Visual Generative Modeling 2025

Nikita Starodubcev

Yandex Research
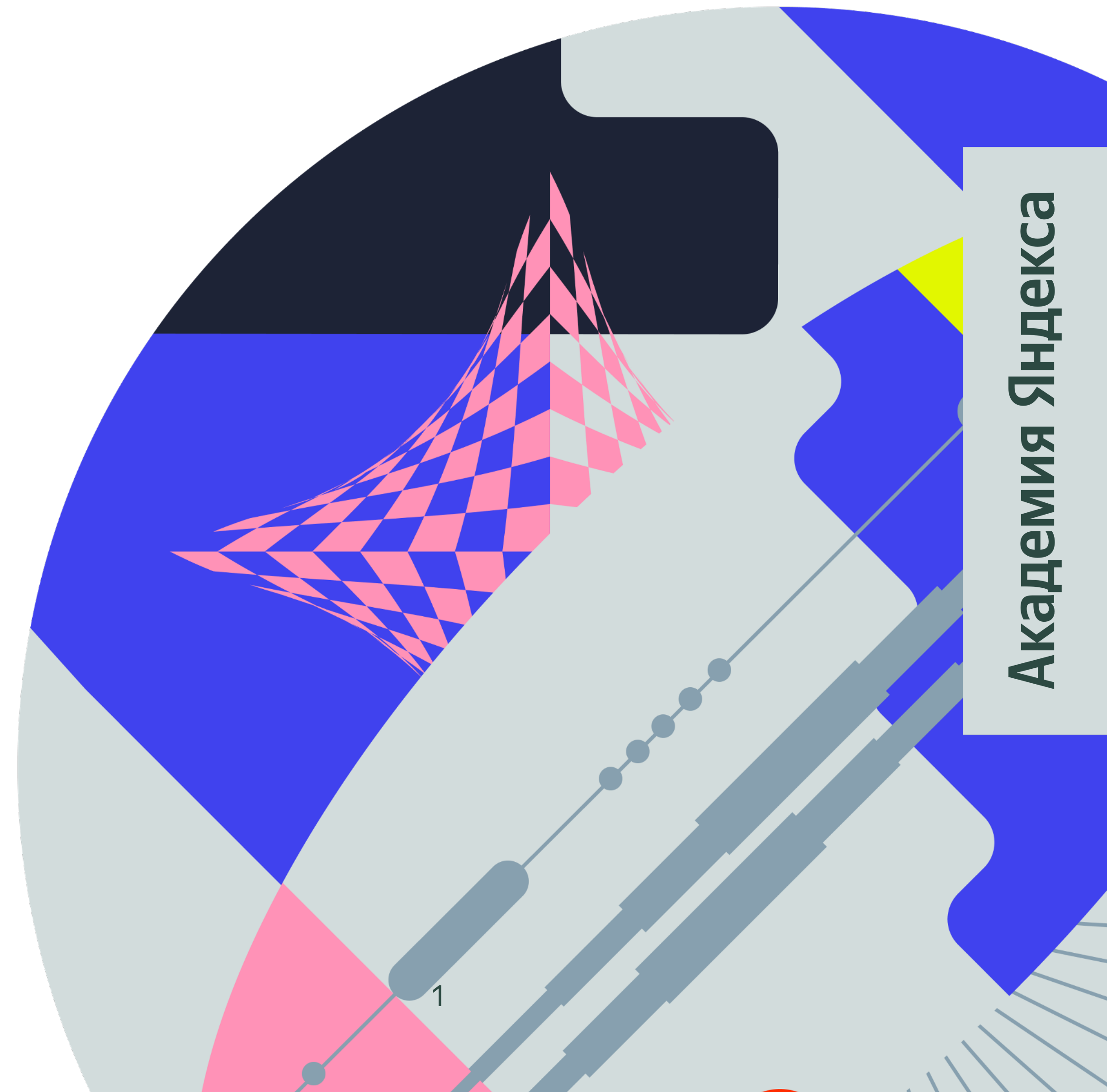
Offtop

$$DM \overset{?}{==} FM$$

Offtop

In the FM, we have the following:

$$\boldsymbol{x} = (1-t)\boldsymbol{x}_0 + \boldsymbol{\epsilon} t, \ t \in [0,1], \ \mathrm{d}\boldsymbol{x} = \boldsymbol{u}_{\boldsymbol{\theta}}(\boldsymbol{x}, t)\mathrm{d}t, \ \boldsymbol{u}_{\boldsymbol{\theta}} := \boldsymbol{\epsilon}_{\boldsymbol{\theta}} - \boldsymbol{x}_{\boldsymbol{\theta}}$$

Let's show that FM is a special case of DMs. The basic equation in DMs:

$$\mathrm{d}\boldsymbol{x} = \left[\boldsymbol{x}f(t) - \frac{1}{2}g(t)^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) \right] \mathrm{d}t \, .$$

Which $f(t)$ and $g(t)$ should be to obtain the FM forward process? To this end, lets remember the connection between $f(t)$, $g(t)$ and statistics of the forward process (expectation and variance).

$$\mathrm{d}\mathbb{E}_{\boldsymbol{x}}\boldsymbol{x} = f(t)\mathbb{E}_{\boldsymbol{x}}\boldsymbol{x}\mathrm{d}t, \ \mathbb{E}_{\boldsymbol{x}}\boldsymbol{x}(0) = \boldsymbol{x}_0 \, .$$

$$\mathrm{d}\mathbb{D}_{\boldsymbol{x}}\boldsymbol{x} = \left(2f(t)\mathbb{D}_{\boldsymbol{x}}\boldsymbol{x} + g(t)^2 \right) \mathrm{d}t, \ \mathbb{D}_{\boldsymbol{x}}\boldsymbol{x}(0) = 0.$$

Offtop

$$f(t) = -\frac{1}{1-t}. \qquad g(t)^2 = \frac{2t}{1-t}.$$

$$\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) \approx -\frac{1}{t^2}\left(\boldsymbol{x} - (1-t)\boldsymbol{x_\theta}\right) = -\frac{\boldsymbol{\epsilon_\theta}}{t}$$

$$\mathrm{d}\boldsymbol{x} = \left[-\boldsymbol{x}\frac{1}{1-t} + \frac{1}{2}\frac{2t}{1-t}\frac{\boldsymbol{\epsilon_\theta}}{t}\right]\mathrm{d}t,$$

$$\mathrm{d}\boldsymbol{x} = \frac{1}{1-t}\left[-(1-t)\boldsymbol{x_\theta} - \boldsymbol{\epsilon_\theta}t + \boldsymbol{\epsilon_\theta}\right]\mathrm{d}t,$$

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{u_\theta}\mathrm{d}t$$

Offtop

DM == FM

FM is special case of DM with specific noising process

Lecture 3 | Numerical solvers

Noising process $\longrightarrow$ ODE (SDE) $\longrightarrow$ Training of NN $\longrightarrow$ Inference using solver

	Noising process	ODE	Parameterization
General case	$\boldsymbol{x}_t = \alpha_t \boldsymbol{x}_0 + \sigma_t \boldsymbol{z},$ $t \in [0,T], \boldsymbol{z} \sim \mathcal{N}(0,1)$	$\mathrm{d}\boldsymbol{x} = \left[\boldsymbol{x} f(t) - \frac{1}{2} g(t)^2 \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) \right] \mathrm{d}t$ $f(t) = \frac{\mathrm{d}\log\alpha_t}{\mathrm{d}t}, g^2(t) = \frac{\mathrm{d}\sigma_t^2}{\mathrm{d}t} - 2\frac{\mathrm{d}\log\alpha_t}{\mathrm{d}t}\sigma_t^2$	$\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) = -\dfrac{\boldsymbol{x} - \alpha_t \mathbb{E}\boldsymbol{x}_0 \mid \boldsymbol{x}}{\sigma_t^2}$ $\boldsymbol{\epsilon} = \dfrac{\boldsymbol{x} - \alpha_t \mathbb{E}\boldsymbol{x}_0 \mid \boldsymbol{x}}{\sigma_t}, \quad \mathbb{E}\boldsymbol{x}_0 \mid \boldsymbol{x} \approx \boldsymbol{x}_0$
VP	$\boldsymbol{x}_t = \sqrt{\alpha_t}\boldsymbol{x}_0 + \sqrt{1-\alpha_t}\boldsymbol{z}$ $t \in [0,T], \boldsymbol{z} \sim \mathcal{N}(0,1)$	$\mathrm{d}\boldsymbol{x} = \frac{1}{2\alpha_t}\left[x + \nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) \right] \mathrm{d}\alpha_t$	$\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) = -\dfrac{\boldsymbol{x} - \sqrt{\alpha_t}\mathbb{E}\boldsymbol{x}_0 \mid \boldsymbol{x}}{1-\alpha_t}$
VE	$\boldsymbol{x}_t = \boldsymbol{x}_0 + t\boldsymbol{z}$ $t \in [0,T], \boldsymbol{z} \sim \mathcal{N}(0,1)$	$\mathrm{d}\boldsymbol{x} = -t\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})\mathrm{d}t$	$\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) = -\dfrac{\boldsymbol{x} - \mathbb{E}\boldsymbol{x}_0 \mid \boldsymbol{x}}{t^2}$
FM (RF)	$\boldsymbol{x}_t = (1-t)\boldsymbol{x}_0 + t\boldsymbol{z}$ $t \in [0,1], \boldsymbol{z} \sim \mathcal{N}(0,1)$	$\mathrm{d}\boldsymbol{x} = -\frac{1}{1-t}\left[x + t\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) \right] \mathrm{d}t$ $\mathrm{d}\boldsymbol{x} = v(x,t)\mathrm{d}t$	$\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x}) = -\dfrac{\boldsymbol{x} - (1-t)\mathbb{E}\boldsymbol{x}_0 \mid \boldsymbol{x}}{t^2}$ $\boldsymbol{v} = \boldsymbol{\epsilon} - \mathbb{E}\boldsymbol{x}_0 \mid \boldsymbol{x}$

$$1.\ \mathbb{E}_{\boldsymbol{x}_0,\ t,\ \boldsymbol{x}_t}\|\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{\epsilon}\| \qquad 2.\ \mathbb{E}_{\boldsymbol{x}_0,\ t,\ \boldsymbol{x}_t}\|\boldsymbol{s}_\theta(\boldsymbol{x}_t, t) - \nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{x}_t)\| \qquad 3.\ \mathbb{E}_{\boldsymbol{x}_0,\ t,\ \boldsymbol{x}_t}\|\boldsymbol{x}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{x}_0\|$$

1. Conditional expectation

$$\mathbb{E}_{\boldsymbol{x}_0,\ t,\ \boldsymbol{x}_t}\|\boldsymbol{x}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{x}_0\|$$

$$\boldsymbol{x}_\theta(\boldsymbol{x}, t) \approx \mathbb{E}\boldsymbol{x}_0\,|\,\boldsymbol{x} = \int \boldsymbol{x}_0 p_t(\boldsymbol{x}_0\,|\,\boldsymbol{x})\mathrm{d}\boldsymbol{x}_0\,.$$

$$p_t(\boldsymbol{x}_0\,|\,\boldsymbol{x}) = \frac{p_t(\boldsymbol{x}\,|\,\boldsymbol{x}_0)p(\boldsymbol{x}_0)}{\int p_t(\boldsymbol{x}\,|\,\boldsymbol{x}_0)p(\boldsymbol{x}_0)\mathrm{d}\boldsymbol{x}_0},\ p_{data}(\boldsymbol{x}_0) = \frac{1}{N}\sum_{j=1}^{N}\delta(\boldsymbol{x}_0 - \boldsymbol{x}_0^j)\,.$$

$$\mathbb{E}\boldsymbol{x}_0\,|\,\boldsymbol{x} = \int \boldsymbol{x}_0\frac{p_t(\boldsymbol{x}\,|\,\boldsymbol{x}_0)p(\boldsymbol{x}_0)}{\int p_t(\boldsymbol{x}\,|\,\boldsymbol{x}_0)p(\boldsymbol{x}_0)\mathrm{d}\boldsymbol{x}_0}\mathrm{d}\boldsymbol{x}_0 = \ldots = \frac{\sum_{j=1}^{N}\boldsymbol{x}_0^j\,\mathcal{N}\left(\boldsymbol{x}\,|\,\alpha_t\boldsymbol{x}_0^j, \sigma_t^2\right)}{\sum_{j=1}^{N}\mathcal{N}\left(\boldsymbol{x}\,|\,\alpha_t\boldsymbol{x}_0^j, \sigma_t^2\right)}\,.$$

1. Conditional expectation

$$\mathbb{E}\boldsymbol{x}_0 \,|\, \boldsymbol{x} = \frac{\sum_{j=1}^{N} \boldsymbol{x}_0^j \,\mathcal{N}\left(\boldsymbol{x} \mid \alpha_t \boldsymbol{x}_0^j, \sigma_t^2\right)}{\sum_{j=1}^{N} \mathcal{N}\left(\boldsymbol{x} \mid \alpha_t \boldsymbol{x}_0^j, \sigma_t^2\right)} \,.$$
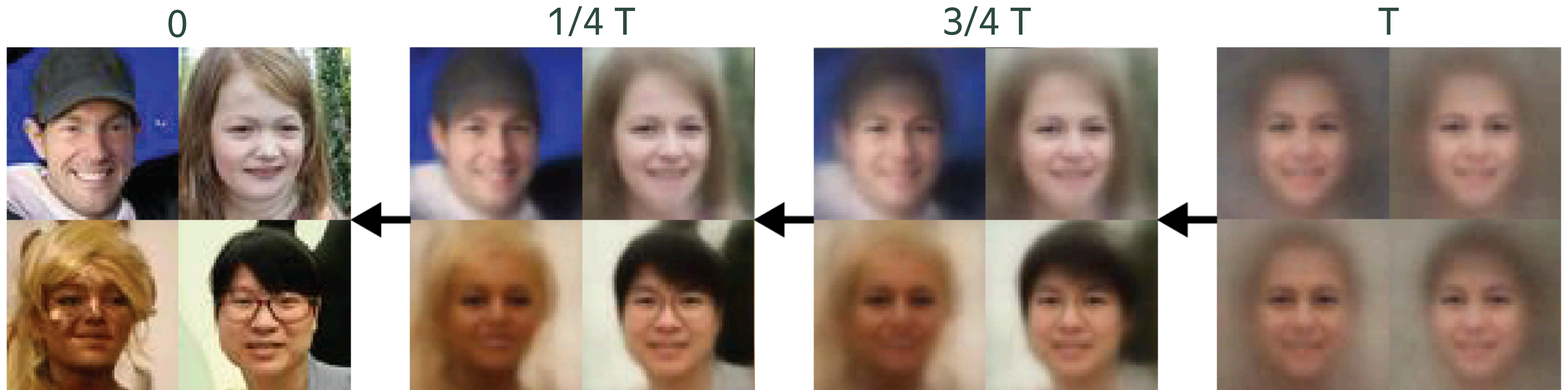
We make the images less and less averaged, coarse-to-fine generation

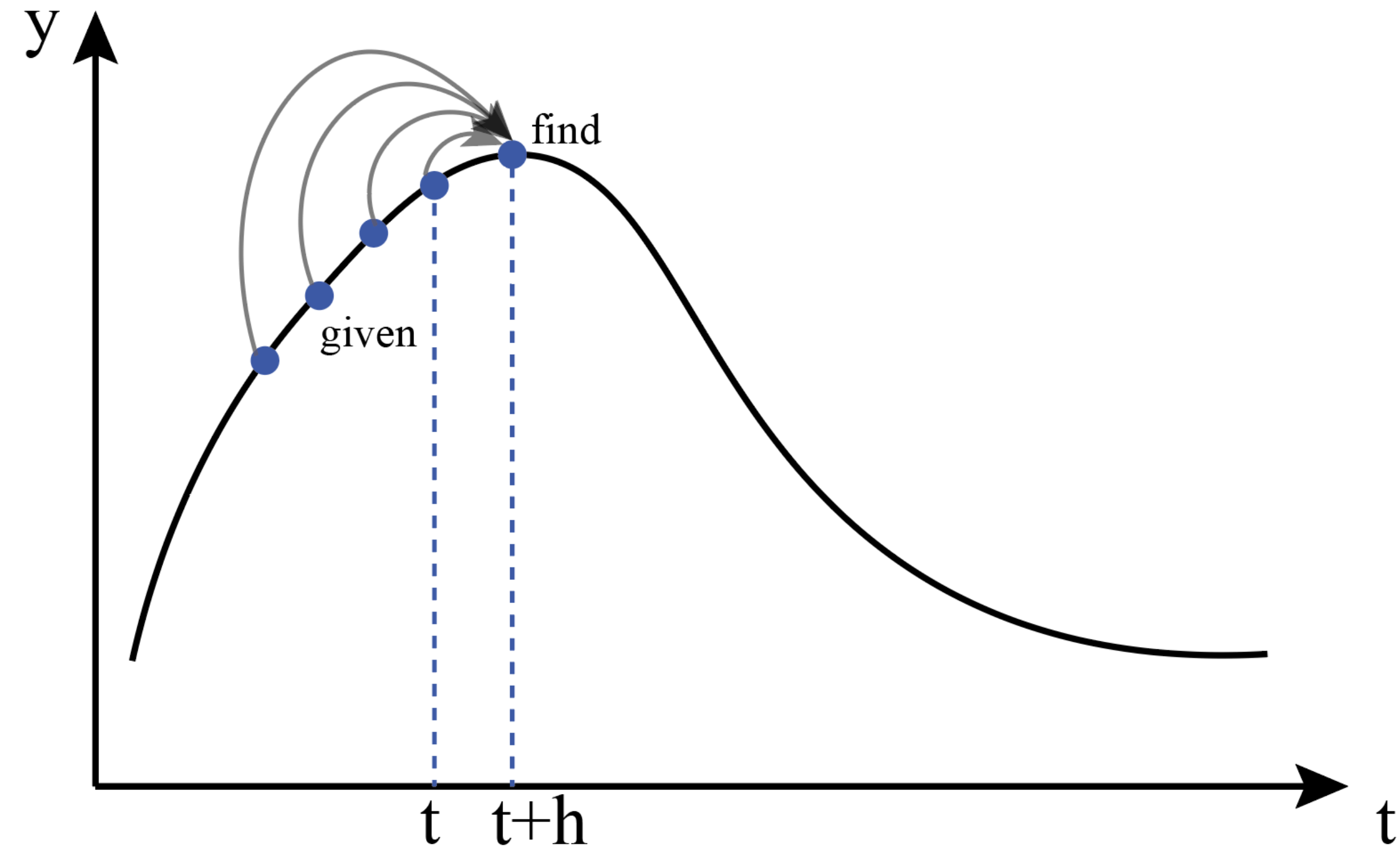# 2. Basics of numerical solution of ODE

Cauchy problem

$$\mathrm{d}y = f(t, y(t))\mathrm{d}t,\ y(0) = y_0$$

Numerical solvers $\quad [t_0, \ldots, t_N], h = t_{i+1} - t_i$

Singlestep $\quad y_{i+1} = y_i + \Phi(y_i, h)$
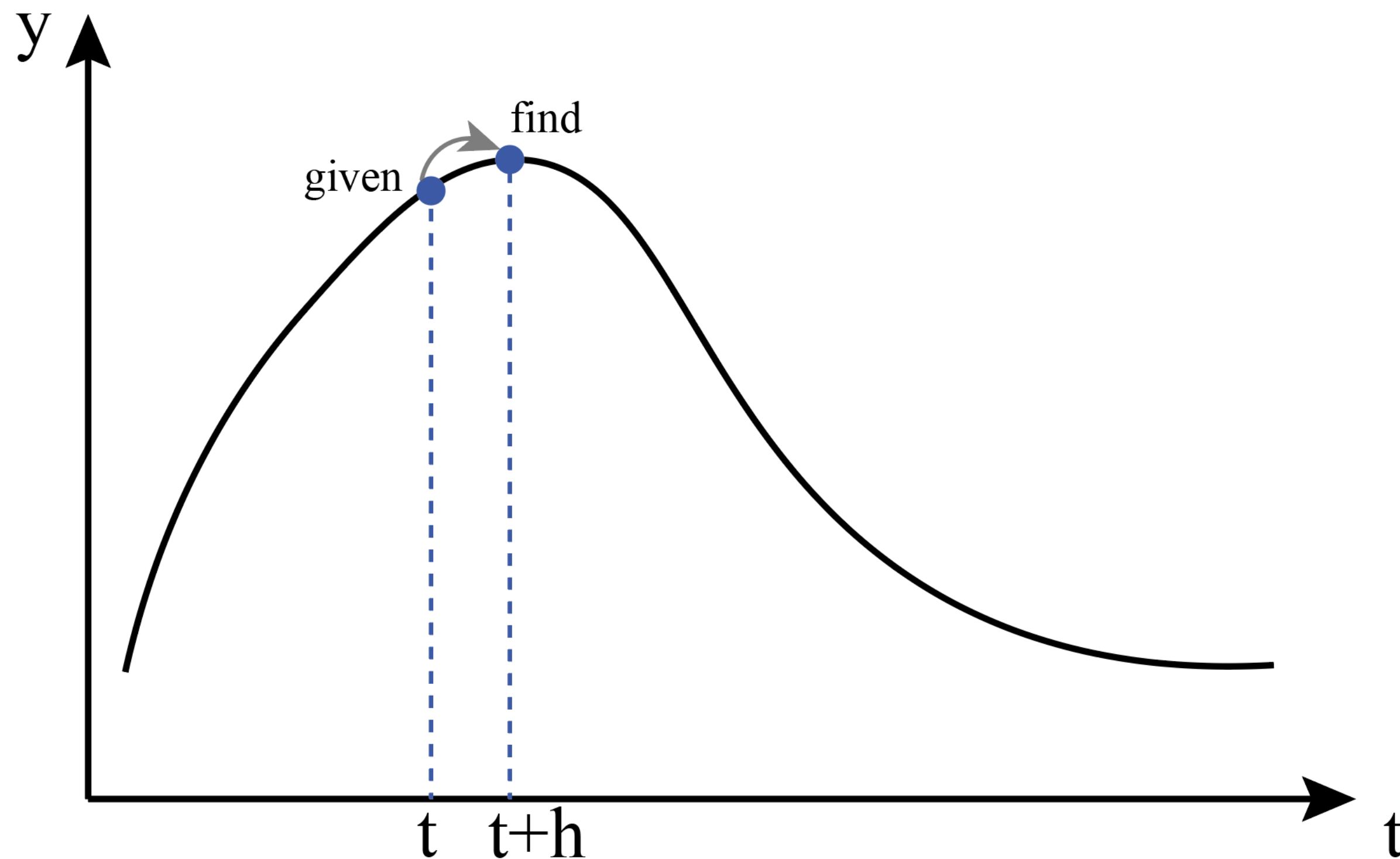
Multistep $\quad y_{i+1} = y_i + \Phi(y_i, y_{i-1}, \ldots, h)$

Singlestep $\quad y_{i+1} = y_i + \Phi(y_i, h)$
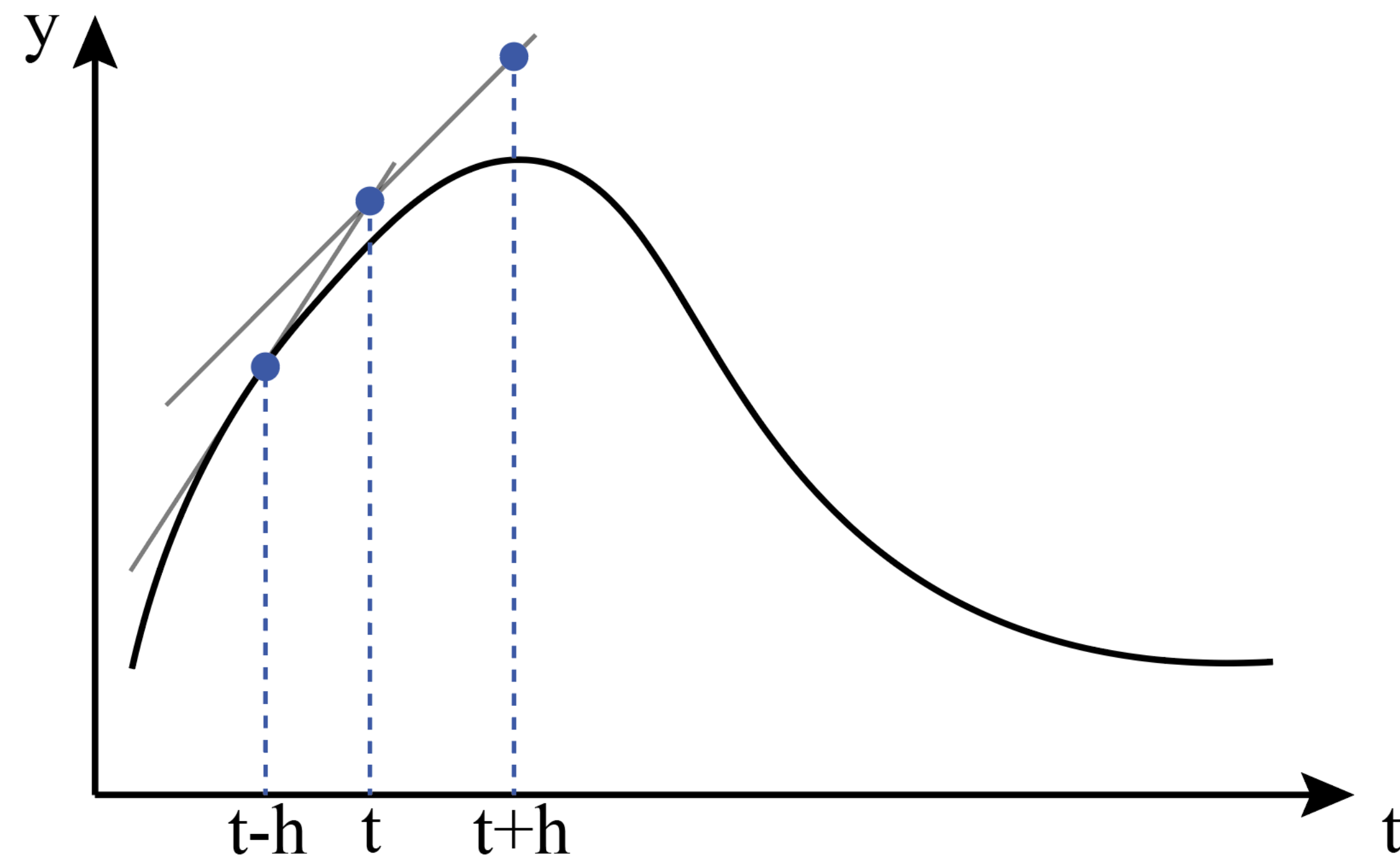
Multistep $\quad y_{i+1} = y_i + \Phi(y_i, y_{i-1}, \ldots, h)$

Implicit $\quad y_{i+1} = y_i + \Phi(y_{i+1}, y_i, h)$

Implicit $\quad y_{i+1} = y_i + \Phi(y_{i+1}, y_i, y_{i-1}, \ldots, h)$
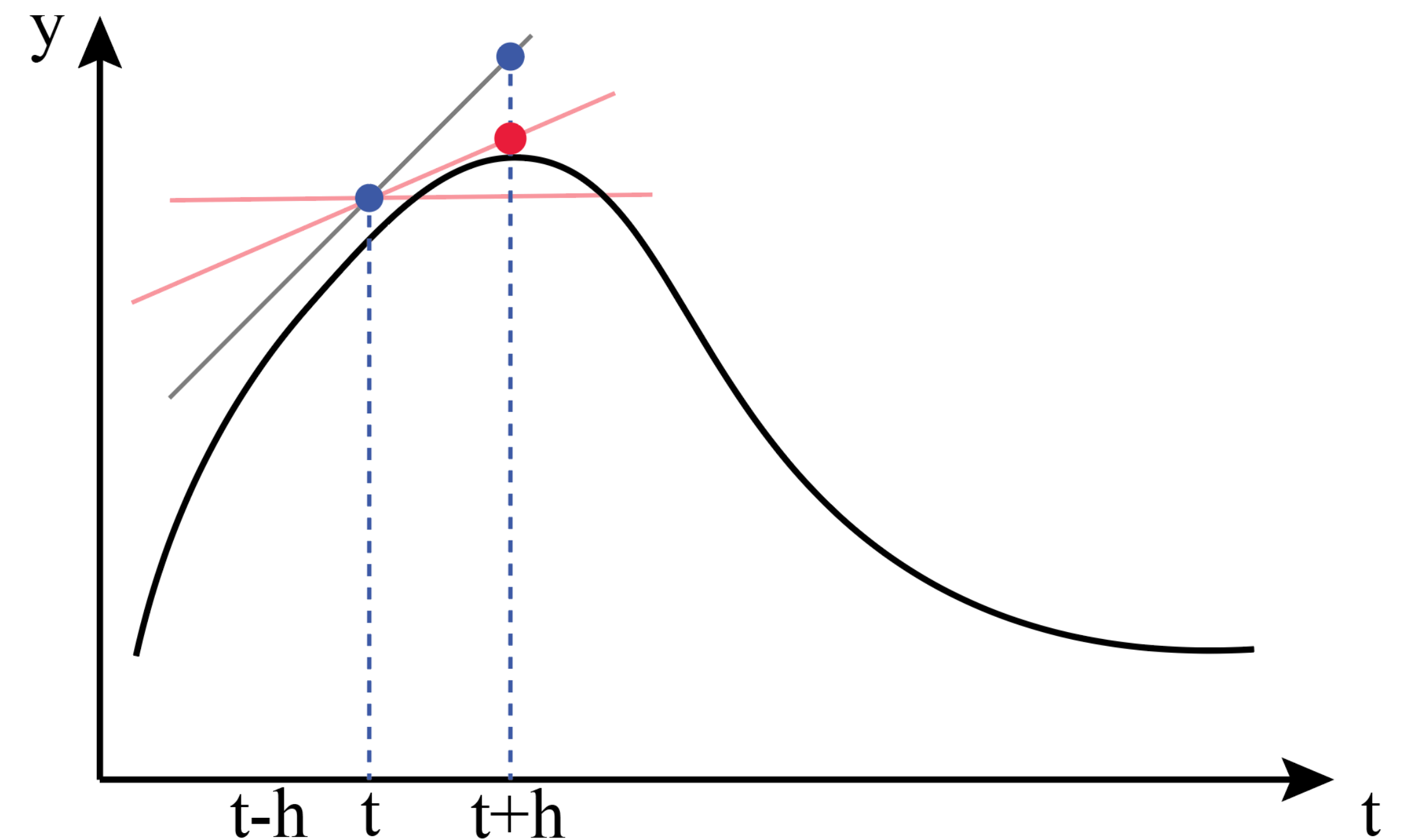
Examples:

* Euler (singlestep, explicit)

$$y_{i+1} = y_i + hf(y_i, t_i)$$

* Trapezoidal rule (singlestep, implicit)

$$y_{i+1} = y_i + \frac{h}{2}\left(f(y_i, t_i) + f(y_{i+1}, t_{i+1})\right)$$

How to understand how accurate is your solver?

1. Local truncation error (we assume previous steps are exact)

$$\tau_{i+1} = \|y_{i+1} - y(t_{i+1})\|$$

* Euler

$$y(t_{i+1}) = y_i + hy_i' + \frac{h^2}{2}y_i'' + O(h^3)$$

$$y_{i+1} = y_i + hf(y_i, t_i), f(y_i, t_i) = y_i'$$

$$\tau_{i+1} = y_i + hy_i' - \left(y_i + hy_i' + \frac{h^2}{2}y_i'' + O(h^3) \right) = -\frac{h^2}{2}y_i'' + O(h^3) \sim O(h^2)$$

* Trapezoidal rule

$$\tau_{i+1} \sim O(h^3)$$

2. Global truncation error (accumulative error after N steps)

$$e_N = \|y_N - y(t_N)\| \qquad e_N \sim N\tau \sim \frac{\tau}{h}$$

* Euler

$$e_N \sim O(h)$$

$$\int_{t_i}^{t_{i+1}} \mathrm{d}y = \int_{t_i}^{t_{i+1}} f(t, y(t))\mathrm{d}t$$

$$y_{i+1} = y_i + \int_{t_i}^{t_{i+1}} f(t, y(t))\mathrm{d}t$$

$$f(t, y(t)) = \sum_{k=0} \frac{h^k}{k!} f^{(k)}(t_i, y_i) = f(t_i, y_i) + O(h)$$

$$y_{i+1} = y_i + f(t_i, y_i) + O(h^2)$$

* Trapezoidal rule

$$e_N \sim O(h^2)$$

Solver converges if:

1. LTE → 0, as h → 0

2. It is stable
small perturbations do not grow exponentially.

Multistep explicit methods (Adams-Bashforth methods)

Use past solution values and past derivative evaluations to compute the next step

$$y_{i+1} = y_i + \int_{t_i}^{t_{i+1}} f(t, y(t))\mathrm{d}t \quad \text{Cannot integrate } f(t, y(t))$$

$$f(t, y(t)) \approx p_{s-1}(t), \quad p_{s-1}(t) = \sum_{j=0}^{s-1} \frac{(-1)^{s-j-1} f(t_{n+j}, y_{n+j})}{j!(s-j-1)!h^{s-1}} \prod_{i=0}^{s-1} (t - t_{n+i}) \quad \text{Lagrange polynom}$$

Lagrange polynom
Next point
Existed points
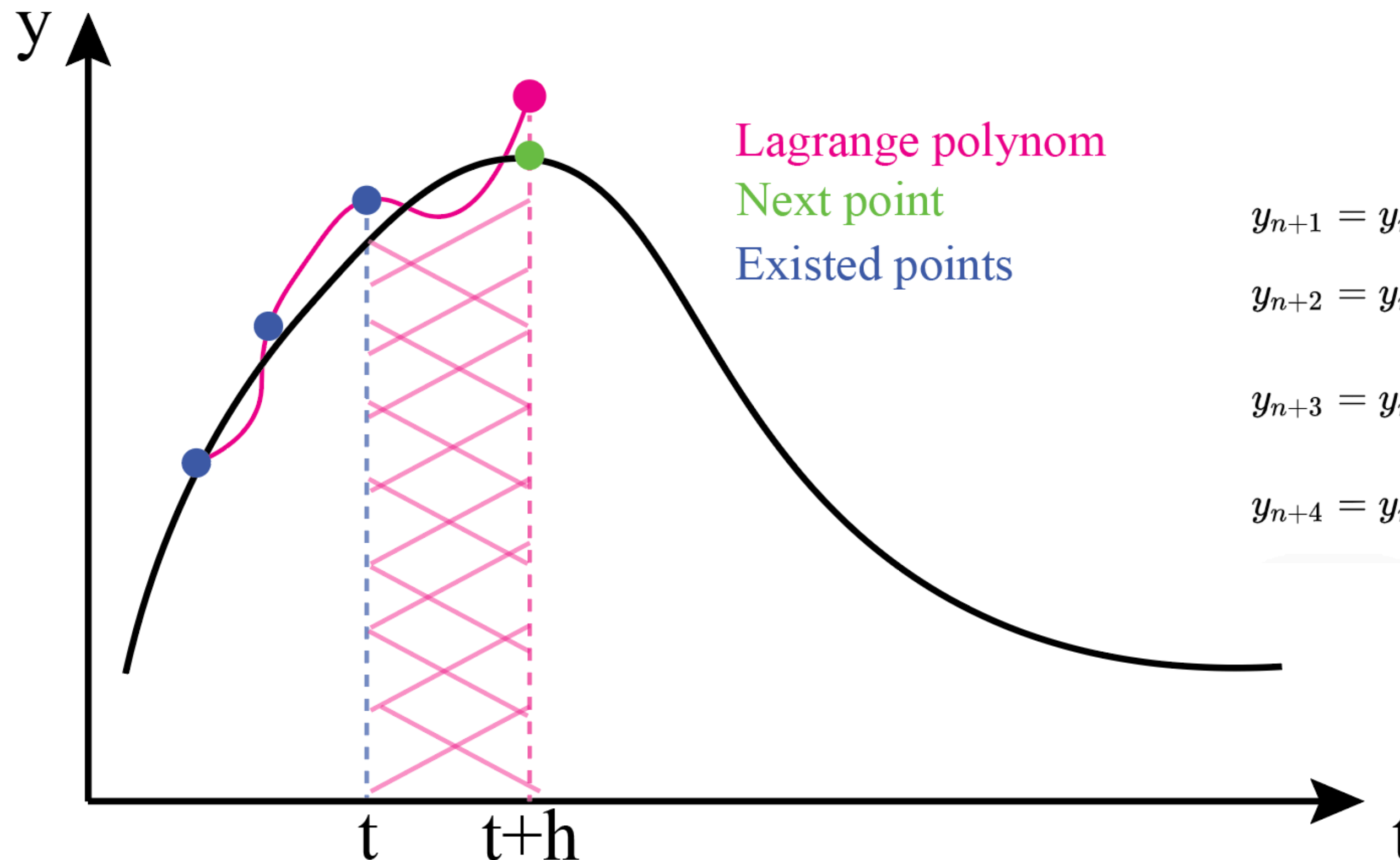
$$y_{n+1} = y_n + hf(t_n, y_n), \qquad \text{(This is the Euler method)}$$

$$y_{n+2} = y_{n+1} + h\left(\frac{3}{2} f(t_{n+1}, y_{n+1}) - \frac{1}{2} f(t_n, y_n)\right),$$

$$y_{n+3} = y_{n+2} + h\left(\frac{23}{12} f(t_{n+2}, y_{n+2}) - \frac{16}{12} f(t_{n+1}, y_{n+1}) + \frac{5}{12} f(t_n, y_n)\right),$$

$$y_{n+4} = y_{n+3} + h\left(\frac{55}{24} f(t_{n+3}, y_{n+3}) - \frac{59}{24} f(t_{n+2}, y_{n+2}) + \frac{37}{24} f(t_{n+1}, y_{n+1}) - \frac{9}{24} f(t_n, y_n)\right),$$

Multistep vs singlestep (DPM-solver specific)

$$y_{i+1} = y_i + \int_{t_i}^{t_{i+1}} f(t, y(t))\mathrm{d}t$$

$$f(t, y(t)) = \sum_{k=0}^{\infty} \frac{h^k}{k!} f^{(k)}(t_i, y_i) = f(t_i, y_i) + hf'(t_i, y_i) + O(h^2)$$

How to approximate $f'(t_i, y_i)$?

Singlestep (use next intermediate point)

$$f'(t_i, y_i) \approx \frac{f(t_{i+\delta_i}, y_{i+\delta_i}) - f(t_i, y_i)}{t_{i+\delta_i} - t_i}$$

Multistep (use previous point)

$$f'(t_i, y_i) \approx \frac{f(t_i, y_i) - f(t_{i-1}, y_{i-1})}{h}$$

Summary

I) Singlestep explicit $\quad y_{i+1} = y_i + \Phi(y_i, h)$
 * Euler (1 NFE per step, 1st order)
 * RK 2 (2 NFE per step, 2nd order)
Use the intermediate steps

II) Singlestep implicit $\quad y_{i+1} = y_i + \Phi(y_{i+1}, y_i, h)$
 * Backward Euler
 * RK 2 implicit, Heun (2 NFE per step, 2nd order)
Use the next steps

III) Multistep explicit $y_{i+1} = y_i + \Phi(y_i, y_{i-1}, \ldots, h)$
 * Adams-Bashforth (1 NFE, any order)
Use the previous steps

IV) Multistep implicit $y_{i+1} = y_i + \Phi(y_{i+1}, y_i, y_{i-1}, h)$
 * Adams-Moulton
Use the next and previous steps

V) Multistep + Singlestep
 * General linear methods
Use the previous and intermediate steps

2. DPM-solver (VP)

$$\mathrm{d}\boldsymbol{x} = \frac{1}{2\alpha_t}\left[x + \nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x})\right]\mathrm{d}\alpha_t$$

DDIM (Denoising Diffusion Implicit Models)

$$\boldsymbol{x}_t = \sqrt{\alpha_t}\left(\frac{\boldsymbol{x}_s - \sqrt{1-\alpha_s}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_s, s)}{\sqrt{\alpha_s}}\right) + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_s, s)$$

What is DDIM? Just Euler?

2. DPM-solver (VP)

$$dx = \frac{1}{2\alpha_t} \left[x + \nabla_x \log p_t(x) \right] d\alpha_t$$

DDIM (Denoising Diffusion Implicit Models)

$$x_t = \sqrt{\alpha_t} \left(\frac{x_s - \sqrt{1 - \alpha_s} \epsilon_\theta(x_s, s)}{\sqrt{\alpha_s}} \right) + \sqrt{1 - \alpha_t} \epsilon_\theta(x_s, s)$$

What is DDIM? Just Euler?

$$dx = \frac{1}{2\alpha_t} \left[x - \frac{\epsilon(x, t)}{\sqrt{1 - \alpha_t}} \right] d\alpha_t$$

$$x_t = \frac{\alpha_t + \alpha_s}{2\alpha_s} x_s - \frac{\alpha_t - \alpha_s}{2\alpha_s \sqrt{1 - \alpha_s}} \epsilon_\theta(x_s, s)$$

$$x_t = x_s + \int_s^t \frac{1}{2\alpha_\tau} \left[x - \frac{\epsilon(x, \tau)}{\sqrt{1 - \alpha_t}} \right] d\alpha_\tau$$

$$\mathrm{d}\boldsymbol{x} = \frac{1}{2\alpha_t}\left[\boldsymbol{x} - \frac{\boldsymbol{\epsilon}(\boldsymbol{x},t)}{\sqrt{1-\alpha_t}}\right]\mathrm{d}\alpha_t$$

$$\mathrm{d}\boldsymbol{x} = \left[\frac{\boldsymbol{x}}{2\alpha_t} - \frac{\boldsymbol{\epsilon}(\boldsymbol{x},t)}{2\alpha_t\sqrt{1-\alpha_t}}\right]\mathrm{d}\alpha_t, s \to t$$

$$\phi(t,s)\mathrm{d}\boldsymbol{x} = \phi(t,s)\left[\frac{\boldsymbol{x}}{2\alpha_t} - \frac{\boldsymbol{\epsilon}(\boldsymbol{x},t)}{2\alpha_t\sqrt{1-\alpha_t}}\right]\mathrm{d}\alpha_t,$$

$$\mathrm{d}\left[\boldsymbol{x}\phi(t,s)\right] = -\phi(t,s)\frac{\boldsymbol{\epsilon}(\boldsymbol{x},t)}{2\alpha_t\sqrt{1-\alpha_t}}\mathrm{d}\alpha_t,$$

$$\mathrm{d}\phi(t,s) = -\phi(t,s)\frac{1}{2\alpha_t}\mathrm{d}\alpha_t$$

$$\int_s^t \frac{\mathrm{d}\phi(t,\tau)}{\phi(t,\tau)} = -\int_s^t \frac{\mathrm{d}\alpha_\tau}{2\alpha_\tau} \to \ln\phi(t,s) = \frac{1}{2}\left(\ln\alpha_t - \ln\alpha_s\right)$$

$$\phi(t,s) = \sqrt{\frac{\alpha_t}{\alpha_s}}$$

$$\boldsymbol{x}_t = \sqrt{\frac{\alpha_t}{\alpha_s}}\boldsymbol{x}_s - \frac{1}{2}\int_s^t \phi(t,\tau)\frac{\boldsymbol{\epsilon}(\boldsymbol{x},\tau)}{\alpha_\tau\sqrt{1-\alpha_\tau}}\mathrm{d}\alpha_\tau,$$

$$\boldsymbol{x}_t = \sqrt{\frac{\alpha_t}{\alpha_s}}\boldsymbol{x}_s - \frac{\sqrt{\alpha_t}}{2}\int_s^t \frac{\boldsymbol{\epsilon}(\boldsymbol{x},\tau)}{\sqrt{\alpha_\tau}\alpha_\tau\sqrt{1-\alpha_\tau}}\mathrm{d}\alpha_\tau,$$

$$\int_s^t \frac{\boldsymbol{\epsilon}(\boldsymbol{x},\tau)}{\sqrt{\alpha_\tau}\alpha_\tau\sqrt{1-\alpha_\tau}}\mathrm{d}\alpha_\tau \approx \boldsymbol{\epsilon}(\boldsymbol{x}_s,s)\int_s^t \frac{1}{\sqrt{\alpha_\tau}\alpha_\tau\sqrt{1-\alpha_\tau}}\mathrm{d}\alpha_\tau$$

$$\int \frac{\mathrm{d}\alpha}{\sqrt{\alpha}\,\alpha\sqrt{1-\alpha}}$$

$$1-\alpha = \cos^2 u, \quad \alpha = \sin^2 u, \quad \mathrm{d}\alpha = 2\sin u\cos u\,\mathrm{d}u$$

$$\int \frac{2\mathrm{d}u}{\sin^2 u} = -2\mathrm{ctg}u = -2\mathrm{ctg}\arcsin\sqrt{\alpha} = -2\frac{\sqrt{1-\alpha}}{\sqrt{\alpha}}$$

DDIM = DPM-1

$$x_t = \sqrt{\frac{\alpha_t}{\alpha_s}}x_s - \frac{\sqrt{\alpha_t}}{2}\left(-2\frac{\sqrt{1-\alpha_t}}{\sqrt{\alpha_t}} + 2\frac{\sqrt{1-\alpha_s}}{\sqrt{\alpha_s}}\right)\epsilon(x_s,s)$$

$$\boxed{x_t = \sqrt{\alpha_t}\left(\frac{x_s - \sqrt{1-\alpha_s}\epsilon_\theta(x_s,s)}{\sqrt{\alpha_s}}\right) + \sqrt{1-\alpha_t}\epsilon(x_s,s)}$$

$$x_t = \sqrt{\frac{\alpha_t}{\alpha_s}} x_s - \frac{\sqrt{\alpha_t}}{2} \int_s^t \frac{\epsilon(x, \tau)}{\sqrt{\alpha_\tau} \alpha_\tau \sqrt{1 - \alpha_\tau}} \mathrm{d}\alpha_\tau,$$

$$\epsilon(x, \tau) = \sum_{k=0}^{n-1} \frac{(\alpha_\tau - \alpha_s)^k}{k!} \epsilon^{(k)}(x_s, s)$$
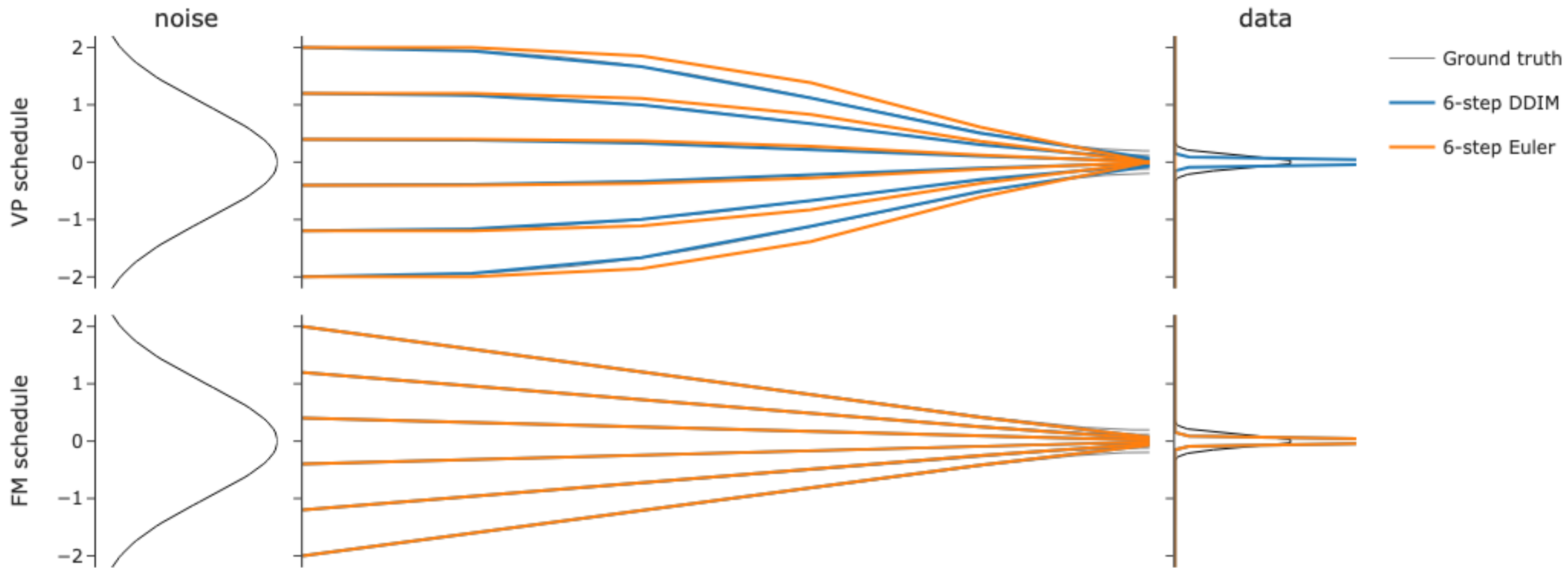
Singlestep (use the intermediate point)

$$\epsilon^{(1)}(x_s, s) \approx \frac{\epsilon(x_{s-\delta_s}, s - \delta_s) - \epsilon(x_s, s)}{\alpha_{s-\delta_s} - \alpha_s}$$

Multistep (use the previous point)

$$\epsilon^{(1)}(x_s, s) \approx \frac{\epsilon(x_s, s) - \epsilon(x_{s+h}, s + h)}{\alpha_s - \alpha_{s+h}}$$
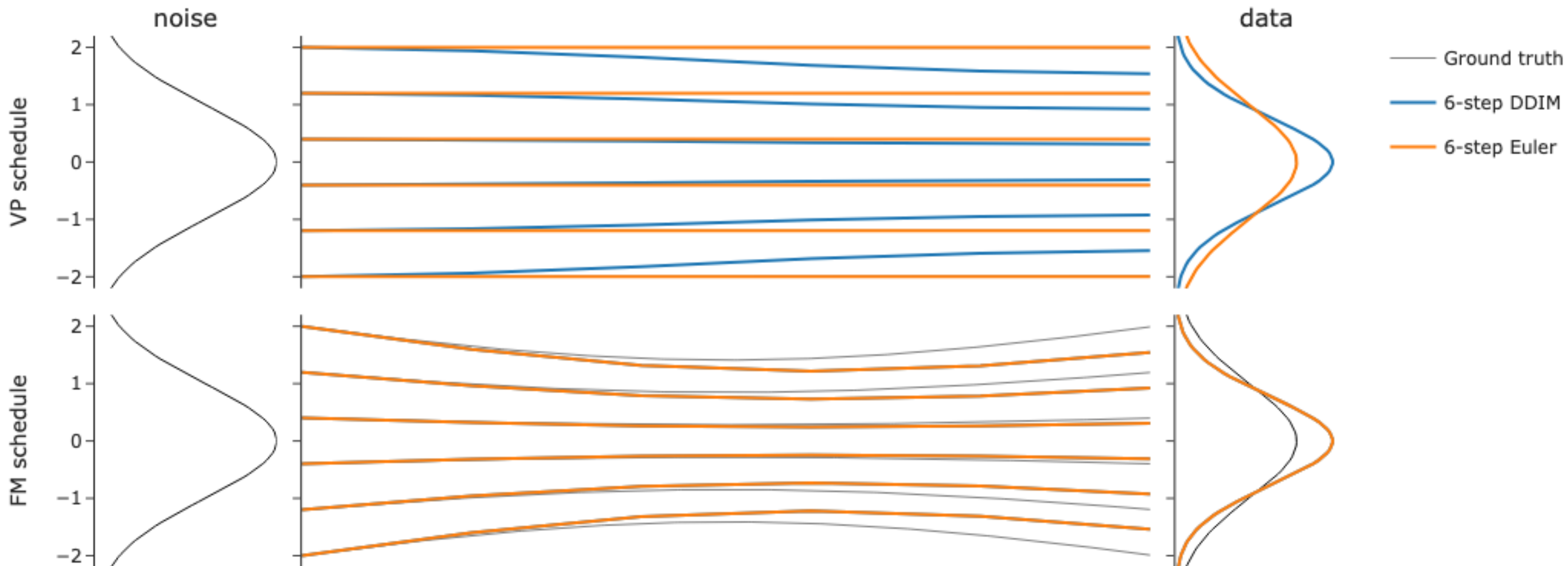
2. Sampling schedules

Variance Preserving vs Flow Matching schedules for varying data distributions

2. Sampling schedules

Variance Preserving vs Flow Matching schedules for varying data distributions

2. Sampling schedules

$$t_m = \frac{st_n}{1 + (s - 1)t_n}$$