Rekurrente Neuronale Netze

Rekurrente Netze: Abkühlungsgesetz

Ein Körper der Temperatur ϑ_0 wird in eine Umgebung der Temperatur ϑ_A eingebracht.

Die Abkühlung/Aufheizung des Körpers kann beschrieben werden durch das **Newtonsche Abkühlungsgesetz**:

$$\frac{\mathrm{d}\vartheta}{\mathrm{d}t} = \dot{\vartheta} = -k(\vartheta - \vartheta_A).$$

Exakte analytische Lösung:

$$\vartheta(t) = \vartheta_A + (\vartheta_0 - \vartheta_A)e^{-k(t-t_0)}$$

Ungefähre Lösung mit Hilfe des **Euler-Cauchyschen Polygonzuges**:

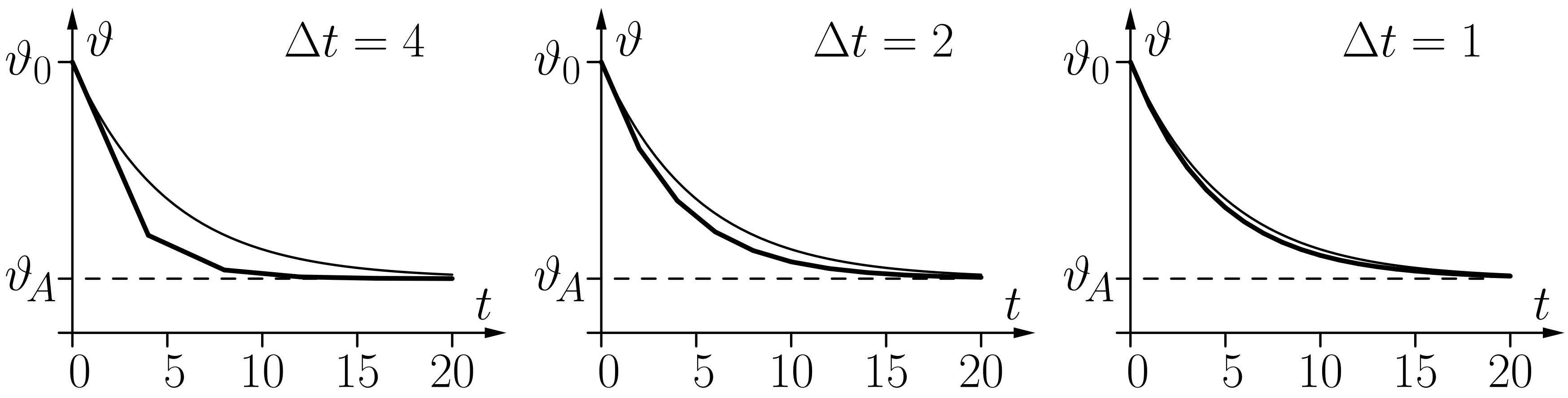
$$\vartheta_1 = \vartheta(t_1) = \vartheta(t_0) + \dot{\vartheta}(t_0)\Delta t = \vartheta_0 - k(\vartheta_0 - \vartheta_A)\Delta t.$$

$$\vartheta_2 = \vartheta(t_2) = \vartheta(t_1) + \dot{\vartheta}(t_1)\Delta t = \vartheta_1 - k(\vartheta_1 - \vartheta_A)\Delta t.$$

Allgemeine rekursive Gleichung:

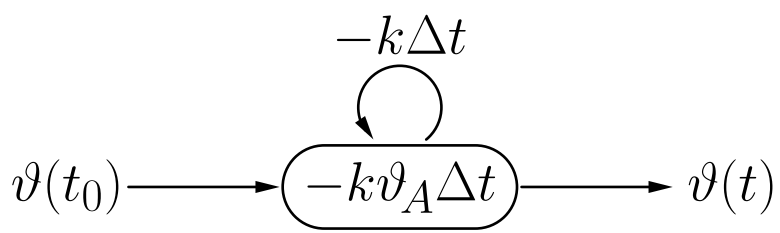
$$\vartheta_i = \vartheta(t_i) = \vartheta(t_{i-1}) + \dot{\vartheta}(t_{i-1})\Delta t = \vartheta_{i-1} - k(\vartheta_{i-1} - \vartheta_A)\Delta t$$

Euler–Cauchy-Polygonzüge für verschiedene Schrittweiten:



Die dünne Kurve ist die genaue analytische Lösung.

Rekurrentes neuronales Netz:



Rekurrente Netze: Abkühlungsgesetz

Formale Herleitung der rekursiven Gleichung:

Ersetze Differentialquotient durch **Differenzenquotient**

$$\frac{\mathrm{d}\vartheta(t)}{\mathrm{d}t} \approx \frac{\Delta\vartheta(t)}{\Delta t} = \frac{\vartheta(t+\Delta t) - \vartheta(t)}{\Delta t}$$

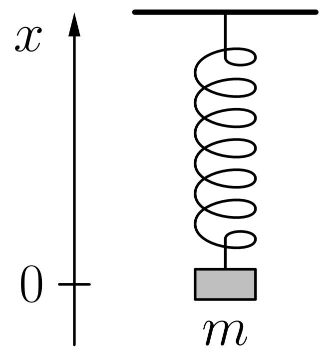
mit hinreichend kleinem Δt. Dann ist

$$\vartheta(t+\Delta t) - \vartheta(t) = \Delta\vartheta(t) \approx -k(\vartheta(t) - \vartheta_A)\Delta t,$$

$$\vartheta(t+\Delta t) - \vartheta(t) = \Delta\vartheta(t) \approx -k\Delta t\vartheta(t) + k\vartheta_A\Delta t$$

und daher

$$\vartheta_i \approx \vartheta_{i-1} - k\Delta t\vartheta_{i-1} + k\vartheta_A\Delta t.$$



Zugrundeliegende physikalische Gesetze:

- **Hooke'sches Gesetz**: $F = c\Delta l = -cx$ (c ist eine federabhängige Konstante)
- **Zweites Newton'sches Gesetz**: $F = ma = m\ddot{x}$ (Kraft bewirkt eine Beschleunigung)

Resultierende Differentialgleichung:

$$m\ddot{x} = -cx \qquad \text{oder} \qquad \ddot{x} = -\frac{c}{m}x.$$

Rekurrente Netze: Masse an einer Feder

Allgemeine analytische Lösung der Differentialgleichung:

$$x(t) = a \sin(\omega t) + b \cos(\omega t)$$

mit den Parametern

$$\omega = \sqrt{\frac{c}{m}}, \qquad \begin{aligned} a &= x(t_0) \sin(\omega t_0) \; + \; v(t_0) \cos(\omega t_0), \\ b &= x(t_0) \cos(\omega t_0) \; - \; v(t_0) \sin(\omega t_0). \end{aligned}$$

Mit gegebenen Initialwerten $x(t_0) = x_0$ und $v(t_0) = 0$ und
der zusätzlichen Annahme $t_0 = 0$ bekommen wir den einfachen Ausdruck

$$x(t) = x_0 \cos\left(\sqrt{\frac{c}{m}}\, t\right).$$

Rekurrente Netze: Masse an einer Feder

Wandle Differentialgleichung in zwei gekoppelte Gleichungen um:

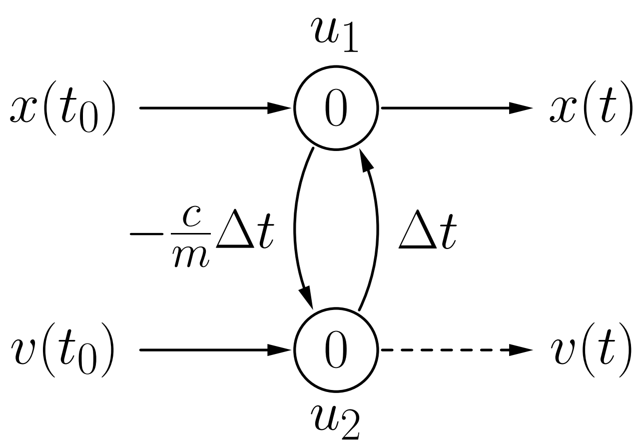
$$\dot{x} = v \qquad \text{and} \qquad \dot{v} = -\frac{c}{m}x.$$

Approximiere Differentialquotient durch Differenzenquotient:

$$\frac{\Delta x}{\Delta t} = \frac{x(t + \Delta t) - x(t)}{\Delta t} = v \qquad \text{and} \qquad \frac{\Delta v}{\Delta t} = \frac{v(t + \Delta t) - v(t)}{\Delta t} = -\frac{c}{m}x$$

Resultierende rekursive Gleichungen:

$$\begin{aligned} x(t_i) &= x(t_{i-1}) + \Delta x(t_{i-1}) &&= x(t_{i-1}) + \Delta t \cdot v(t_{i-1}) \qquad \text{und} \\ v(t_i) &= v(t_{i-1}) + \Delta v(t_{i-1}) &&= v(t_{i-1}) - \frac{c}{m}\Delta t \cdot x(t_{i-1}). \end{aligned}$$



Neuron u_1:

$$f_{\text{net}}^{(u_1)}(v, w_{u_1u_2}) = w_{u_1u_2}v = -\frac{c}{m}\Delta t\ v \qquad \text{und}$$

$$f_{\text{act}}^{(u_1)}(\text{act}_{u_1}, \text{net}_{u_1}, \theta_{u_1}) = \text{act}_{u_1} + \text{net}_{u_1} - \theta_{u_1},$$

Neuron u_2:

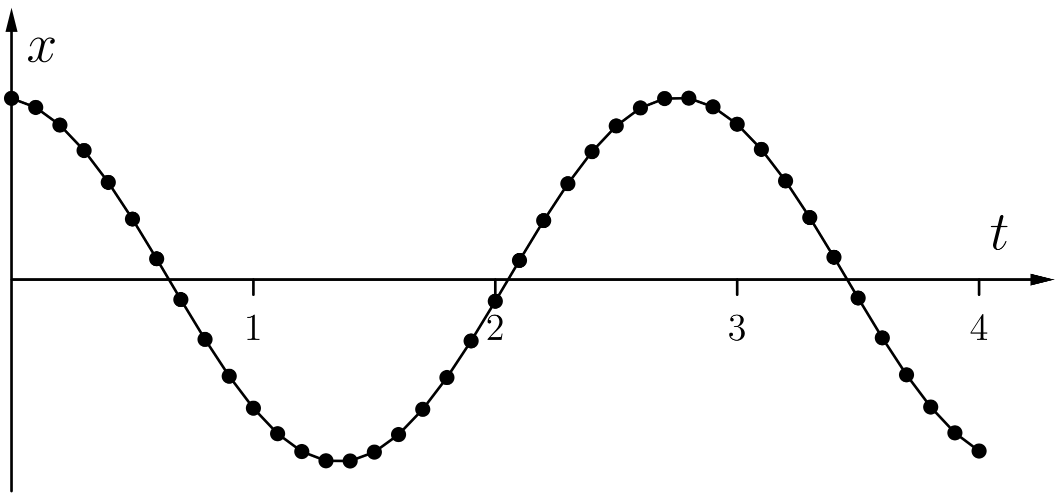
$$f_{\text{net}}^{(u_2)}(x, w_{u_2u_1}) = w_{u_2u_1}x = \Delta t\ x \qquad \text{und}$$

$$f_{\text{act}}^{(u_2)}(\text{act}_{u_2}, \text{net}_{u_2}, \theta_{u_2}) = \text{act}_{u_2} + \text{net}_{u_2} - \theta_{u_2}.$$

Rekurrente Netze: Masse an einer Feder

Einige Berechnungsschritte des neuronalen Netzes:

t	v	x
0.0	0.0000	1.0000
0.1	−0.5000	0.9500
0.2	−0.9750	0.8525
0.3	−1.4012	0.7124
0.4	−1.7574	0.5366
0.5	−2.0258	0.3341
0.6	−2.1928	0.1148



- Die resultierende Kurve ist nah an der analytischen Lösung.
- Die Annäherung wird mit kleinerer Schrittweite besser.

Allgemeine Darstellung expliziter Differentialgleichungen n-ten Grades:

$$x^{(n)} = f(t, x, \dot{x}, \ddot{x}, \ldots, x^{(n-1)})$$

Einführung von $n - 1$ Zwischengrößen

$$y_1 = \dot{x}, \qquad y_2 = \ddot{x}, \qquad \ldots \qquad y_{n-1} = x^{(n-1)}$$

Gleichungssystem

$$\begin{aligned} \dot{x} &= y_1, \\ \dot{y}_1 &= y_2, \\ &\vdots \\ \dot{y}_{n-2} &= y_{n-1}, \\ \dot{y}_{n-1} &= f(t, x, y_1, y_2, \ldots, y_{n-1}) \end{aligned}$$

von n gekoppelten Differentialgleichungen ersten Grades.

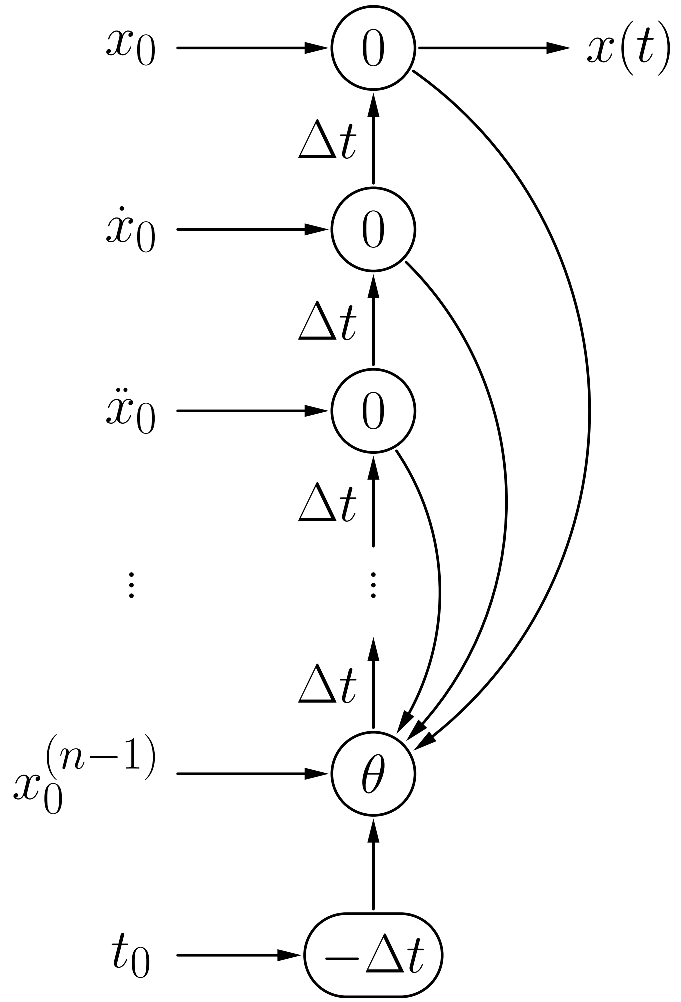
Rekurrente Netze: Differential Equations

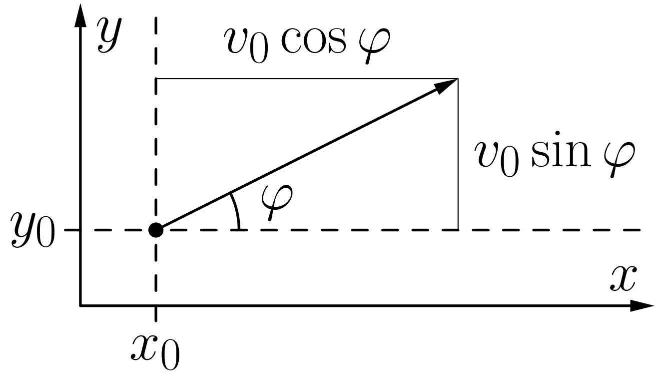
Ersetze Differentialquotient durch Differenzenquotient, um die folgenden rekursiven Gleichungen zu erhalten:

$$
\begin{aligned}
x(t_i) &= x(t_{i-1}) + \Delta t \cdot y_1(t_{i-1}), \\
y_1(t_i) &= y_1(t_{i-1}) + \Delta t \cdot y_2(t_{i-1}), \\
&\vdots \\
y_{n-2}(t_i) &= y_{n-2}(t_{i-1}) + \Delta t \cdot y_{n-3}(t_{i-1}), \\
y_{n-1}(t_i) &= y_{n-1}(t_{i-1}) + f(t_{i-1}, x(t_{i-1}), y_1(t_{i-1}), \ldots, y_{n-1}(t_{i-1}))
\end{aligned}
$$

- Jede dieser Gleichungen beschreibt die Aktualisierung eines Neurons.
- Das letzte Neuron benötigt eine spezielle Aktivierungsfunktion.

Rekurrente Netze: Differentialgleichungen





Schräger Wurf eines Körpers.

Zwei Differentialgleichungen (eine für jede Koordinatenrichtung):

$$\ddot{x} = 0 \qquad \text{und} \qquad \ddot{y} = -g,$$

wobei $g = 9.81\,\mathrm{ms}^{-2}$.

Anfangsbedingungen $x(t_0) = x_0$, $y(t_0) = y_0$, $\dot{x}(t_0) = v_0 \cos\varphi$ und $\dot{y}(t_0) = v_0 \sin\varphi$.

Rekurrente Netze: Schräger Wurf

Führe Zwischenbedingungen ein:

$$v_x = \dot{x} \qquad \text{und} \qquad v_y = \dot{y}$$

um das System der folgenden Differentialgleichungen zu erhalten:

$$\begin{aligned} \dot{x} &= v_x, & \qquad \dot{v}_x &= 0, \\ \dot{y} &= v_y, & \qquad \dot{v}_y &= -g, \end{aligned}$$

aus dem wir das System rekursiver Anpassungsformeln erhalten

$$\begin{aligned} x(t_i) &= x(t_{i-1}) + \Delta t\ v_x(t_{i-1}), & \qquad v_x(t_i) &= v_x(t_{i-1}), \\ y(t_i) &= y(t_{i-1}) + \Delta t\ v_y(t_{i-1}), & \qquad v_y(t_i) &= v_y(t_{i-1}) - \Delta t\ g. \end{aligned}$$

Rekurrente Netze: Schräger Wurf

Bessere Beschreibung: Benutze **Vektoren** als Eingaben und Ausgaben

$$\ddot{\boldsymbol{r}} = -g\boldsymbol{e}_y,$$

wobei $\boldsymbol{e}_y = (0, 1)$.

Anfangsbedingungen sind $\boldsymbol{r}(t_0) = \boldsymbol{r}_0 = (x_0, y_0)$ und $\dot{\boldsymbol{r}}(t_0) = \boldsymbol{v}_0 = (v_0 \cos\varphi, v_0 \sin\varphi)$.

Führe eine **vektorielle** Zwischengröße $\boldsymbol{v} = \dot{\boldsymbol{r}}$ ein, um

$$\dot{\boldsymbol{r}} = \boldsymbol{v}, \qquad \dot{\boldsymbol{v}} = -g\boldsymbol{e}_y$$

zu erhalten.

Das führt zu den rekursiven Anpassungsregeln

$$\begin{aligned}
\boldsymbol{r}(t_i) &= \boldsymbol{r}(t_{i-1}) + \Delta t\ \boldsymbol{v}(t_{i-1}), \\
\boldsymbol{v}(t_i) &= \boldsymbol{v}(t_{i-1}) - \Delta t\ g\boldsymbol{e}_y
\end{aligned}$$

Die Vorteile vektorieller Netze werden offensichtlich, wenn Reibung mit in Betracht gezogen wird:

$$\boldsymbol{a} = -\beta \boldsymbol{v} = -\beta \dot{\boldsymbol{r}}$$

β ist eine Konstante, die von Größe und Form des Körpers abhängt.
Dies führt zur Differentialgleichung

$$\ddot{\boldsymbol{r}} = -\beta \dot{\boldsymbol{r}} - g\boldsymbol{e}_y.$$

Führe die Zwischengröße $\boldsymbol{v} = \dot{\boldsymbol{r}}$ ein, um

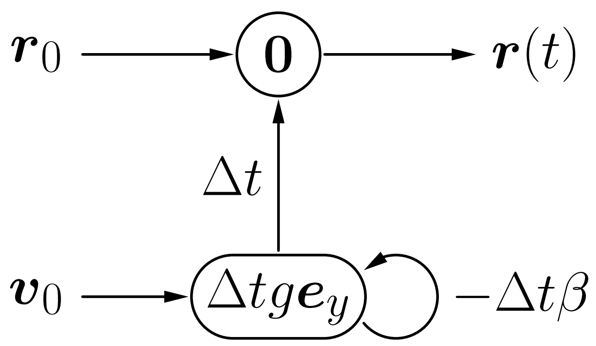
$$\dot{\boldsymbol{r}} = \boldsymbol{v}, \qquad \dot{\boldsymbol{v}} = -\beta \boldsymbol{v} - g\boldsymbol{e}_y,$$

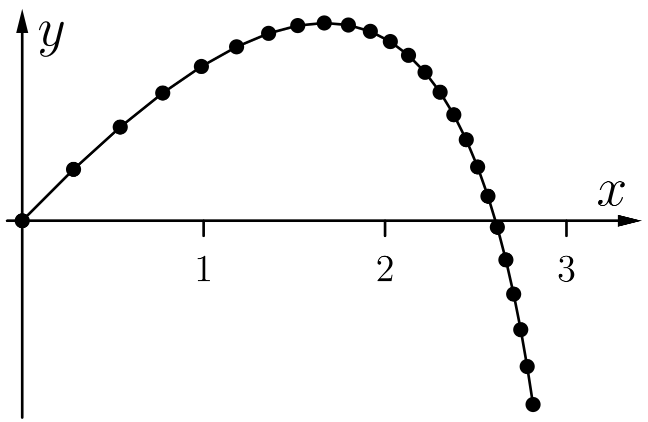
zu erhalten,

woraus wir die folgenden rekursiven Anpassungsformeln bekommen:

$$\begin{aligned} \boldsymbol{r}(t_i) &= \boldsymbol{r}(t_{i-1}) + \Delta t\, \boldsymbol{v}(t_{i-1}), \\ \boldsymbol{v}(t_i) &= \boldsymbol{v}(t_{i-1}) - \Delta t\, \beta\, \boldsymbol{v}(t_{i-1}) - \Delta t\, g\boldsymbol{e}_y. \end{aligned}$$

Sich ergebendes rekurrentes neuronales Netz:





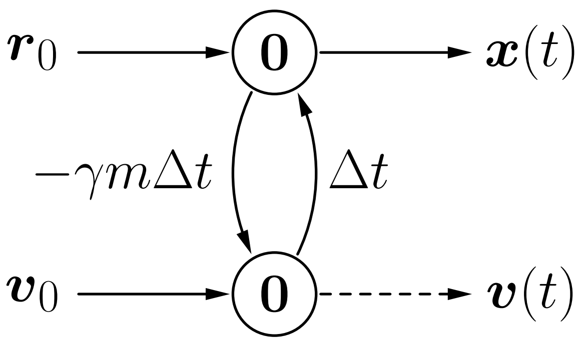
- Es gibt keine “seltsamen” Kopplungen wie in einem nicht-vektoriellen Netz.
- Man beachte die Abweichung von der Parabel, die durch die Reibung bewirkt wird.

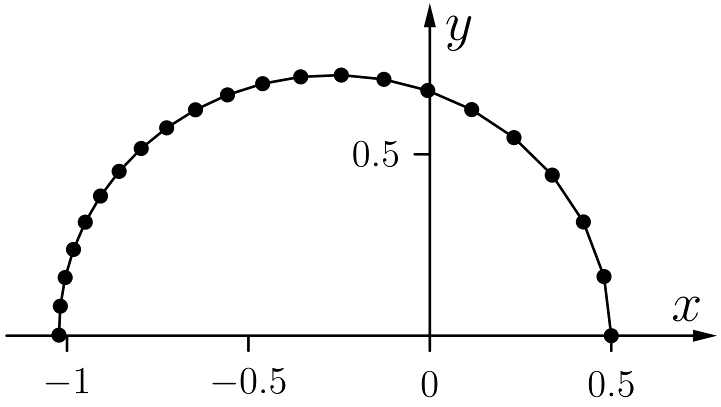
Rekurrente Netze: Umlaufbahnen der Planeten

$$\ddot{\boldsymbol{r}} = -\gamma m \frac{\boldsymbol{r}}{|\boldsymbol{r}\,|^3} \qquad \Rightarrow \qquad \dot{\boldsymbol{r}} = \boldsymbol{v} \qquad \dot{\boldsymbol{v}} = -\gamma m \frac{\boldsymbol{r}}{|\boldsymbol{r}\,|^3}$$

Rekursive Anpassungsregeln:

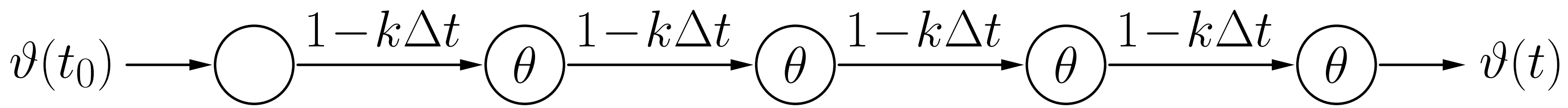
$$\begin{aligned}
\boldsymbol{r}(t_i) &= \boldsymbol{r}(t_{i-1}) + \Delta t \; \boldsymbol{v}(t_{i-1}) \\
\boldsymbol{v}(t_i) &= \boldsymbol{v}(t_{i-1}) - \Delta t \; \gamma m \frac{\boldsymbol{r}(t_{i-1})}{|\boldsymbol{r}(t_{i-1})|^3}
\end{aligned}$$





Idee: Entfalte das Netzwerk zwischen Trainingsmustern,
d.h. lege ein Neuron für jeden Zeitpunkt an.

Beispiel: **Newton'sches Abkühlungsgesetz**



Entfalten in vier Schritten. Es ist $\theta = -k\vartheta_A \Delta t$.

- Training: Standard-Backpropagation im entfalteten Netzwerk.
- Alle Anpassungen beziehen sich auf dasselbe Gewicht.
- Anpassungen werden ausgeführt, wenn das erste Neuron erreicht wird.