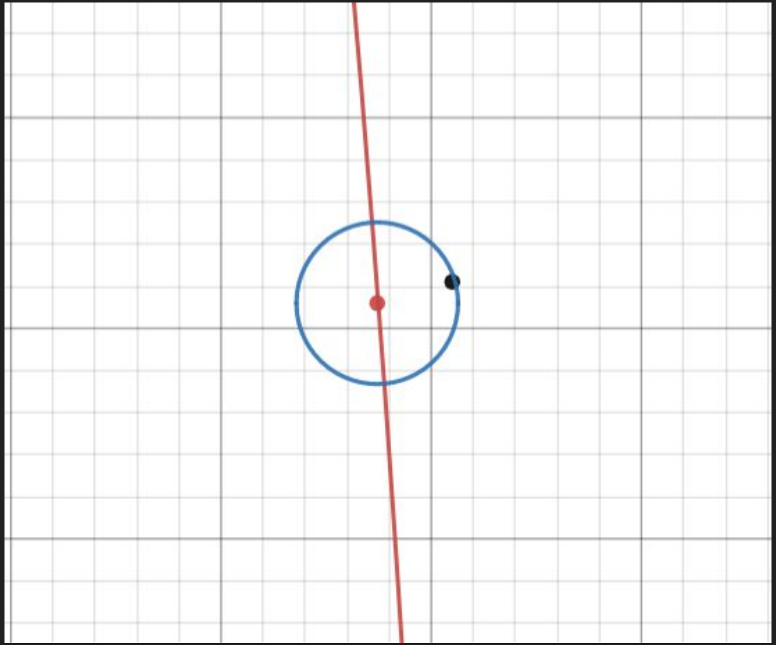
Initial Conditions: Aphelion and Perihelion

Year	Date/Time of Perihelion	Distance	Relative to Mean Per.	Date/Time of Aphelion	Distance	Relative to Mean Ap.	Perihelion Interval
2001	Jan 04 08:52	0.9832860 AU	-582 km	Jul 04 13:37	1.0166426 AU	-10122 km	367.15 days
2002	Jan 02 14:09	0.9832898 AU	-20 km	Jul 06 03:47	1.0166882 AU	-3304 km	363.22 days
2003	Jan 04 05:02	0.9833204 AU	4556 km	Jul 04 05:40	1.0167282 AU	2683 km	366.62 days
2004	Jan 04 17:42	0.9832648 AU	-3752 km	Jul 05 10:54	1.0166937 AU	-2485 km	365.53 days
2005	Jan 02 00:35	0.9832968 AU	1032 km	Jul 05 04:58	1.0167416 AU	4683 km	363.29 days
2006	Jan 04 15:30	0.9833270 AU	5543 km	Jul 03 23:10	1.0166973 AU	-1949 km	367.62 days
2007	Jan 03 19:43	0.9832602 AU	-4449 km	Jul 06 23:53	1.0167059 AU	-649 km	364.18 days
2008	Jan 02 23:51	0.9832801 AU	-1476 km	Jul 04 07:41	1.0167535 AU	6465 km	364.17 days
2009	Jan 04 15:30	0.9832730 AU	-2526 km	Jul 04 01:40	1.0166664 AU	-6563 km	367.65 days
2010	Jan 03 00:09	0.9832897 AU	-38 km	Jul 06 11:30	1.0167020 AU	-1246 km	363.36 days

http://www.astropixels.com/ephemeris/perap2001.html

Initial Conditions: Initial Position Jan 4 2022





Initial Conditions: Initial Velocity

$$v^2 = \mu \left(\frac{2}{r} - \frac{1}{a} \right)$$

Instantaneous velocity is tangent to orbit. Solve and convert to x-y coordinates.

Initial Conditions

x_E		1.467425829E11
$\dot{x}_E$		-2083.5317978
y_E		1.0120227668E10
$\dot{y}_E$		30211.0631877
x_M	=	1.4709119758E11
$\dot{x}_M$		-2158.59593173
y_M		1.0221534081E10
$\dot{y}_M$		31291.26048

Finding our orbit

$$q = \begin{bmatrix} X_E \\ Y_E \\ X_M \\ Y_M \\ X_S \\ Y_S \\ Vx_E \\ Vy_E \\ Vx_M \\ Vy_M \\ Vx_S \\ Vy_S \end{bmatrix}, Vx = \dot{x} \qquad \dot{q} = \begin{bmatrix} Vx_E \\ Vy_E \\ Vx_M \\ Vy_M \\ Vx_S \\ Vy_S \\ Ax_E \\ Ay_E \\ Ax_M \\ Ay_M \\ Ax_S \\ Ay_S \end{bmatrix}, Ax = \ddot{x}$$

$$q = \begin{bmatrix} X_E \\ Y_E \\ X_M \\ Y_M \\ X_S \\ Y_S \\ Vx_E \\ Vy_E \\ Vx_M \\ Vy_M \\ Vx_S \\ Vy_S \end{bmatrix} \qquad \dot{q} = \begin{bmatrix} Vx_E \\ Vy_E \\ Vx_M \\ Vy_M \\ Vx_S \\ Vy_S \\ Ax_E \\ Ay_E \\ Ax_M \\ Ay_M \\ Ax_S \\ Ay_S \end{bmatrix}$$

Computing our Acceleration

$$F = ma, \quad F_g = G\frac{m\,M}{r^2}$$

$$ma = G\frac{m\,M}{r^2}$$

$$a = G\frac{M}{r^2}$$

$$Our\ acceleration\ must\ have\ direction\ (unit\ vector)$$

$$\vec{d}\ (distance) = \begin{bmatrix} x \\ y \end{bmatrix}, r = \|\vec{d}\|$$

$$direction\ of\ a = \frac{\vec{d}}{\|\vec{d}\|}$$

$$\vec{a} = \frac{\vec{d}}{\|\vec{d}\|} \cdot G\frac{M}{\|\vec{d}\|^2} = \vec{d} \cdot G\frac{M}{\|\vec{d}\|^3}$$

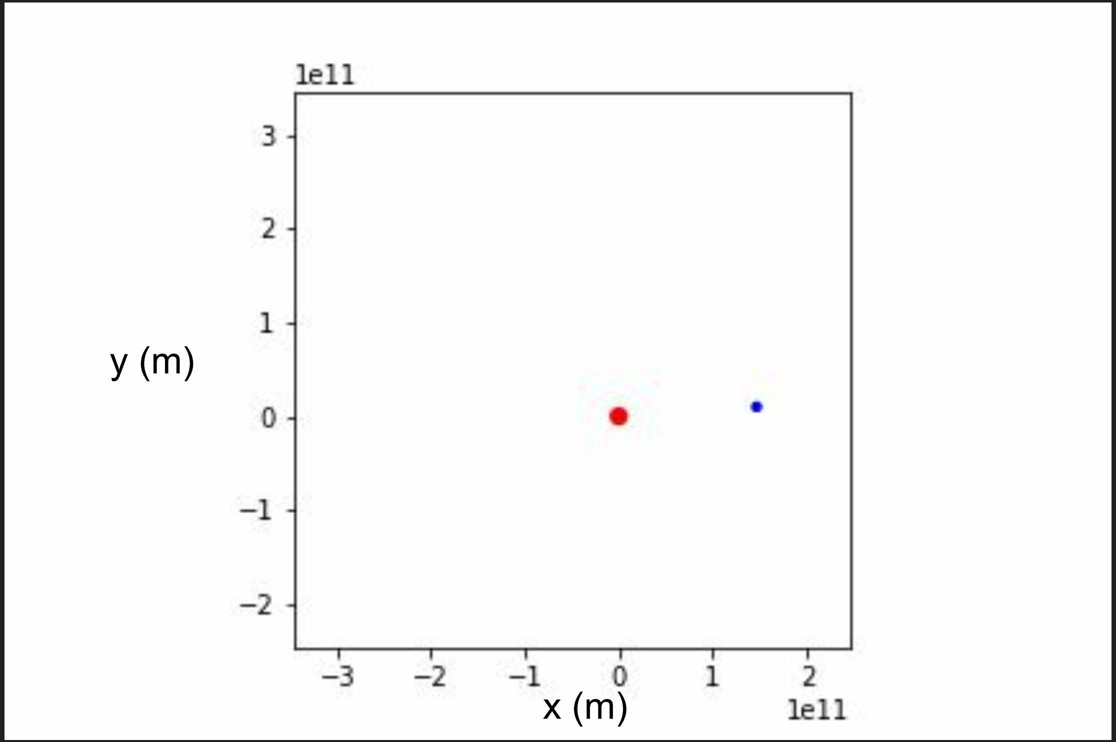
Using ODE solver

```
def F(q0):
    q=q0.copy()
    #new distances
    xe=q[0]
    ye=q[1]

    xm=q[2]
    ym=q[3]

    xs=q[4]
    ys=q[5]
    #distances
    des=np.array([xe-xs, ye-ys])
    dsm=np.array([xm-xs, ym-ys])
    dem=np.array([xm-xe, ym-ye])
    #acceleration
    Ae=-A(ms, des)+A(mm, dem)
    As=A(me, des)+A(mm, dsm)
    Am=-A(me, dem)-A(ms, dsm)
    #q dot
    q_2=np.array([q[6], q[7], q[8], q[9], q[10], q[11], \
                  Ae[0], Ae[1], Am[0], Am[1], As[0], As[1]])
```

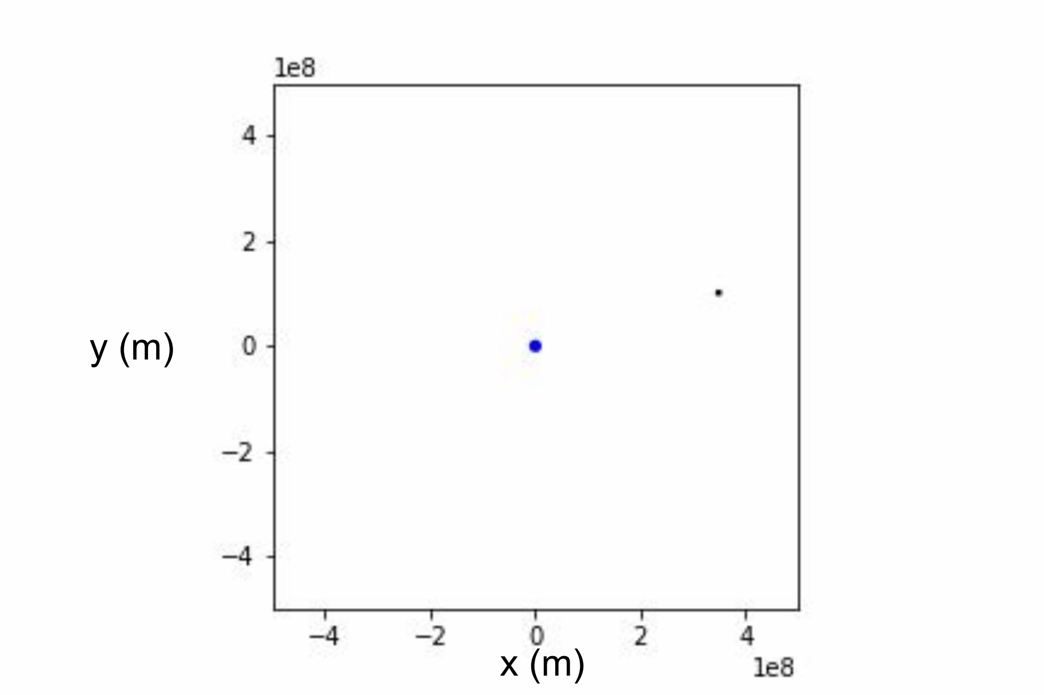
```
def integral(F, t, q, tmax, tstep):

    def runkut(F, t, q, tstep):
        k0 = tstep * F(q)
        k1 = tstep * F(q + k0/2.0)
        k2 = tstep * F(q + k1/2.0)
        k3 = tstep * F(q + k2)
        return (k0 + 2.0 * k1 + 2.0 * k2 + k3)/6.0

    T = []
    Q = []
    T.append(t)
    Q.append(q)
    while t < tmax:
        tstep = min(tstep, tmax - t)
        q = q + runkut(F, t, q, tstep)
        t = t + tstep
        T.append(t)
        Q.append(q)
    return np.array(T), np.array(Q)
```

```
def A(m_op, d):
    d1=d.copy()
    dmag=np.linalg.norm(d)

    return d1*((m_op*G)/(dmag**3))
```

Sun, Earth and Moon system…

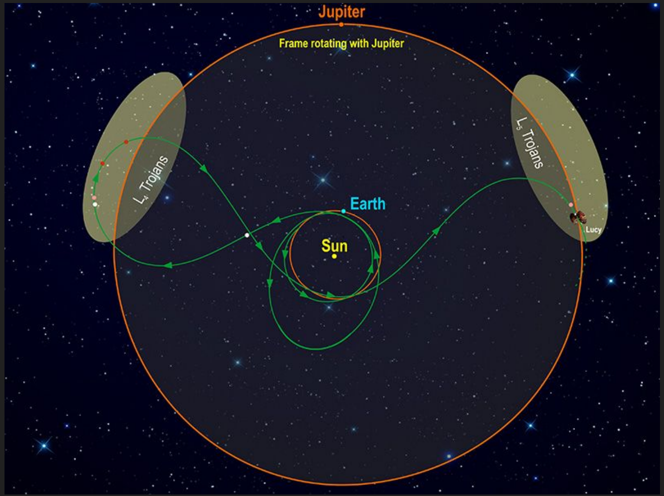


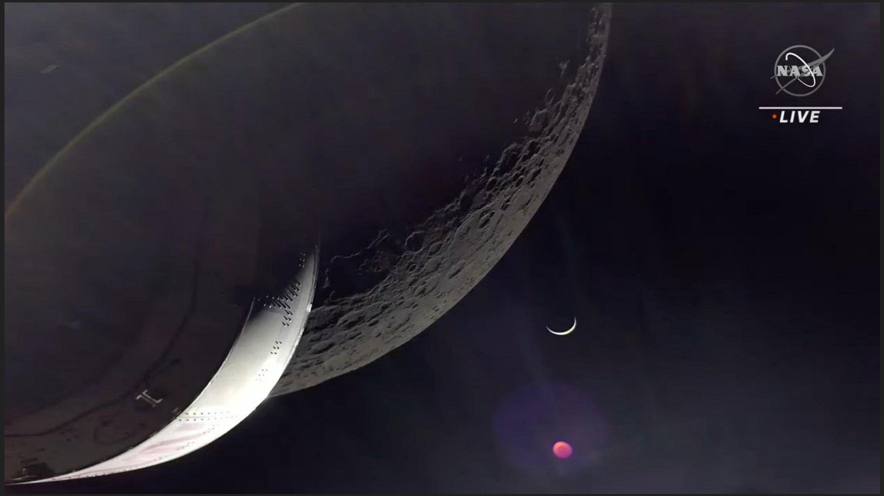
Zoom in on Earth and Moon:



Future Work

- Can scale up to multiple satellites
  - Predicting orbits of potentially dangerous objects
- Predicting eclipses





Acknowledgement

- Prof. Zucker
- https://www.khanacademy.org/computing/computer-programming/programming-natural-simulations/programming-forces/a/gravitational-attraction
- https://www.nasa.gov/sites/default/files/orbit-3.jpg
- https://eclipse.gsfc.nasa.gov/SEhelp/image/Fig4-2a.png
- https://solarsystem.nasa.gov/internal_resources/5121


thank you