Aktives Lernen in der sensormotorischen Schleife

Georg Martius

Institut für Informatik, Universität Leipzig

23. Februar 2005

Inhaltsverzeichnis

Backward Modelling

Lernaufgabe

- Lernaufgabe: Gutes mapping von Eingängen $x \in \mathcal{X}$ zu Ausgängen $y \in \mathcal{Y}$
- Minimierung der einer Fehlerfunktion. Z.B.:

$$E = \sum_{i \in I} (y_i - \hat{y}_i)^2,$$

wobei $\hat{y}_i = \psi(x_i)$ und I Indexmenge aller Paare (x_i, y_i)

Motivation/Ziele

- Situated Robotics: Cognitives Bootstrapping Problem
  Bei NN: kleine Gewichtsinitialisierung
- Homeokinesis: vorwärts und **rückwärts** in der Zeit
- Spontane Symmetriebrechung
  - Roboter bewegt sich im freien Raum: Sensorraum invariant
  - Brechung von Symmetrie in der Umgebung hilfreich (auch bei klassischen Lernaufgaben)
- Anwendbarkeit auf Neuronale Netze (Feed-forward)
- Tiefe Netzwerke
- Biologisch plausibel (zumindestes nur lokales Wissen)

Idee: Invertierung

- Normal: Ermittelung des Ausgangs bei gegebenem Eingang
- Jetzt: Ermittelung **eines** Eingangs bei gegebenem Ausgang
- Im Allgemeinen: stark unterbestimmtes Problem.
- Man erhält **virtuelle Eingänge**.
- Lernen mit virtuellen Eingängen.

Berechnung virtueller Eingänge

Finde: $\hat{x}$ für das gilt: $\psi(\hat{x}) = \psi(x) + \xi$

Betrachte shift: $\hat{x} = x + v$

Gradientenabstieg auf Fehler F:

$$F(v) = ||y - \psi(x+v)||^2$$

$$= \epsilon \frac{\partial\, \psi(\hat{x})}{\partial x} \cdot (y - \underbrace{\psi(x+v)}_{\hat{y}_v})^T$$

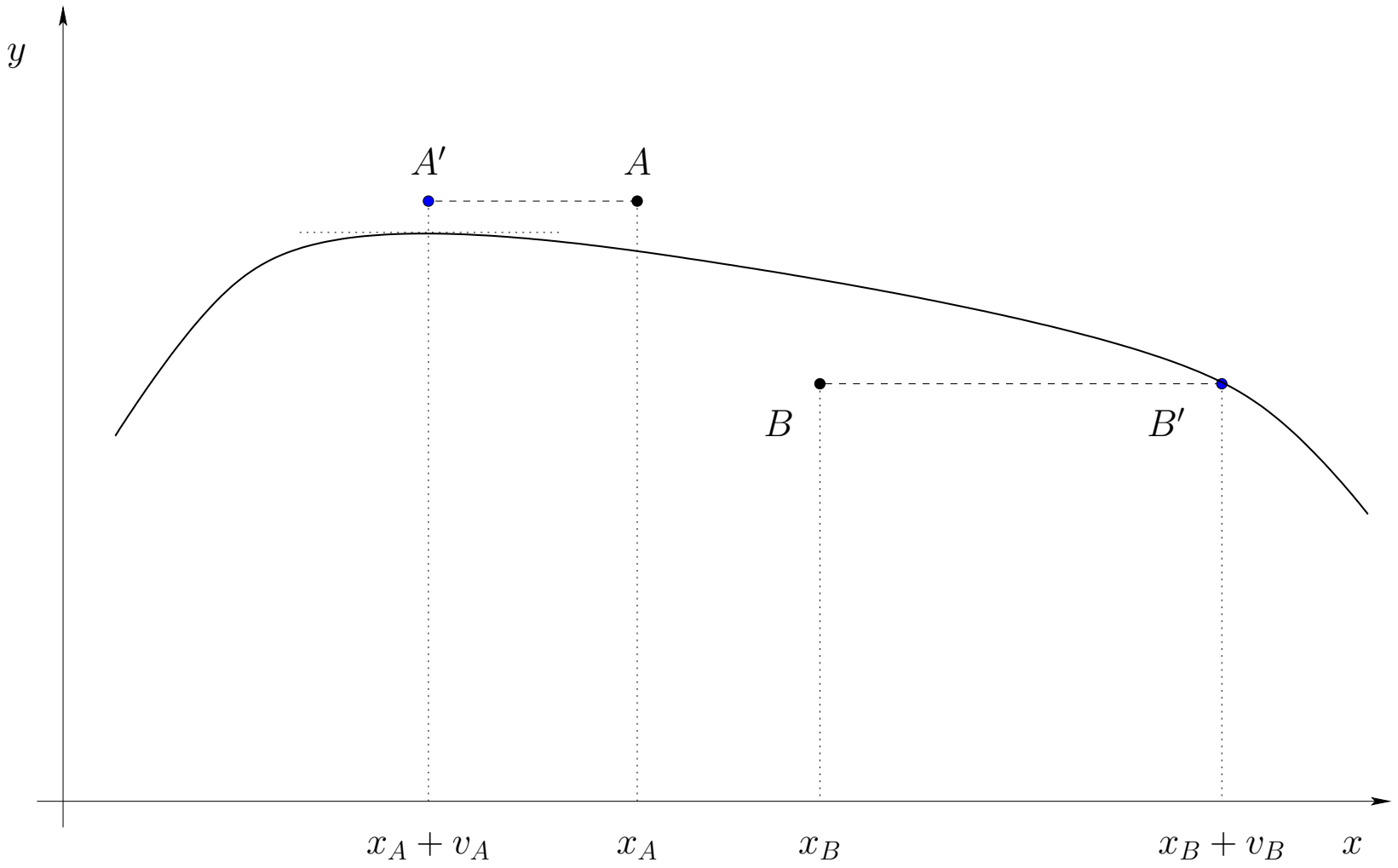


Abbildung 1: Reale und virtuelle Datenpunkte. A' erreicht Extrema, B' liegt auf Modell

Partielle Shifts

Vollständige Shifts bringen kein Lernsignal, da $y - \psi(\hat{x}) = 0$.

- Penalty Term λ:

$$F_\lambda(v) = ||y - \psi(x + v)||^2 + \lambda||v||^2$$

- Partial shift ratio γ: Beende Gradientenabstieg wenn:

$$\frac{F(v)}{F(0)} \leq \gamma$$

  Vorteil: schneller, erwünschtere Shifts und einfacher zu implementieren

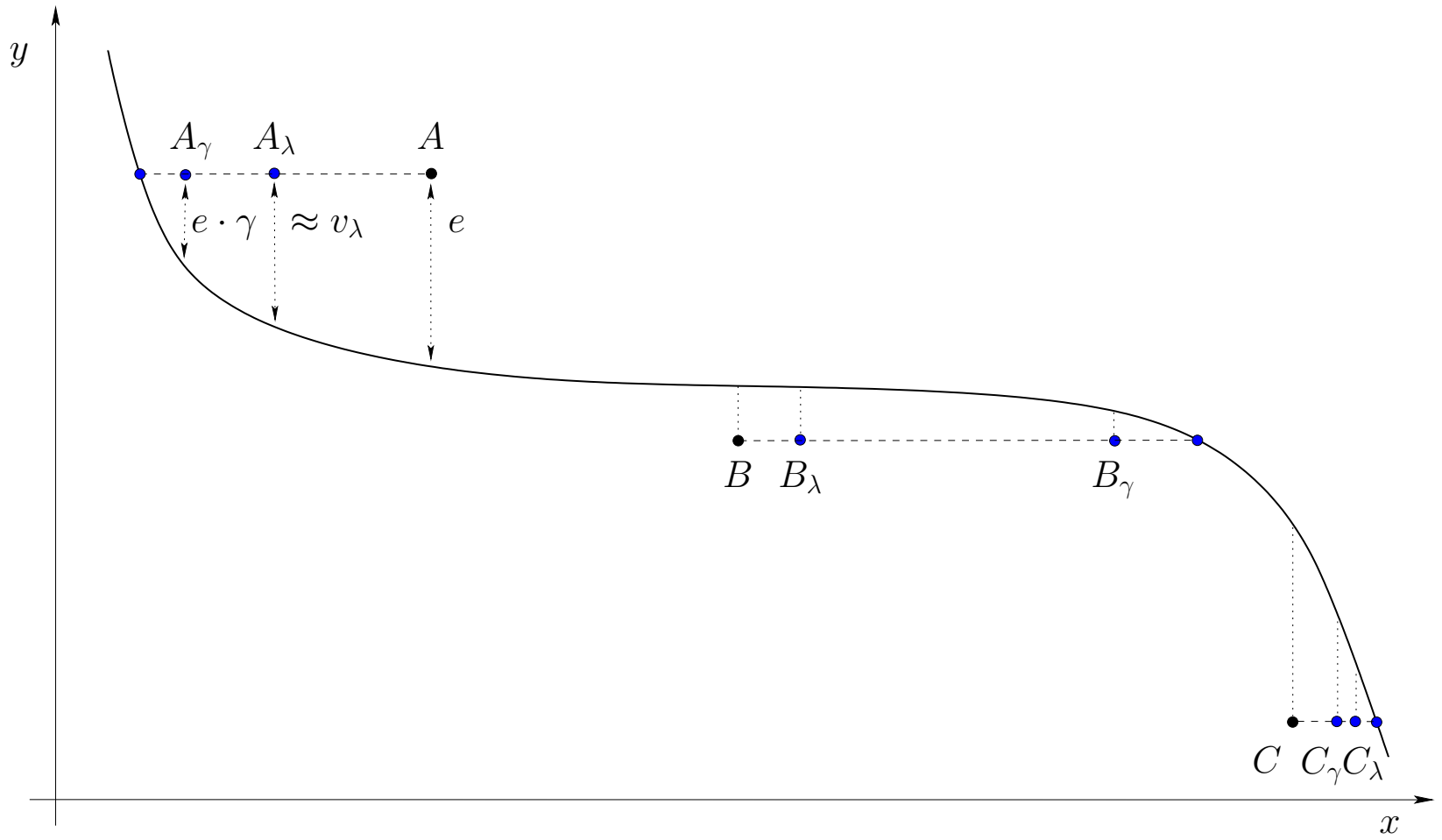


Abbildung 2: Reale und virtuelle Datenpunkte. Die Linie ist das Modell. A, B, C: Datenpunkte; X_λ: partieller Shift mit Penalty Term; X_γ partieller Shift mit Abbruchkriterium

Neuronale Netze

Shiftberechnung: δ-Werte von Backpropagation als Gradient.

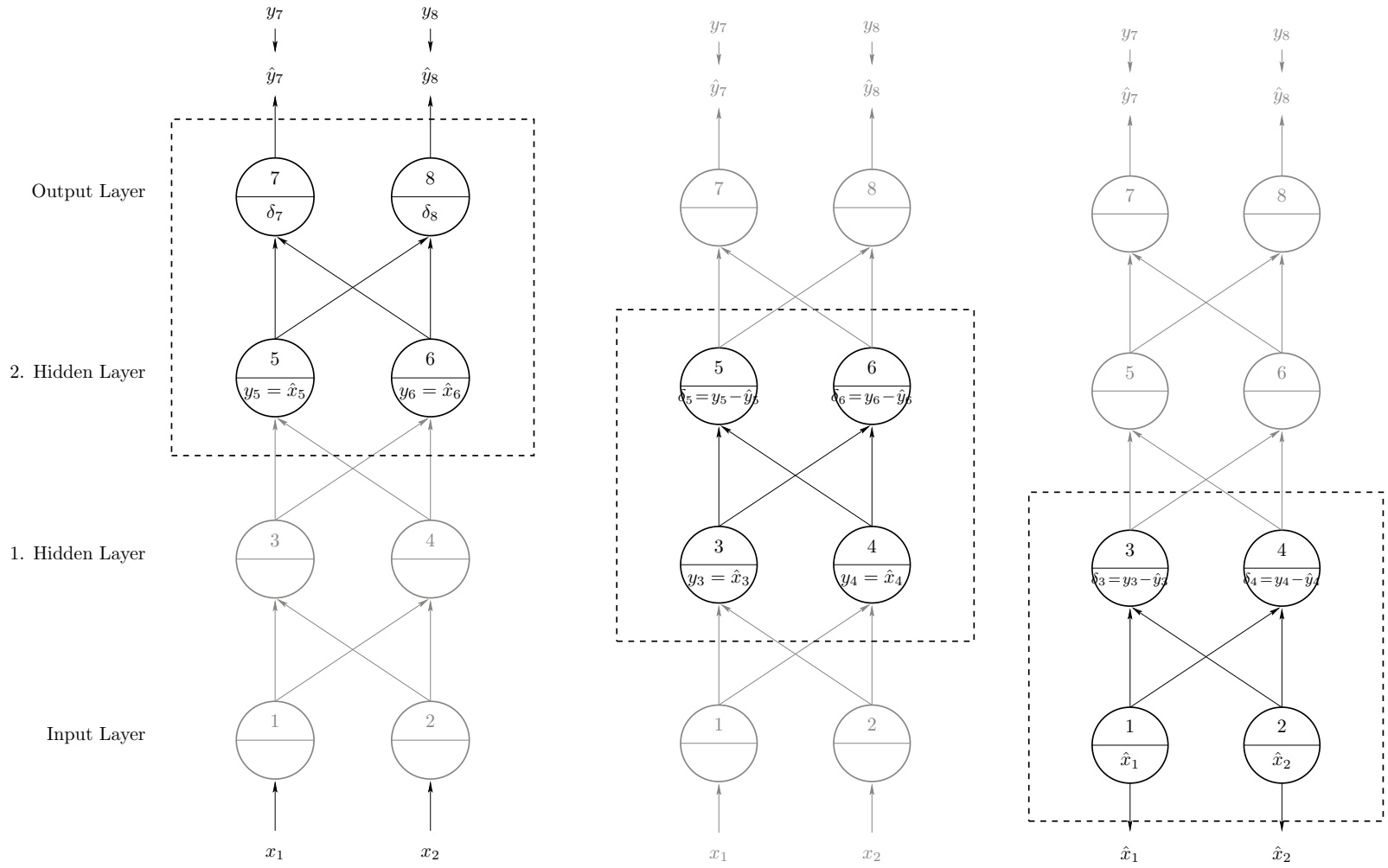
- Rückpropagieren der δ-Werte bis in die Eingabe Schicht.

$$\delta_j = \begin{cases} g'(\hat{y}_j)(y_j - \hat{y}_j) & \text{für } j \text{ Ausgabenneuron,} \\ g'(\hat{y}_j) \sum_k \delta_k w_{jk} & \text{sonst} \end{cases}$$

- Iteration zur Bestimmung von v:

$$v_i^{t+1} = v_i^t + \epsilon \delta_i$$

Schichtweise



After forward activation, calc shift in highest subnetwork | Virtual input on second subnetwork is calculated | Virtual inputs reach input layer

Abbildung 3: Skizzierung des schichtweise Backward Modelling Verfahrens

Effekte

Response Increasing & Spontane Symmetriebrechung

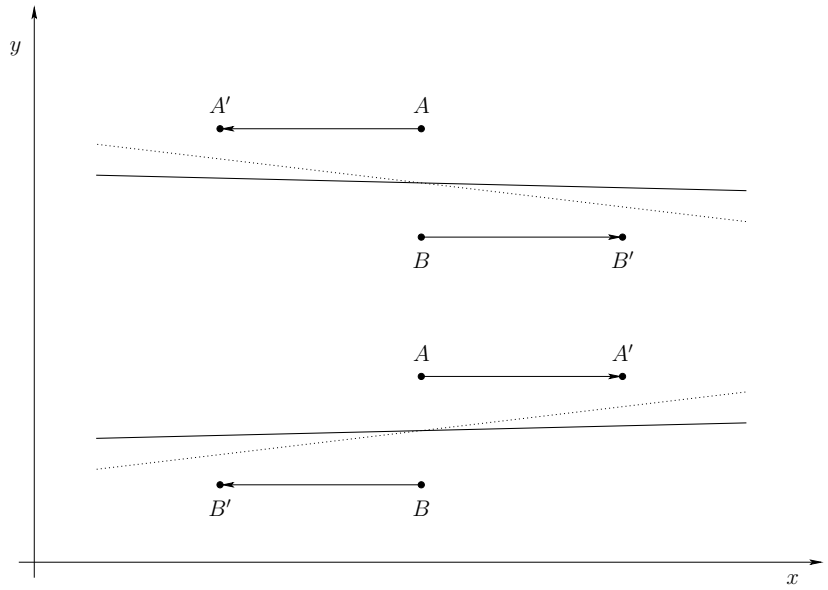


Abbildung 4: Spontane Symmetriebrechung

Tiefe Netwerke & Kleine Initialisierung

- Backpropagation:
  - δ-Werte verschwinden
  - Gewichtsupdate minimal: $\Delta w_{ij} = \eta\ \hat{y}_i \delta_j$
- Backward Modelling:
  - Virtuelle Eingänge sorgen für höhere Aktivierung ($\hat{y}$)
  - Schichtweise Verfahren: große δ-Werte und große Aktivierung

Algorithmen

- BackModelGiant: erzeugt Shifts in Eingabeschicht
- BackModelSim: gleichzeitiges Ändern der Gewichte und der Shifts (nicht erfolgreich)
- BackModel: erzeugt Shifts in jeder Schicht
  - Netzwerk wird in aufeinanderforlgende Subnetzwerke zerlegt
  - Zwischenschichten arbeiten nur noch mit virtuellen Sollwerten

Schrittweitensteuerung sorgt fuer schnelle Konvergenz des Gradientenabstiegs.

Ergebnisse

XOR und n- Parity

- n-Parity: Verallgemeinung von XOR
- Hochsymmetrisches Klassifikationsproblem.

Partial shift ratio

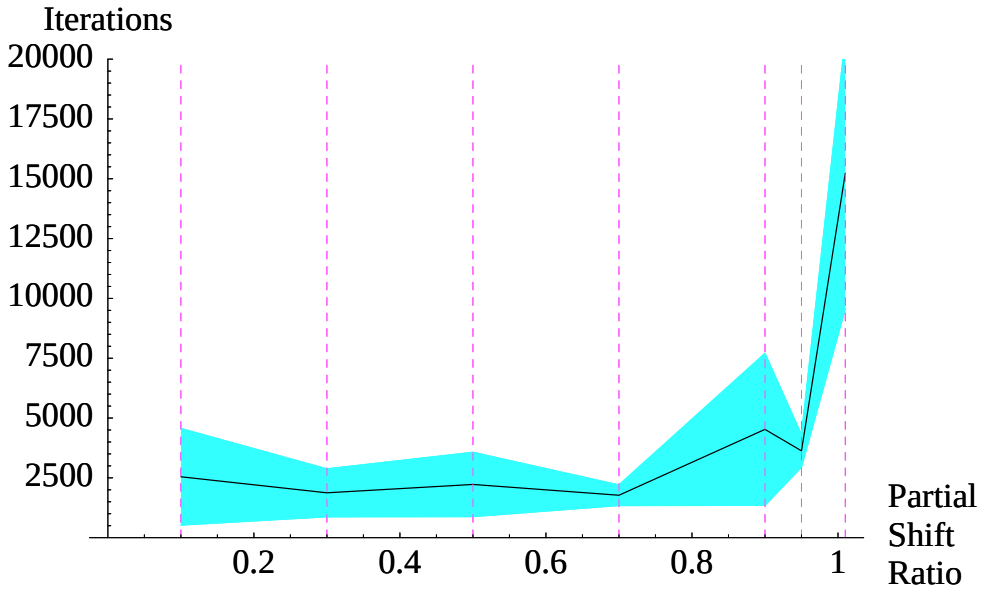


Abbildung 5: BackModel partial shift ratio (Init=0.01, $\eta > 0.1$)

BackProp vs. BackModel

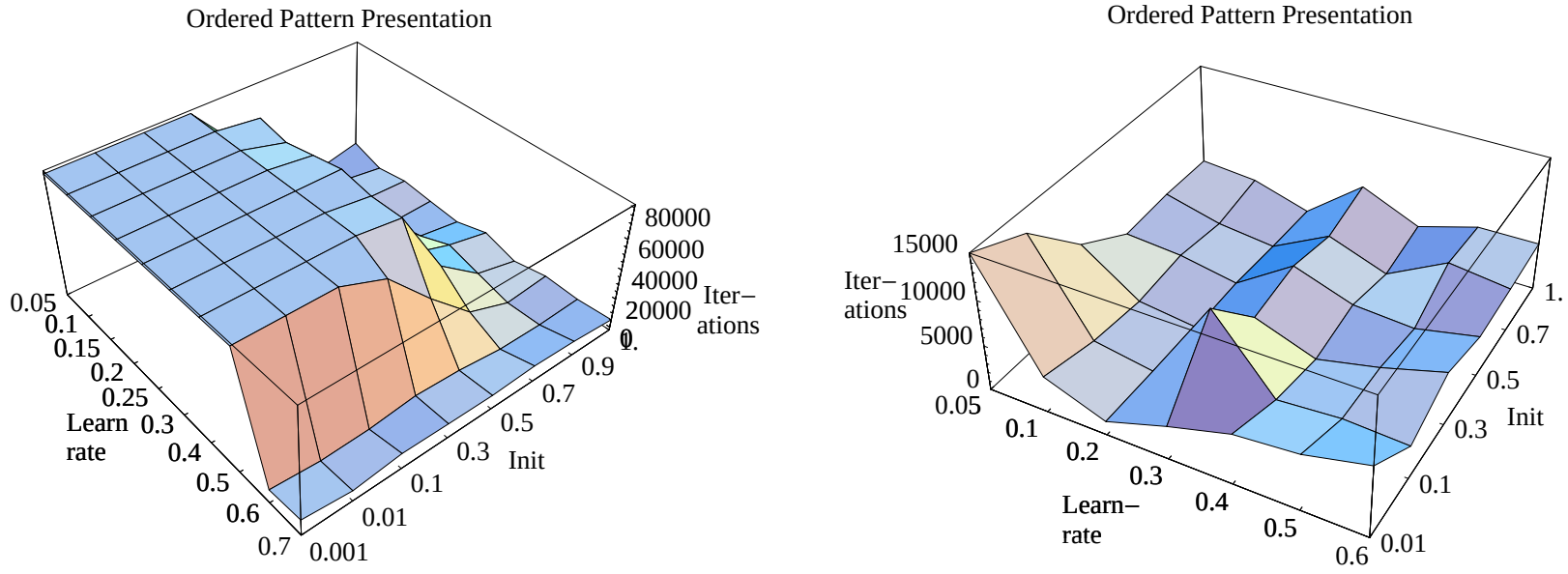


Abbildung 6: Lernrate-Init-Iterationen, 4-Parity mit 6 versteckten Neuronen. Links BackProp; Rechts BackModel

Encoder 10-5-10

- Lernverhalten auf vielschichtige Netzwerken üeberprüfen

Drei Versteckte Schichten

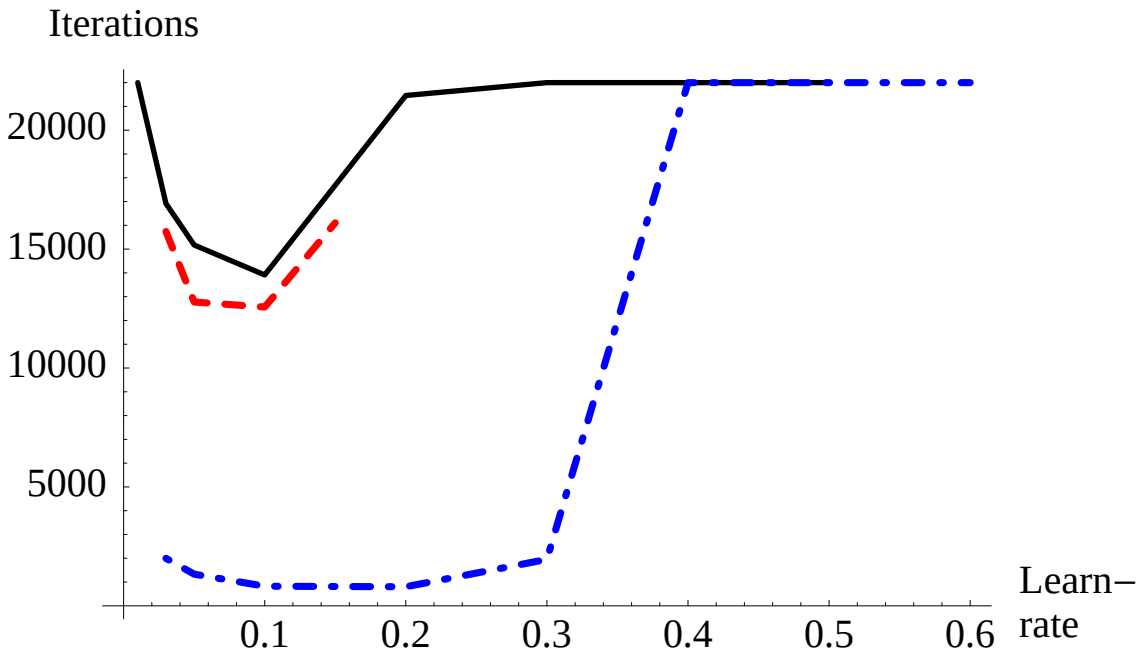


Abbildung 7: Lernrate-Iternationen. Schwarz: BACKPROP; Red: BACKMODELGIANT; Blau: BACKMODEL (Init < 0.1)

h Versteckte Schichten

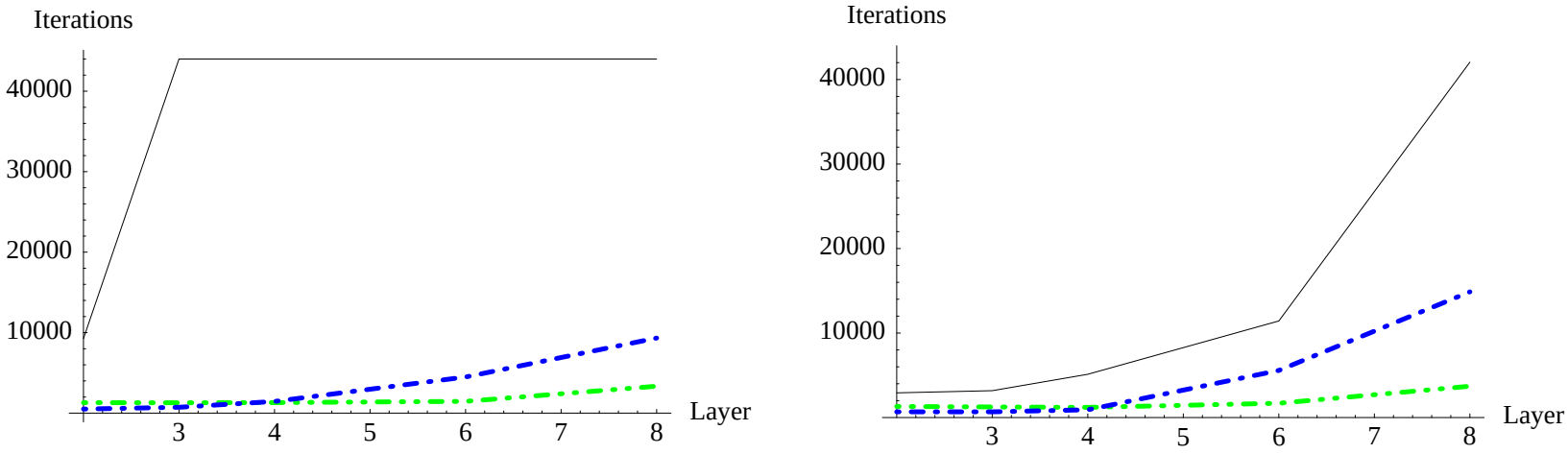


Abbildung 8: h-Iterationen, Encoder 10-5-10 mit h versteckten Schichten. Schwarz: BACKPROP; Blau: BACKMODEL; Grün: BACKMODELFULLBP Links: Niedige Initialisierung; Rechts: Hohe InitialisierungBACKMODEL

Aktives Lernen

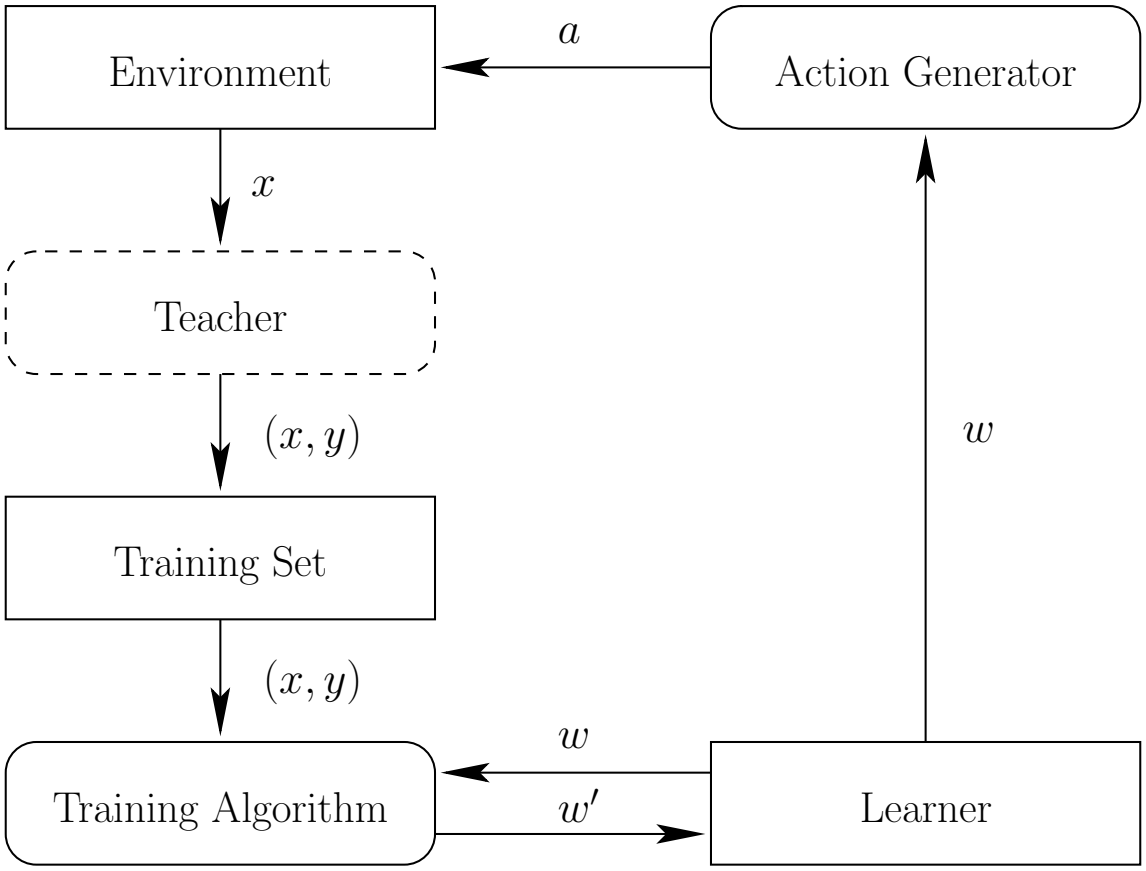


Abbildung 9: Architektur für aktives Lernen

Aktives Lernen mit Backward Modelling

Generell zwei Ansätze:

- Selektion: Zusammenstellung von Trainingsmengen durch Maß der Informativität
  - Größe des Shifts als Maß
- Queries: Anfrage von Pattern
  - Virtuelle Eingänge als Query

Sensormotorische Schleife

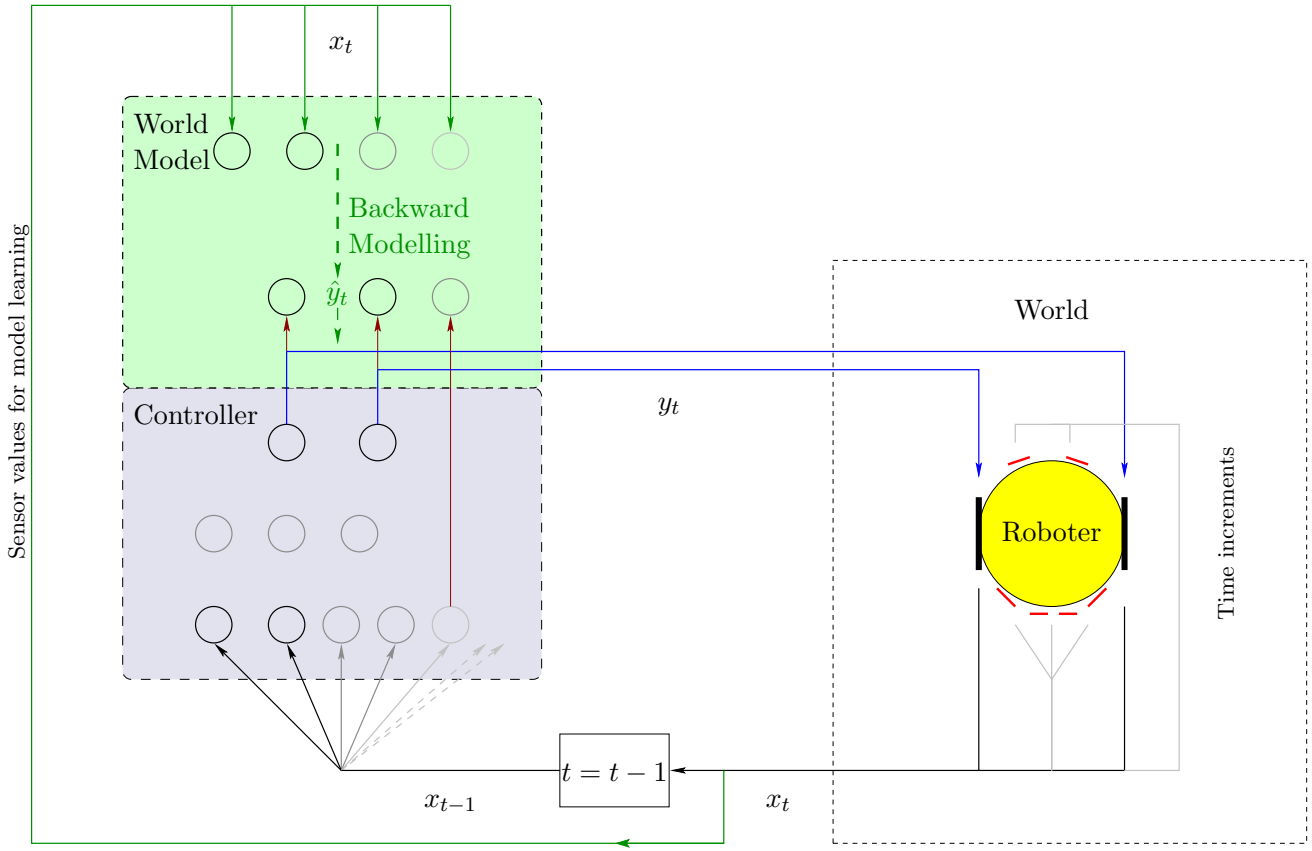


Abbildung 10: Sensormotorische Schleife mit Weltmodell und Backward Modelling

Modell generiert Kontroller-Werte

- Backward Modelling im Weltmodell $\Longrightarrow$ virtuelle Kontroller-Werte
- Strategien zur Auswahl der letztlich geltenden Aktuatoren:
  - gewichtetes Mittel: $y = y_K \cdot f + \hat{y} \cdot (1 - f)$
  - Abstandsschwellwert th:

$$y = \begin{cases} y_K & \text{wenn } v < th \\ \hat{y} & \text{wenn } v >= th \end{cases}$$

Beispiel: *Turni*

Szenario:

- Sehr einfacher Roboter:
  - Aktuatoren: zwei Motoren: rechtes und linkes Rad
  - Sensoren: zwei Radsensoren: Radgeschwindigkeit
  - Sensorspace = Actuatorspace
  - $x^{t+1} = A \cdot y^t + \xi \qquad A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
- Kontroller:
  - Lässt den Roboter nur im Kreis fahren.
  - $y_1 = -y_2$, $y_1^t = \sin(\omega\, t)$

Modellparameter

W sind die Verbindungen innerhalb des Modells.

$$x_{t+1} = y \cdot W$$

$$W = \begin{pmatrix} w_{13} & w_{23} \\ w_{14} & w_{24} \end{pmatrix}$$

Richtiges Modell: Degeneriertes Modell:

$$W = A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad W = \tfrac{1}{2} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

Aktives Lernen bemerkt Singularitäten im Modell

- Degeneriertem Modell ist nahe der Singularität
- Rauschen ξ der Sensoren sorgt für Modellierungsfehler.
- Betrachte Fehler-Komponente $\xi_\perp$ die senkrecht auf $\binom{1}{-1}$ steht.
- Erzeugt großen Shift in Richtung $\binom{1}{1}$
- $\Longrightarrow$ Modell wird korrigiert

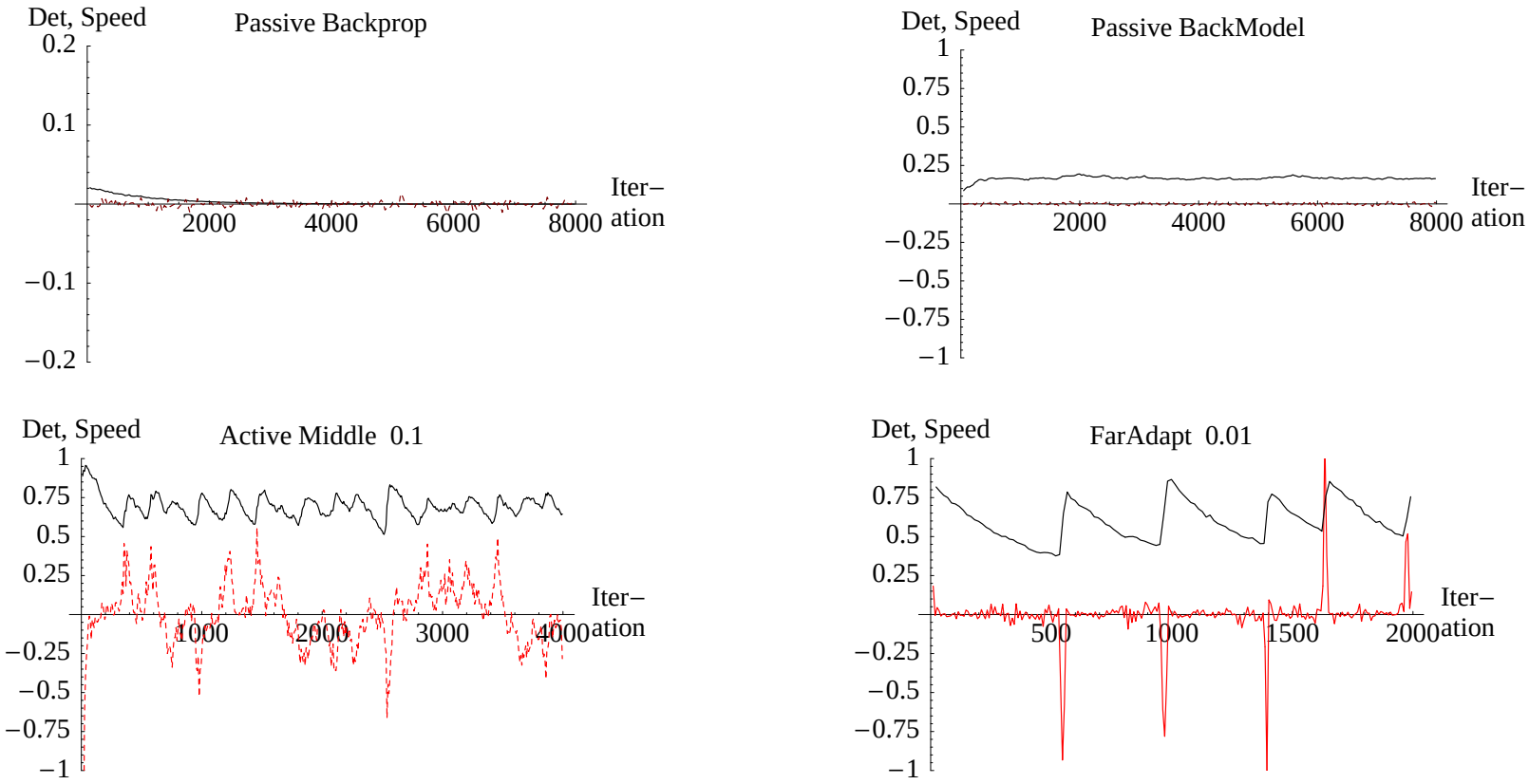


Abbildung 11: Verhalten des Roboters

Zusammenfassung

- Backward Modelling:
  - allgemeines Verfahren
  - anwendbar auf NN
  - ermöglicht direkte Queries für Aktives Lernen
- Sensormotorische Schleife:
  - Backward Modelling im Modell erzeugt virtuelle Kontroller-Werte
  - Active Exploration
  - verbessert degenerierte Weltmodelle
  - Zielanwendung: Cognitives Bootstrapping, Homeokinesis