

# Learning Classifier Systeme in kollaborativen, dynamischen Multiagentenszenarien

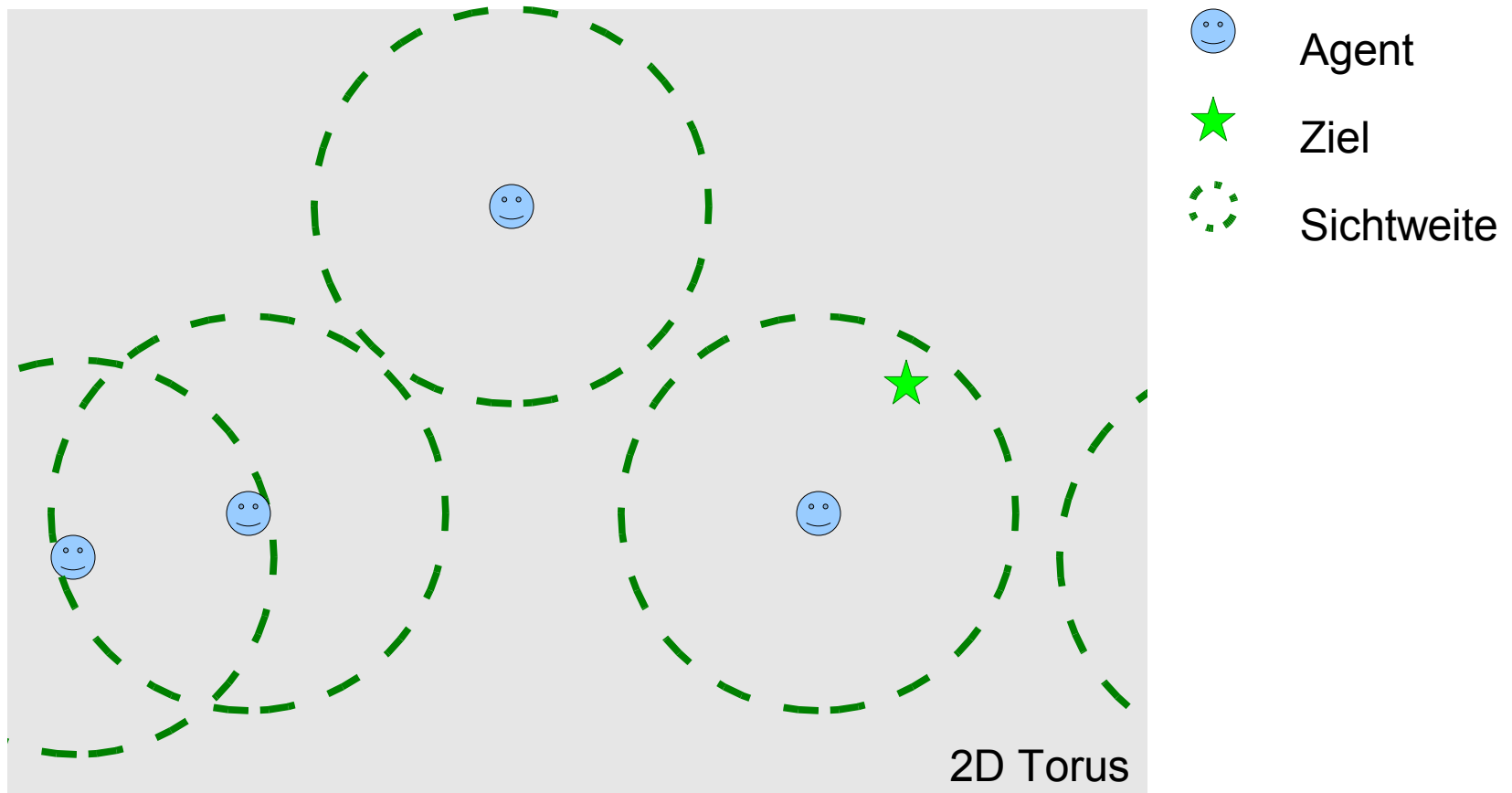
Clemens Lode  
29.10.08

# Überblick

- Testszenario
  - Multiagentensystem (Agenten, Grid, Kommunikation)
  - Observation eines sich bewegenden Ziels
- Ablauf des Learning Classifier Systems
- Beispiele für Classifier, Generierung von Classifier
- Reward (single-/multistep)
- Bewegungen des Ziels
- Kommunikation und Organisation der Agenten
- Verteilung des Rewards zwischen Agenten
- Offene Fragen und weitere Testszenarien

# 1. Szenario

- Observiere Ziel möglichst lange mit beliebigem Agenten

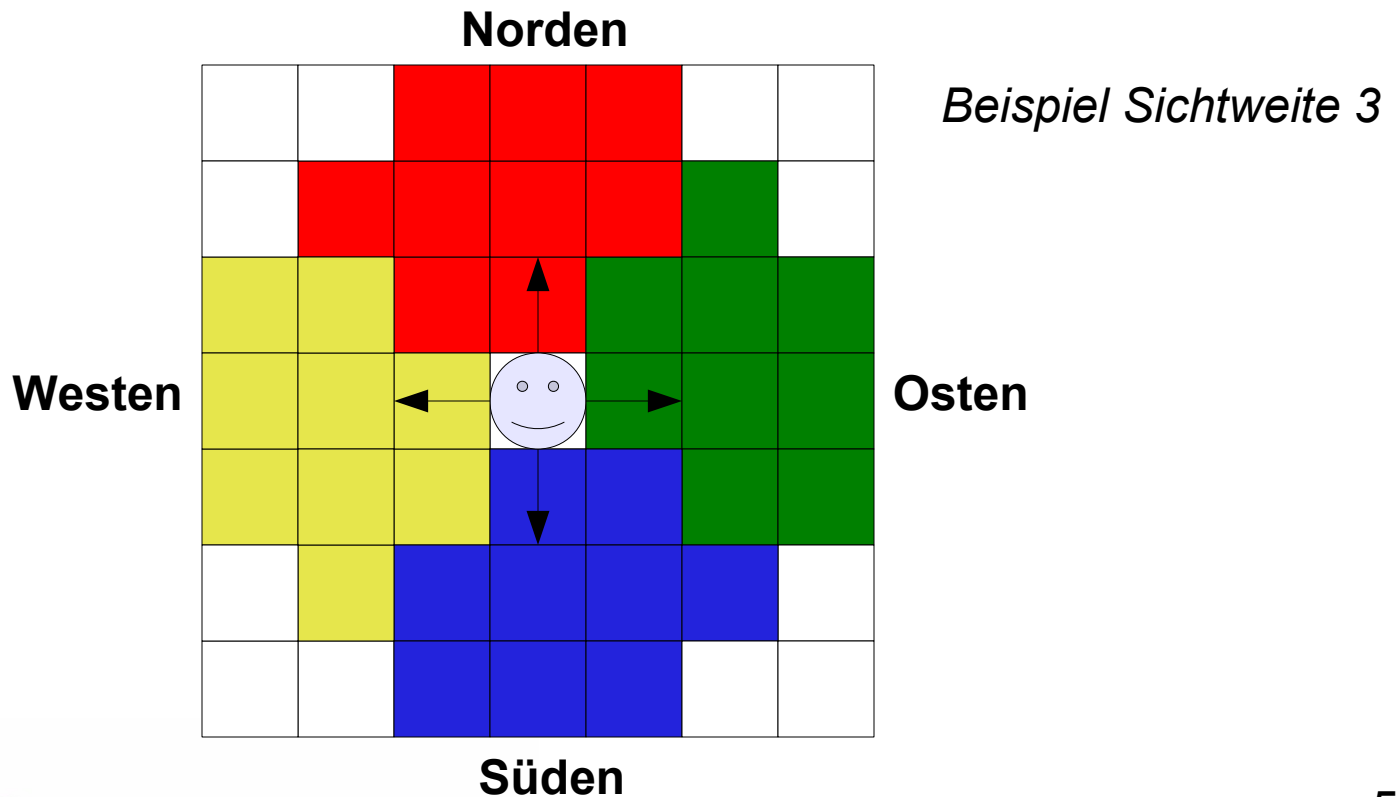


# Szenario Besonderheiten

- Begrenzte Sichtweite
- Einteilung in ein Grid
- Hindernisse
- Ziel bewegt sich
  - Schneller als einzelner Agent
  - Unvorhersehbar
- Keinerlei globale Kommunikation

# Implementation Agenten

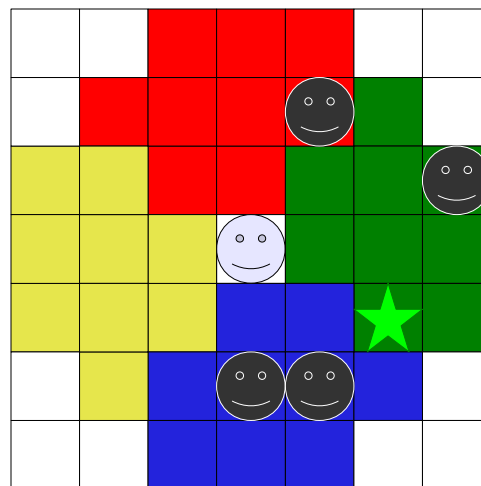
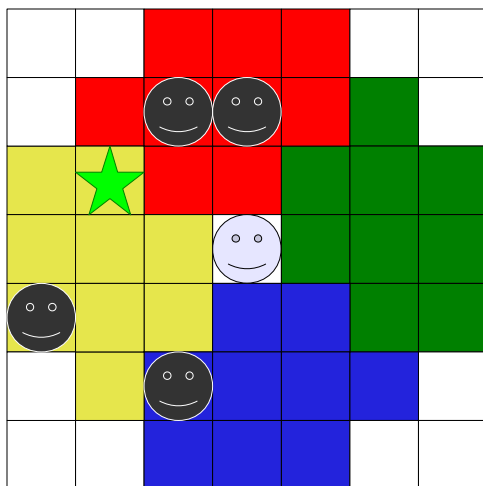
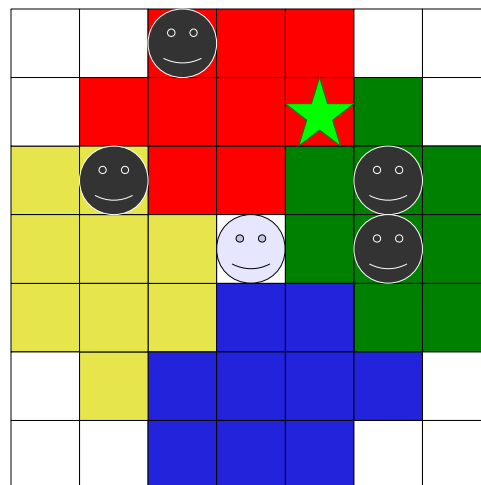
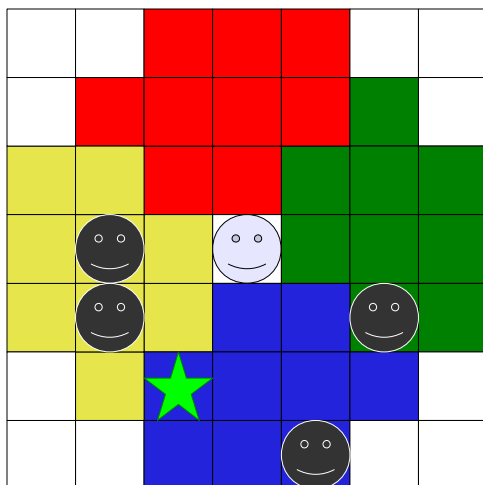
- 4 Sichtrichtungen
- Binäre Sensoren (Agent/Ziel in Sicht oder nicht in Sicht)
- 6 Aktionen (4 Richtungen, zufällig, stehenbleiben)



# Darstellung Input

- Binärstring NOSW (Sicht andere Agenten) + Bit (Sicht Ziel)
- Ein Bit für Ziel ausreichend da absolute Richtungen nicht entscheidend
  - Sei Aktion X optimal für Situation A
  - Sei Situation B um 90 Grad gedrehte Situation A
  - Um 90 Grad gedrehte Aktion X optimal für Situation B
- Optimierung: Nur ein Bit für Ziel, Rest nach dieser Richtung ausrichten

# Identische Szenarien

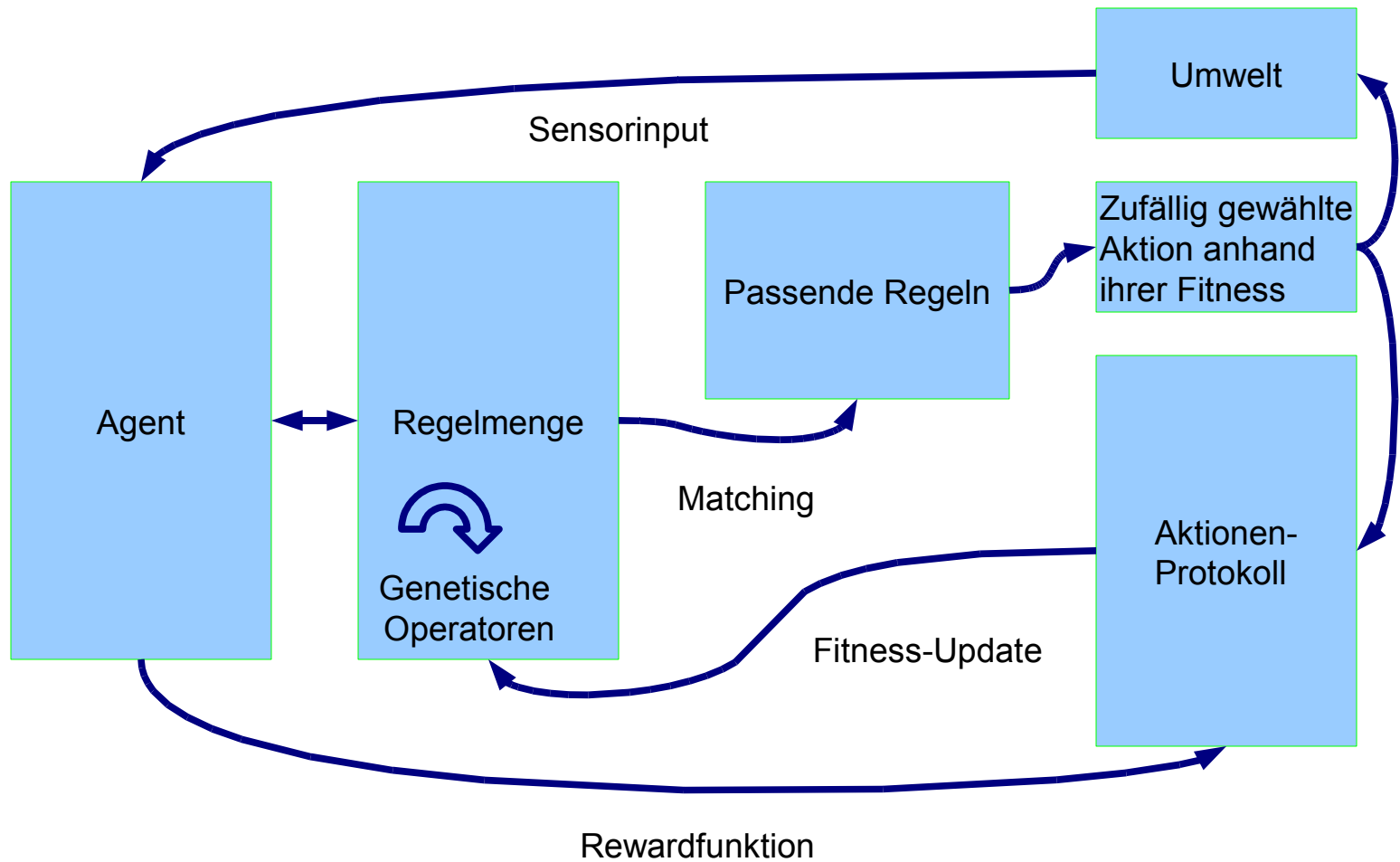


# Grundlegender Ablauf des Classifier Systems

- Jeder Agent besitzt Regelmenge
- Regel besteht aus Bedingung und Aktion
- Input wird mit Bedingung verglichen, Aktion wird ausgeführt
- Besonderheit: Bedingungen können aus Wildcards bestehen
  - Mehrere passende Bedingungen möglich
  - Wähle daraus eine zufällige Regel, gewichtet mit deren Fitness
- Fitness einer Regel wird später angepasst
- Lernen: Generiere neue Classifier (z.B. mittels genetischer Operatoren, Mutation)

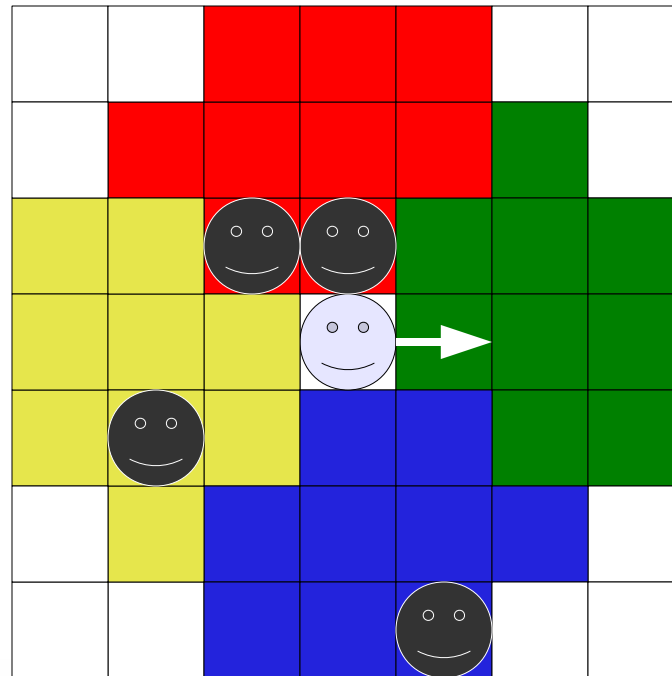


# Darstellung LCS



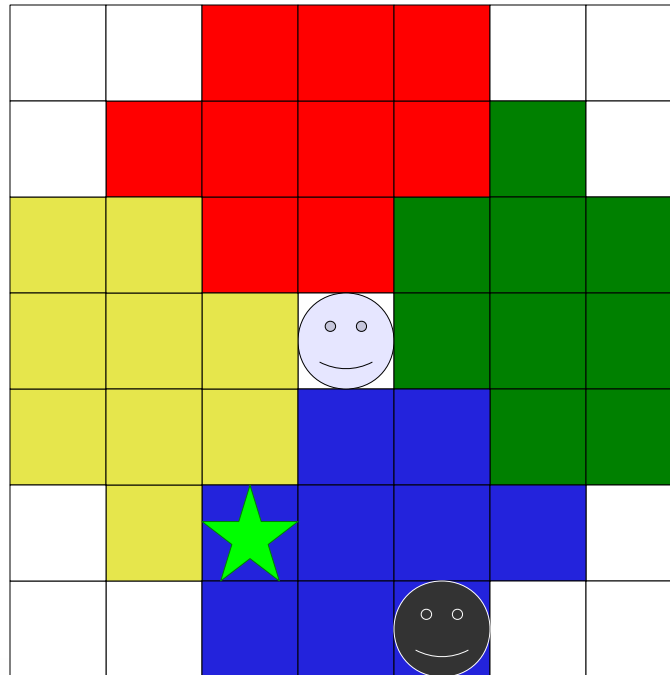
# Beispiele für Classifier

- 0.1011 ► 1: Bewege in die Richtung in der kein Agent in Sicht ist (Classifier ist identisch zu 0.1101 ► 2, 0.1110 ► 3 und 0.0111 ► 0)



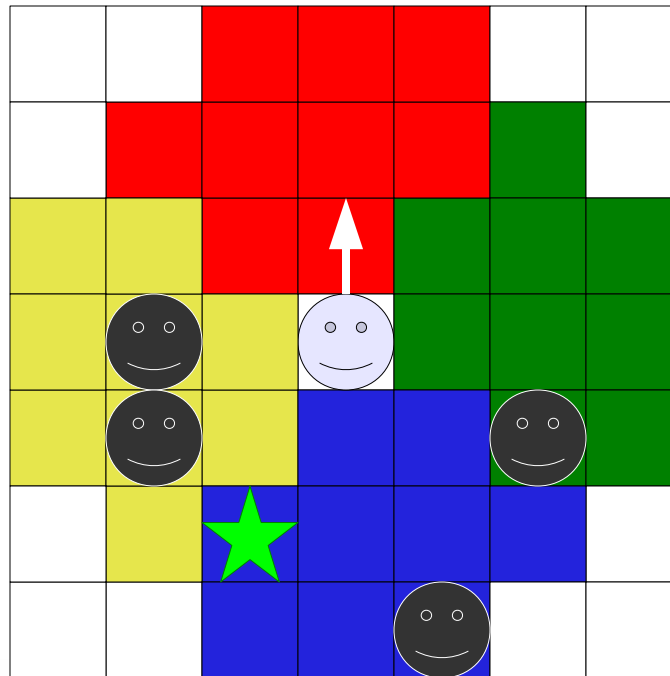
# Beispiele für Classifier

- 1.1000 ► 5: Keine Aktion (5) wenn Agent und Ziel in der selben Richtung in Sicht
  - Ist Ziel in Sicht entspricht die erste Richtung der Richtung in der das Ziel zu sehen ist



# Beispiele für Classifier

- 1.1101 ► 2: Bewege in eine Richtung, in der kein Agent in Sicht ist

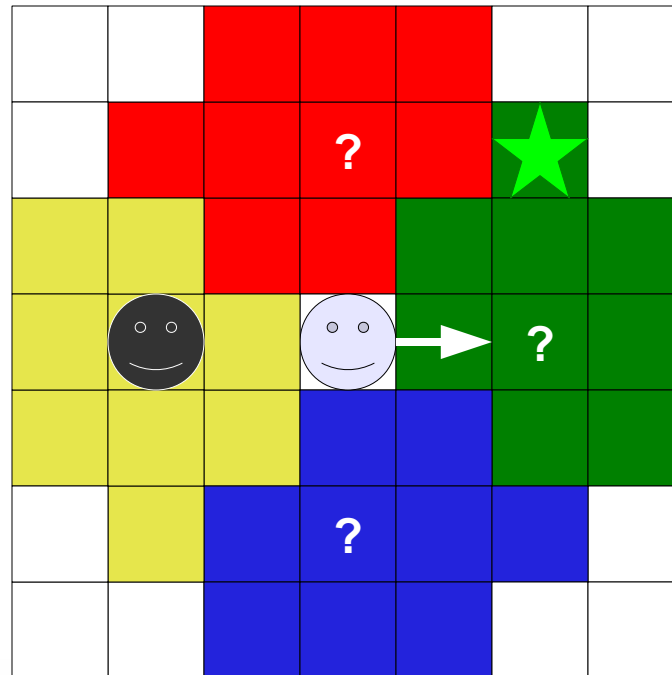


# Generalisierung: Wildcards

- Menge der Classifier kleiner (Effizienz)
- Vermeidung redundanter Informationen
- Beispiel: 1.0111 ▶ 2 und 0.0111 ▶ 2
  - Benutze # als “wildcard”: #.0111 ▶ 2
- Problem: Sicherstellen, dass Information nicht verloren geht
  - Ignoriert wurde bei obigem Beispiel die Rolle der Fitness der beiden Classifier
- Offene Frage: Je nach Szenario unterschiedliche Arten von Generalisierung denkbar (z.B. Angabe eines begrenzten Zahlenbereichs bei mehr als 2 Belegungsmöglichkeiten)

# Beispiele für Classifier

- 1.##1#►0: Bewege auf Ziel zu, bewege von Agenten weg, ignoriere restliche Umgebung



# Generierung neuer Classifier

- Drei Quellen für neue Classifier:
  - Covering
    - Falls kein Matching für ein Classifier gefunden wurde
    - Erstelle neuen, zufälligen Classifier mit passendem Matching
  - Genetischer Algorithmus, Mutation
    - Crossover zwischen bestehenden Classifiern
    - Mutation von bestehenden Classifiern
  - Austausch zwischen Agenten
    - Crossover oder direkte Kopie

# Reward

- Zentrale Frage: Wie werden Classifier bewertet?
- Verschiedene Ansätze
  - Single-Step
    - Nach jeder Aktion muss Reward bekannt sein.
    - Problem: Globale Information notwendig
  - Multi-Step
    - Reward wird an vergangene Aktionen bei Erreichen des Ziels verteilt
    - Problem: Kein konkretes Ziel, dauerhafte Überwachung, sich verändernde Umwelt



# Zielagent

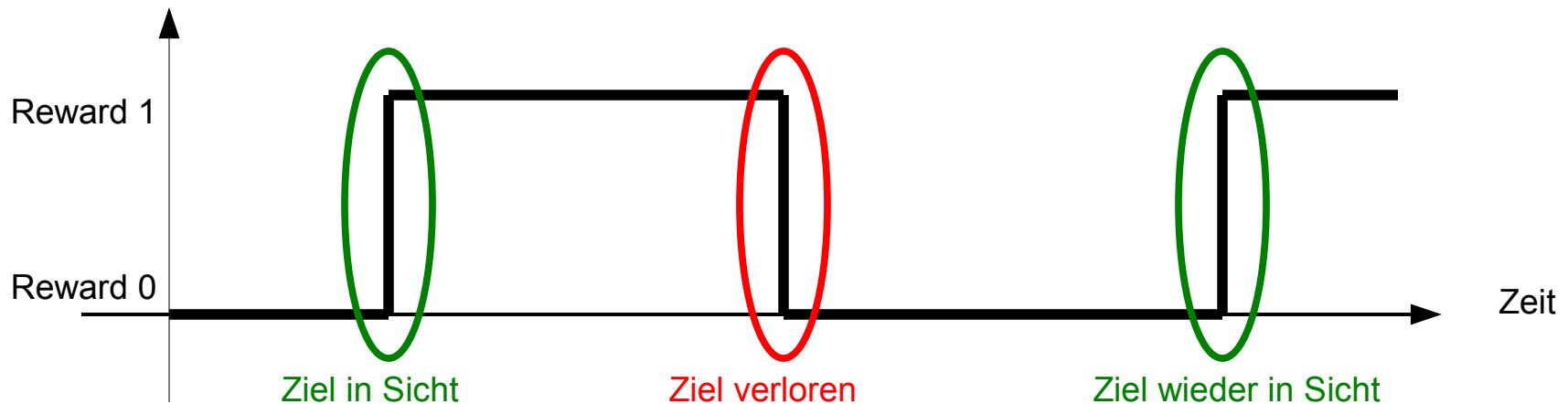
- Unbewegt: Problem ist identisch mit dem Maze Problem.
- Vorhersehbare Bewegung: Ähnlich dem Maze Problem, Bewegung der Agenten verschiebt sich.
- Interessanter: Unvorhersehbare Bewegung
  - Weder über Single-Step noch Multi-Step vernünftig lösbar.
- Zusätzlich: Zielagent bewegt sich schneller als die einzelnen Agenten.
  - Konstantes Verfolgen also unmöglich
  - Ziel ist eher, kollaborativ die Spielfläche abzudecken.

# Ereignisse aus Sicht eines Agenten

- Mögliche Ereignisse:
  - Ziel bleibt außer Sichtweite.
  - Ziel bleibt in Sichtweite.
  - Ziel wurde gerade aus Sichtweite verloren.
    - Absteigende Bestrafung der vorangegangenen Aktionen
  - Ziel kommt gerade in Sichtweite.
    - Absteigende Belohnung der vorangegangenen Aktionen

# Rewardfunktion

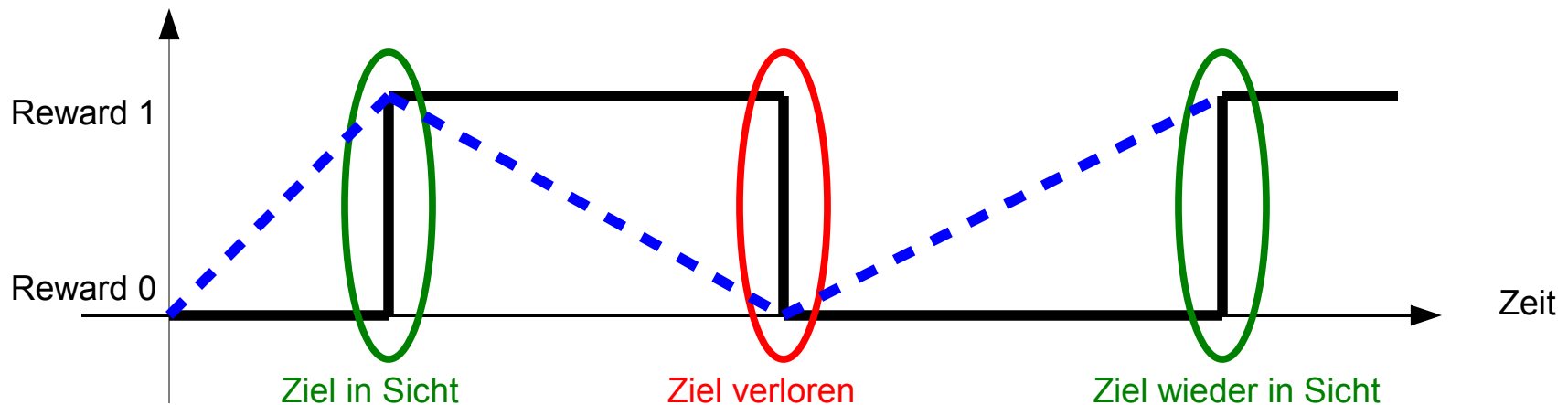
- Zu erwarten: Verteilung des Rewards in längeren Zeitabschnitten von 0 (Ziel nicht in Sicht) und 1 (in Sicht)
- In jedem Zeitabschnitt werden eine Anzahl von Classifier aktiviert und protokolliert
- Irrelevant: Ziel momentan in Sicht / nicht in Sicht
- Relevant: Ziel kommt in Sicht / Sicht zum Ziel verloren



# Rewardfunktion

- Ziel kommt in Sicht (oder verlieren wir es aus der Sicht)
  - Annahme: Protokollierten Aktionen seit dem letzten Ereignis waren absteigend daran beteiligt
  - Belohnung der zugehörigen Classifier

■ ■ ■ ■ ■ ■ ■ Tatsächlich vergebener Reward an einzelne Classifier

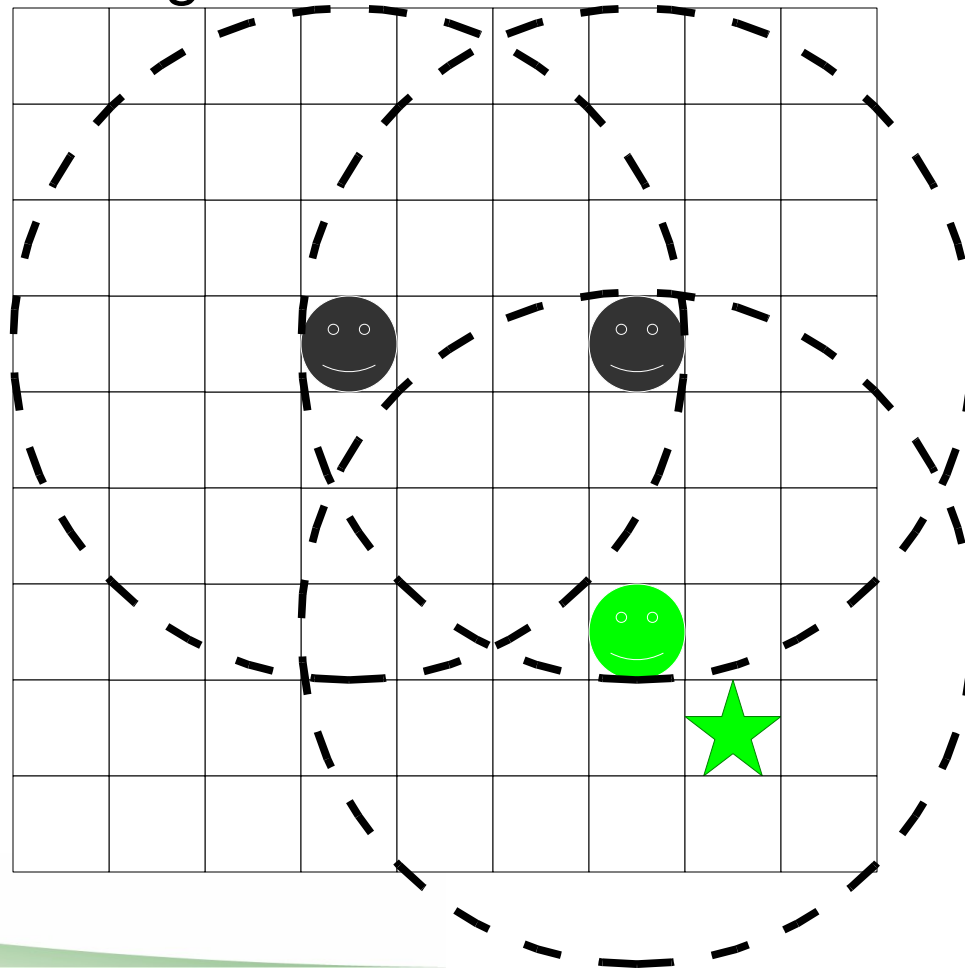


# Verteilung des Rewards zwischen Agenten

- Problem:
  - Kollaboration wird nicht honoriert
  - Keine globale Organisationseinheit
- In einem Netz aus Agenten müsste jeder Agent abhängig von der Aufenthaltswahrscheinlichkeit des Ziels im überwachten Gebiet belohnt werden.
  - Problem: Bewegung des Ziels ist grundsätzlich unvorhersehbar
    - Aufenthaltswahrscheinlichkeit unbekannt
  - Idee:
    - Bildung lokaler Populationen
    - Verteilung des Rewards innerhalb der jeweiligen Population

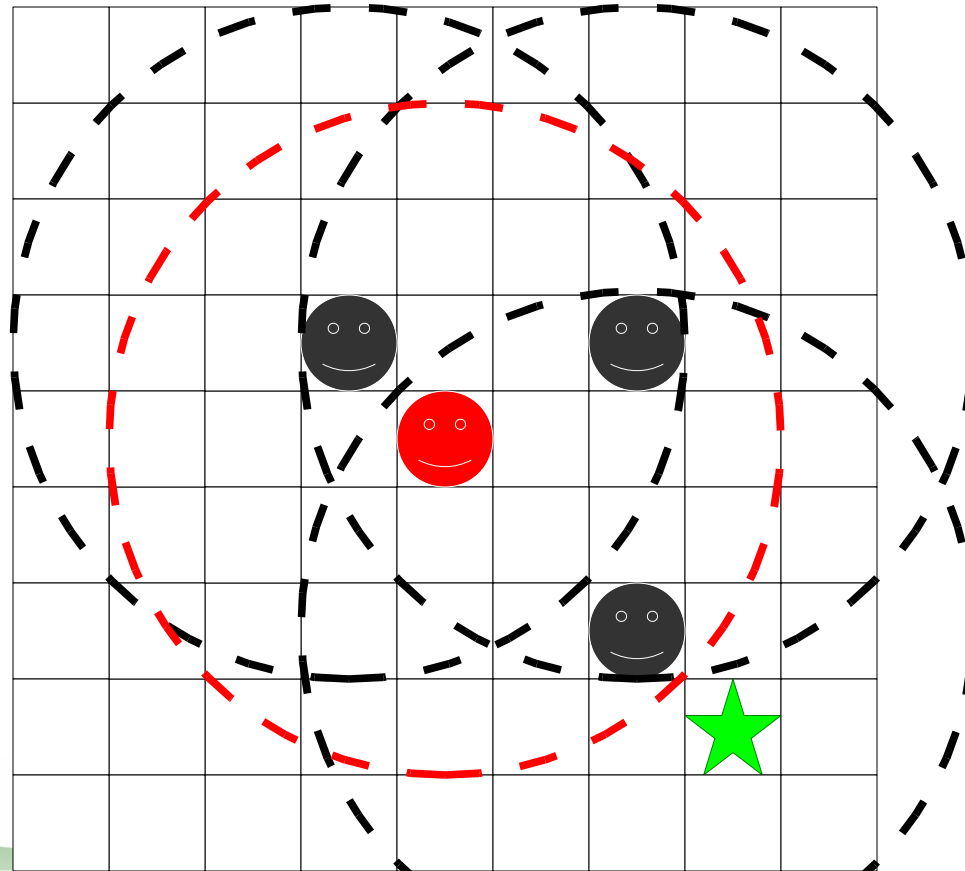
# Verteilung des Rewards zwischen Agenten

- Nur grüner Agent wird belohnt, schwarze Agenten gehen trotz zusätzlich abgedecktem Gebiet leer aus



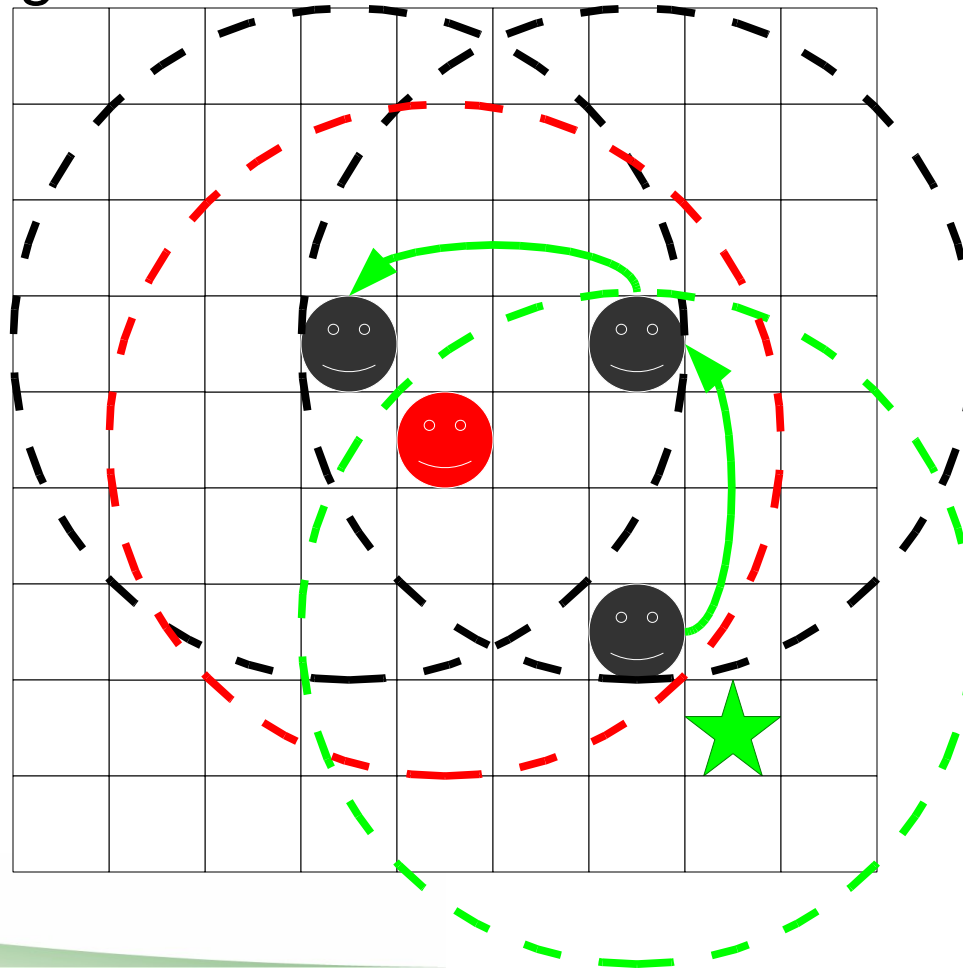
# Möglicher Lösungsansatz

- Verteilung des Rewards an verwandte Agenten
  - Lokalpopulationen bilden eine Gruppe, sieht ein Mitglied das Ziel, erhalten alle den Reward.



# Möglicher Lösungsansatz

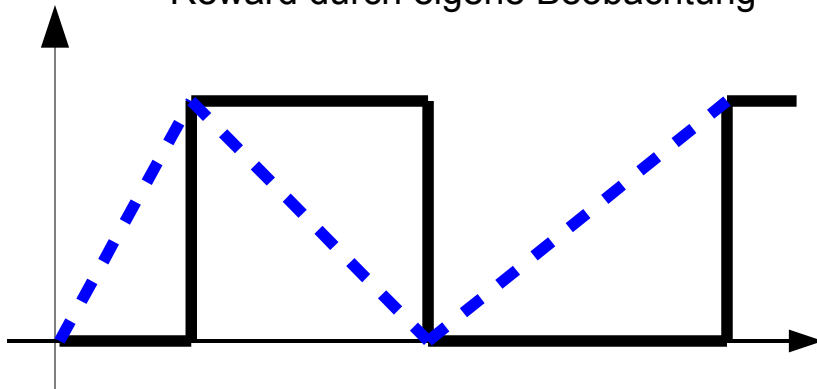
- Weitergabe des Rewards nur an die beiden schwarzen Agenten, rot geht leer aus



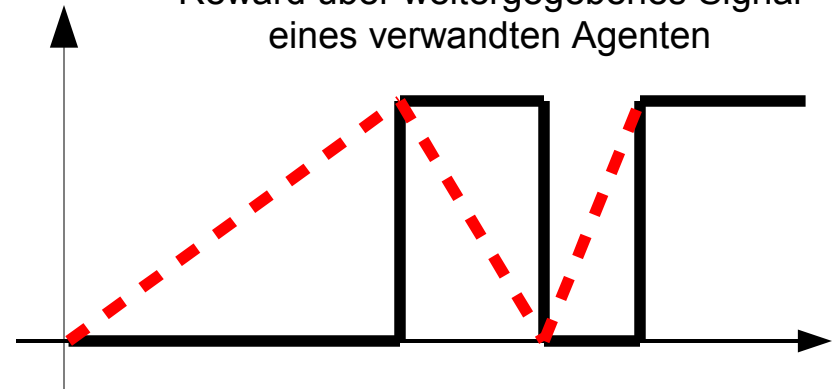


# Alternative Rewardfunktion

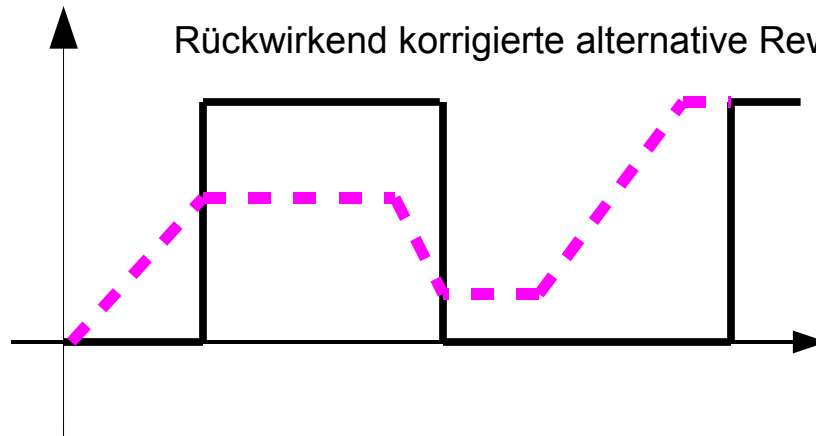
Reward durch eigene Beobachtung



Reward über weitergegebenes Signal eines verwandten Agenten



Rückwirkend korrigierte alternative Rewardfunktion



# Mögliche Probleme der Idee

- Homogenisierung der Regeln wird begünstigt.
- Aufwand zur Bestimmung des Verwandtschaftsgrads
- Begrenztheit der Kommunikationsmittel
- Gewisse Verzögerung bis sich Information ausgebreitet hat
- Aber:
  - Keine Übertragung der Sensorinformation nötig
  - Kommunikation ist selten notwendig
  - Nicht zeitkritisch, sofern bisher aktivierte Classifier protokolliert werden
    - Weitertransport des Rewards von Agent zu Agent
    - Markierung mit Timestamp

# Offene Fragen und Testszenarien

- Vergleich mit üblichen Vorgehensweisen (Heuristiken, single-step LCS)
- Test der Kooperationsfähigkeit (Austausch von Regeln)
- Test der Reaktionsfähigkeit des Systems auf Veränderungen der Umwelt (Hindernisse, Änderung des Bewegungsmusters des Ziels, Hinzufügen/Entfernen von Agenten)
- Nicht erwähnt: Hindernisse (Hinzunahme 4 weiterer Sensorbits)
- Komplexere Sensoren (Abstände)

Vielen Dank für die  
Aufmerksamkeit :)