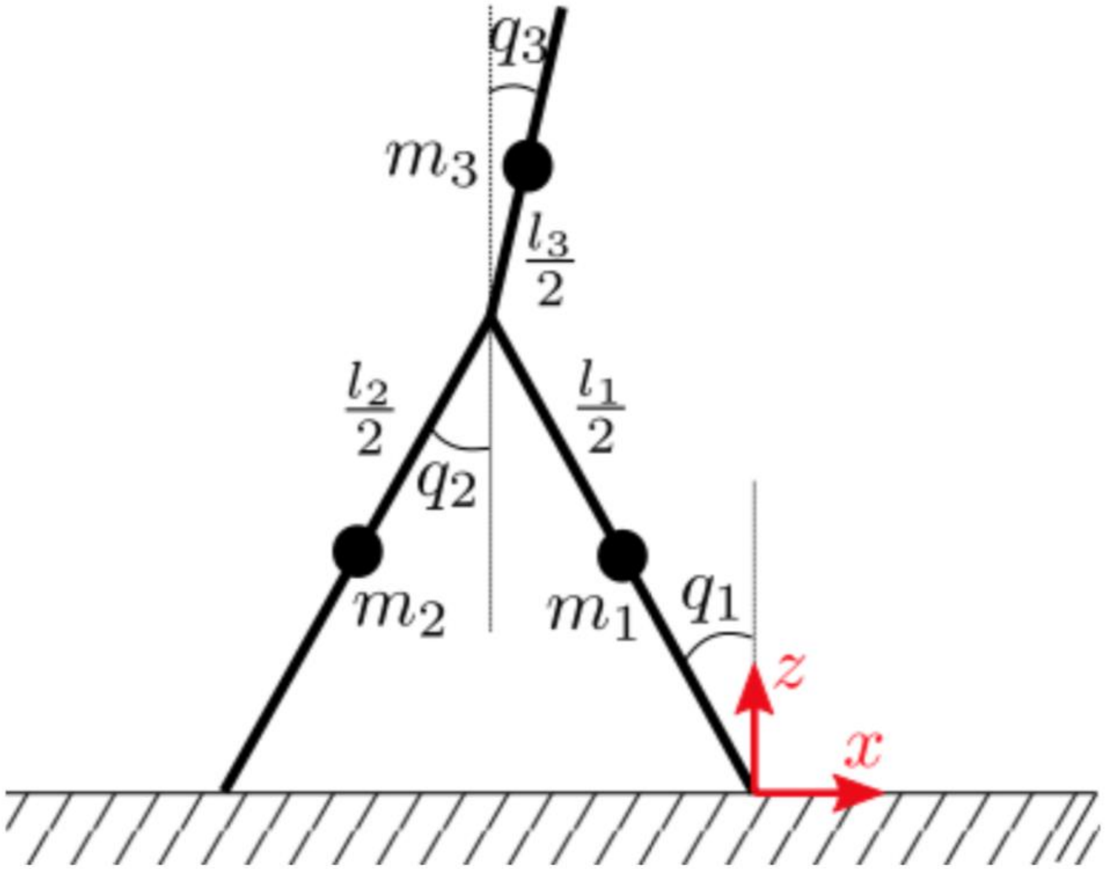
Simulation of a three-link 2D biped

Legged Robots

Overview

- Transform equations of motion into first order differential equation
- Numerical integration of the swing phase model
- Handling of discrete events (ground contact)



First order ODE

$$\boldsymbol{m\ddot{x} + b\dot{x} + kx = f}$$

$$\boldsymbol{\dot{y} = f(t, y)}$$

Solvers can solve first order ODE

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$

Use two new states

$$\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix}$$

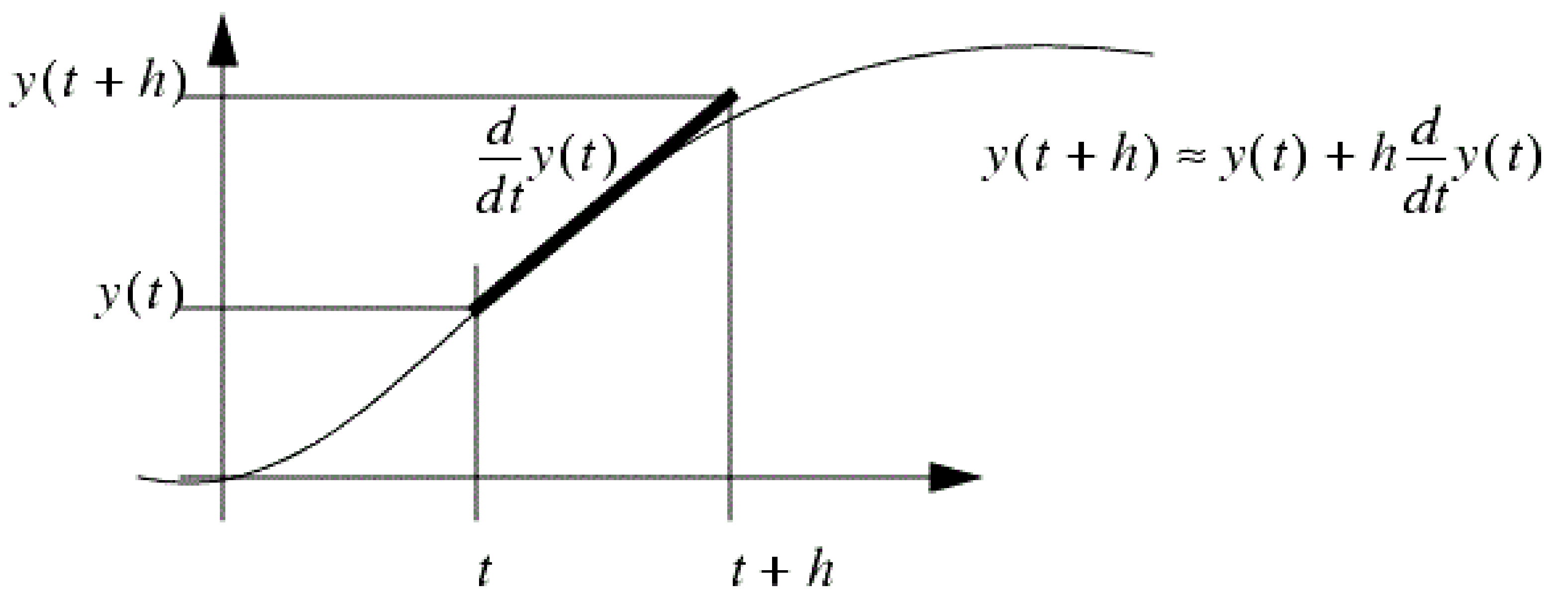
Differentiate

$$= \begin{bmatrix} y_2 \\ \dfrac{(f - ky_1 - by_2)}{m} \end{bmatrix}$$

Right hand side is only a function of y

Do the same for the equations of motion of our model!

Numerical integration

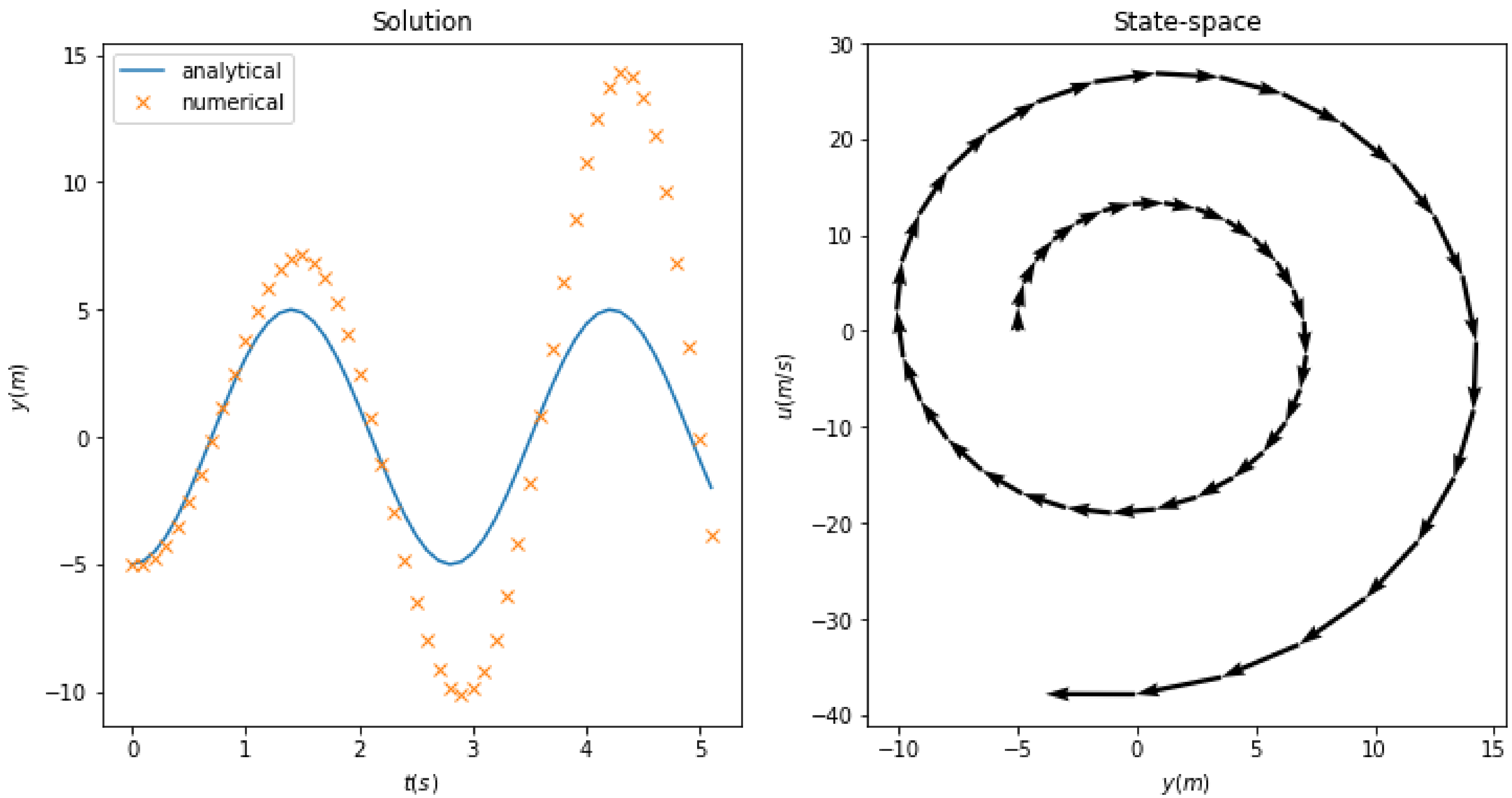


$$y(t+h) = y(t) + h\,\dot{y}(t) + \frac{h^2}{2!}\ddot{y}(t) + \cdots$$

What about stability?

$$m\ddot{x} + b\dot{x} + kx = 0$$

You can make an undammed oscillator marginally stable

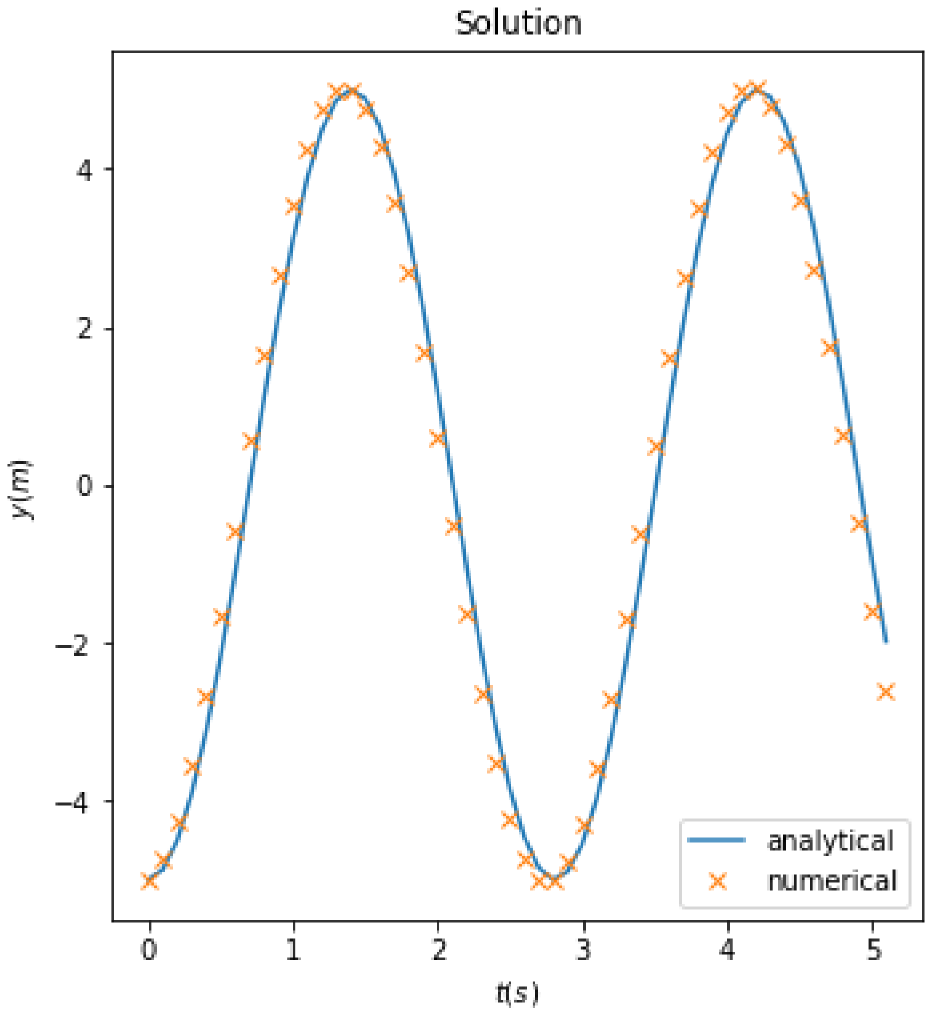


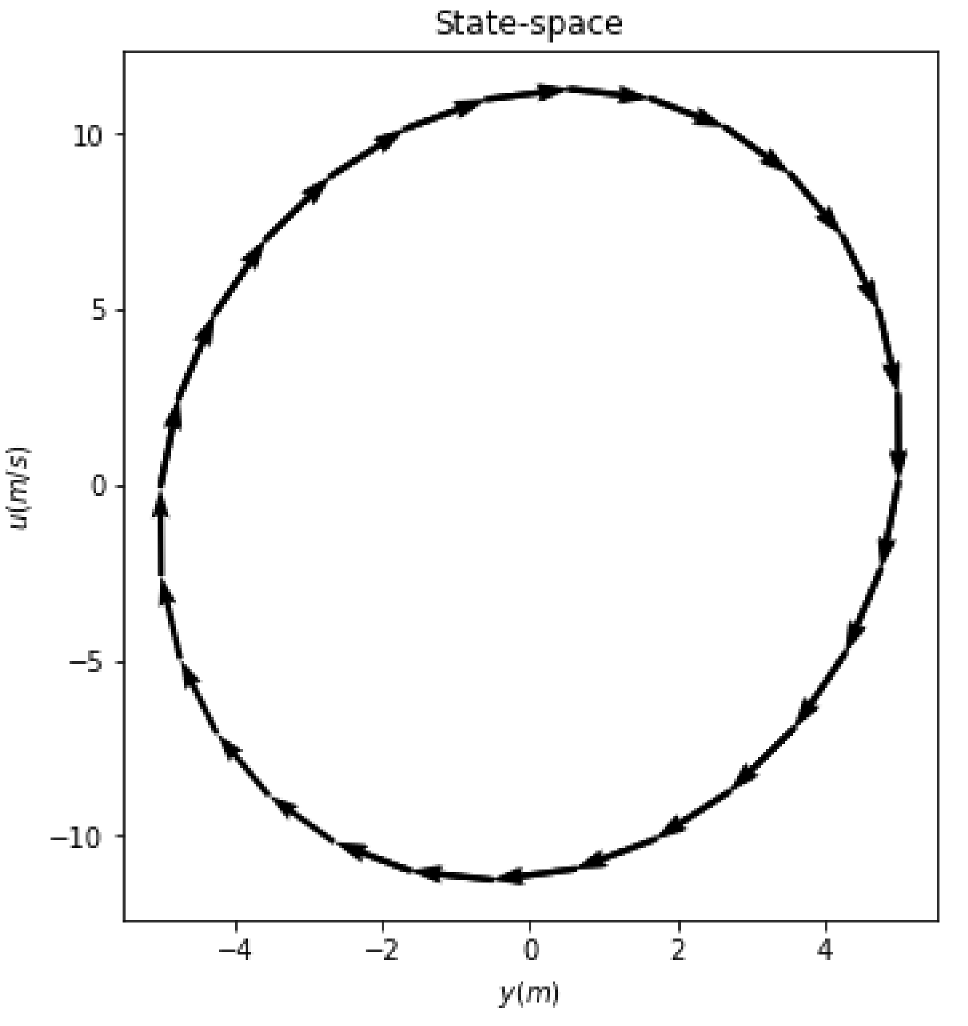
$$y(t+h) = y(t) + h\,\dot{y}(t)$$

Explicit Euler is unstable!

https://en.wikipedia.org/wiki/Damping_ratio

Implicit Euler is Stable



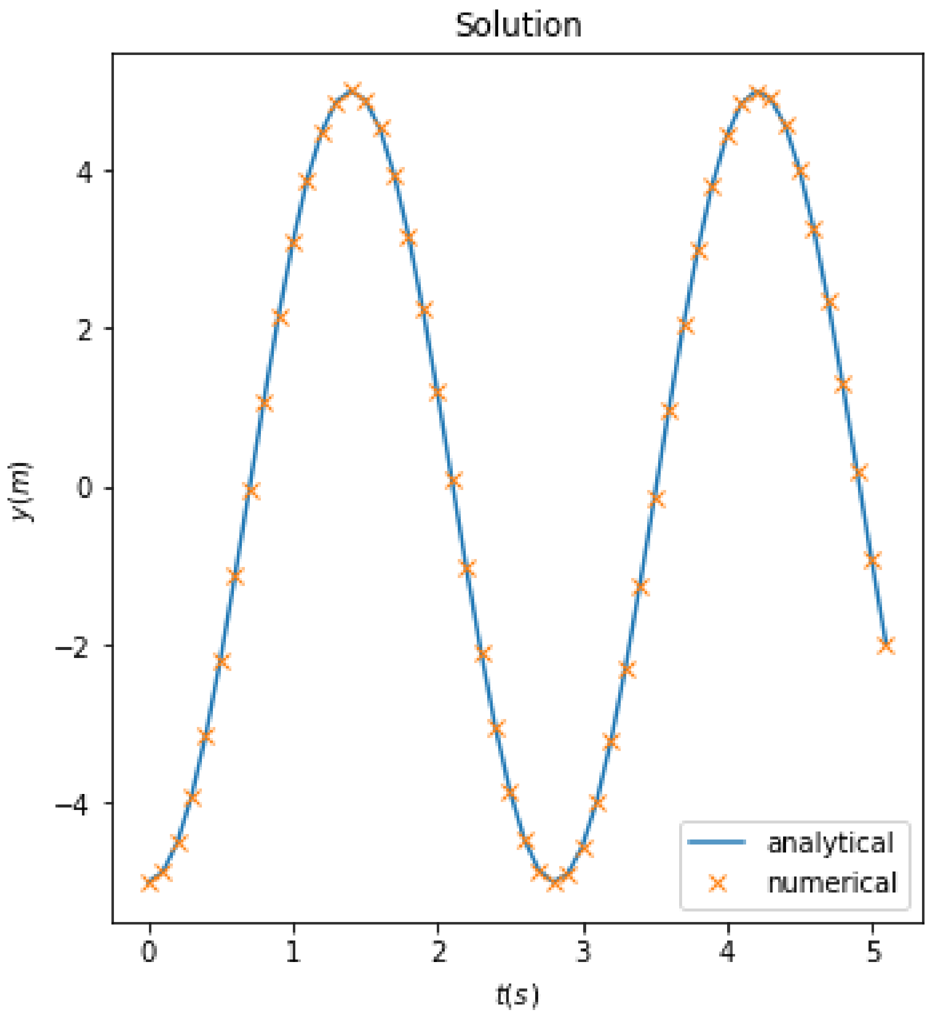


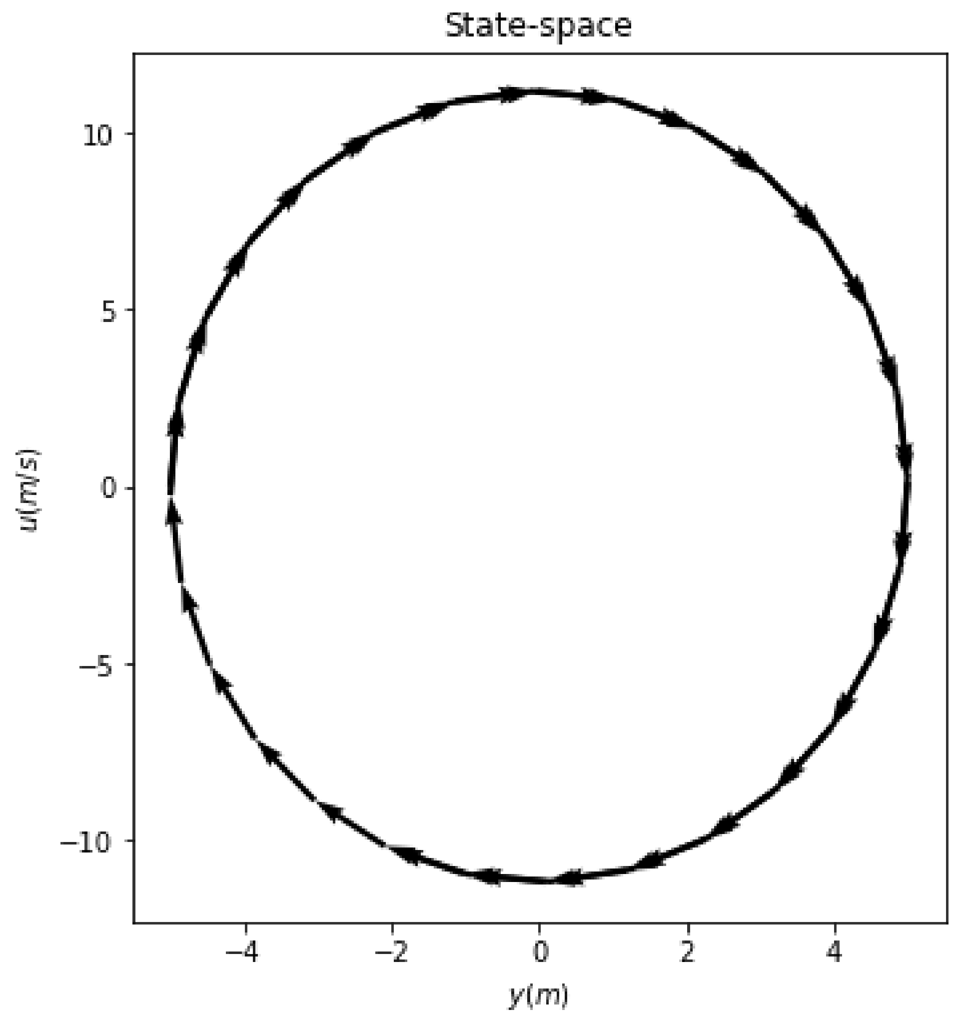
$$y(t+h) = y(t) + h\,\dot{y}(t+h)$$

Depends on future values of derivative!

https://en.wikipedia.org/wiki/Backward_Euler_method

Midpoint methods





$$\dot{y}(t) = f(t, y(t))$$

$$y(t+h) = y(t) + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}$$

$$\frac{error}{step} \sim O(h^5)$$

https://en.wikipedia.org/wiki/Midpoint_method

$$k_1 = hf(t, y(t))$$

$$k_2 = hf\left(t + \frac{h}{2}, y(t) + \frac{k_1}{2}\right)$$

$$k_3 = hf\left(t + \frac{h}{2}, y(t) + \frac{k_2}{2}\right)$$

$$k_4 = hf(t + h, y(t) + k_3)$$

Adaptive time stepping

- Whatever the underlying method, a major problem lies in determining a good step size

- Ideally, one wants to choose h as large as possible, but not so large as to give an unreasonable amount of error, or worse still, to induce instability

- Adaptive time stepping

"Many small steps should tiptoe through treacherous terrain,
while a few great strides should speed through smooth uninteresting countryside."
~ Numerical Recipes

MATLAB ode45

```
[T, Y, TE, YE] = ode45(@eqns, t_end, y0, options);
```

$$\dot{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{t}, \boldsymbol{y})$$

```
options = odeset('RelTol', 1e-5, 'Events', @event_func);
```

```
function [value, isterminal, direction] = event_func(t, y)
```

```
value: zero crossing to trigger an event

isterminal: [1|0] indicates termination of integrator

direction: [1|0|-1] which type of zero crossing triggers an event
```

https://www.mathworks.com/help/matlab/ref/ode45.html

https://www.mathworks.com/help/matlab/math/ode-event-location.html