

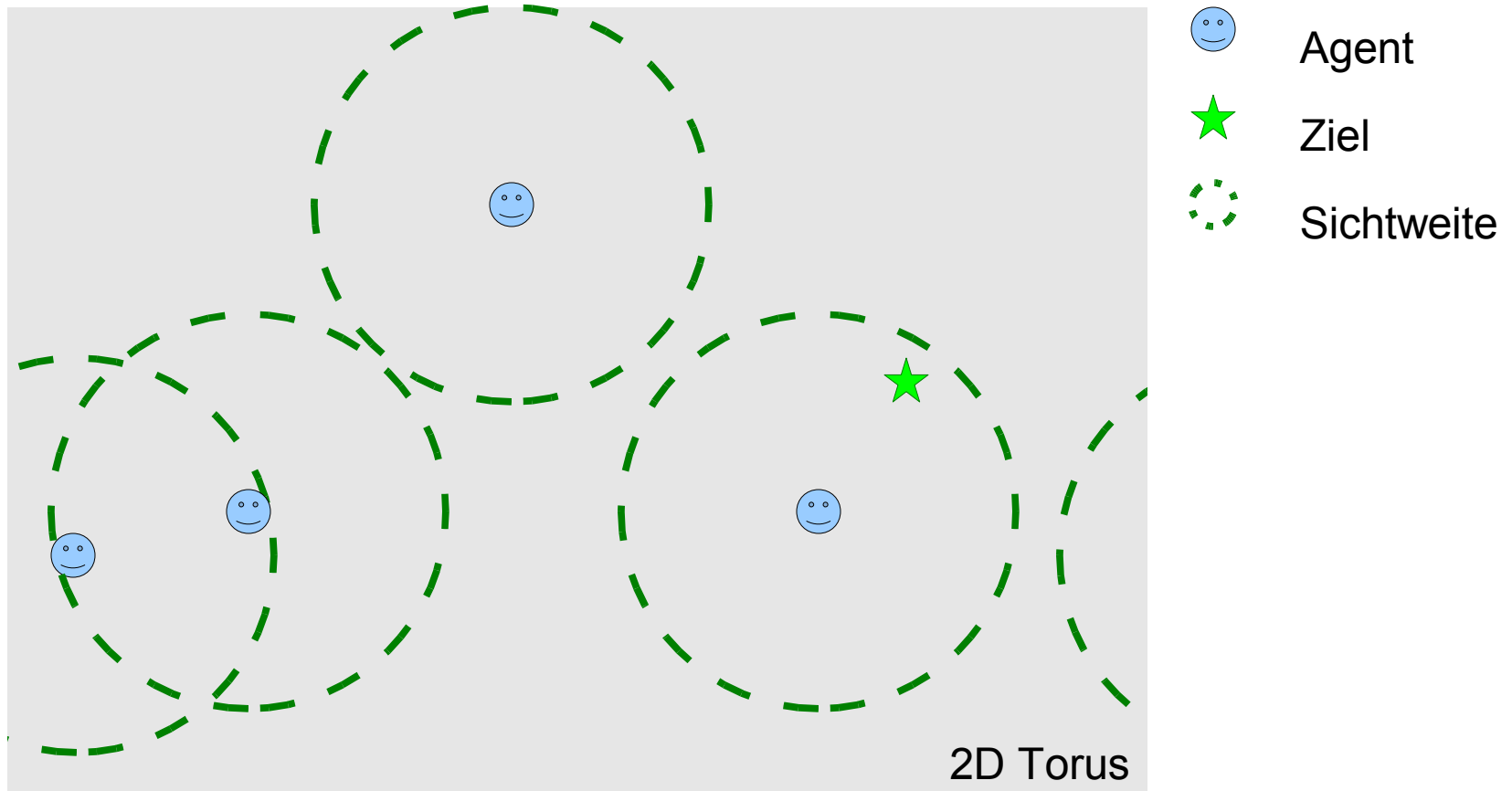
Learning Classifier Systems in kollaborativen, dynamischen Multiagentenszenarien

Überblick

- Testszenario
 - Multiagentensystem (Grid, Hindernisse)
 - Observation eines bewegenden Ziels
- Ablauf des des Learning Classifier Systems
- Beispiele für Classifier, Generierung von Classifier
- Bewegungen des Ziels
- Reward (single-/multistep, sharing between agents)
- Kommunikation und Organisation der Agenten
- Offene Fragen und Testszenarien

Szenario

- Observiere Ziel möglichst lange mit beliebigem Agenten

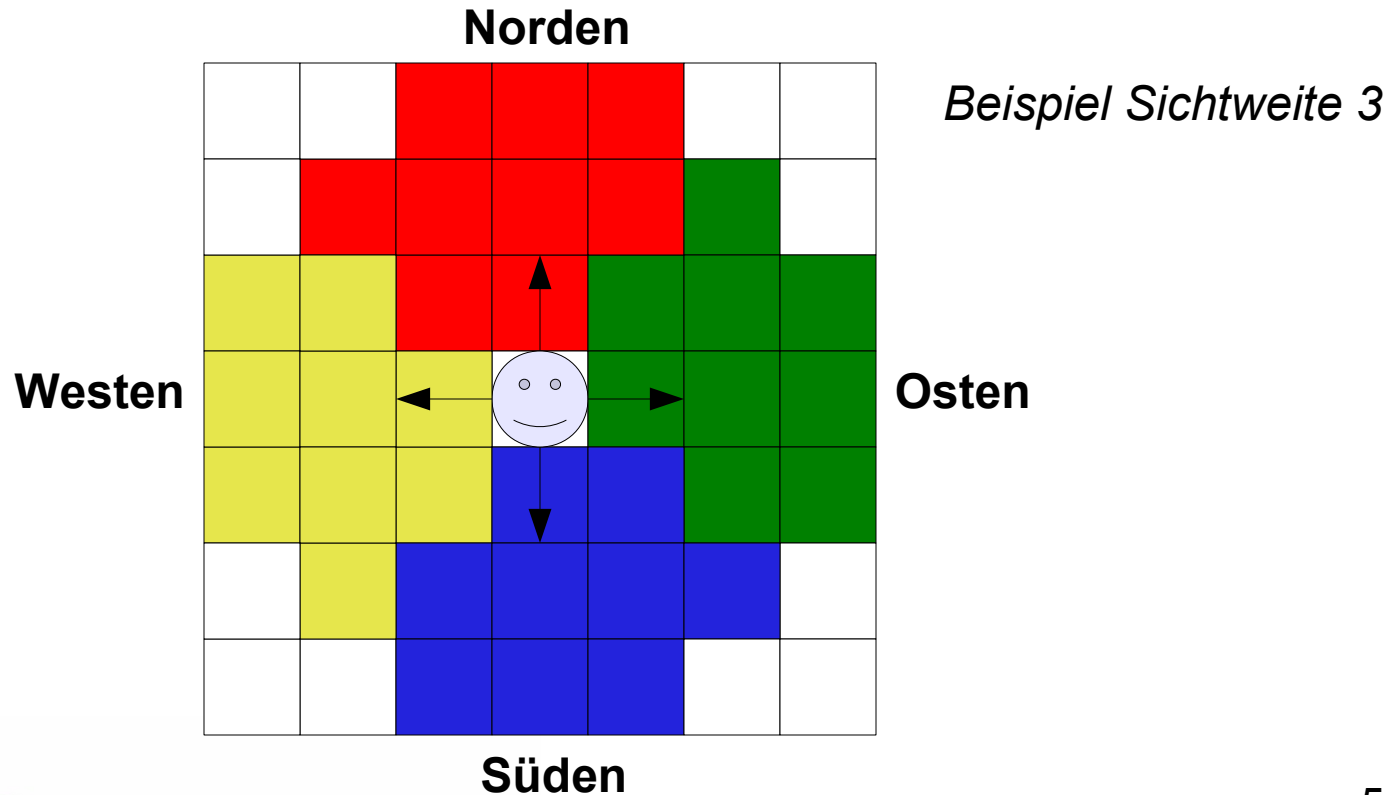


Szenario Besonderheiten

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

Agenten, konkrete Umsetzung

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



Darstellung Input

- Binärstring NESW (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

Grundlegender Ablauf des Classifier Systems

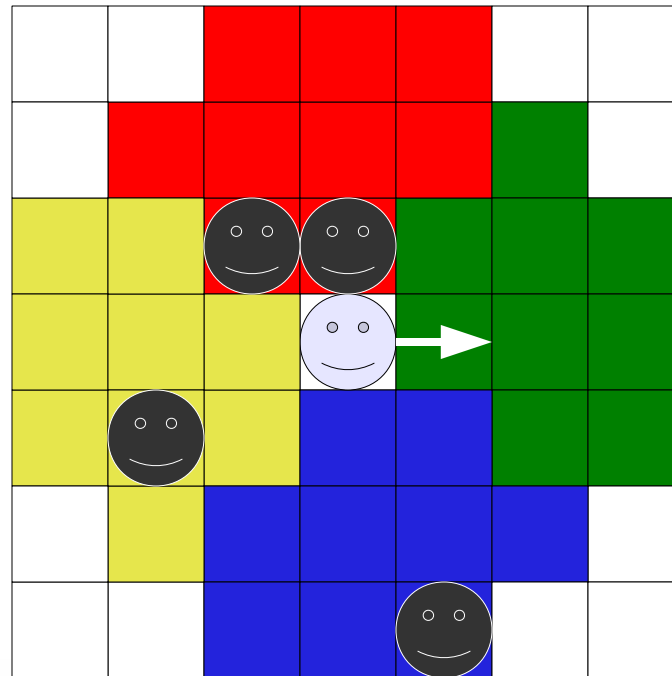
- Jeder Agent besitzt Regelmenge
- Regel besteht aus Matching und Aktion
- Input wird mit Matchings verglichen, Aktion wird ausgeführt
- Besonderheit: Matchings können aus Wildcards bestehen, mehrere Matchings pro Input möglich
- Mehrere Matchings
 - wähle daraus eine zufällige Regel, gewichtet mit deren Fitness
- Fitness einer Regel wird später angepasst
- Lernen: Generiere neue Classifier (mittels GA, Mutation)

Vereinfachte Darstellung LCS

- TODO Grafik Ablauf LCS

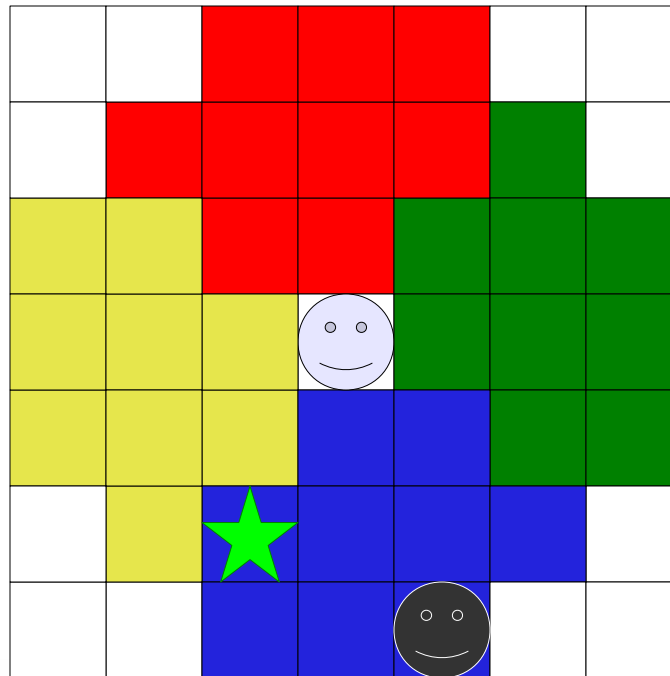
Beispiele für Classifier

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



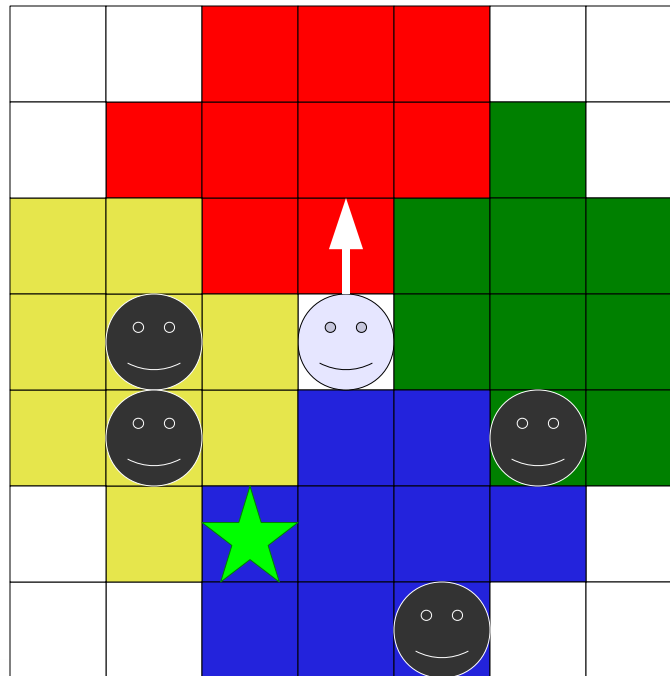
Beispiele für Classifier

- 1.1000 ▶ 5: Keine Aktion (5) wenn Agent und Ziel in der selben Richtung in Sicht



Beispiele für Classifier

- 1.1101 ► 2: Bewege in eine Richtung in der kein Agent in Sicht ist (Classifier ist identisch zu 1.1110 ► 3, 1.0111 ► 0 und 1.1011 ► 1)

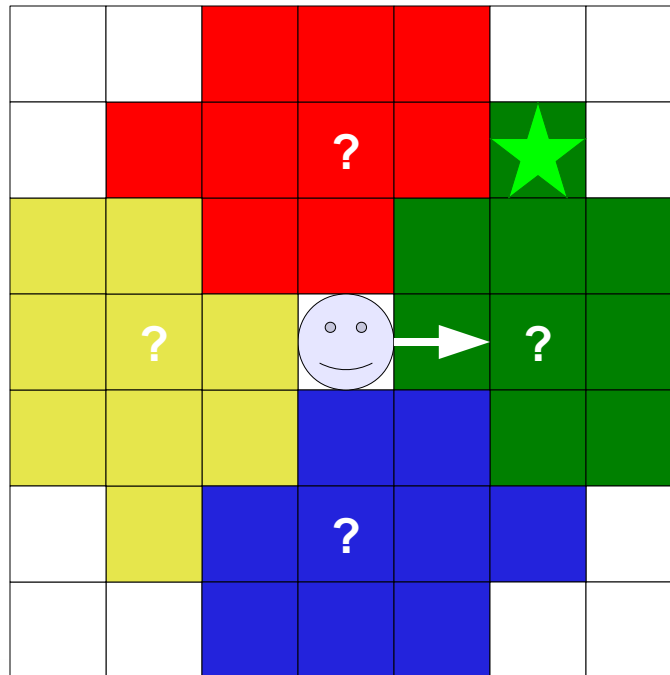


Generalisierung: Wildcards

- 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)

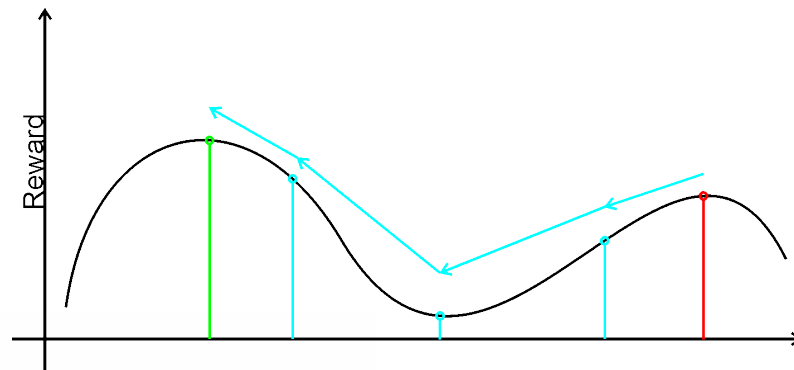
Beispiele für Classifier

- 1.####►0: Bewege auf Ziel zu, ignoriere Umgebung



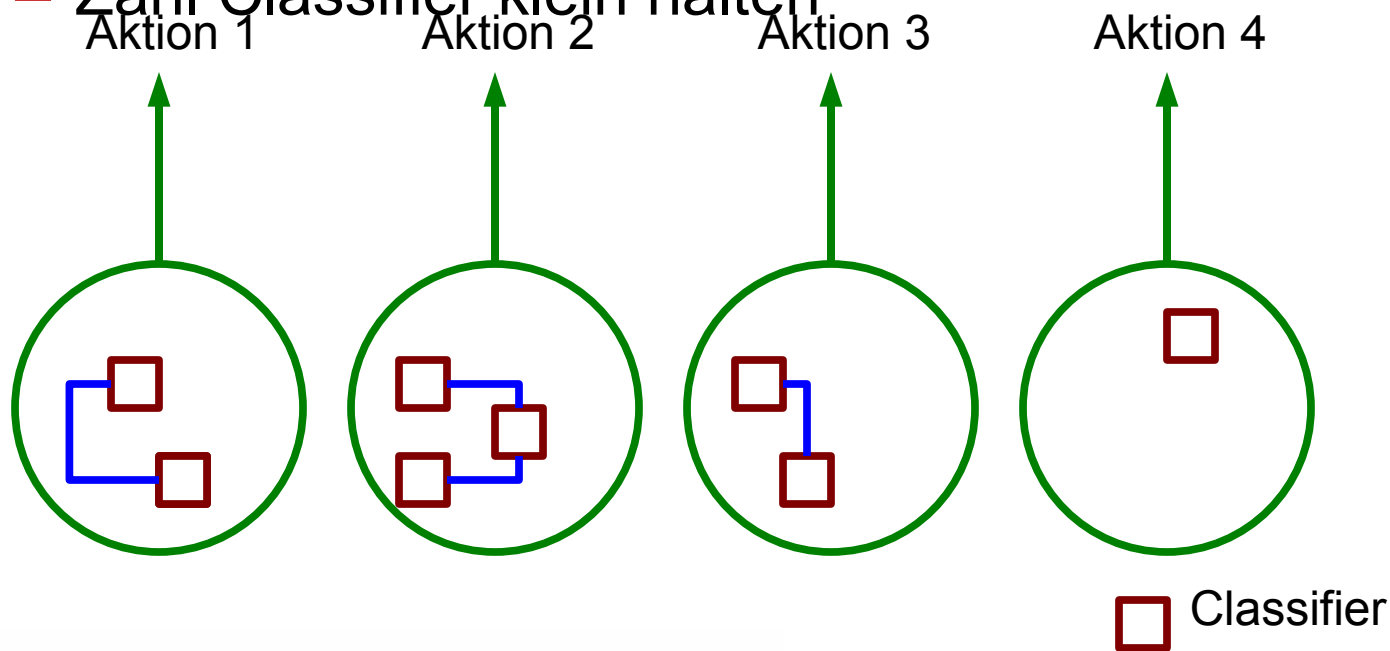
Reward und Wildcards

- Classifiers mit geringer Fitness werden eher entfernt
- Low-reward actions might be necessary to reach global optimum, but are removed due to low fitness.
- Very general classifiers might get higher average reward than more specific classifiers with actually better action.
- Extension: Fitness for reproduction and deletion based on **accuracy of prediction** → XCS (Wilson).
-
- TODO



TODO

- Ziel:
 - Gemeinsamkeiten zwischen Classifiern finden, verallgemeinern, auf unbekannte / wenig getestete Fälle schließen
 - Zahl Classifier klein halten



Generierung neuer Classifier

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

Reward

- Zentrale Frage: Wie werden Classifier bewertet?
- Verschiedene Ansätze
 - Single-Step
 - Problem: Nach jeder Aktion muß Reward bekannt sein
 - Multi-Step
 - Problem: Kein konkretes Ziel, dauerhafte Überwachung, sich verändernde Umwelt

Reward (single-step)

- Bedingung: Nach jeder Aktion muß bekannt sein, wie gut die Aktion war
- Unmöglich in diesem Fall, außer mit lokalen Heuristiken (z.B. Distanz zum Ziel und zu anderen Agenten)
- Aber: Bei Benutzung lokaler Heuristiken erhält man mit LCS eine an diese Heuristiken (und nicht an die Situation) angepasste (wahrscheinlich nicht optimale) Lösung

Reward (multi-step)

- Wann immer das Ziel eines Problems erreicht wurde nimmt man an, dass alle vorangegangenen Aktionen zur Lösung beigetragen haben
- Hier nicht wirklich anwendbar:
 - “Ziel” des Problems ist angepasste, dauerhafte Überwachung
 - Gebiet verändert sich (Hindernisse, andere Agenten, sich bewogender Zielagent)

Zielagent

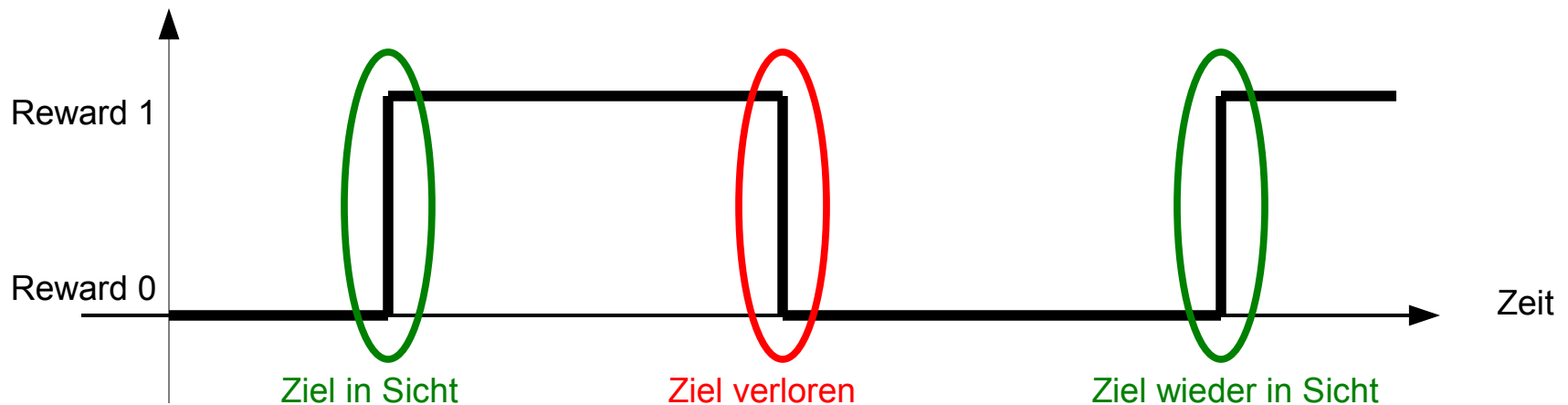
- Unbewegt: Problem ist identisch mit dem Maze Problem
- Vorhersehbare Bewegung: Ähnlich dem Maze Problem, Bewegung der Agenten verschiebt sich
- Interessanter: Unvorhersehbare Bewegung
 - weder über Single-Step noch Multi-Step vernünftig lösbar
- Zusätzlich: Zielagent bewegt sich schneller als die einzelnen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

Ereignisse

- 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

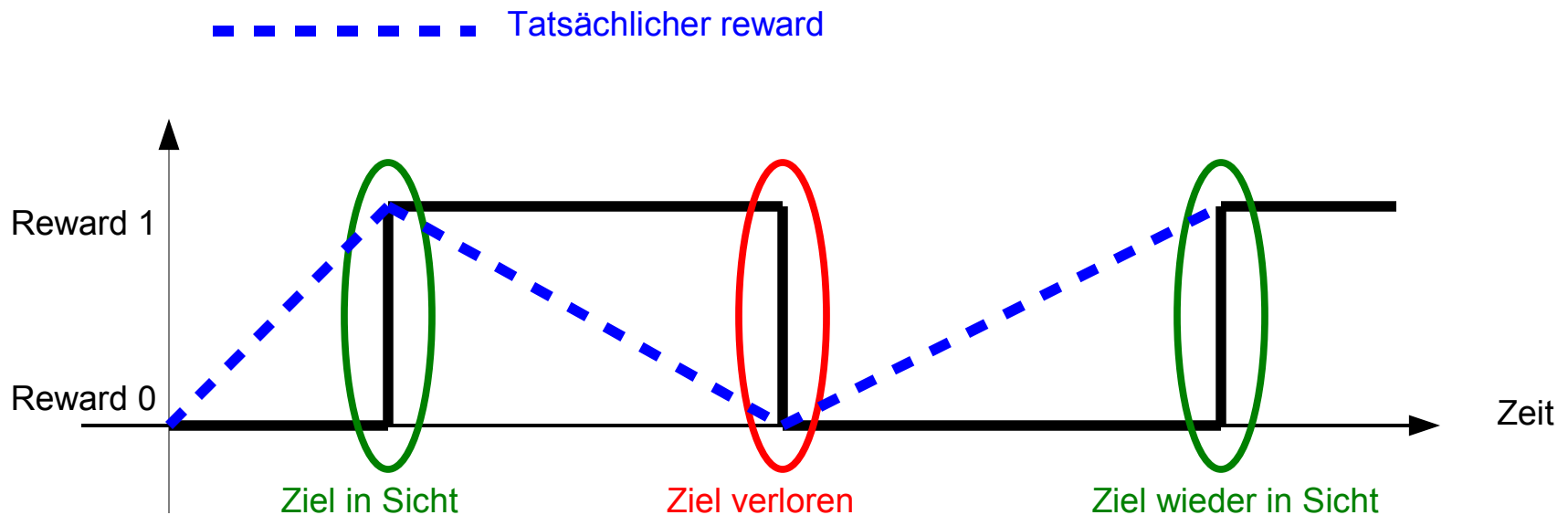
Verfahren in diesem Szenario

- Zu erwarten ist eine Verteilung des Rewards in längeren Zeitabschnitten von 0 (Ziel nicht in Sicht) und 1 (Ziel in Sicht)
- Relevante “Ereignisse” sind nicht, dass momentan das Ziel in Sicht ist, sondern dass das Ziel sich in oder ausser Sichtweite begibt:



Verteilung des Rewards

- Kommt das Ziel in Sicht (oder verlieren wir es aus der Sicht), gehen wir davon aus, dass die Aktionen seit dem letzten Ereignis absteigend daran beteiligt waren



Kommunikation und Organisation der Agenten

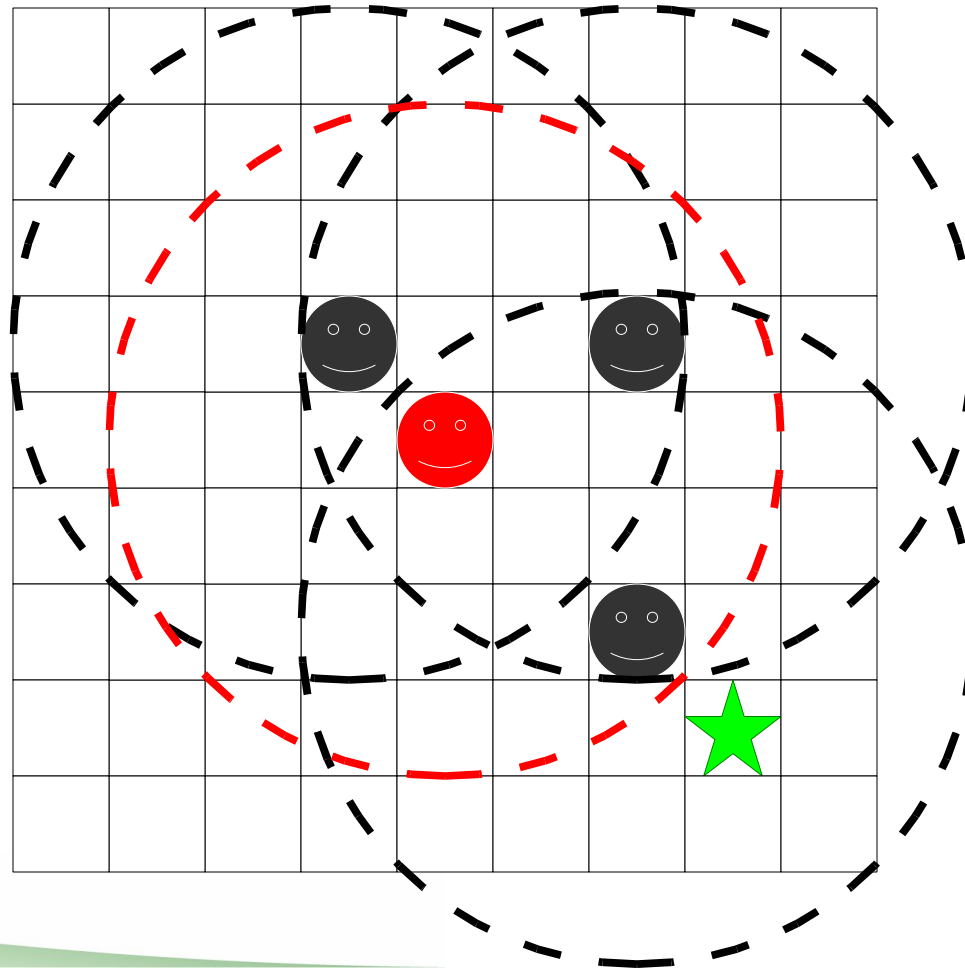
- Keine zentrale Organisationseinheit, keine globale Kommunikation
- Globaler Austausch von Regeln würde lokale Optimierungen (Hindernisse) ignorieren
 - Begrenzte lokale Kommunikation
 - Bildung lokaler Populationen
 - Verteilung des Rewards innerhalb der jeweiligen Population
- Momentan noch offen inwiefern dies zielführend ist

Verteilung des Rewards

- Bewegung des Zielagents ist grundsätzlich zufällig
- Problem: z.B. 3 Agenten die zusammen ein nicht-überlappendes Gebiet abdecken
 - Nur jeweils ein Agent erhält Reward obwohl alle 3 Agenten gleichermaßen beigetragen haben
 - Keine Basis für Kollaboration

Verteilung des Rewards

- Schwarze Agenten miteinander verwandt bzw. ähnliche Classifier

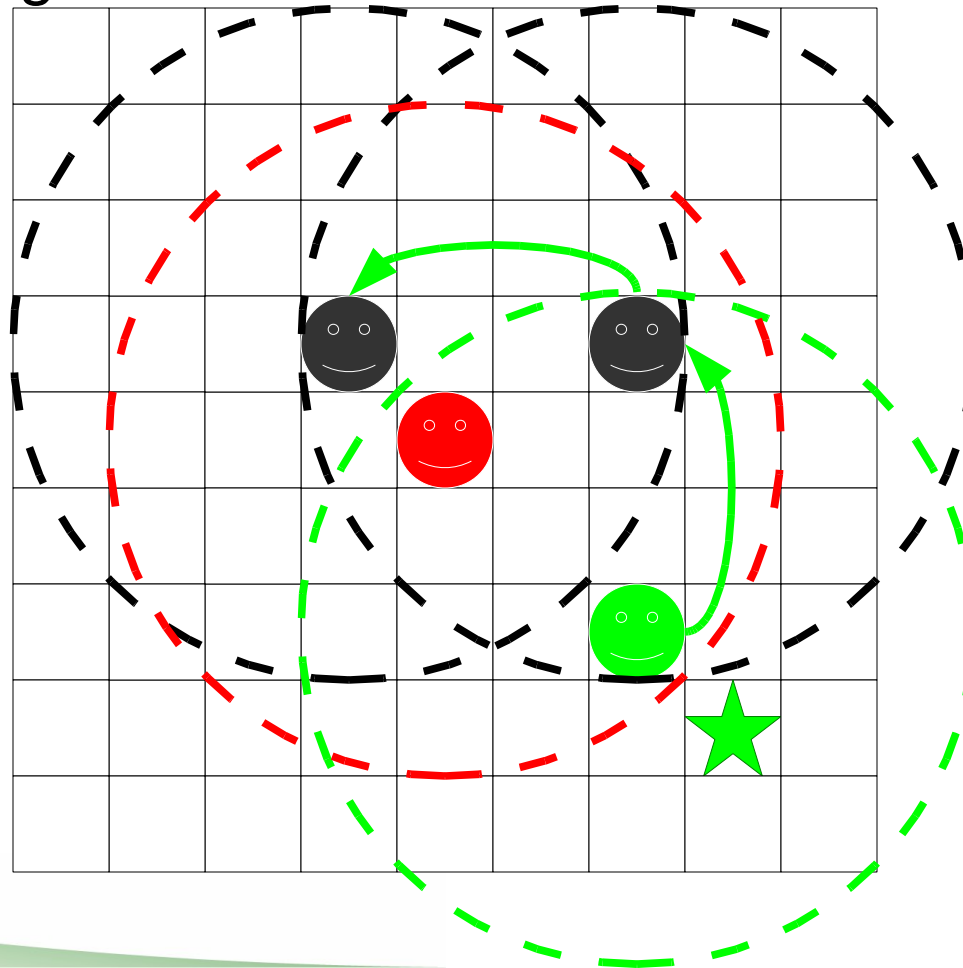


Möglicher Lösungsansatz

- Verteilung des Rewards an verwandte Agenten
 - Lokalpopulationen bilden eine Gruppe, sieht ein Mitglied das Ziel, erhalten alle den Reward
 - Mögliches Problem: Homogenisierung der Regeln wird begünstigt
- Begrenzung der Kommunikationsmittel
 - Aufwand zur Bestimmung des Verwandtschaftsgrad
 - Aber:
 - Keine Übertragung der Sensorinformation nötig
 - Nicht zeitkritisch (genügend Speicherplatz vorausgesetzt kann das Signal bzw. der Reward von Agent zu Agent weitertransportiert werden, auch mit größeren Zeitabständen)

Verteilung des Rewards

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



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)
- Komplexere Sensoren (Abstände, Richtungen)

Überlegungen zur Anwendung

- TODO

Vielen Dank für die
Aufmerksamkeit :)