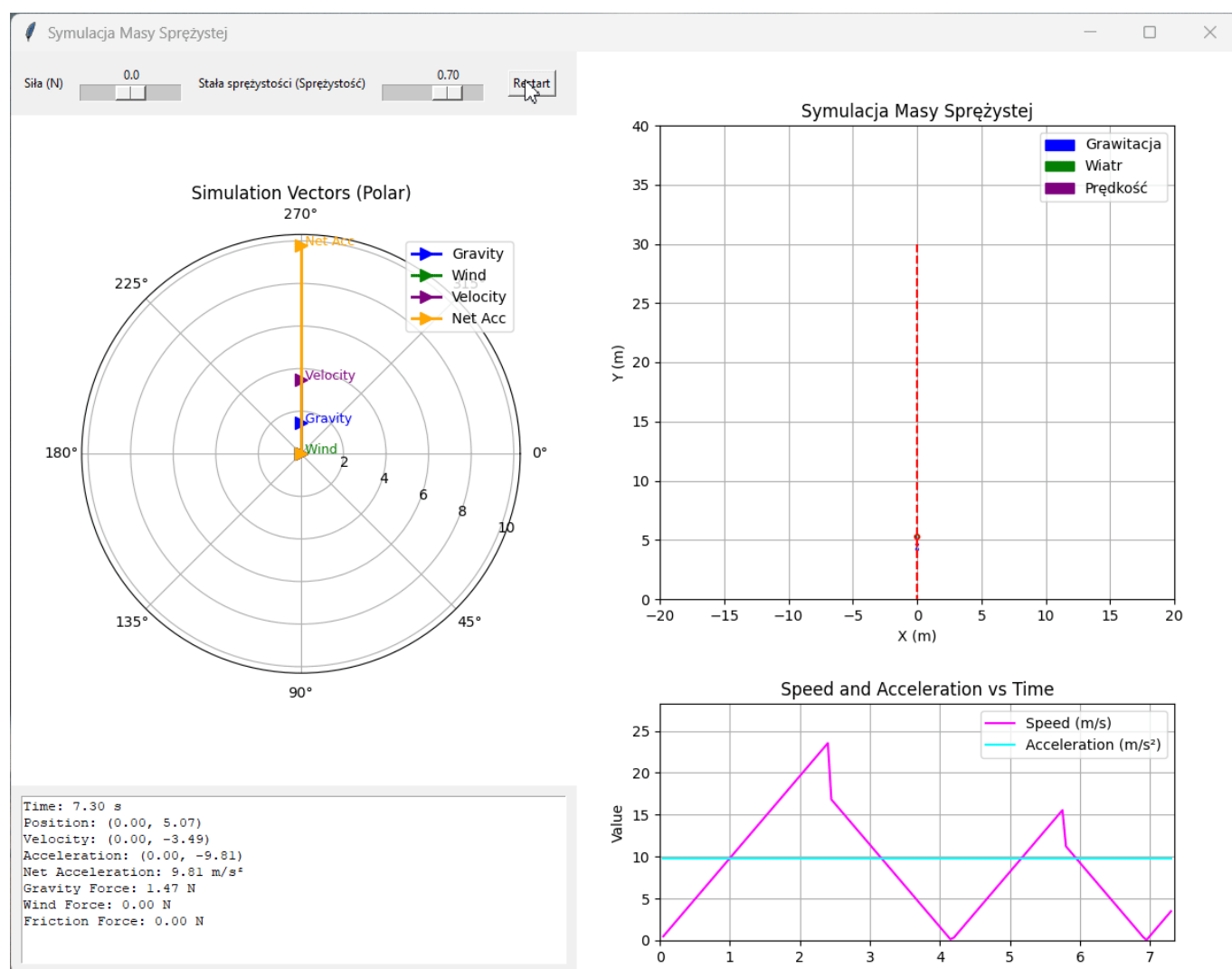


Skrypt do pobrania:

apple_sim.py



Symulacja Masy Sprężystej

Ten artykuł wyjaśnia, jak działa skrypt symulujący ruch masy sprężystej pod wpływem grawitacji, wiatru oraz siły tarcia, z użyciem bibliotek Tkinter, Matplotlib i NumPy.

⚙️ Stałe fizyczne i warunki początkowe

```
g = 9.81          # Przyspieszenie ziemskie (m/s^2)
m = 0.15          # Masa piłki (kg)
restitution = 0.7  # Współczynnik sprężystości (odbicia)
friction_coefficient = 0.5 # Współczynnik tarcia
```

Używamy podstawowych stałych z fizyki:

- Siła grawitacji: ($F_g = m g$)
- Odbicie piłki od ziemi następuje z redukcją prędkości pionowej przez współczynnik sprężystości:

$$v_{y, \text{po}} = -v_{y, \text{przed}} \cdot e$$

▢ Obliczenia sił i przyspieszeń

Kod:

```
Fw = wind_force
ax_acc = Fw / m
ay_acc = -g
```

Zgodnie z II zasadą dynamiki Newtona:

- Siła wiatru generuje przyspieszenie poziome: $a_x = \frac{F_{\text{wiatr}}}{m}$
- Przyspieszenie pionowe jest równe przyspieszeniu ziemskiemu: $a_y = -g$

▢ Kinematyka ruchu

```
vx += ax_acc * time_step
vy += ay_acc * time_step
x += vx * time_step + 0.5 * ax_acc * time_step**2
y += vy * time_step + 0.5 * ay_acc * time_step**2
```

Używamy podstawowych równań kinematyki:

$$v = v_0 + a \cdot \Delta t \quad s = s_0 + v \cdot \Delta t + \frac{1}{2} a \cdot \Delta t^2$$

▢ Odbicie od ziemi i tarcie

```
if y <= y_min:
    vy = -vy * restitution
    if abs(vy) < 0.5:
        vx -= \dots \text{(siła tarcia)}
```

Siła tarcia kinetycznego:

$$F_{\text{tarcia}} = \mu \cdot m \cdot g \quad a_{\text{tarcia}} = \frac{F_{\text{tarcia}}}{m} = \mu \cdot g$$

▣ Wektory sił i wykresy

Kod rysuje wektory:

- Grawitacji – zawsze w dół
- Wiatru – poziomo
- Prędkości – dynamicznie

```
draw_arrow(x, y, vx * scale, vy * scale, 'purple')
```

▣ Wektory na wykresie biegunowym

```
draw_polar_arrow(polar_ax, angle, magnitude, color)
```

Ten fragment przelicza wartości sił i prędkości do współrzędnych biegunowych. Dzięki temu użytkownik może obserwować ich kierunki i względne wartości.

▣ Wykresy prędkości i przyspieszenia

```
current_speed = np.sqrt(vx**2 + vy**2)  
net_acc = np.sqrt(ax_acc**2 + ay_acc**2)
```

Wartości te są dodawane do list i rysowane w czasie rzeczywistym.

▣ Podsumowanie

Ten skrypt to przykład dynamicznej symulacji ruchu w dwuwymiarowym polu sił, uwzględniającej:

- Grawitację
- Wiatr jako siłę zewnętrzną
- Sprężystość przy zderzeniu z podłożem
- Tarcie dynamiczne przy niskiej prędkości

Dzięki interfejsowi graficznemu użytkownik może manipulować parametrami i obserwować efekty fizyczne w czasie rzeczywistym.

From:

<https://wiki.ostrowski.net.pl/> - **Kacper's Wiki**

Permanent link:

https://wiki.ostrowski.net.pl/doku.php?id=narzedzia:apple_sim_py&rev=1746613280

Last update: **2025/05/07 12:21**