

# Projektdokumentation Enactive Learning

Stefan Simmeth, Maximilian Meyer, Felix Brabänder

Version 1:

Grundlage: Structural Turing

Agenten sind zufällig auf Welt verteilt und verfügen über eine positive, konstante Sekretionsrate. Die Zellen der Welt produzieren sowohl Aktivator, als auch Inhibitor über die Turing-Gleichungen. Muster bilden sich.

Problem: Der Einfluss der Agenten war aufgrund der zu unkontinuierlichen Abstufungen zu gering.

Version 2:

Grundlage: Turing (Diffusion/Evaporation)

Agenten und Welt wie in Version 1. Implementierung eines Feedback-Loops. Die neu eingeführte „checkFlow“ Methode überprüft die Aktivatorkonzentration zum jetzigen Zeitpunkt und vergleicht den Wert mit der Konzentration des vorherigen Timesteps (Flow). Wird eine Zunahme der Konzentration festgestellt, wird die Geschwindigkeit der Agenten verlangsamt und der Winkel des Tumbings verringert sich stochastisch. Somit soll gewährleistet werden, dass Agenten sich bei der (für sie günstigeren) höheren Aktivatorkonzentration sammeln.

Problem: Modell stabilisiert sich nicht, aufgrund der konstanten Sekretionsrate der Agenten und des explosionsartigen Inhibitorwachstums zu Beginn der Simulation.

Version 3:

Grundlage: Turing (Diffusion/Evaporation)

Agenten und Welt wie in Version 2. Lösung des Inhibitor-Problems aus Version 2, durch modifizierte Evaporation-Berechnung (Sigmoidale Form). Lösung zu Stabilitäts-Problem aus Version 2, durch Suche nach niedriger Inhibitor-Konzentration (Version 2: hohe Aktivator-Konzentration) und Veränderung der Agentengeschwindigkeit durch extrem stark gestauchte Parabel.

Problem: Division durch 0 (Inhibitor)

-> Veränderung der Rahmenbedingungen (dynamische Anpassung der Inhibitor-Werte)

Version 4:

Grundlage: Turing (Diffusion/Evaporation) durch Hill-Funktionen ersetzt

Agenten und Welt wie in Version 3. Hill-Funktion zur Lösung des Problems aus Version 3 (Division durch 0). Agenten festigen sich nun weniger stark.

Problem: Beeinflusste Agentenstabilität durch veränderte Form der Hill-Funktion

-> Veränderung der Rahmenbedingungen (Trennung der Produktion von Aktivator und Inhibitor auf Agenten und Welt)

Version 5:

Grundlage: Turing (Diffusion/Evaporation) durch Hill-Funktionen ersetzt

Agenten und Welt wie in Version 4. Auslagerung der Aktivator-Produktion ausschließlich an Agenten, Inhibitor wird weiterhin von Welt erzeugt. semi-stabile Punkte entstehen (Punkte bleiben stabil, Agenten immer in Bewegung).

Problem: Beeinflusste Agentenstabilität durch veränderte Form der Hill-Funktion

-> Veränderung der Rahmenbedingungen (Schrittweise Abwandlung eines funktionierenden Systems)

Version 6.1:

Grundlage: Turing (Diffusion/Evaporation)

Agenten sind gleichmäßig in Welt verteilt (Jede Zelle der Welt). Aktivator wird von Agenten / Inhibitor von Welt produziert. Lösung des Problems aus Version 5, durch entfernen der Hill-Funktion (Division durch 0 wird durch Vorinitialisieren der Inhibitor-Werte verhindert.) Entfernung der Sekretionsrate, um Turing-Muster zu erzeugen (siehe Constructive Alignment). Inhaltlich wie Turing, nur mit getrennter Zuständigkeit der Ausschüttung.

-> Turing Muster (Punkte) entstehen.

Version 6.2

Grundlage: Turing (Diffusion/Evaporation)

Szenario wie in 6.1 mit zusätzlicher konzentrationsabhängiger Bewegung der Agenten. Kontinuierliche Ausbreitung der Bewegung der Agenten, welche nach längerer Laufzeit Semi-stabile Punkte (Punkte bleiben stabil, Agenten immer in Bewegung) erzeugt. (Siehe Bilder Seite 6)

Im Constructive Alignment sind die verschiedenen, zwischen Version 5 und Version 6.2, durchgeführten Schritte beschrieben.

## Constructive Alignment:

Nullhypothese 1.0:

*Wir platzieren auf jeder Zelle der Welt einen Agenten, um so die Erzeugung von Aktivator durch die Zellen der Welt, auf die Agenten auslagern zu können. Das System erzeugt Turing-Muster, wie in der Ursprungsversion des Turing-Moduls*

Wir starten mit einer Welt (100\*100) und auf jeder Zelle in der Welt wird ein Agent platziert. Inhibitor wird von der Welt erzeugt und Aktivator ausschließlich von den Agenten. Aufgrund der Positionierung der Agenten erwarten wir Turing-Muster wie in der Ursprungsversion des Turingmoduls. Dieses Verhalten beobachten wir jedoch nur, wenn wir die Aktivatorkonzentration in der Welt mit kleinen zufälligen Werten vorinitialisieren.

Im nächsten Schritt entfernen wir die zufällige Initialisierung des Aktivators in der Welt. Nun ist es nichtmehr möglich die erwarteten Turing-Muster zu beobachten. Das erklären wir durch die Gleichverteilung der Agenten, welche über ihre Sekretionsrate auf allen Zellen der Welt gleichmäßig Aktivator erzeugt. Da die Sekretionsrate jedoch von der Aktivatorkonzentration in der Umgebung abhängt, wird diese fortlaufend, bei allen Agenten und somit an allen Positionen in der Welt gleich bleiben. Die hemmende Wirkung des Inhibitors auf der Welt ist mit zufälligen vorinitialisierten Werten ( $1e-4$ ) zu klein und kann deshalb nicht ausreichend mit der Sekretionsrate (Aktivator) Wechselwirken -> keine Turing Muster (Unterschiede zu klein = nicht - kaum sichtbar) Nach langer Laufzeit ist die Verteilung auf der ganzen Welt nahezu identisch und somit kommt es in der grafischen Darstellung zu Anzeige problemen.

Um die Idee der Gleichverteilungsprobleme zu testen, stellen wir folgende *Nullhypothesen* auf:

Nullhypothese 2.1: *Wir erhöhen die Initialwerte des Inhibitors in der Welt (verschiedene Werte) um somit eine verstärkte Interaktion zwischen Aktivator und Inhibitor zu ermöglichen.* Nun kommt es wieder zu einer Beeinflussung auf den Aktivator durch die hemmende Wirkung des Inhibitors und Turing-Muster können erzeugt werden.

Nullhypothese 2.2: *Wir ändern die Positionierung der Agenten von einer Gleichverteilung auf eine zufällige Anordnung auf der Welt,* somit verhindern wir das Problem der fortlaufend nahezu gleichen Verteilung von Aktivator und sind wieder in der Lage die erwarteten Turing-Muster zu erzeugen.

2.1: Wir bleiben bei der Ausgangssituation (1.0) und initialisieren die Inhibitorwerte vorerst, mit zufälligen Werten zwischen 0 und 10 vor. Entgegen der Erwartung verändert sich das Ergebnis nicht. Nach wie vor werden keine Turing-Muster erzeugt, da sich eine Gleichverteilung von Inhibitor und Aktivator aufgrund der Berechnungsweise (Turing-Gleichungen) ergibt. -> Sowohl kleine, als auch große Inhibitorwerte erzeugen langfristig eine Gleichverteilung von Inhibitor in der Welt. (Problem = Turing-Gleichungen)

Anmerkung: Hier wären noch weitere Tests nötig um die genauen Ursachen zu finden und eventuell zu beheben.

2.2: Wir bleiben bei der Ausgangssituation (1.0) und ändern die Positionierung der Agenten von einer Gleichverteilung zu einer zufälligen. Das Problem des Wechselwirkungskonflikts zwischen Aktivator und Inhibitor Konzentration scheint behoben. Allerdings entstehen nun leere Zellen, auf denen zwar Inhibitor von der Welt, jedoch kein Aktivator seitens der Agenten erzeugt wird. Die Inhibitorwerte an diesen Stellen

können jedoch aufgrund der ausbleibenden Aktivatorproduktion nichtmehr ansteigen und evaporieren gegen Null. Sollten Agenten sich nun zufällig wieder auf eben jene Zellen der niedrigen Inhibitor-Konzentrationen bewegen, entsteht wie bei Version 3, das Problem der Division durch 0.

Da die Nullhypothesen allesamt widerlegt wurden, gehen wir nun, ausgehend von der Startsituation, einen weiteren Schritt zurück, indem wir die Aktivator-Konzentration aus der Welt vorinitialisieren ( $1e-4 \cdot \text{rand}$ ) und die Sekretions-Rate der Agenten entfernen.

Nullhypothese 3.1: *Wir positionieren die Agenten weiterhin gleichverteilt über die gesamte Welt, initialisieren den Aktivator vor und entfernen die Sekretions-Rate der Agenten um uns dem ursprünglichen Turing-Modell zu nähern. Die Trennung zwischen Aktivator- und Inhibitorproduktion durch Agenten und Welt bleibt bestehen und Turing-Muster können nun beobachtet werden.*

3.1: Wir bleiben bei der Ausgangssituation (1.0), aber ändern den Aktivatorinitialisierung auf einen kleinen zufälligen Wert und entfernen die Sekretionsrate. Das System erzeugt nun die erwarteten Turing-Muster (Punkte)

Anmerkung(Sekretionsrate): Auch hier wären weitere Untersuchungen nötig, um die genaue Ursache, des von der Sekretionsrate verursachten Verhaltens, zu erklären

Nullhypothese 3.2: *Da die Turing-(ähnlichen)-Muster (in 3.1) nun erzeugt werden können, erweitern wir das System nun wieder durch die ursprünglich verwendete konzentrationsbasierte Bewegungsfunktion (Version 2) und erwarten eine Ansammlung der Agenten entlang der Turing(ähnlichen)-Muster.*

3.2: Wir bleiben bei der Ausgangssituation (3.1) und erweitern wie in der Nullhypothese 3.2 beschrieben das System um die Bewegungsfunktion. Wir beobachten zunächst keine Bewegung der Agenten, bis sich die Konzentrationswerte stark reduziert haben. Anschließend beobachten wir eine sich kontinuierlich ausbreitende Bewegung über die gesamte Welt, bis sich letztendlich Semi-stabile Punkte bilden.

Für weitere Modell-Optimierungen, Tests, Untersuchungen und Ursachenfindung von Problemen ist mehr Zeit erforderlich. Im Rahmen unseres Projekt konnten wir einige Probleme lokalisieren:

- Berechnungsprobleme mit Turing (Division durch 0)
- Hill-Funktion (möglich für Turing-Muster?)
- Trennbarkeit der Produktionsaufgaben (Aktivator / Inhibitor)
- Einfluss der Sekretionsrate

- Extremwerte (Aktivator / Inhibitor):

Das wiederholte Auftreten von Problemen im Zusammenhang mit Extremwerten stellt eine spezielle Herausforderung dar. Aufgrund der Division durch einen Inhibitor in der Turing-Gleichung resultieren aus einer minimalen Vorinitialisierung sehr große Werte.

Diese Extremwerte sind unter anderem verantwortlich für:

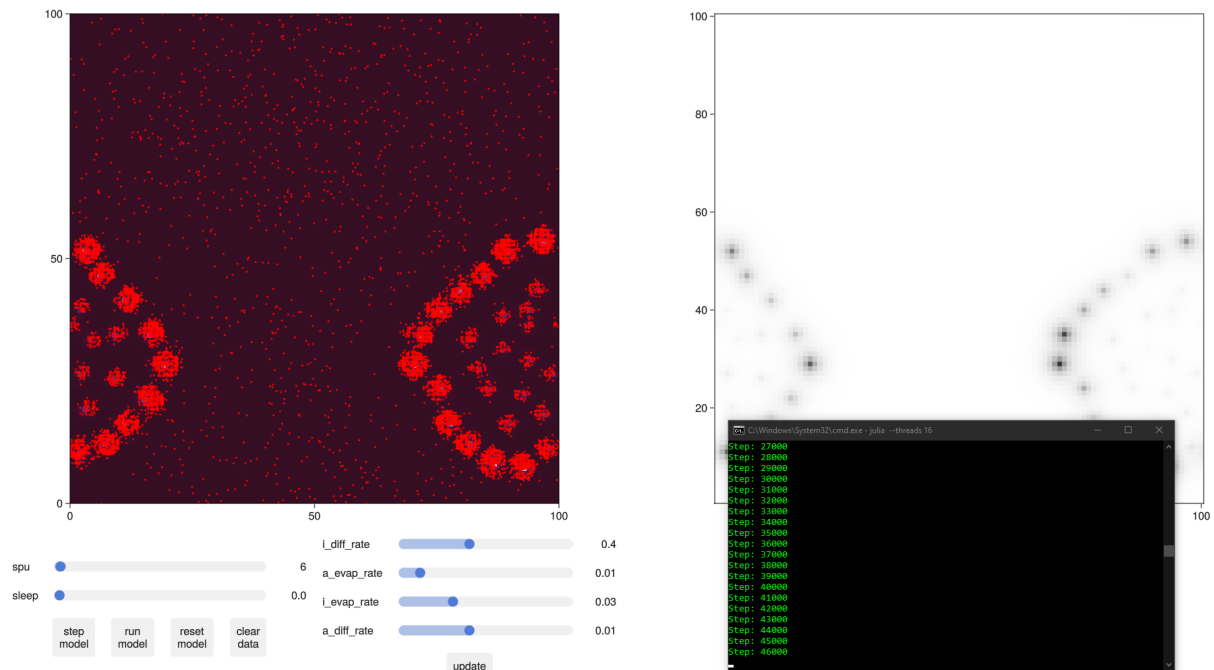
- eine erheblich längere Laufzeit und eine hohe Anzahl von Generationen, bis eine Stabilität erkennbar wird
- eine verzögerte Aktivierung der Agenten, was wiederum zu einer verzögerten Bildung von Mustern führt.

Es wird vermutet, dass diese beobachteten Extremwerte künstliche Artefakte der Simulation sind. Um solches Verhalten in zukünftigen Untersuchungen zu vermeiden, können verschiedene Schritte unternommen werden:

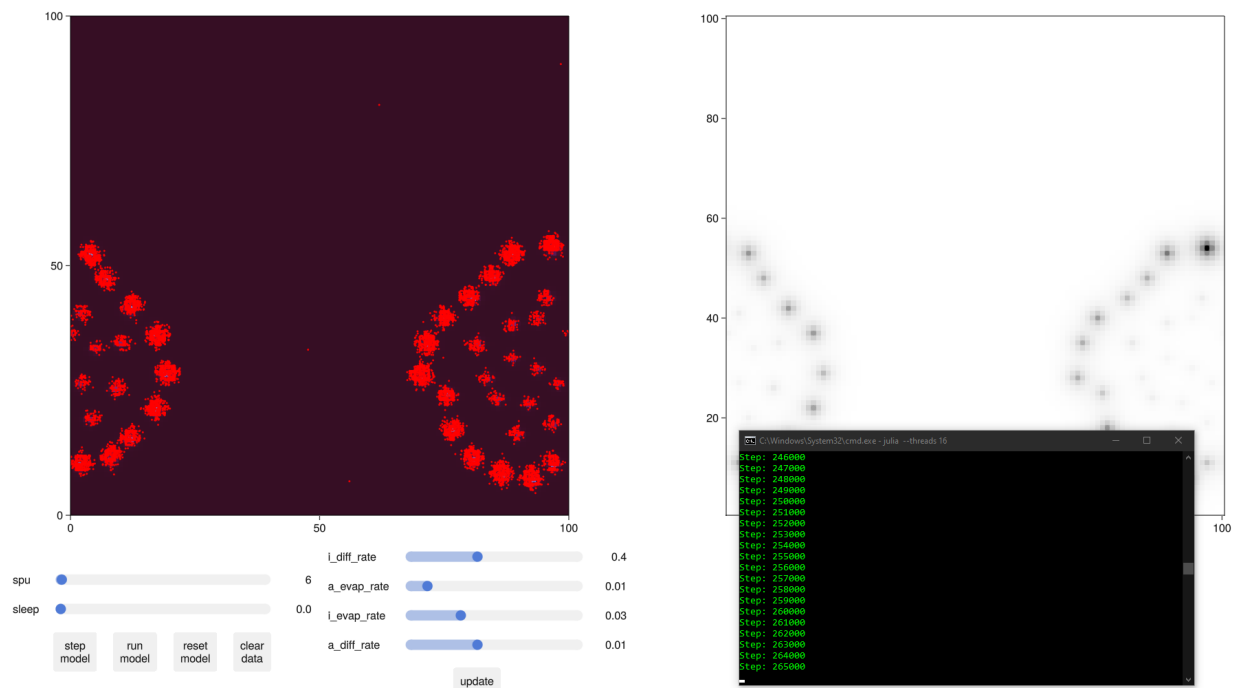
- Eine Implementierung der Berechnungen (Turing-Gleichungen) auf einer anderen Grundlage, beispielsweise Structural-Turing.
- Die Verwendung anderer Funktionen zur Berechnung der Turing-Gleichung, wie zum Beispiel die Hill-Funktion.

Gesamtzeitaufwand des Projekts: 36h Programmier- und Entwicklungsaufwand

Da dieses Projekt in konstanter Zusammenarbeit durchgeführt wurde, ist eine klare Abtrennung der einzelnen Arbeitsschritte bzw. eine Personenzuordnung, kaum bis nicht möglich.



Exemplarische Darstellung der finalen Version mit Bewegung nach 46000 Generationen



Exemplarische Darstellung der finalen Version mit Bewegung nach 265000 Generationen