

Trajectory-Generation mit Reinforcement Learning

Tim Büttel, Jonas Peltzer, Pierre Pospiech, Jannik Schremmer



Idee

- ☞ Ist es möglich Reinforcement Learning und eine Trajektorien-Generierung zu kombinieren, um zufälligen Hindernissen auszuweichen?
- ☞ Im Rahmen dieser Arbeit soll ein System implementiert und getestet werden, welches ein Arm-Modell erzeugt, das dem menschlichen Arm gleicht und auf Basis des Reinforcement-Learnings seinen Weg bestimmt um ein Ziel zu erreichen und dabei Hindernissen auszuweichen.
- ☞ Dazu soll eine geeignete Trajektorie erstellt werden, dessen Parameter vom Reinforcement-Learning bestimmt werden. Mit dem bestimmten Arm-Modell soll über einen Algorithmus die inverse Kinematik bestimmt und eine Kollisionserkennung zwischen Hindernissen und Arm durchgeführt werden.
- ☞ Aus dem erfolgreichen Erreichen des Ziels und dem Ausweichen der Hindernisse soll ein Reward ermittelt werden, durch den das neuronale Netz mithilfe eines Reinforcement Learning Algorithmus trainiert wird.

Entwicklungsansatz

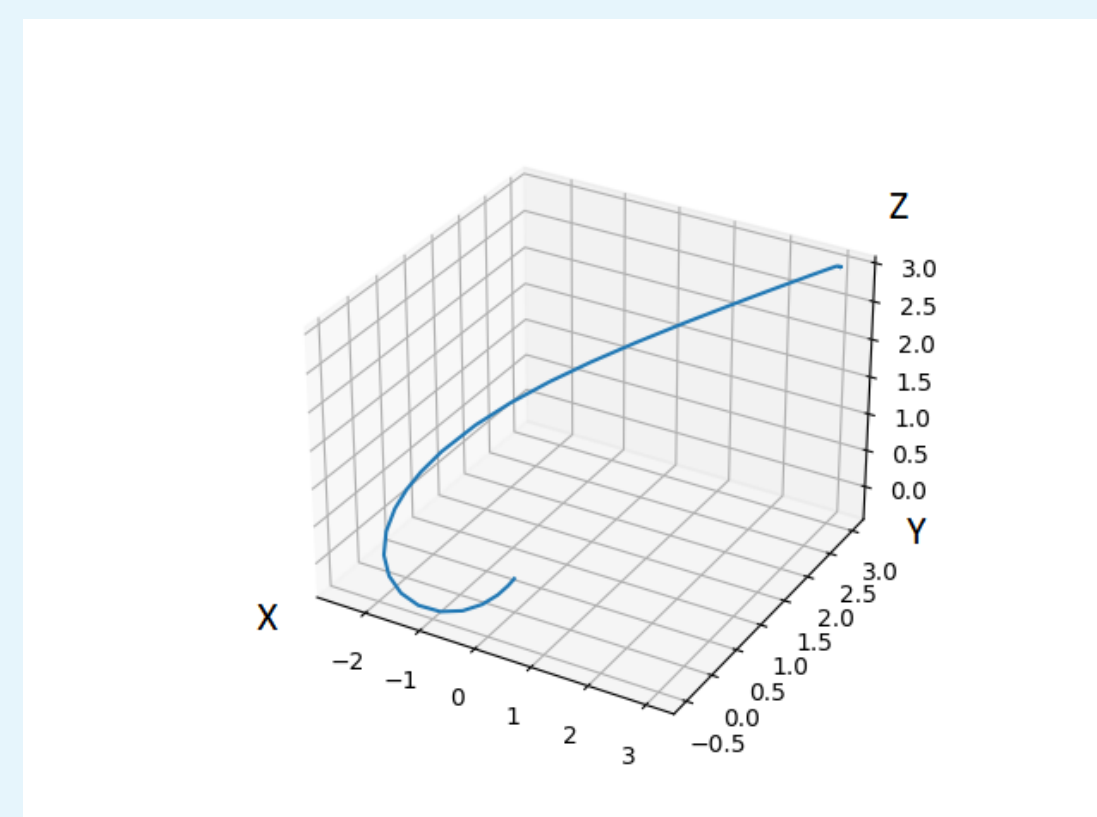
- ☞ Trajektoriengenerierung basierend auf "Modeling Human Arm Motion by Means of Attractor Dynamics Approach"
- ☞ Inverse Kinematik eines 4 DOF Arm berechnen
- ☞ Kollisionserkennung zwischen Hindernissen und Arm
- ☞ Parameter durch Reinforcement lernen lassen

Reinforcement Learning

Der genutzte REINFORCE-Algorithmus ist eine einfache Implementierung der Policy-Gradient-Methode. Eine Policy-Gradient-Methode ist ein Ansatz des Reinforcement Learning, bei dem versucht wird, eine Policy direkt zu erlernen, indem beispielsweise ein neuronales Netz als Policy-Funktion verwendet und so trainiert wird, dass es die Wahrscheinlichkeit von Aktionen auf der Grundlage der beobachteten Belohnungen erhöht.

Trajektorien-Generierung

Zur Trajektoriengenerierung wurde der Ansatz aus dem Paper „Modeling Human Arm Motion through the Attractor Dynamics Approach“ übernommen. Bei dieser Methode der Trajektoriengenerierung nähert sich der Endeffektor durch einen Attraktor dem Ziel an. Diese Methode hat sich im Paper als sehr menschennahe Bewegung des Endeffektors erwiesen, weshalb wir uns für diese entschieden haben.



Inverse Kinematik

Bei diesem Ansatz wird zunächst der Winkel des Ellenbogens berechnet. Damit können dann alle möglichen Positionen auf einer Kreisbahn mit einer gegebenen Auflösung erfasst werden. Mit den nun gegebenen potenziellen Ellenbogenpositionen können die restlichen Gelenkparameter berechnet werden.

$$\begin{aligned}\theta_2 &= \text{atan2}(r_{u3}, -r_{u1}) \\ \theta_3 &= \text{atan2}(\sqrt{r_{u1}^2 + r_{u3}^2}, r_{u2}) \\ c_4 &= \frac{-r_{f2}/L_2 - c_3 * s_5}{s_3 * c_5} \\ s_4 &= \frac{r_{f1}/L_2 + c_2 * c_3 * c_4 * c_5 - c_2 * s_3 * s_5}{s_2 * c_5} \\ \theta_4 &= \text{atan2}(s_4, c_4)\end{aligned}$$

Abbildung: Formeln zum Berechnen der Gelenkparameter

Daraus lassen sich schließlich die Ellenbogenpositionen rausfiltern, dessen Gelenkparameter im gültigen Bereich liegen. Schlussendlich wird die genaue Position im gültigen Wertebereich mit Hilfe eines Parameters bestimmt, wobei -1 die minimal mögliche Position, und 1 die maximal mögliche Position auf der Kreisbahn widerspiegelt.

Ergebnisse

Wenn der Algorithmus ohne Beschränkung des Actionspaces trainiert, ist zwar zunächst eine Steigerung des Trainingserfolgs zu erkennen, jedoch steigt der Wert der gewählten Aktionen irgendwann so stark an, dass sich der Reward minimiert. Wie in der Abbildung zu sehen, steigt sowohl die Anzahl an Episoden mit positivem Reward, als auch der gesamte Reward bis 1000 Episoden, an. Gleichzeitig sinkt die Anzahl der Episoden mit negativem Reward. Wenn man allerdings, wie ebenfalls in der Abbildung zu sehen, eine deutlich höhere Episodenanzahl wählt, ist die Belohnung aufgrund der stark gestiegenen Parameter negativ.

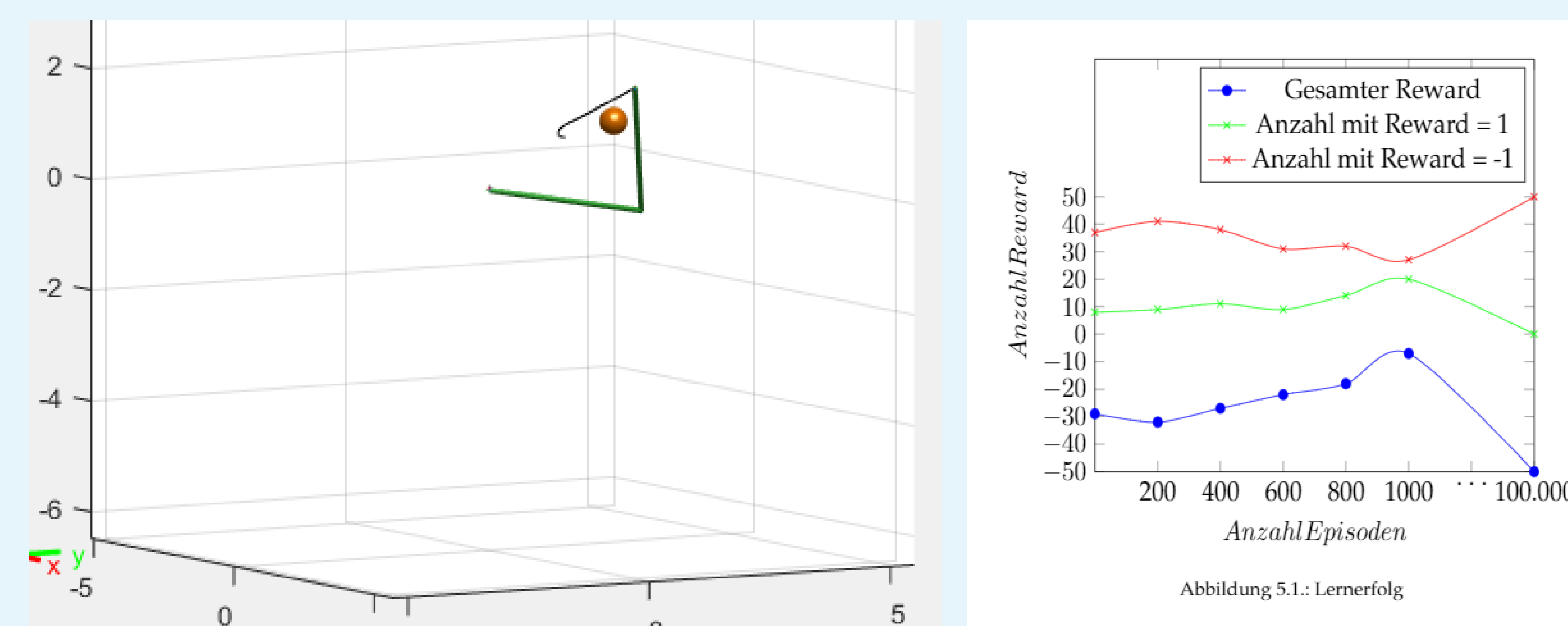


Abbildung 5.1.: Lernerfolg

Quellen

- ☞ I. Rano und I. Iossifidis, „Modelling human arm motion through the attractor dynamics approach,“ in 2013 IEEE International Conference on Robotics and Biomimetics, ROBIO 2013, 2013, S. 2088–2093, ISBN: 9781479927449
- ☞ R. S. Sutton und A. G. Barto, Reinforcement Learning Second Edition: An Introduction. 2018, ISBN: 9780262039246
- ☞ „pytorch-REINFORCE,“ (2017), Adresse: <https://github.com/chingyaoc/pytorch-REINFORCE/>