

eXtended Classifier Systems (XCS) in dynamischen Multiagenten- Überwachungsszenarien

Clemens Lode

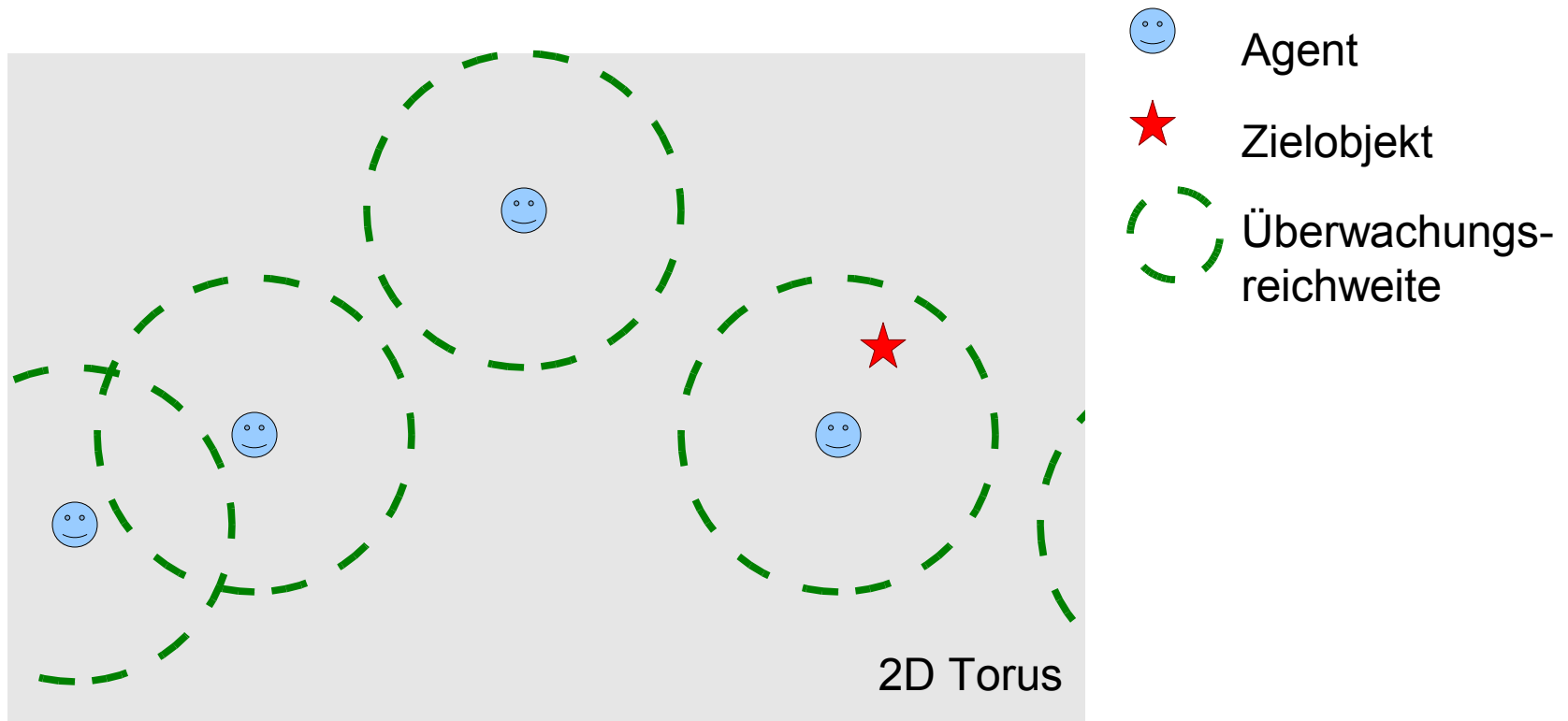
09.04.2009

Überblick

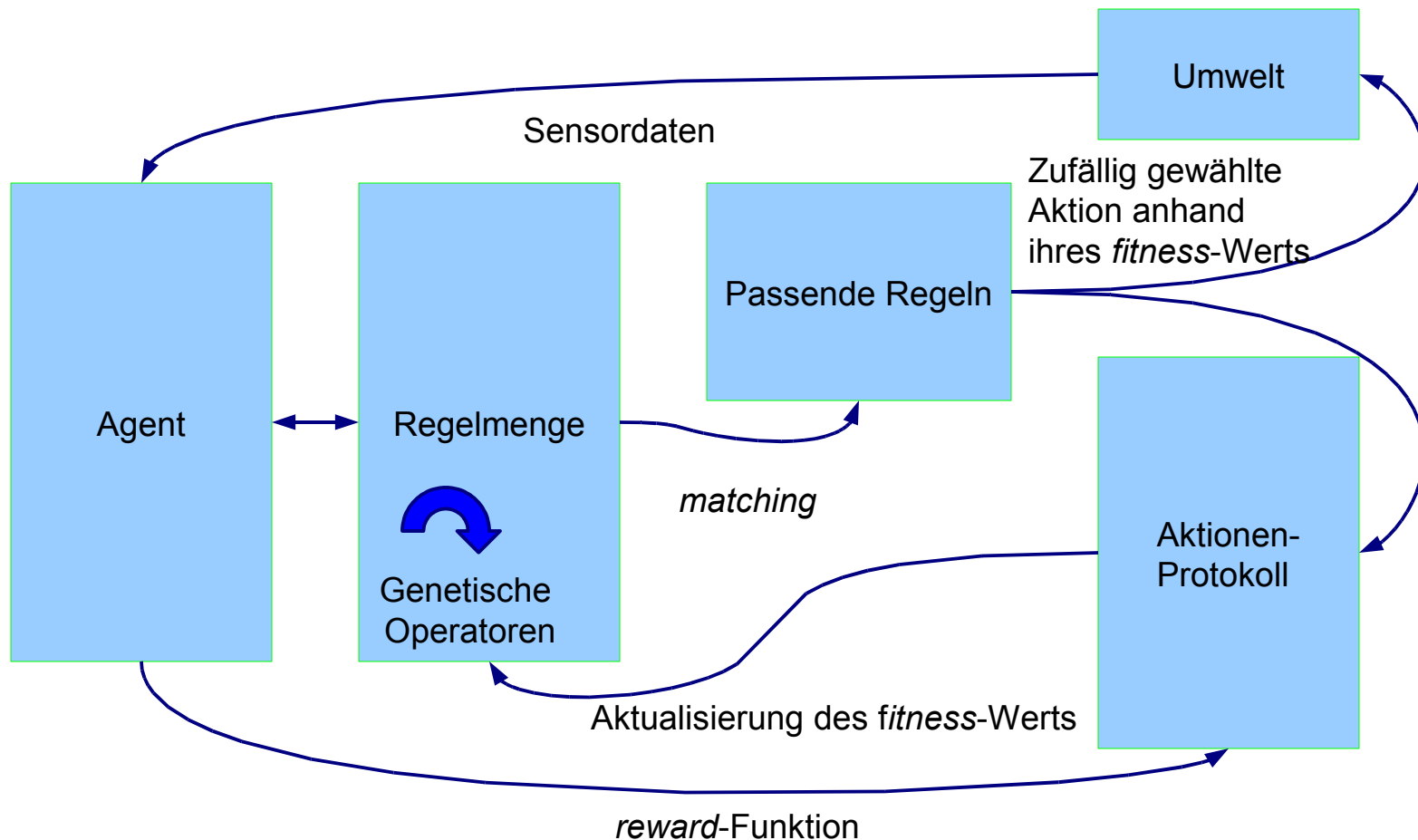
- Problemstellung und Übersicht zu XCS
- Beispiele für *multi-* und *single-step* Verfahren in XCS
- Markow-Eigenschaft
- Überwachungsszenario
- Szenarien- und Sensorenbeschreibung
- Heuristiken des Zielobjekts und der Agenten
- Analyse der lokalen Bewertungsfunktion
- XCS-Variante für Überwachungsszenarien (SXCS)
- Vergleich XCS Standardimplementation mit SXCS
- Demonstration des Simulationsprogramms
- *Ausblick*: Delayed SXCS (DSXCS), DSXCS mit Kommunikation und sonstige Erweiterungen

Allgemeine Problemstellung

- Überwache Zielobjekt möglichst lange mit beliebigem Agenten.
- **Gesamtqualität** = Anteil der überwachten Zeit an Gesamtzeit

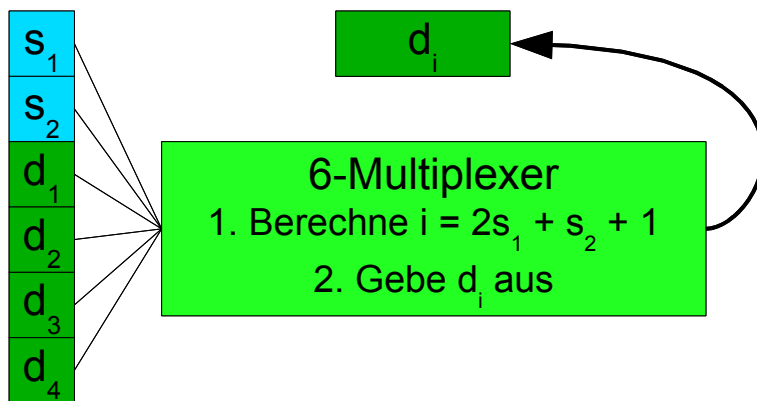


Schematische Darstellung eines XCS



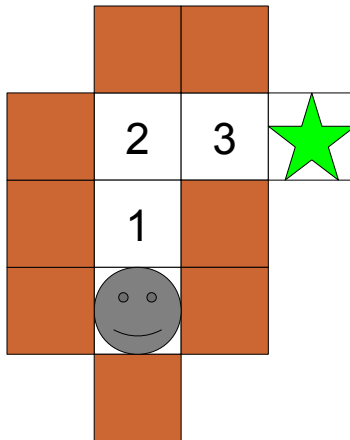
Beispiel für XCS *single-step* Verfahren

- 6-Multiplexer:
 - zwei Steuerbits wählen aus 4 Datenbits ein Datenbit
 - Ausgabe des Datenbits
- Ziel:
 - Darstellung als *classifier system*
- Merkmale:
 - Globale Information verfügbar
 - Nur ein Lernschritt pro Problem nötig / möglich
 - Sofortige Bewertung jeder Aktion



Beispiel XCS *multi-step* Verfahren

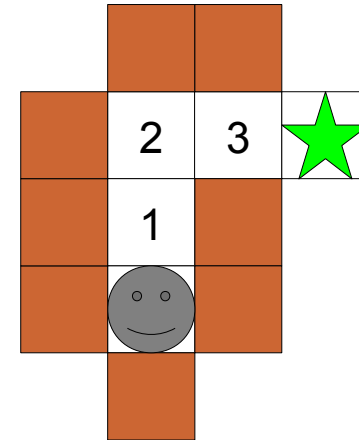
- *Maze-Problem*
 - Finde Weg zum Zielobjekt in einem Labyrinth
- Ziel:
 - Optimale Belegung für *classifier set* Liste für minimalen Weg vom Start zum Ziel
- Merkmale:
 - Begrenzte, lokale Information
 - Bewertung nur im letzten Schritt möglich
 - Zielobjekt bewegt sich nicht
 - Transformation in *single-step* Verfahren durch sukzessive Weitergabe der Bewertung



Beispiel XCS *multi-step* Verfahren

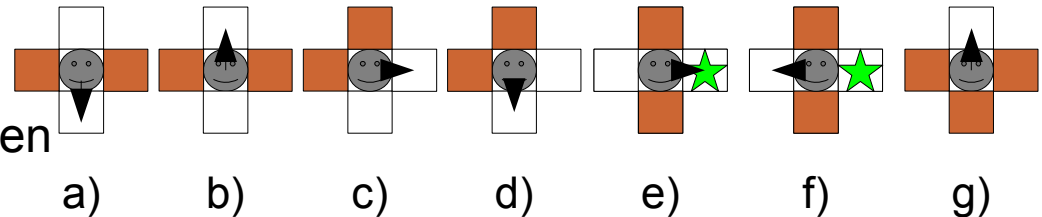
- Ablauf:

- Zufällige Wahl der *classifier*
- **e)** erhält positive Bewertung.
- **e)** wird mit höherer Wahrscheinlichkeit als **f)** an Position 3 gewählt.
- **c)** erhält von **e)** positive Bewertung.
- **c)** wird an Position 2 mit höherer Wahrscheinlichkeit als **d)** gewählt.



– ...

→ optimales Verhalten gefunden

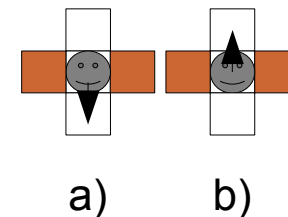
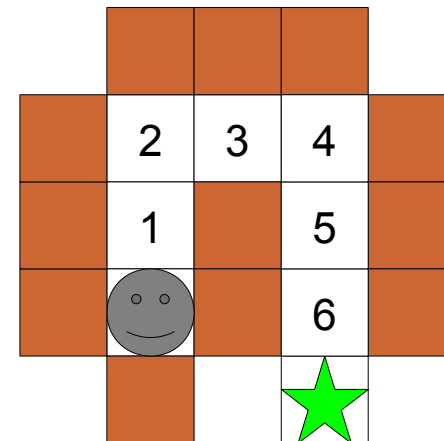


classifier set

Markow-Eigenschaft (1)

- Definition:
 - Vergangene Entscheidungen / Sensordaten sind für optimale Entscheidung nicht relevant.
 - Mit Markow-Eigenschaft kann optimales Verhalten erreicht werden
- Szenario besitzt keine Markow-Eigenschaft:
 - Identische Sensordaten auf Position 1 und 5
→ *classifier a)* und *b)* in Konkurrenz
 - Optimales Verhalten erfordert Unterscheidung beider Positionen

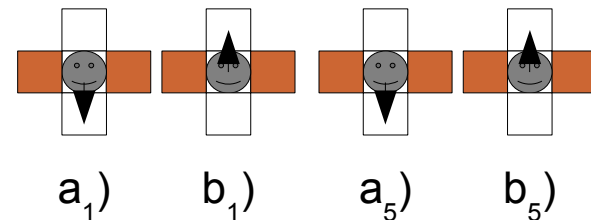
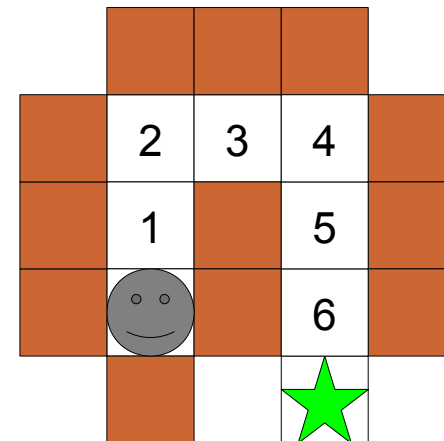
- Szenario ohne Markow-Eigenschaft



Markow-Eigenschaft (2)

- Idee:
 - Interner Zustand, der durch Regeln erkannt und verändert werden kann
- Einfachstes Beispiel:
 - Zähler der Schritte
 - Hohe Bewertung von $b_1)$ und $a_5)$ führt zum Erfolg
 - Optimales Verhalten kann über Darstellung mit Zähler erreicht werden

- Szenario ohne Markow-Eigenschaft



Überwachungsszenario (1)

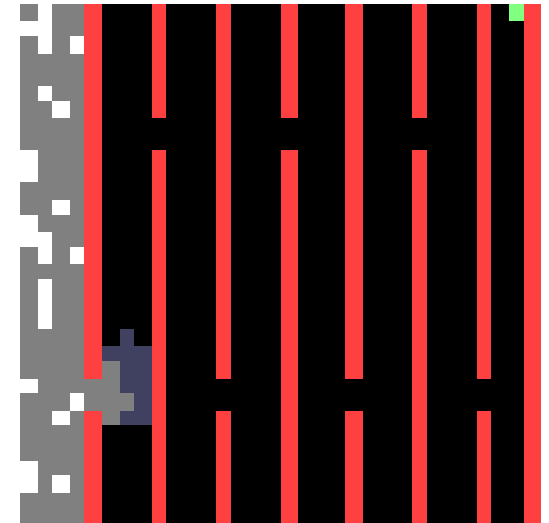
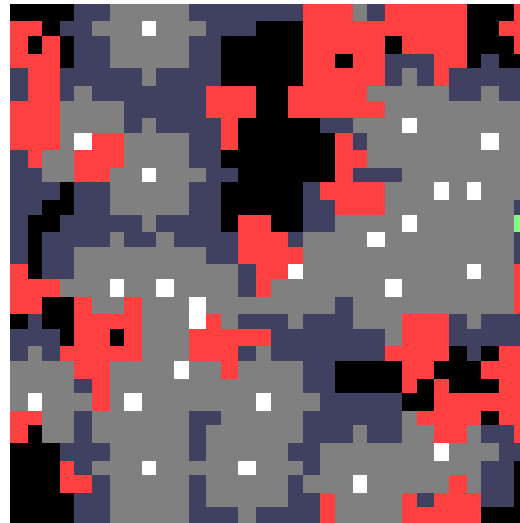
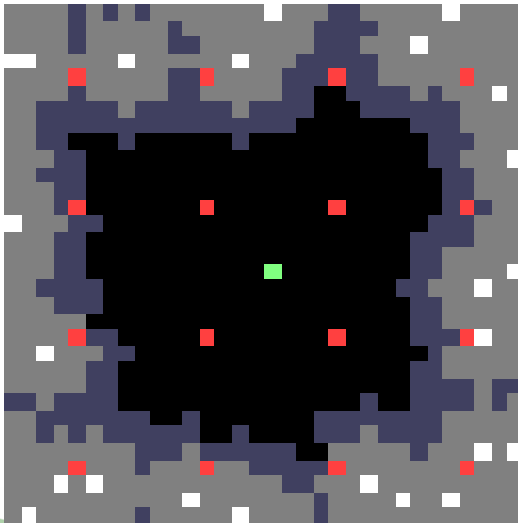
- Multiagenten-Überwachungsszenario:
 - Zielobjekt von diesem oder von anderen Agenten in Überwachungsreichweite
- Ziel:
 - Optimale Belegung für *classifier set* Liste für Maximierung der Überwachungszeit
- Ansatz:
 - Speicherung bisheriger Aktionen, direkte Weitergabe bei positiver Bewertung
- Merkmale:
 - Zielobjekt bewegt sich
 - Begrenzte, lokale Information
 - Bewertung nur selten möglich
 - Allgemeines Verhalten muss gelernt werden.
 - Szenario läuft kontinuierlich weiter.
 - Gleiche Situationen wiederholen sich nur bedingt.
 - Markow-Eigenschaft nicht herstellbar:
 - Globale Information nie verfügbar (dynamisch)!

Überwachungsszenario (2)

- Multiagenten-Überwachungsszenario:
 - **Zielobjekt von diesem oder von anderen Agenten in Überwachungsreichweite**
- Ziel:
 - Optimale Belegung für *classifier* set Liste für Maximierung der Überwachungszeit
- Ansatz:
 - Speicherung bisheriger Aktionen, direkte Weitergabe bei positiver Bewertung
- Schwierigkeiten:
 - Darstellung eines globalen “Ziels” in lokaler Bewertungsfunktion
 - Kein konkretes Ziel sondern dauerhaftes Verhalten soll erreicht werden.
 - Welche lokale Bewertungsfunktion löst das Problem “gut”?
- Ansatz:
 - Untersuchung spezieller Szenarien anhand von Heuristiken

Szenarien

- Legende:
 - Weiß (Agenten)
 - Grün (Zielobjekt)
 - Grau (überwachtes Gebiet)
 - Blau (maximale Sichtweite)
 - (abgebildet sind 32x32 mit 24 Agenten)
- Untersuchte Szenarien:
 - Säulenszenario
 - Szenario mit zufällig verteilten Hindernissen
 - Schwieriges Szenario
 - 16x16 mit 8 Agenten



Sensoren

- Erkennung

- Sensordaten werden von *classifier* erkannt, die jeweils identische Einträge oder Einträge mit Platzhalter “#” besitzen

- Beispiel:

Sensordaten

10 00 00 00 . 00 00 11 00 . 00 11 00 11

werden erkannt von z.B.

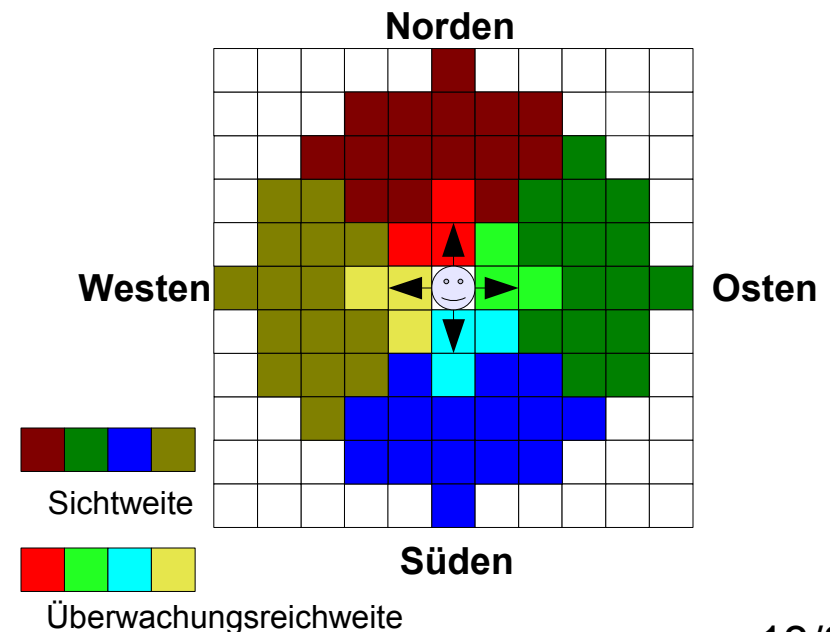
10 00 00 00 . ## ## ## ## . 00 ## ## ##

. ## ## #1 00 . 00 11 ##

#0 ## ## ## . ## ## 01 ## . ## 11 ## 11

- Sensoren:

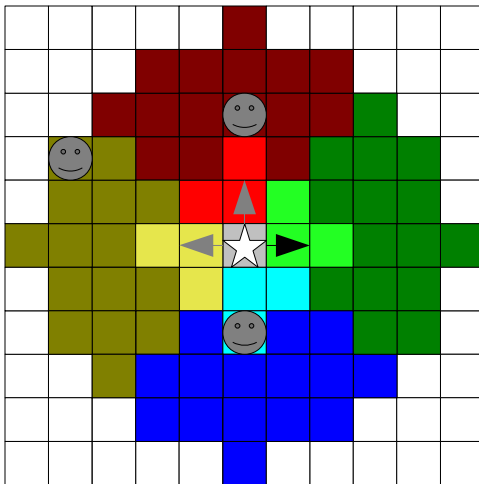
- Binärsensor für Sichtweite und Überwachungsreichweite
- jeweils in 4 Richtungen
- jeweils für 3 Objekttypen
→ differenzierteres Verhalten



Zielobjekt und Agenten

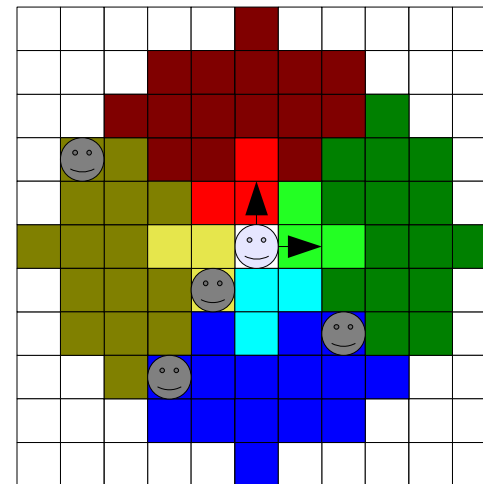
- Typen des Zielobjekts

- Einfache Richtungsänderung
- Intelligentes Zielobjekt
- Ohne Richtungsänderung



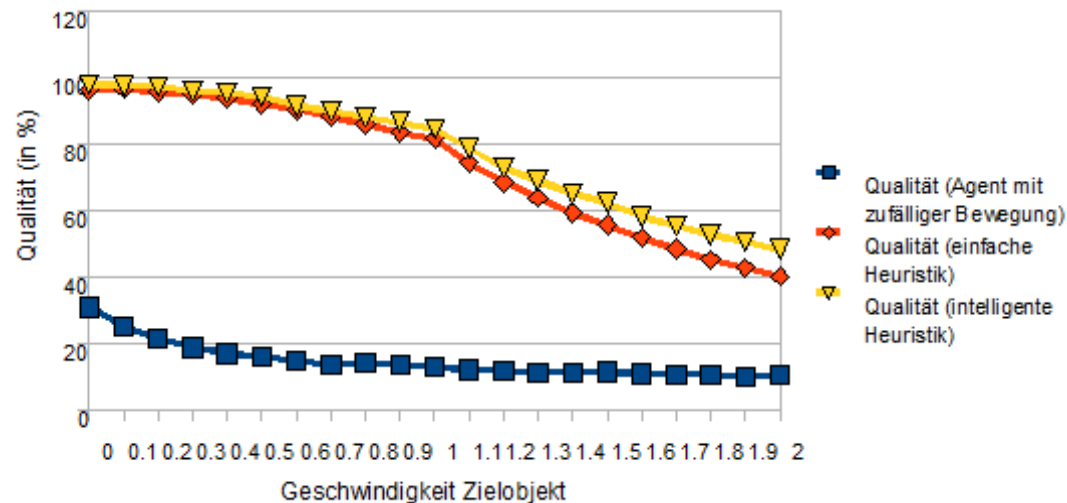
- Typen von Agenten

- Zufällige Bewegung
- Einfache Heuristik
- Intelligente Heuristik



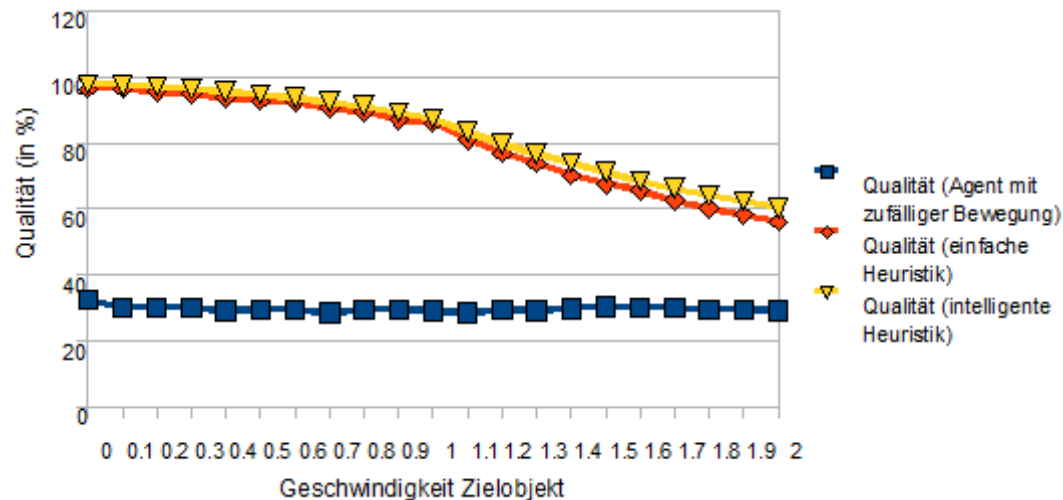
Vergleich der Heuristiken (1)

- Ergebnis:
 - Intelligente Heuristik deutlich besser als einfache Heuristik, besonders bei höheren Geschwindigkeiten
 - Knick bei Geschwindigkeit 1, nicht mehr alleinige Verfolgung
- Szenario
 - Säulenszenario
 - Intelligentes Zielobjekt



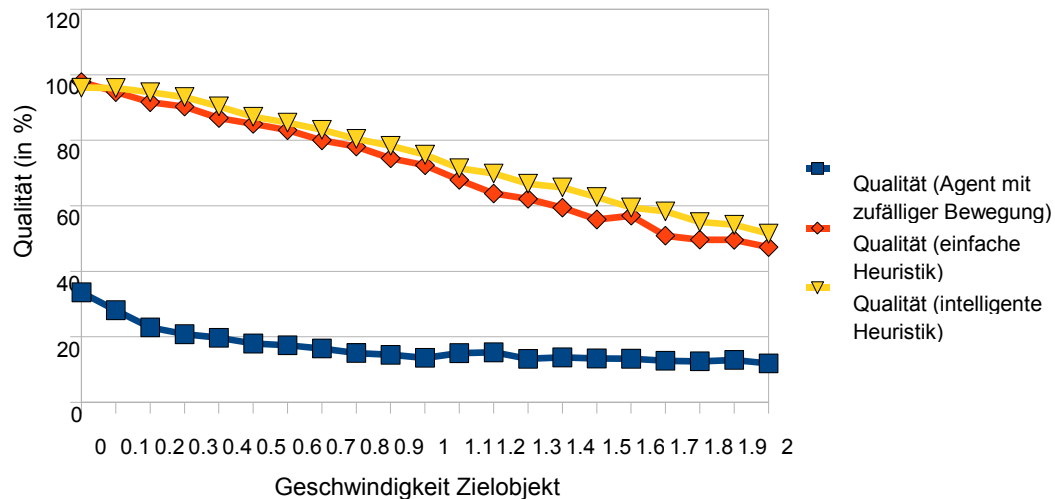
Vergleich der Heuristiken (2)

- Ergebnis:
 - Intelligente Heuristik deutlich besser als einfache Heuristik, besonders bei höheren Geschwindigkeiten
 - Knick bei Geschwindigkeit 1, nicht mehr alleinige Verfolgung
- Szenario
 - Säulenszenario
 - Zielobjekt mit einfacher Richtungsänderung



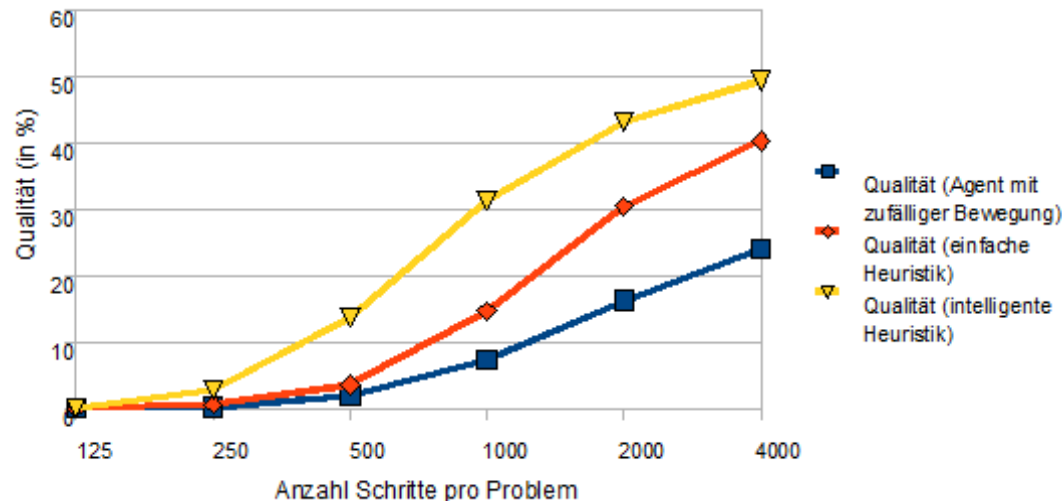
Vergleich der Heuristiken (3)

- Ergebnis:
 - Intelligente Heuristik etwas besser als einfache Heuristik
 - Knick bei Geschwindigkeit 1 kaum auszumachen
- Szenario
 - Szenario mit zufällig verteilten Hindernissen
 - Intelligentes Zielobjekt



Vergleich der Heuristiken (5)

- Ergebnis:
 - Intelligente Heuristik deutlich besser als einfache Heuristik
 - Versucht sich von Agenten zu entfernen und erreicht so automatisch den Zielbereich
- Szenario
 - Schwieriges Szenario
 - Zielobjekt ohne Richtungsänderung
 - Betrachtung unterschiedlicher Schrittzahlen pro Durchgang

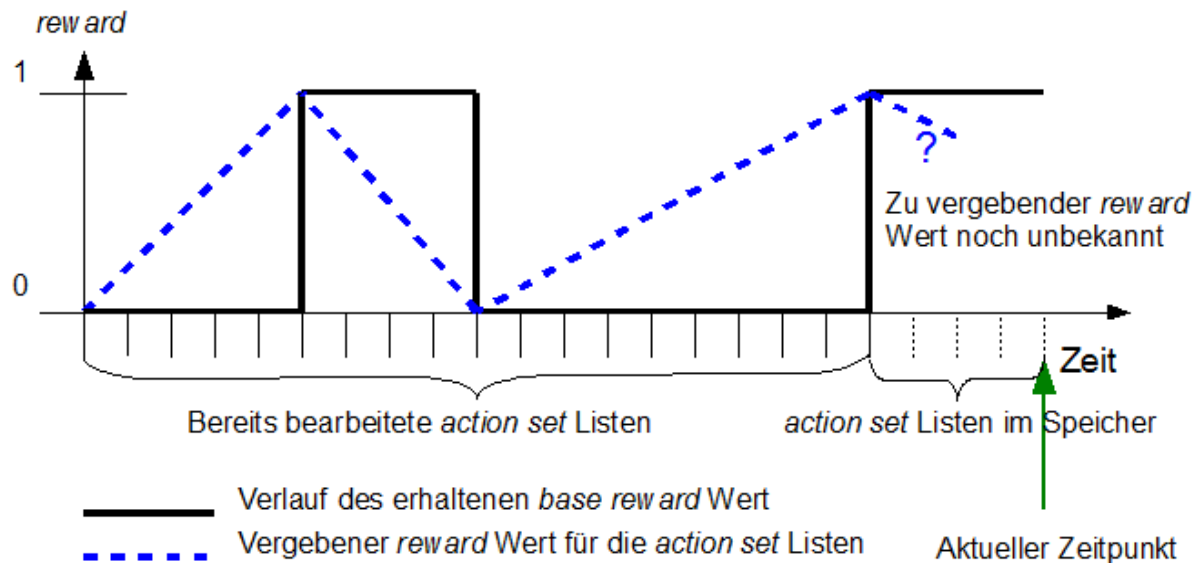


Analyse der lokalen Bewertungsfunktion

- Einfache Heuristik
 - bewertet Nähe zum Zielobjekt als “gut”
- Intelligente Heuristik
 - bewertet (zusätzlich) Abwesenheit von Agenten als “gut”
- Umsetzung für XCS:
 - Wahl intelligenter Heuristik als Modell (beste Qualität)
 - *Allerdings*: Mehrwertige nicht vollständig auf binäre Bewertungsfunktion abbildbar
- Bewertungsfunktion:
 - Bewertung “1” wenn Zielobjekt in Sichtweite oder kein Agent in Sichtweite,
 - Bewertung “0” sonst.

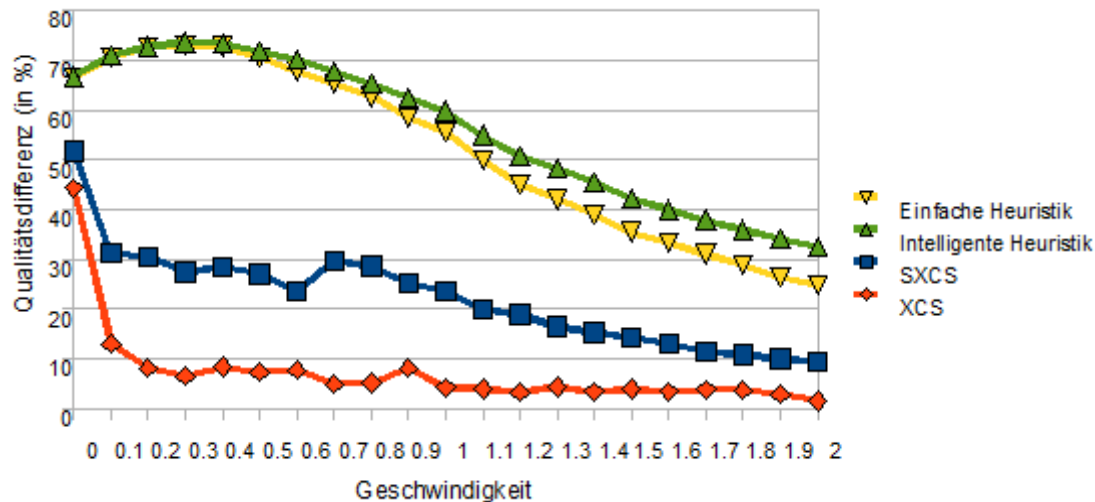
Supervision eXtended Classifier System (SXCS)

- **Ansatz:**
 - Speicherung bisheriger Aktionen, direkte Weitergabe bei positiver Bewertung
- **Erweiterung**
 - ab- bzw. aufsteigender *reward* Wert nach Änderung der Bewertung



Vergleich XCS, SXCS

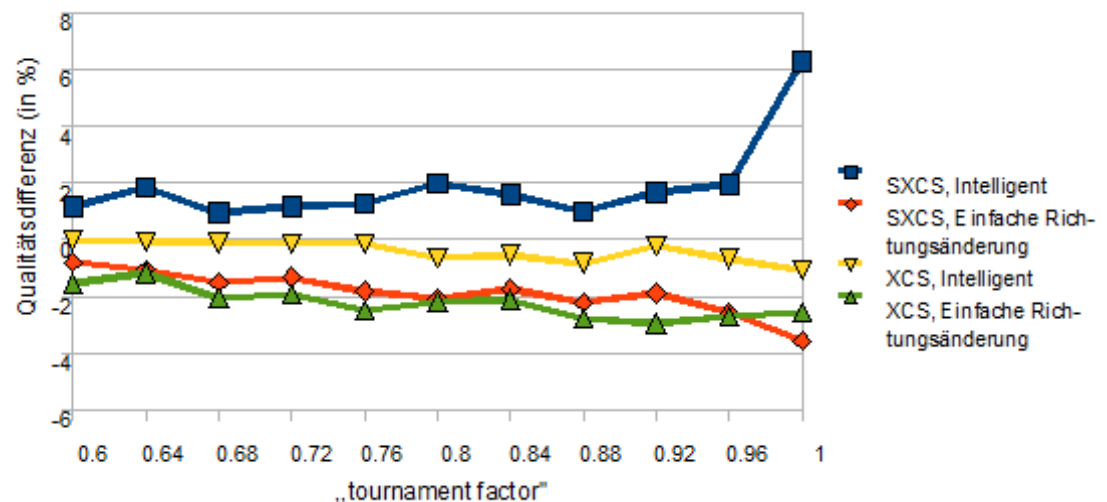
- Ergebnisse:
 - Knick früher bei 0,7
 - SXCS deutlich besser als XCS
 - XCS fast wie Algorithmus mit zufälliger Bewegung
- Qualitätsdifferenz:
 - Differenz der Qualität zur Qualität des Algorithmus mit zufälliger Bewegung



Vergleich XCS, SXCS

Szenario mit zufällig verteilten Hindernissen

- Ergebnis:
 - Geringe Qualitätsdifferenz für SXCS und XCS
 - Großer Anteil blockierter Bewegungen (bis 70%)
- Durchführung:
 - Abwechselnde Turnier- und proportionale Auswahl (jeweils bei einem Ereignis)
- Problem:
 - durch Auswahl der besten Regel und ohne Malus für Kollisionen
 - Fähigkeiten des Zielobjekts Hindernissen auszuweichen



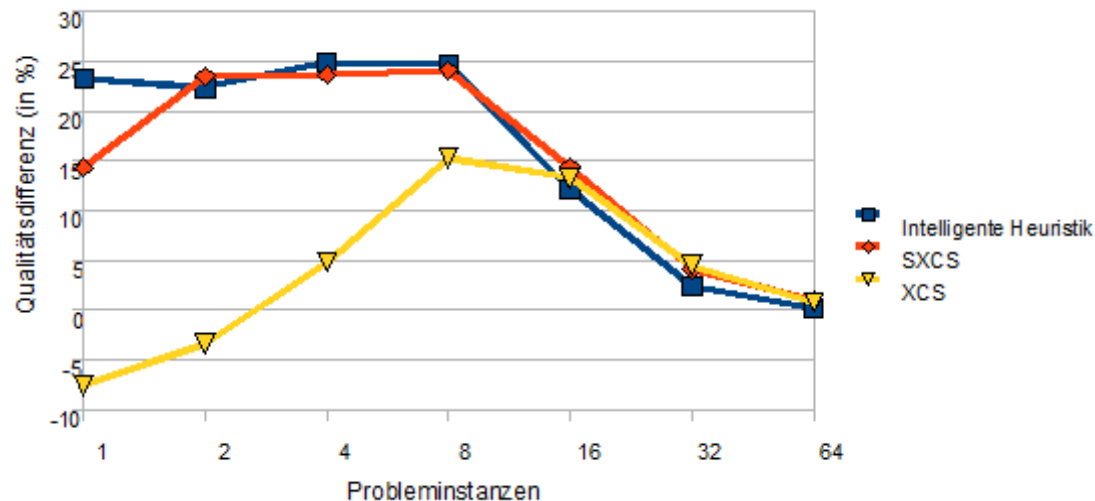
Vergleich XCS, SXCS (schwieriges Szenario)

- Ergebnisse:

- SXCS und intelligente Heuristik gleichauf
- XCS lernt langsamer, ist bei hoher Zahl Probleminstanzen gleichauf

- Vergleich:

- 8000 Schritte
- Unterschiedliche Zahl Probleminstanzen (= Neustart und Beginn aller Agenten am linken Rand)



Simulationsprogramm

Agent Configuration File Editor v1.00

Problem definition

Tests

Random Seed

Experiments

Problems

Steps

Number of agents

Grid

Sight range

Reward range

Max X / Max Y

☐ Random scenario

Grid percentage

Connection factor

☐ Pillar scenario

☒ Difficult scenario

Goal Agent Movement

☐ Total random

☐ Random neighbor

☐ One direction change

☐ Intelligent

☒ Always same direction

☐ SXCS

Speed

Agent type

Reward model

Stack size

☐ Use max prediction?

☒ Use quadratic reward

Algorithm

☐ Randomized

☐ Simple heuristic

☐ Intelligent heuristic

☐ Standard XCS

☒ Surveillance XCS

☐ Delayed SXCS

Exploration Mode

☐ Always explore

☐ Always explore (random)

☐ Always exploit

☐ Always exploit (best)

☒ Switch (explore)

☐ Switch (exploit)

☐ Random explore/exploit

Communication

☒ No external reward

☐ Reward all equally

☐ Egoistic relation

☐ Log output ☐ Create GIF

LCS parameters

Classifier subsumption and deletion

Max population N

Fraction mean fitness δ

Deletion threshold θ_{del}

Subsumption threshold θ_{sub}

Covering # probability P_g

Tournament probability

☒ Action set subsumption

☐ Random start

GA parameters

☒ Use genetic algorithm?

☒ GA subsumption

GA threshold θ_{GA}

Mutation probability μ

Prediction error reduction

Fitness Reduction

Fitness init and update

Fitness init F_i

Prediction init p_i

☐ Prediction init adaption

Prediction error init ϵ_i

Accuracy equal below ϵ_0

Accuracy calculation a

Accuracy power v

Prediction discount y

Learning rate β

| Conf... | Configu... | Spread | Average | Spread | Average | Spread | Average | Covered | Wasted | Goal ju... | Wasted | Half goa... | Goal pe... | |
|---------|------------|--------|---------|--------|---------|--------|---------|---------|--------|------------|--------|-------------|------------|--------|
| agent-2 | 108293 | 136.24 | 233.85 | 2.2903 | 5.8243 | 5.0070 | 5.9978 | 0.0 | 47.490 | 7.5079 | 0.0025 | 25.188 | 25.821 | 18.276 |
| agent-2 | 108293 | 140.54 | 256.91 | 2.2589 | 5.7982 | 5.0173 | 6.1413 | 0.0 | 50.714 | 5.8466 | 0.0025 | 23.537 | 26.438 | 20.239 |
| agent-2 | 108293 | 145.55 | 249.58 | 2.2815 | 5.7648 | 1.9991 | 6.1670 | 0.0 | 50.082 | 5.7839 | 0.0015 | 25.602 | 26.401 | 19.100 |
| agent-2 | 108293 | 168.17 | 226.98 | 2.3031 | 5.8047 | 1.9832 | 6.0145 | 0.0 | 47.319 | 7.5049 | 0.0020 | 25.672 | 25.038 | 17.560 |
| agent-2 | 108293 | 137.75 | 280.59 | 2.2618 | 5.8465 | 2.0171 | 6.2036 | 0.0 | 51.396 | 6.4316 | 0.0025 | 23.303 | 28.399 | 20.283 |
| agent-2 | 108293 | 170.23 | 413.38 | 2.2859 | 5.8244 | 2.0381 | 6.0200 | 0.0 | 49.378 | 6.6608 | 0.0200 | 29.823 | 33.162 | 29.012 |
| agent-2 | 108293 | 160.99 | 340.87 | 2.2485 | 5.6511 | 2.0088 | 5.9250 | 0.0 | 47.406 | 7.4064 | 0.0085 | 25.195 | 27.916 | 24.719 |
| agent-2 | 108293 | 187.71 | 382.65 | 2.2927 | 5.8096 | 2.0307 | 6.0594 | 0.0 | 49.362 | 6.6850 | 0.0135 | 29.194 | 33.409 | 27.950 |
| agent-2 | 108293 | 173.00 | 399.12 | 2.2786 | 5.8442 | 2.0491 | 6.0258 | 0.0 | 49.983 | 6.5764 | 0.0120 | 28.060 | 35.260 | 29.244 |
| agent-2 | 108293 | 163.27 | 326.93 | 2.2565 | 5.6692 | 2.0235 | 5.9192 | 0.0 | 48.066 | 7.3389 | 0.0070 | 24.367 | 30.638 | 23.774 |
| agent-2 | 108293 | 173.76 | 306.89 | 2.3147 | 5.7109 | 2.0109 | 6.0067 | 0.0 | 48.332 | 6.9959 | 0.0055 | 28.956 | 31.129 | 23.609 |
| agent-2 | 108293 | 179.76 | 278.40 | 2.3037 | 5.7397 | 2.0167 | 6.0393 | 0.0 | 48.926 | 6.8516 | 0.0020 | 29.545 | 28.779 | 21.586 |
| agent-2 | 108293 | 152.26 | 283.32 | 2.2703 | 5.7910 | 2.0288 | 6.0759 | 0.0 | 50.263 | 6.6102 | 0.0040 | 26.932 | 30.479 | 22.067 |
| agent-2 | 108293 | 139.50 | 276.24 | 2.2568 | 5.8722 | 2.0285 | 6.1849 | 0.0 | 51.527 | 6.3845 | 0.0040 | 23.039 | 30.328 | 21.192 |
| agent-2 | 108293 | 166.50 | 314.01 | 2.2586 | 5.8765 | 2.0319 | 6.1649 | 0.0 | 51.570 | 6.3647 | 0.0034 | 23.163 | 31.802 | 22.819 |
| agent-2 | 108293 | 147.97 | 272.05 | 2.2638 | 5.8145 | 2.0230 | 6.1474 | 0.0 | 50.829 | 6.6066 | 0.0025 | 24.545 | 28.038 | 19.597 |
| agent-2 | 108293 | 129.16 | 266.37 | 2.2503 | 5.8850 | 2.0333 | 6.2164 | 0.0 | 51.939 | 6.3524 | 5.0E-4 | 20.808 | 27.405 | 19.784 |
| agent-2 | 108293 | 144.71 | 291.01 | 2.2542 | 5.8849 | 2.0331 | 6.2420 | 0.0 | 51.823 | 6.3112 | 0.0045 | 23.525 | 29.475 | 20.864 |
| agent-2 | 108293 | 123.45 | 321.14 | 2.2459 | 5.8745 | 2.0305 | 6.2034 | 0.0 | 51.852 | 6.4824 | 0.0049 | 22.151 | 33.908 | 23.743 |
| agent-2 | 108293 | 117.57 | 306.11 | 2.2468 | 5.8808 | 2.0314 | 6.2485 | 0.0 | 52.198 | 6.4352 | 0.0040 | 21.148 | 32.122 | 22.825 |
| agent-2 | 108293 | 129.60 | 306.80 | 2.2452 | 5.8813 | 2.0257 | 6.2746 | 0.0 | 52.409 | 6.4064 | 0.0045 | 20.033 | 30.465 | 21.544 |
| agent-2 | 108293 | 123.74 | 294.32 | 2.2411 | 5.8915 | 2.0227 | 6.2407 | 0.0 | 52.688 | 6.3789 | 0.0059 | 19.433 | 31.721 | 21.109 |
| agent-2 | 108293 | 116.58 | 260.19 | 2.2336 | 5.8701 | 2.0086 | 6.2929 | 0.0 | 52.762 | 6.3578 | 0.0029 | 19.033 | 27.887 | 18.390 |
| agent-2 | 108293 | 123.72 | 258.72 | 2.2263 | 5.8646 | 2.0183 | 6.3224 | 0.0 | 52.862 | 6.3489 | 0.0034 | 18.830 | 28.485 | 18.390 |
| agent-2 | 108293 | 112.24 | 273.51 | 2.2180 | 5.8602 | 2.0160 | 6.2819 | 0.0 | 53.089 | 6.3386 | 0.0020 | 18.618 | 28.455 | 18.647 |
| agent-2 | 108293 | 118.91 | 253.30 | 2.2280 | 5.8663 | 2.0082 | 6.3303 | 0.0 | 52.950 | 6.3177 | 0.0034 | 18.681 | 27.685 | 18.160 |
| agent-2 | 108293 | 119.91 | 216.26 | 2.2166 | 5.8570 | 2.0040 | 6.3572 | 0.0 | 53.138 | 6.2906 | 0.0010 | 18.492 | 25.152 | 16.311 |
| agent-2 | 108293 | 133.64 | 244.57 | 2.2189 | 5.8564 | 1.9992 | 6.3585 | 0.0 | 53.040 | 6.2984 | 0.0010 | 18.491 | 24.772 | 16.836 |
| agent-2 | 108293 | 126.64 | 252.11 | 2.2201 | 5.8787 | 2.0166 | 6.3366 | 0.0 | 53.149 | 6.2493 | 0.0020 | 18.621 | 26.658 | 17.887 |
| agent-2 | 108293 | 112.53 | 237.52 | 2.2276 | 5.8618 | 1.9993 | 6.3572 | 0.0 | 53.001 | 6.3226 | 0.0034 | 18.554 | 24.891 | 16.619 |
| agent-2 | 108293 | 117.49 | 240.60 | 2.2208 | 5.8502 | 2.0079 | 6.3307 | 0.0 | 52.981 | 6.3329 | 0.0034 | 18.586 | 26.138 | 17.102 |
| agent-2 | 108293 | 147.71 | 448.64 | 2.2739 | 5.8936 | 2.0680 | 6.0433 | 0.0 | 50.656 | 6.4418 | 0.025 | 27.740 | 38.18 | 31.569 |
| agent-2 | 108293 | 233.10 | 150.76 | 2.2106 | 6.0776 | 2.0241 | 6.2935 | 0.0 | 53.579 | 5.2214 | 0.0 | 42.975 | 13.200 | 10.9 |
| agent-2 | 108293 | 233.10 | 150.76 | 2.2106 | 6.0776 | 2.0241 | 6.2935 | 0.0 | 53.579 | 5.2214 | 0.0 | 42.975 | 13.200 | 10.9 |
| agent-2 | 108293 | 159.62 | 223.01 | 2.3505 | 5.6323 | 1.9610 | 6.2014 | 0.0 | 46.730 | 7.0085 | 0.025 | 41.129 | 22.689 | 19.71 |
| agent-2 | 108293 | 1049.1 | 1327.0 | 2.4059 | 5.8333 | 2.4178 | 5.3098 | 0.0 | 47.495 | 7.1178 | 0.0050 | 53.221 | 65.86 | 48.54 |
| agent-2 | 108254 | 51.404 | 393.97 | 2.4628 | 5.8750 | 1.9705 | 6.1833 | 0.0 | 63.213 | 7.4361 | 0.0 | 43.674 | 25.912 | 26.513 |
| agent-2 | 108254 | 54.575 | 448.07 | 2.2783 | 6.0819 | 2.0445 | 6.0711 | 0.0 | 68.844 | 6.6186 | 5.0E-4 | 12.493 | 30.813 | 30.815 |
| agent-2 | 108254 | 60.399 | 440.79 | 2.2799 | 6.0855 | 2.0424 | 6.0832 | 0.0 | 68.863 | 6.5881 | 0.0 | 12.625 | 30.144 | 30.253 |
| agent-2 | 108254 | 57.477 | 448.38 | 2.2758 | 6.0847 | 2.0435 | 6.0795 | 0.0 | 68.977 | 6.5967 | 0.0 | 11.947 | 30.376 | 30.854 |
| agent-2 | 108254 | 63.435 | 441.92 | 2.2796 | 6.0847 | 2.0429 | 6.0875 | 0.0 | 68.893 | 6.6052 | 0.0 | 12.052 | 30.724 | 30.472 |
| agent-2 | 108254 | 65.535 | 480.95 | 2.2761 | 6.0810 | 2.0503 | 6.0606 | 0.0 | 69.915 | 6.6348 | 0.0 | 11.837 | 31.325 | 31.266 |
| agent-2 | 108254 | 55.453 | 451.92 | 2.2764 | 6.0809 | 2.0478 | 6.0777 | 0.0 | 68.942 | 6.6230 | 0.0 | 11.592 | 30.685 | 30.906 |
| agent-2 | 108254 | 57.325 | 442.61 | 2.2770 | 6.0812 | 2.0442 | 6.0852 | 0.0 | 69.962 | 6.6074 | 0.0 | 11.606 | 30.907 | 30.679 |
| agent-2 | 108254 | 60.949 | 433.47 | 2.2722 | 6.0729 | 2.0376 | 6.0824 | 0.0 | 69.860 | 6.6388 | 0.0 | 11.618 | 30.303 | 30.142 |

Ausblick: Delayed SXCS (DSXCS)

- Idee:
 - Erweiterung des Speichers und verzögerte Bewertung
- Vorteile:
 - Analyse aller bisherigen Bewertungen möglich
 - Zeitgerechte Behandlung von externen Ereignissen (Kommunikation) möglich
- Nachteil:
 - Verzögerung der Aktualisierung, Agent handelt mit veralteten Werten
- Umsetzung:
 - Bei einem Ereignis werden nur die errechneten *reward* Werte gespeichert
 - Ist der Speicher eines DSXCS voll wird nur der letzte classifizierte aktualisiert und entfernt

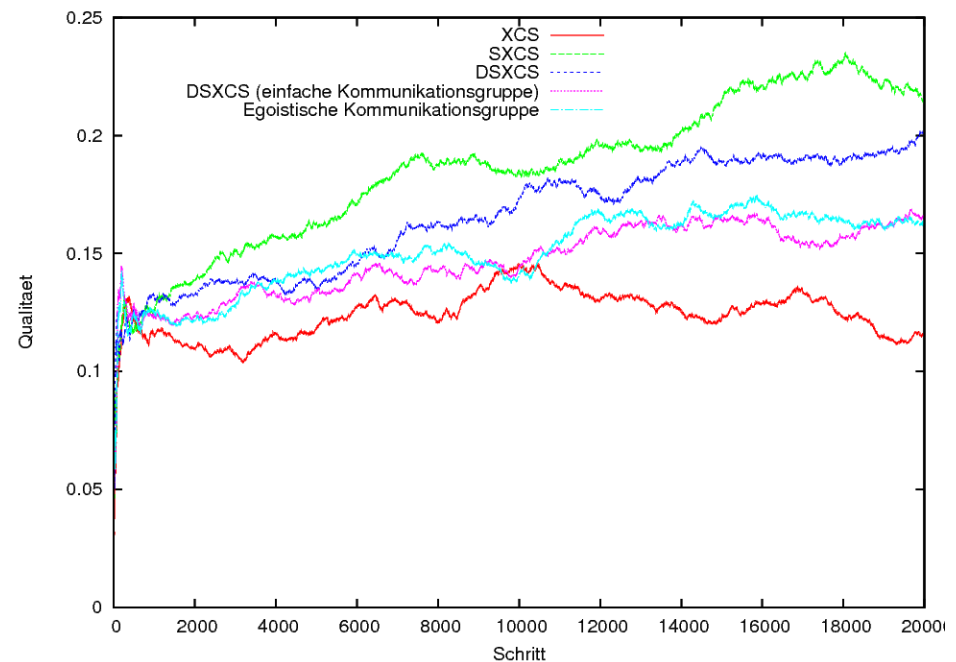
Ausblick: DSXCS mit Kommunikation (1)

- **Idee:**
 - Verhalten von Agenten ohne Kontakt zum Zielobjekt trägt u.U. trotzdem zur Qualität bei
→ Weitergabe der Bewertung an andere Agenten
- **Problem:**
 - Ohne Differenzierung entspricht dies z.T. zufälliger Bewertung von Aktionen
→ Gruppenbildung
- **Gruppenbildung**
 - Vergleich des Verhaltens gegenüber anderen Agenten
 - Bevorzugung von Agenten mit ähnlichem Verhalten
- **Ergebnisse**
 - Keine Unterschiede bezüglich der Qualität bemerkbar
 - Insgesamt schlechtere Ergebnisse als SXCS
 - Allerdings deutlich niedrigere Varianz von individuellen Punktzahlen

Ausblick: DSXCS mit Kommunikation (2)

- Ergebnisse:

| Algorithmus | Varianz Punkte | Qualität |
|-------------------------|----------------|----------|
| XCS | 53,96 | 12,41% |
| SXCS | 78,51 | 19,03% |
| DXCS | 72,85 | 16,96% |
| Einzelne Kom.-Gruppe | 49,73 | 14,91% |
| Egoistische Kom.-Gruppe | 47,70 | 15,30% |



Ausblick: Sonstige Erweiterungen

- **Bestehendes System:**

- Betrachtung der Theorie
- Tiefere Analyse der Ausgabe
- Verbesserung der Auswertung der gespeicherten Aktionen (DSXCS)
- Anpassungsfähigkeit von SXCS testen
- Wechsel zwischen explore/exploit Phasen weiter untersuchen

- **Erweiterungen:**

- Rotation als “Platzhalter”
- Test mit verbesserten Sensoren
- Verwendung mehrwertiger Bewertungsfunktion
- Bessere Heuristiken als Modell und Vergleich (z.B. Einbeziehung von Hindernissen)
- Kommunikation, Austausch von Regeln

Vielen Dank für die
Aufmerksamkeit :)

Überblick

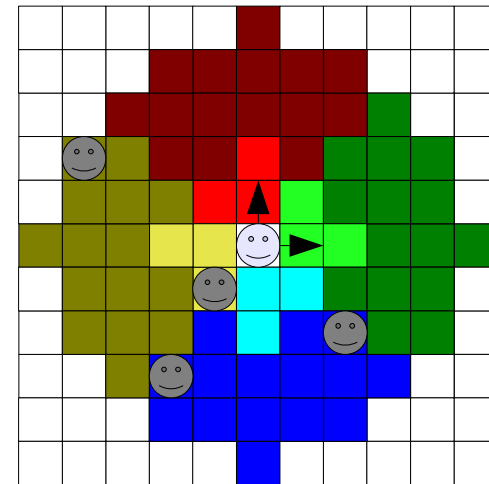
- Problemstellung und Übersicht zu XCS
- Beispiele für *multi-* und *single-step* Verfahren in XCS
- Markow-Eigenschaft
- Überwachungsszenario
- Szenarien- und Sensorenbeschreibung
- Heuristiken des Zielobjekts und der Agenten
- Analyse der lokalen Bewertungsfunktion
- XCS-Variante für Überwachungsszenarien (SXCS)
- Vergleich XCS Standardimplementation mit SXCS
- Demonstration des Simulationsprogramms
- *Ausblick*: Delayed SXCS (DSXCS), DSXCS mit Kommunikation und sonstige Erweiterungen

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)

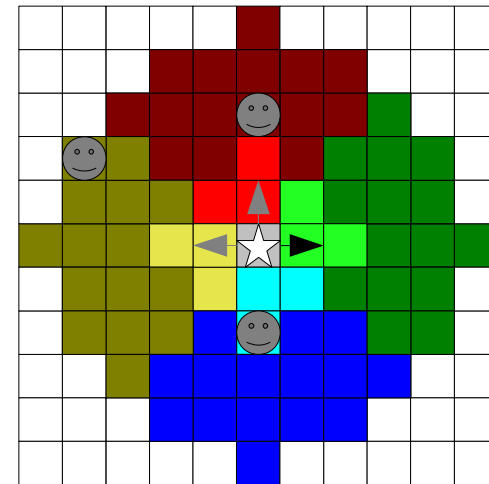
Agent mit intelligenter Heuristik

- **Zielobjekt in Sicht:**
Verhält sich wie einfache Heuristik.
- **Zielobjekt nicht in Sicht:**
Bewegt sich in zufällige Richtung, in der sich kein Agent befindet.



Intelligentes Zielobjekt

- **Keine Agenten in Sicht:**
Bewegt sich zufällig.
- **Agenten in Überwachungsreichweite:** Keine Bewegung in diese Richtung.
- **Agenten in Sichtweite:**
Bewegt sich mit 50% Wahrscheinlichkeit nicht in diese Richtung.



Generalisierung: Wildcards

- Menge der *classifier* kleiner (Effizienz)
- Vermeidung redundanter Informationen
- Beispiel: 1.0111 ► 2 und 0.0111 ► 2
 - Benutze # als “wildcard”: #.0111 ► 2
- Problem: Verhinderung von Informationsverlust
 - 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)

Generierung neuer *classifier*

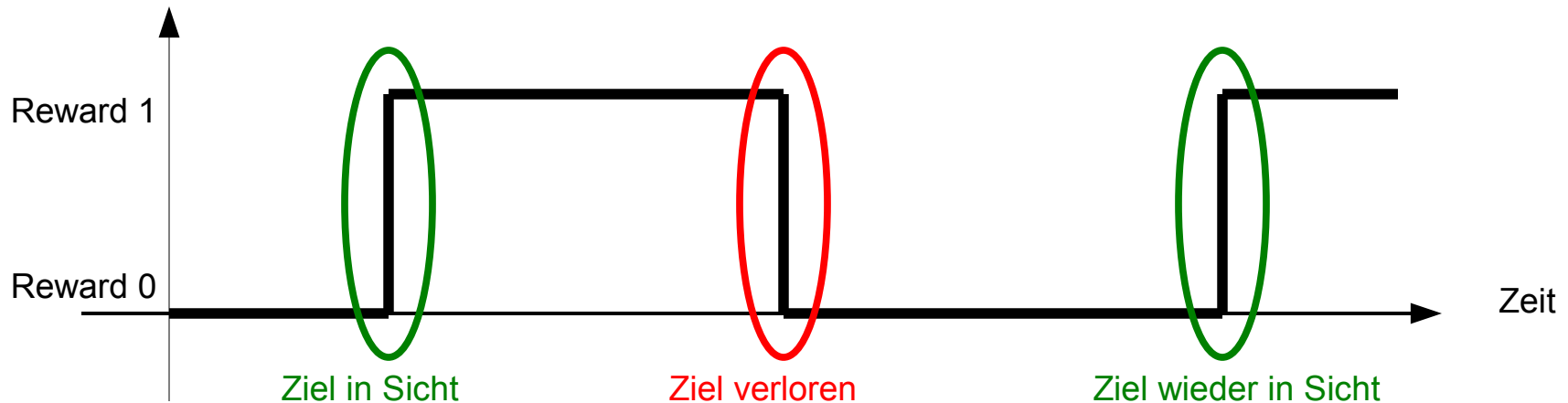
- Drei Quellen für neue *classifier*:
 - Covering
 - Falls kein Matching für ein Classifier gefunden wurde
 - Erstelle neuen, zufälligen Classifier mit passendem Matching
 - Genetischer Algorithmus, Mutation
 - Crossover zwischen bestehenden Classifiern
 - Mutation von bestehenden Classifiern
 - Austausch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

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

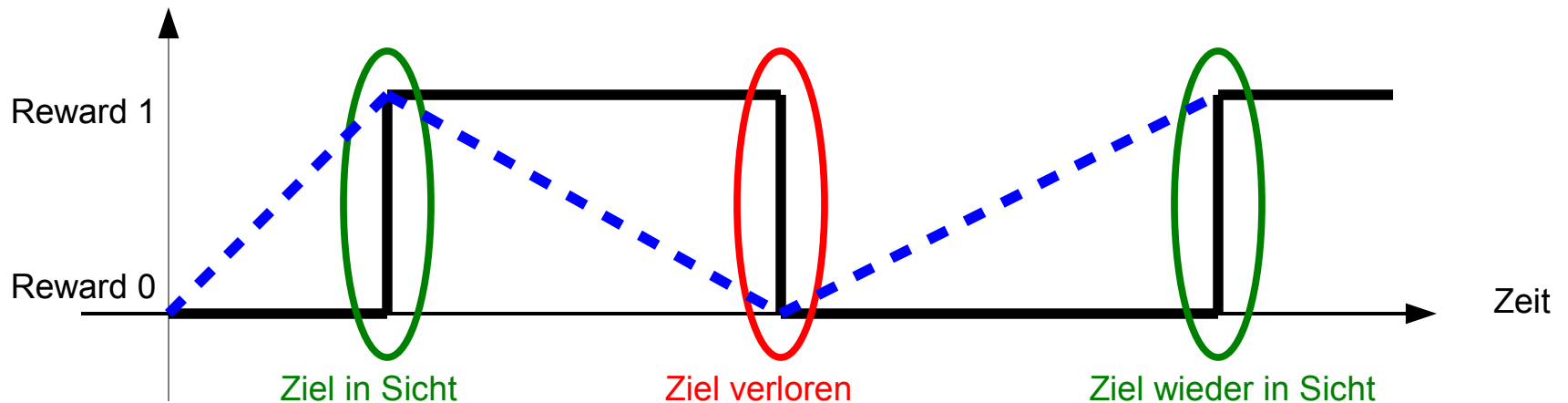
- Zu erwarten: Verteilung des Rewards in längeren Zeitabschnitten von 0 (Ziel nicht in Sicht) und 1 (in Sicht)
- In jedem Zeitabschnitt werden eine Anzahl von Classifier aktiviert und protokolliert
- Irrelevant: Ziel momentan in Sicht / nicht in Sicht
- Relevant: Ziel kommt in Sicht / Sicht zum Ziel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



Rewardfunktion

- Ziel kommt in Sicht (oder wird es aus der Sicht verloren)
 - 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

Mögliche Probleme der Idee

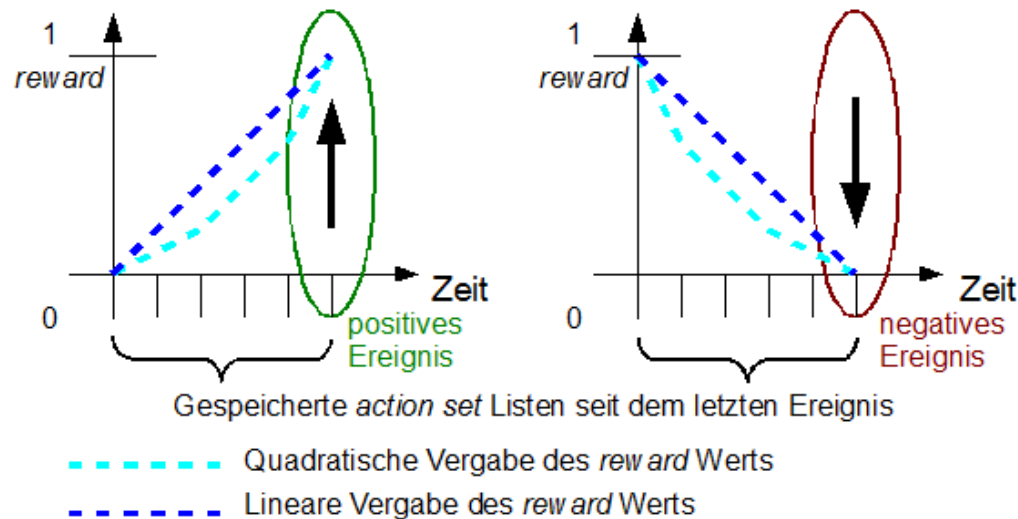
- Homogenisierung der Regeln wird begünstigt.
- Aufwand zur Bestimmung des Verwandtschaftsgrads
- Begrenztheit der Kommunikationsmittel
- Gewisse Verzögerung bis sich Information ausgebreitet hat
- Aber:
 - Keine Übertragung der Sensorinformation nötig
 - Kommunikation ist selten notwendig
 - Nicht zeitkritisch,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

Ablauf der Simulation

Auswahlart (evtl)

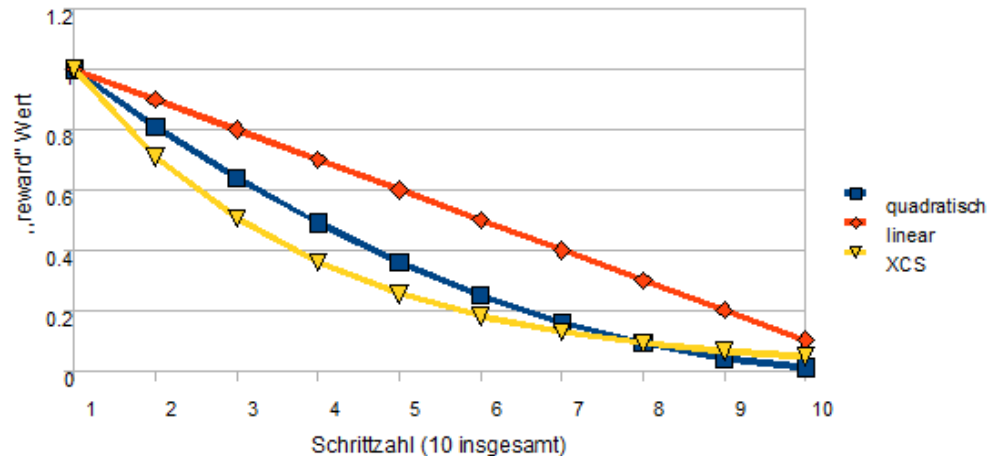
SXCS (1)

- Speicherung vergangener Aktionen
- Einführung von positiven und negativen Ereignissen
- Verteilen der Bewertung an vergangene Aktionen



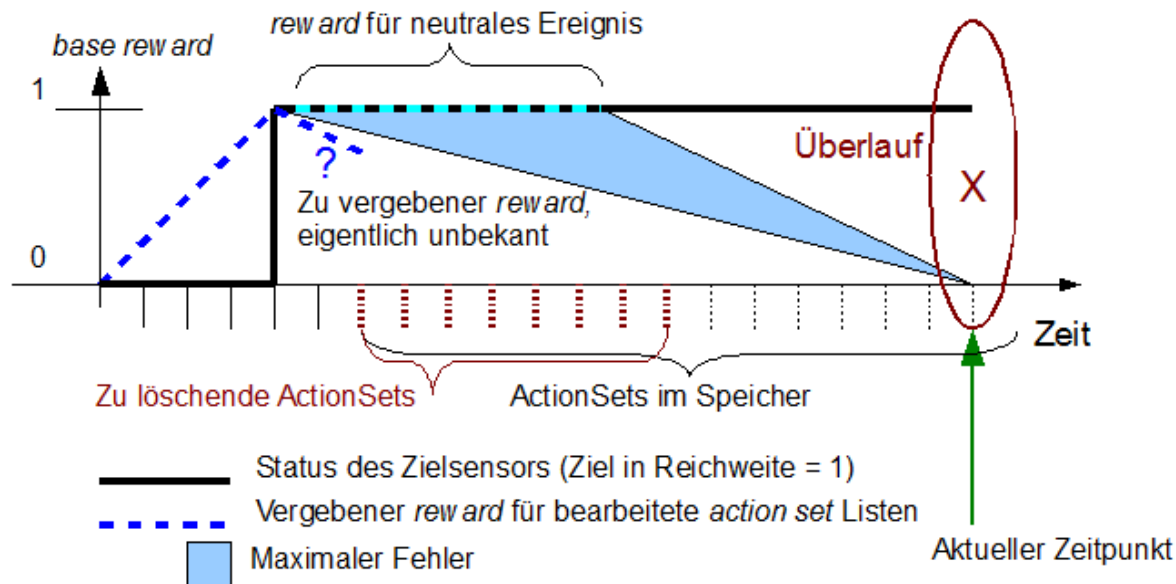
SXCS (2)

- Nachbildung der sukzessiven Weitergabe (XCS) durch ähnliche Funktion mit Nullpunkt
→ Quadratische Funktion

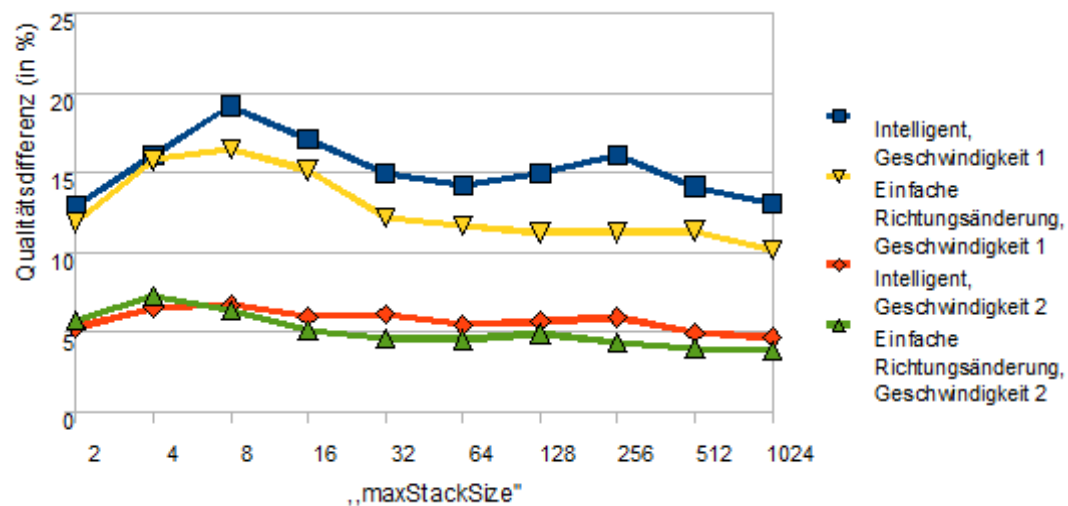


SXCS (3)

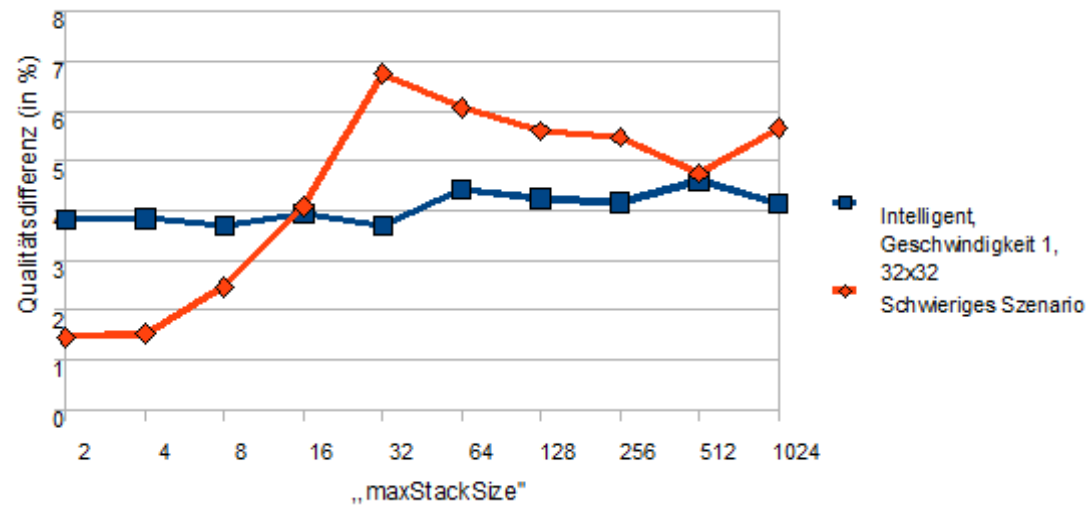
Ablauf und Fehler beim neutralen Ereignis



Stackgröße bei SXCS (1)



Stackgröße bei SXCS (2)



Vergleich der Heuristiken (4)

- Ergebnis:
 - Intelligente Heuristik etwas besser als einfache Heuristik
 - Knick bei Geschwindigkeit 1 kaum auszumachen
- Szenario
 - Szenario mit zufällig verteilten Hindernissen
 - Zielobjekt mit einfacher Richtungsänderung

