General Video Game AI Competition 2016

Teilnahme an einem Wettbewerb der künstlichen Intelligenz für Computerspiele

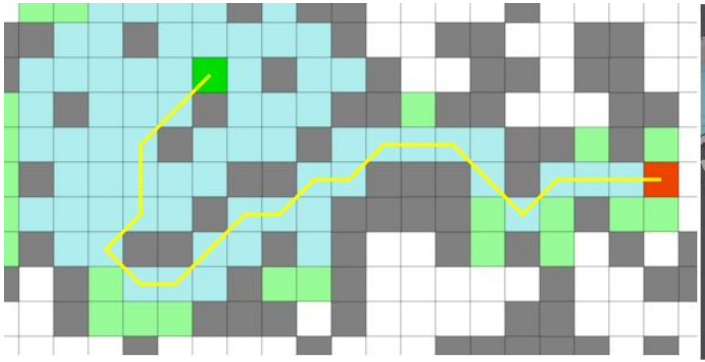
TECHNISCHE UNIVERSITÄT DARMSTADT

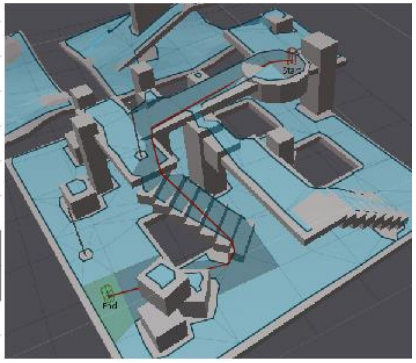
Miriam Moneke, Nils Schröder, Tobias Joppen

Christan Wirth, Prof. J. Fürnkranz

Motivation

- Industrie
  - Verhalten von nicht-spieler Charakteren



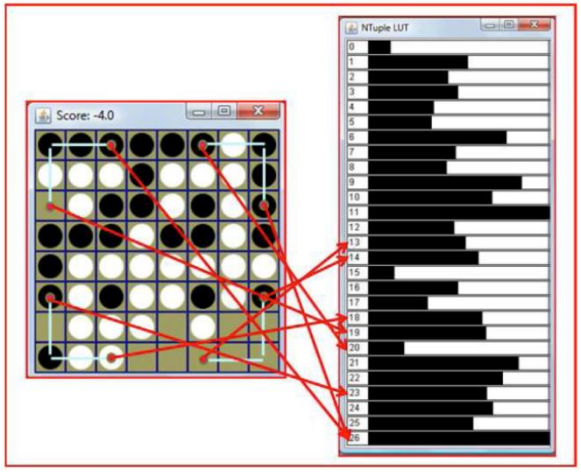


  - Entscheidungsfindung
  - PCG (Procedural Content Generation)

AI Research in Games

TECHNISCHE UNIVERSITÄT DARMSTADT

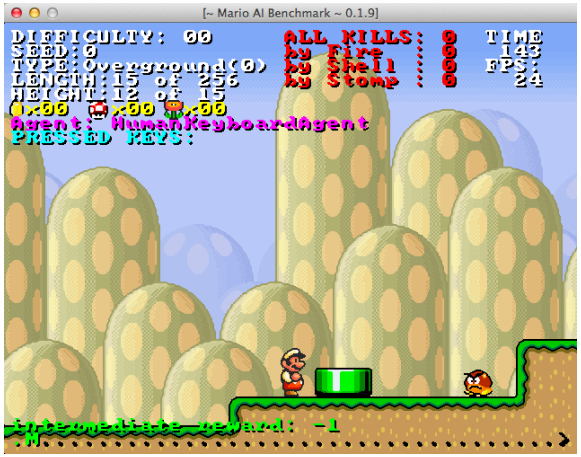
Spiele als Benchmarks für künstliche Intelligenz:



General Video Game Playing

TECHNISCHE UNIVERSITÄT DARMSTADT

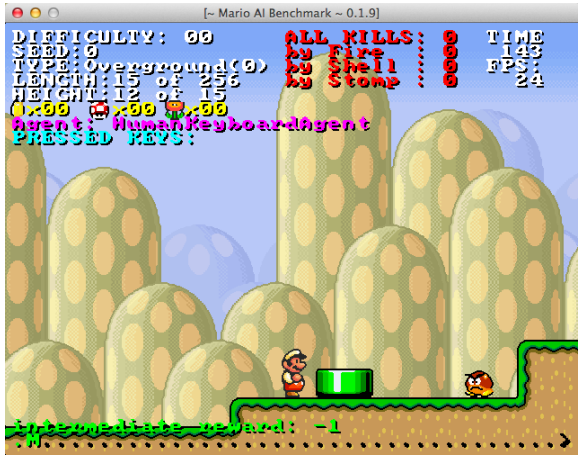
Video game-based AI Competitions



General Video Game Playing

TECHNISCHE UNIVERSITÄT DARMSTADT

General Video Game Playing?



GVGAI und VGDL

TECHNISCHE UNIVERSITÄT DARMSTADT

- Existierendes Framework (GVGAI)
- Teilnehmer reichen Agenten ein
- 110 Spiele insgesamt in Trainings, Validation und Final Sets
- Spiele sind mit der *Video Game Description Language* implementiert

- Fully observable environment
- Agenten haben 40 ms Zeit, um Entscheidungen zu treffen

VGDL and the GVGAI Framework

TECHNISCHE UNIVERSITÄT DARMSTADT

Der erste Agent ist einfach zu erstellen!

```
package controllers.sampleRandom;

import ...

public class Agent extends AbstractPlayer {

    public Agent(StateObservation so, ElapsedCpuTimer elapsedTimer) { }

    public Types.ACTIONS act(StateObservation stateObs, ElapsedCpuTimer elapsedTimer) {
        ArrayList<Types.ACTIONS> actions = stateObs.getAvailableActions();
        return actions.get(new Random().nextInt(actions.size()));
    }
}
```

VGDL and the GVGAI Framework

TECHNISCHE UNIVERSITÄT DARMSTADT

Zu jedem Zeitschritt bekommt der Agent Informationen:

- Spielzustand:
  - Punktzahl, Zeitschritt, Sieger, Spiel beendet, Spielfeldgröße, …
- Zustand des eigenen Avatar:
  - Position, Geschwindigkeit, Blickrichtung, Inventar, Lebenspunkte, …
- Verfügbare Aktionen
- Aktuelle Spielhistorie (Kollisionen)
- Andere Spielobjekte:
  - Spielraster mit Spielobjekten
  - Kategorisierte Spielobjekte (NPCs, statische Objekte, Ressourcen, …)

VGDL and the GVGAI Framework

TECHNISCHE UNIVERSITÄT DARMSTADT

Dem Agenten steht ein forward-model zur Verfügung:

- copy()
- advance(action)
  - Simuliert das Ausführen einer Aktion (Liefert neuen Spielzustand)
  - Dadurch lassen sich Auswirkungen und Effekte ermittlen
  - Spiele sind im allgemeinen stochastisch
    - Der Nachfolgezustand ist nur ein möglicher von mehreren möglichen
    - Der Agent ist dafür verantwortlich diese Ungenauigkeit zu verarbeiten

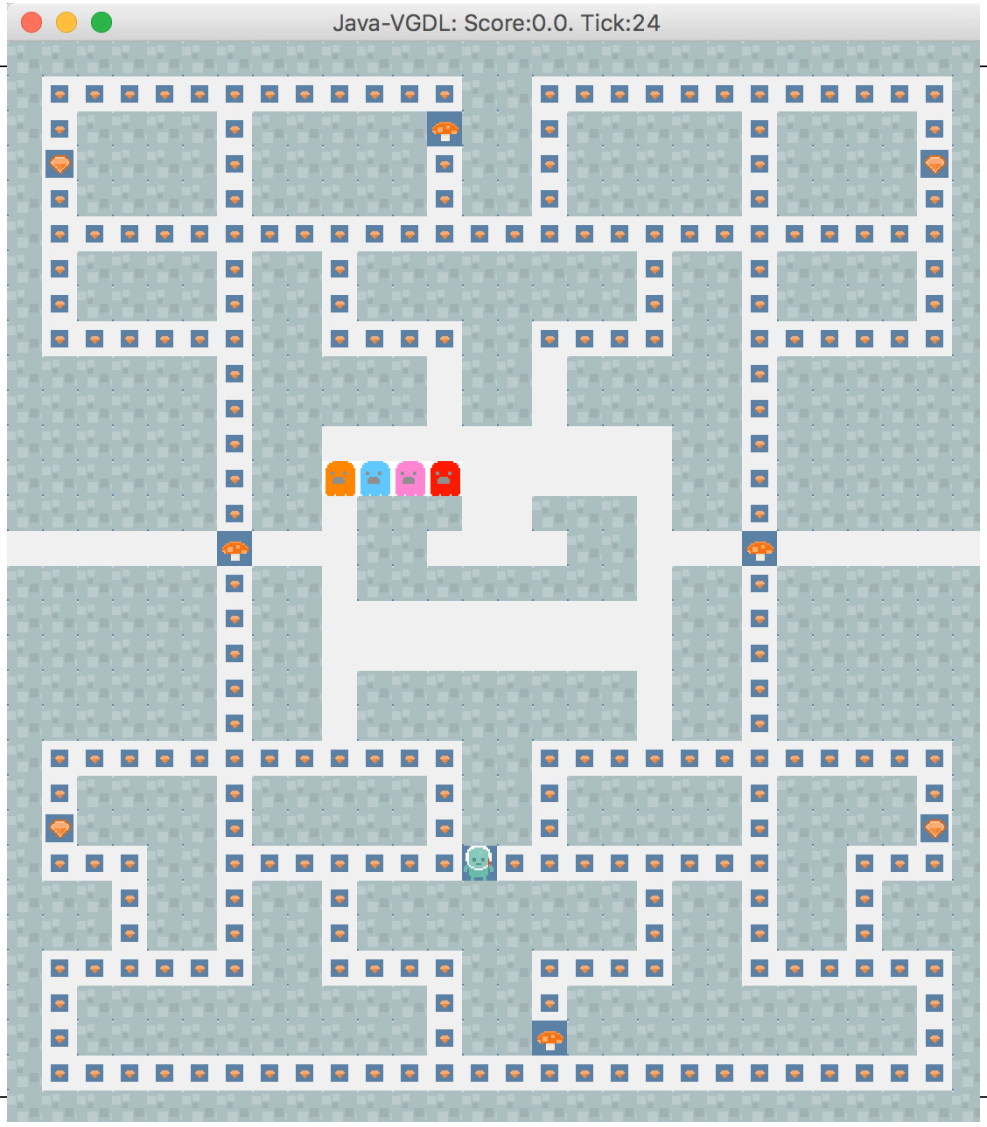
GVGAI und VGDL

TECHNISCHE UNIVERSITÄT DARMSTADT

```
BasicGame key_handler=Pulse square_size=50

    SpriteSet
        hole   > Immovable color=DARKBLUE img=hole
        ground > Immovable img=water
        avatar > MovingAvatar
        bbox > Passive
            box    >  img=box
            boxin  >  img=city

    LevelMapping
        A > avatar ground
        0 > hole
        * > box ground
        . > ground

    InteractionSet
        avatar wall > stepBack
        bbox avatar  > bounceForward
        bbox wall bbox   > undoAll
        box hole    > transformTo stype=boxin scoreChange=1
        boxin ground > transformTo stype=box scoreChange=-1

    TerminationSet
        SpriteCounter stype=box    limit=0 win=True
```

```
wwwww.wwwww
w....www...w
w.**.....wAw
w.*.w000...w
w...wwwwwww
wwww.......
```

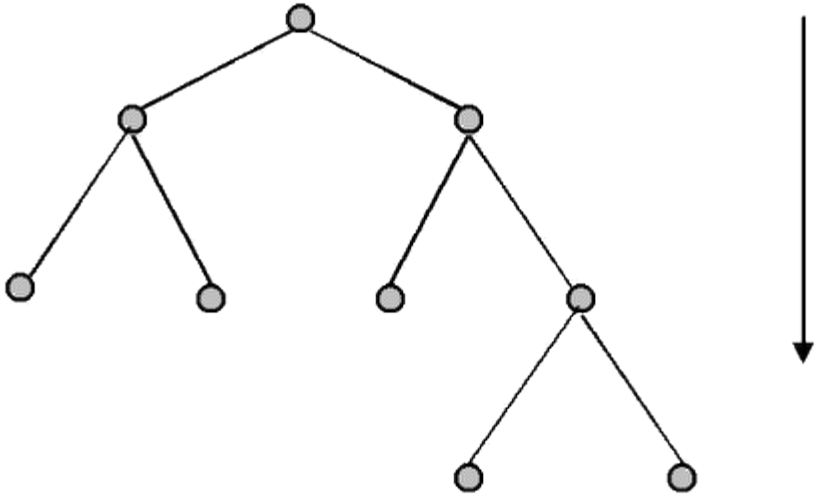
Beispiel (Pacman)



Beispiel (Plants)

Lösungsansätze I (Exhaustive Tree Search)

- Breath first search
- Depth first search
- Best first search
- […]



Lösungsansätze II (Stochastic Tree Search)

TECHNISCHE UNIVERSITÄT DARMSTADT

Monte Carlo Tree Search

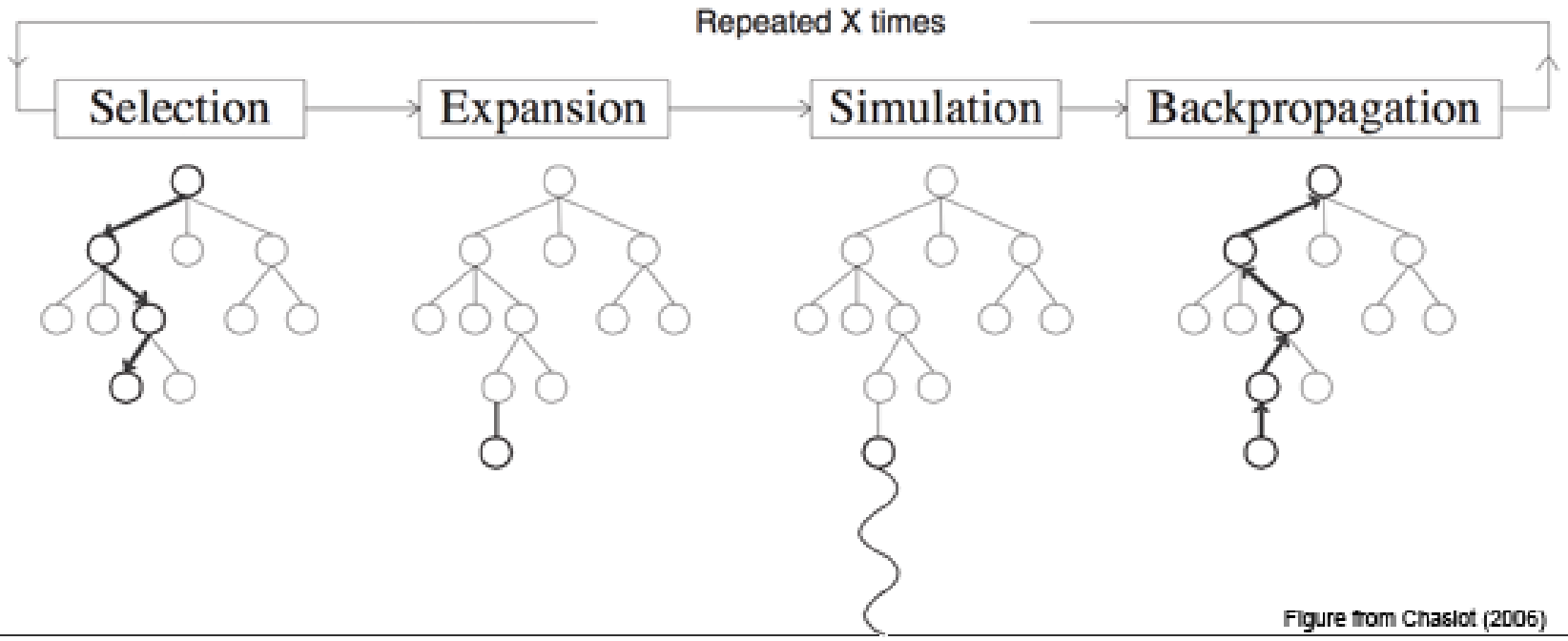


Figure from Chaslot (2006)

Lösungsansätze II (Stochastic Tree Search)

TECHNISCHE UNIVERSITÄT DARMSTADT

Monte Carlo Tree Search

1. Selection:

   Von der Wurzel: wähle Nachfolgeknoten.

   Iteriere bis Blattknoten.

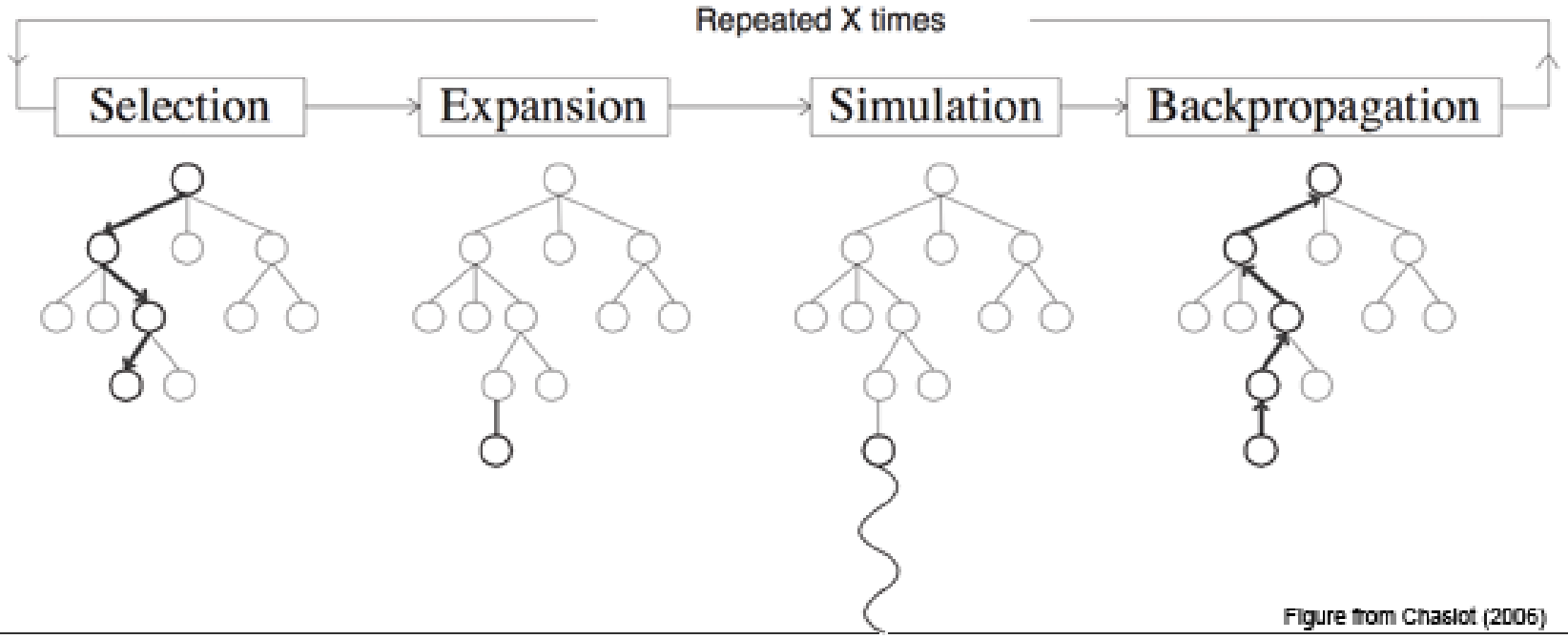


Figure from Chaslot (2006)

Lösungsansätze II (Stochastic Tree Search)

TECHNISCHE UNIVERSITÄT DARMSTADT

Monte Carlo Tree Search

2. Expansion:

   Erstelle neuen Kindknoten.

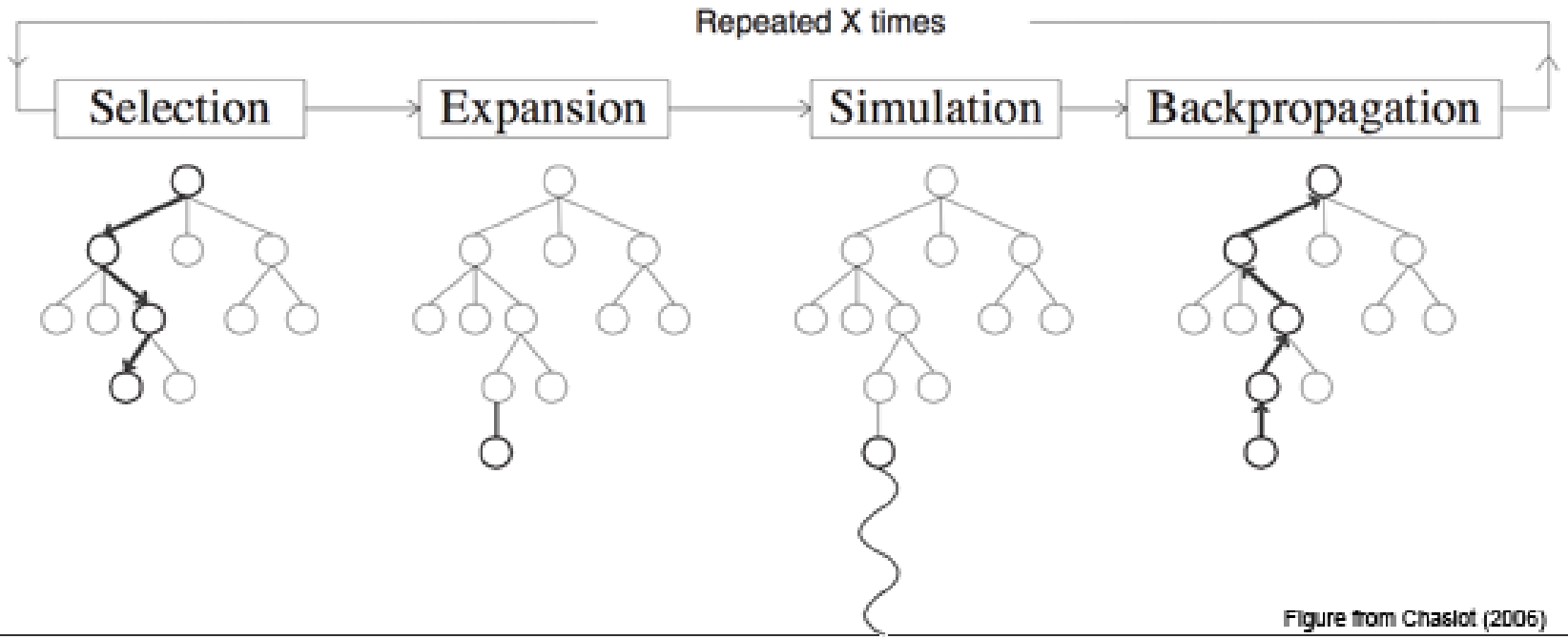


Figure from Chaslot (2006)

Lösungsansätze II (Stochastic Tree Search)

TECHNISCHE UNIVERSITÄT DARMSTADT

Monte Carlo Tree Search

3. Simulation:

   Simuliere Zustand in die Tiefe: führe mehrere zufällige Züge hintereinander aus und bewerte letzten Zustand.

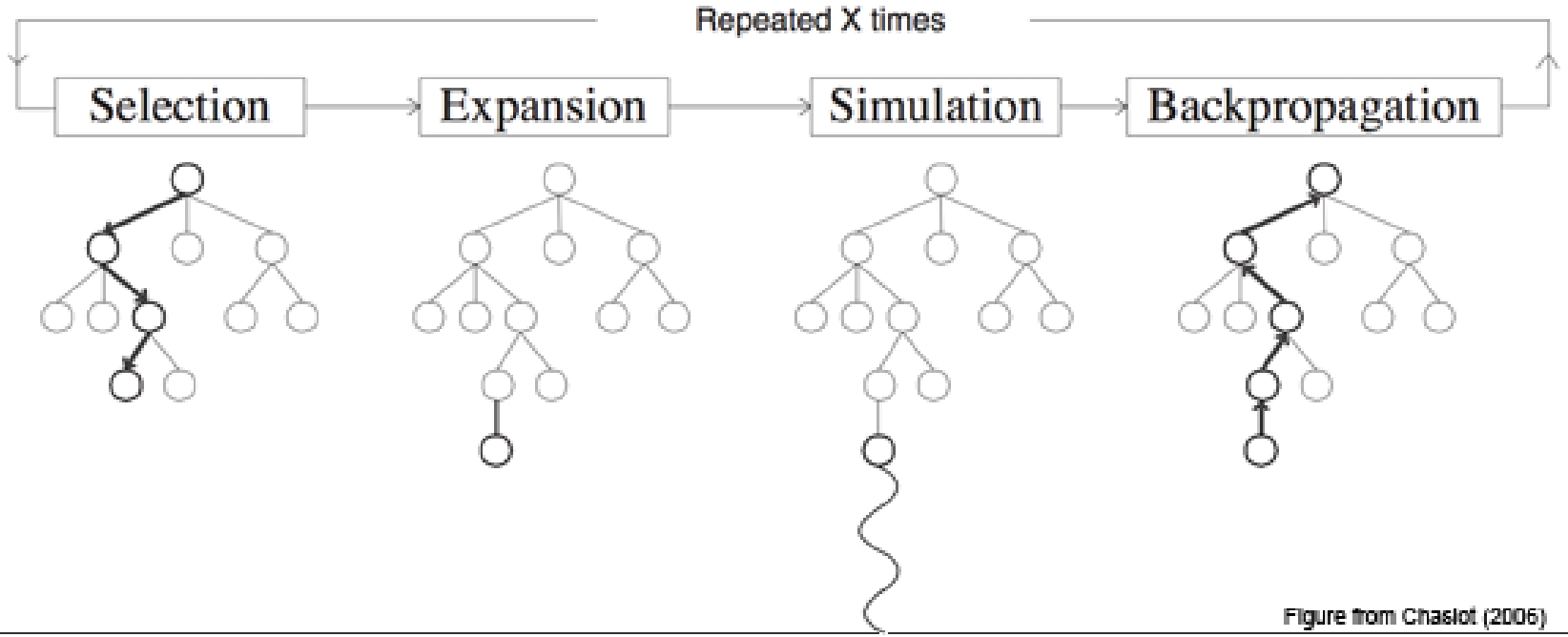


Figure from Chaslot (2006)

Lösungsansätze II (Stochastic Tree Search)

TECHNISCHE UNIVERSITÄT DARMSTADT

Monte Carlo Tree Search

4. Backpropagation:

   Bewerte alle Knoten auf dem Weg zur Wurzel entsprechend der Bewertung des simulierten Zustands

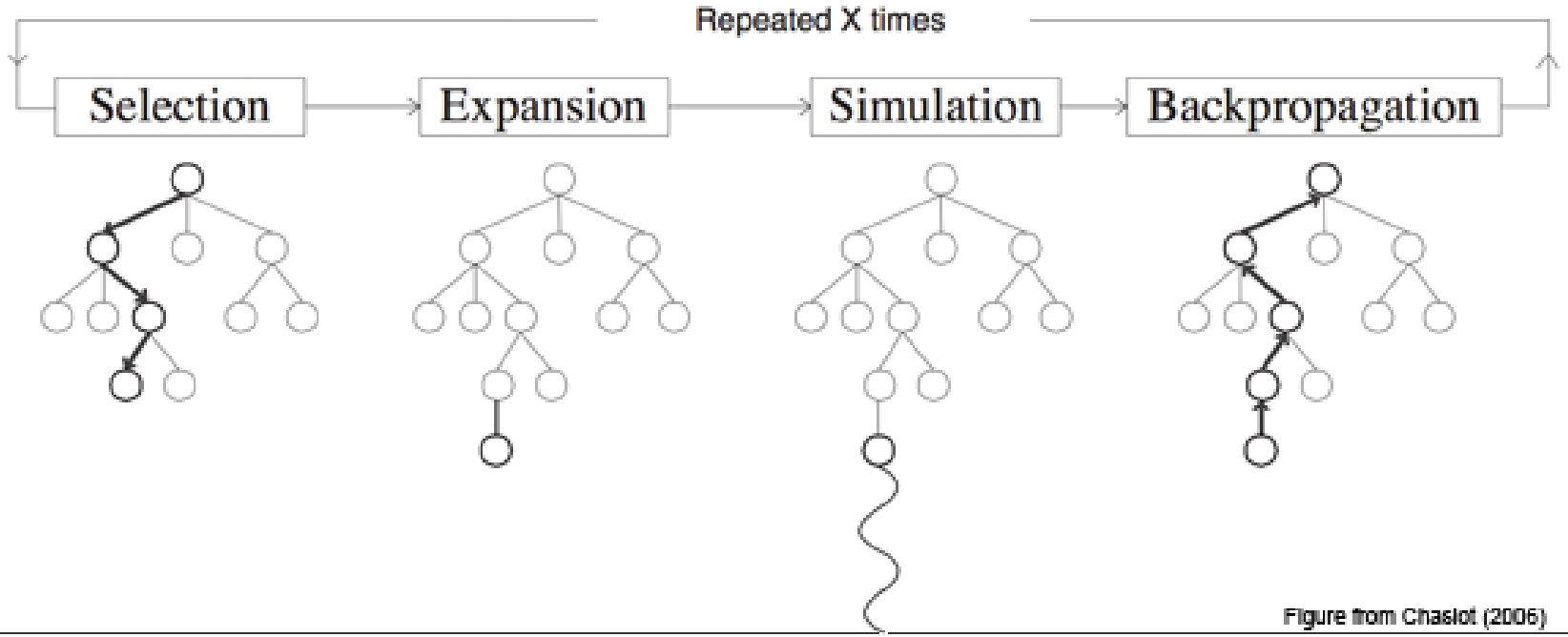


Figure from Chaslot (2006)

Lösungsansätze II (Stochastic Tree Search)

TECHNISCHE UNIVERSITÄT DARMSTADT

Monte Carlo Tree Search

- Wiederhole, bis Zeit abgelaufen

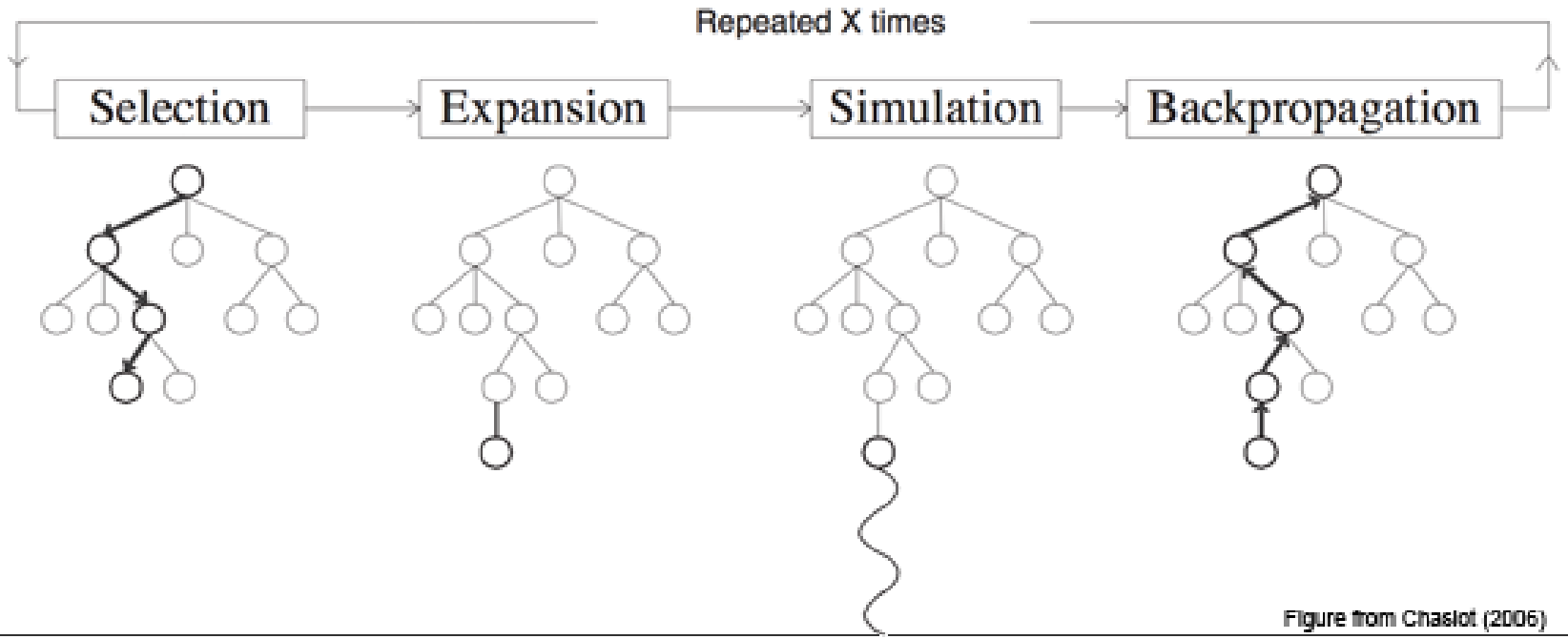
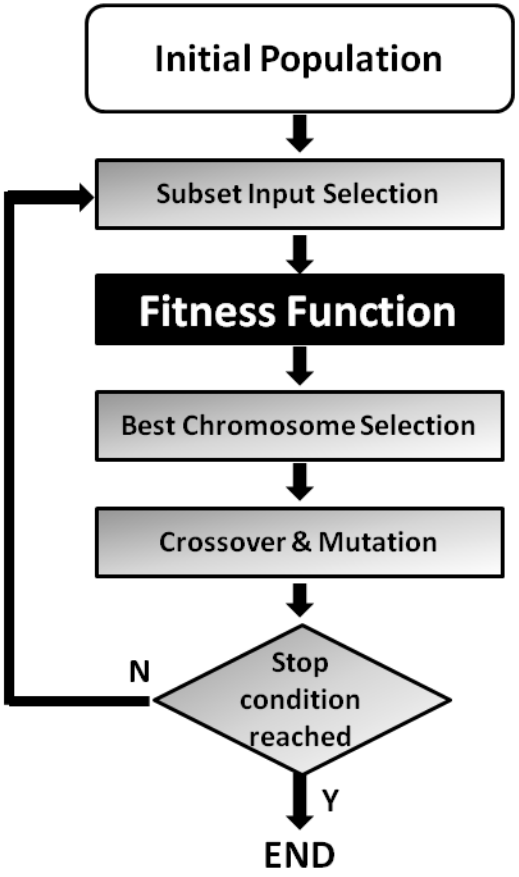


Figure from Chaslot (2006)

Lösungsansätze III (Evolutionary Algorithms)

TECHNISCHE UNIVERSITÄT DARMSTADT

- Individuum: Eine Gewichtung der einzelnen Bestandteile der Fitness Function
- Fitness Function: Auswertung des Spielzustandes nach Ausführen der Aktionen
- Wenn die Zeit abgelaufen ist:
  Führe erste Aktion des besten Individuums aus

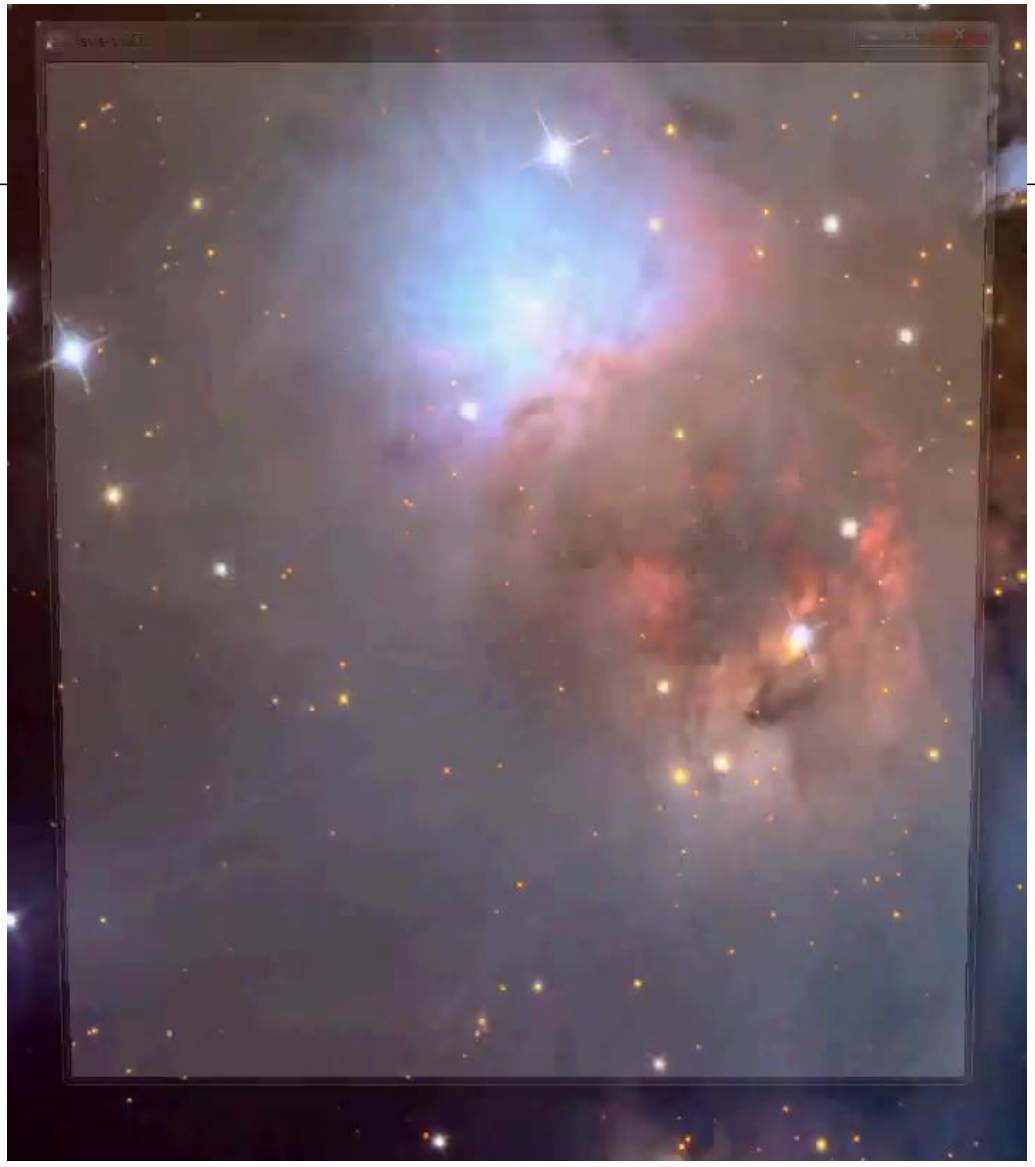


Spielabhängiges Wissen aneignen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Spiele sind aus Regeln aufgebaut (siehe VGDL)
- Regeln können gelernt werden
- ‚Klevere‘ Heuristiken können erzeugt werden
- Lässt sich gut mit den genannten Lösungsansätzen kombinieren

Beispiel



GVGAI 2016 Competition

TECHNISCHE UNIVERSITÄT DARMSTADT

- Single Player Planning Track:
  - Weiterführung des letztjährigen Wettbewerbs
  - Viele Spiele: 80 öffentlich, 20 zur Validierung
  - GECCO 2016 – Denver, US (20^{th} – 24^{th} July)
  - CIG 2016 – Santorini, Greece (20^{th} – 23^{rd} September)

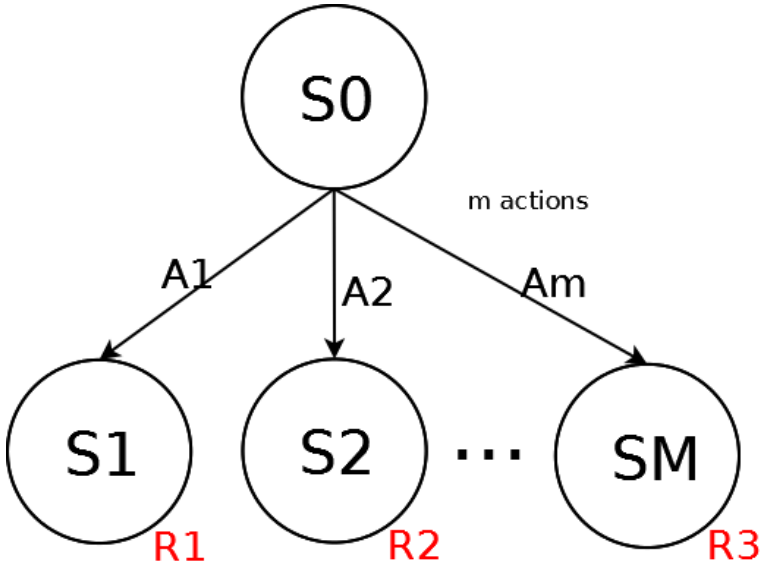
  - Erwartete Abgabefrist: Anfang July

- Bewertungsschema des offiziellen Wettbewerbs:
  - # Wins > avg(Score) > avg(Spielzüge)
  - Formel 1 Rating (1. Platz -> 25, 2. Platz -> 18, 3. Platz -> 15, [..])

Beispielagent (One Step Look Ahead)

Das Framework stellt Beispielagenten zur Verfügung:

- Simple One Step Look Ahead:



Beispielagent (One Step Look Ahead)

TECHNISCHE UNIVERSITÄT DARMSTADT

- Simple One Step Look Ahead:

```
public Types.ACTIONS act(StateObservation stateObs, ElapsedCpuTimer elapsedTimer) {

    Types.ACTIONS bestAction = null;
    double maxQ = Double.NEGATIVE_INFINITY;
    SimpleStateHeuristic heuristic =  new SimpleStateHeuristic(stateObs);
    for (Types.ACTIONS action : stateObs.getAvailableActions()) {

        StateObservation stCopy = stateObs.copy();
        stCopy.advance(action);
        double Q = heuristic.evaluateState(stCopy);

        if (Q > maxQ) {
            maxQ = Q;
            bestAction = action;
        }
    }

    return bestAction;
}
```

Beispielagent (One Step Look Ahead)

TECHNISCHE UNIVERSITÄT DARMSTADT

- Simple One Step Look Ahead:

```
public double evaluateState(StateObservation a_gameState) {

    boolean gameOver = a_gameState.isGameOver();
    Types.WINNER win = a_gameState.getGameWinner();
    double rawScore = a_gameState.getGameScore();

    if(gameOver && win == Types.WINNER.PLAYER_LOSES)
        return HUGE_NEGATIVE;

    if(gameOver && win == Types.WINNER.PLAYER_WINS)
        return HUGE_POSITIVE;

    return rawScore;
}
```

Organisatorisches

TECHNISCHE UNIVERSITÄT DARMSTADT

- Gruppen bis zu 5 Personen
- Wöchentliches Treffen mit Besprechung des Fortschritts
  - Mittwochs 15:20 (wie heute)
  - Gruppen tragen Ergebnisse vor
  - Pro Treffen trägt eine Gruppe die Fortschritte detaillierter vor
- Arbeitsbeginn: Jetzt!
  - Meiste Arbeit fällt ab sofort an, nicht zum Ende des Semesters

Seid ihr interessiert?

Aufgaben für nächste Woche

TECHNISCHE UNIVERSITÄT DARMSTADT

- Teambildung
- Registrieren auf www.gvgai.net (1 Account pro Team!)
- Framework einrichten (… und zum laufen bringen)
  - Auch ohne visuelles Feedback starten (headless)
- Einen Agenten hochladen und online auswerten lassen
- Die Google-Group abonnieren (siehe gvgai.net)

Für Motivierte:

- Ein paar Spiele angucken (gvgai.net / youtube.com)
- Erste eigene Lösungsideen entwickeln