General Video Game AI Competition 2017

Teilnahme an einem Wettbewerb der künstlichen Intelligenz für Computerspiele

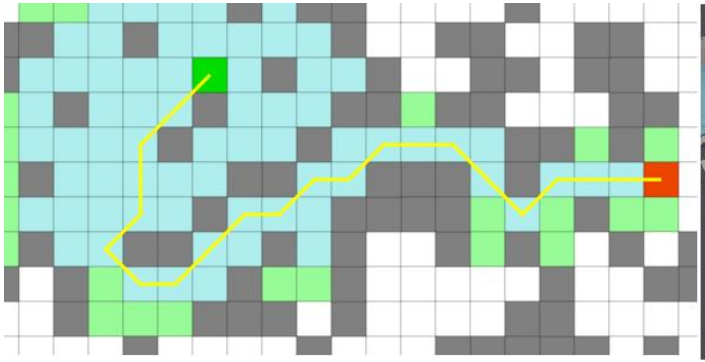
TECHNISCHE UNIVERSITÄT DARMSTADT

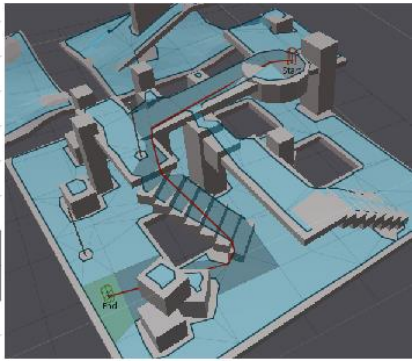
Tobias Joppen, Christan Wirth, Prof. J. Fürnkranz

Motivation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Industrie
  - Verhalten von nicht-spieler Charakteren



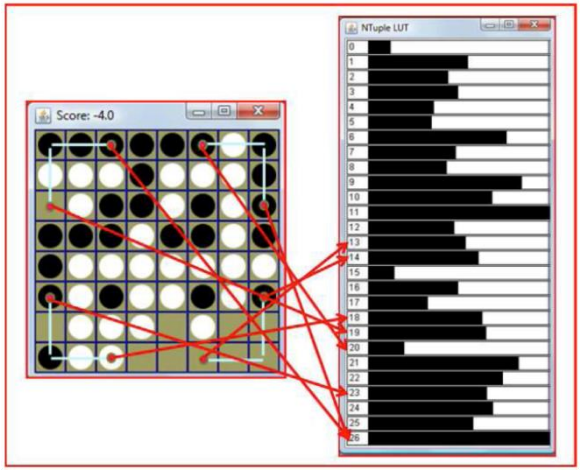


  - Entscheidungsfindung
  - PCG (Procedural Content Generation)

AI Research in Games

TECHNISCHE UNIVERSITÄT DARMSTADT

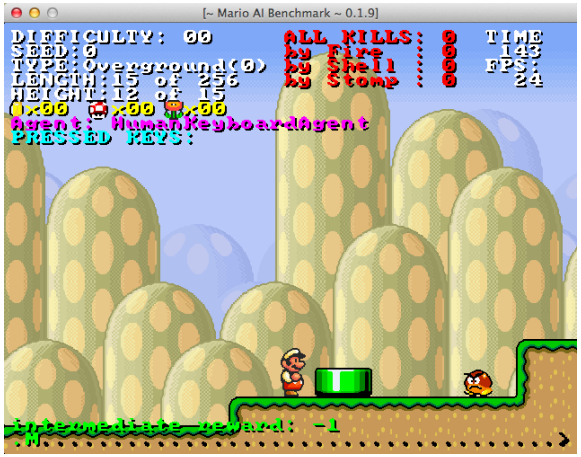
Spiele als Benchmarks für künstliche Intelligenz:



General Video Game Playing

TECHNISCHE UNIVERSITÄT DARMSTADT

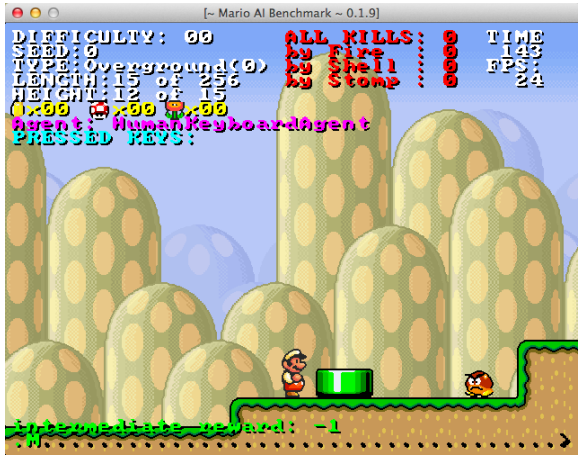
Video game-based AI Competitions



General Video Game Playing

TECHNISCHE UNIVERSITÄT DARMSTADT

General Video Game Playing?



GVGAI und VGDL

TECHNISCHE UNIVERSITÄT DARMSTADT

- Existierendes Framework (GVGAI)
- Teilnehmer reichen Agenten ein
- Über 100 Spiele in Trainings, Validation und Final Sets
- Spiele sind mit der *Video Game Description Language* implementiert

- Fully observable environment (keine versteckten Informationen)
- 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, …)

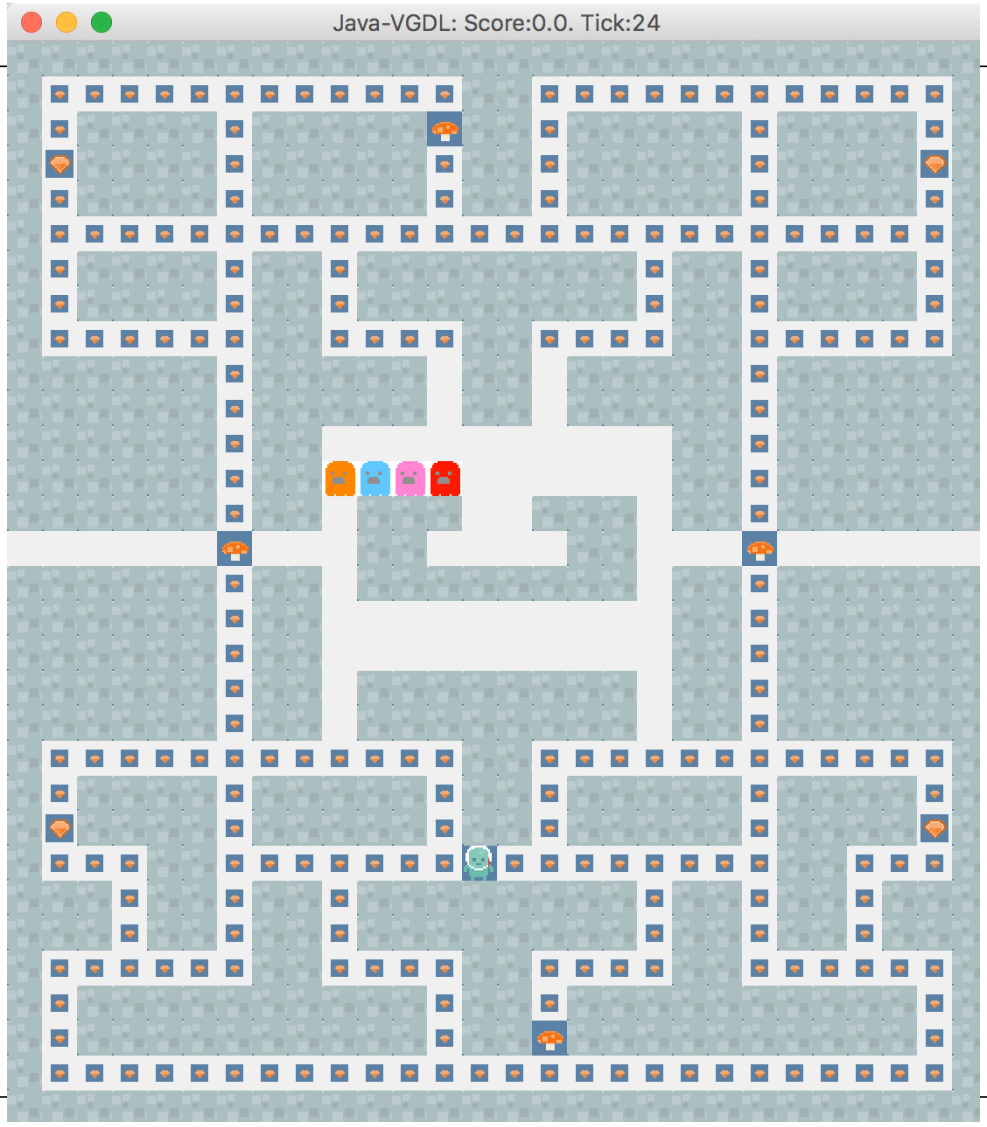
VGDL and the GVGAI Framework

TECHNISCHE UNIVERSITÄT DARMSTADT

Dem Agenten steht ein forward-model zur Verfügung:

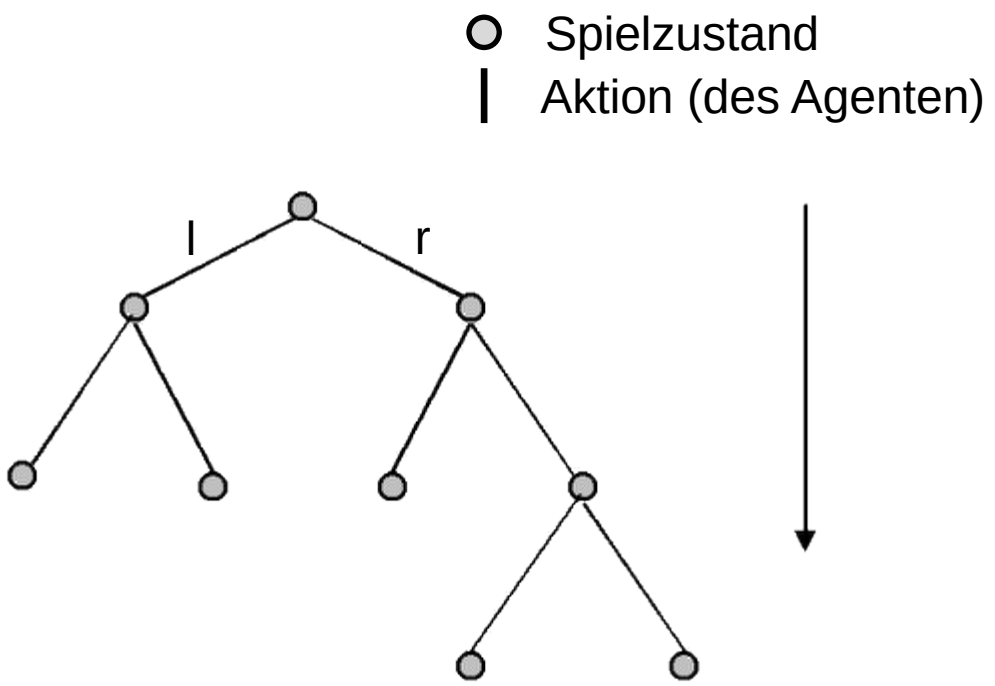
- advance(action)
  - Simuliert das Ausführen einer Aktion (Liefert neuen Spielzustand)
  - Dadurch lassen sich Auswirkungen und Effekte ermittlen
  - Spiele sind im allgemeinen stochastisch
    - Der Nachfolgezustand ist nur einer von mehreren möglichen
    - Der Agent ist dafür verantwortlich diese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

Beispiel (Pacman)



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

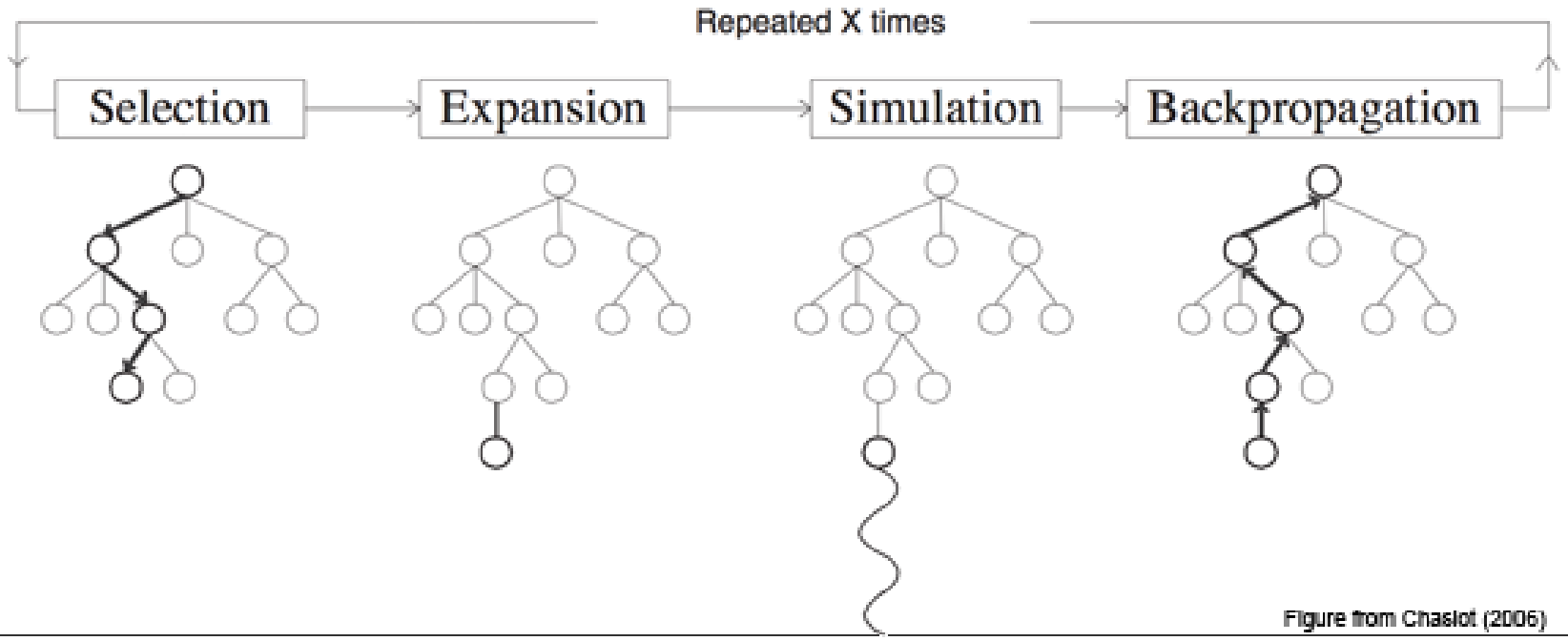


Figure from Chaslot (2006)

Lösungsansätze II (Stochastic Tree Search)

TECHNISCHE UNIVERSITÄT DARMSTADT

Monte Carlo Tree Search

1. Selection:

   Von der Wurzel: wähle Nachfolgeknoten.

   Iteriere bis Blattknoten.

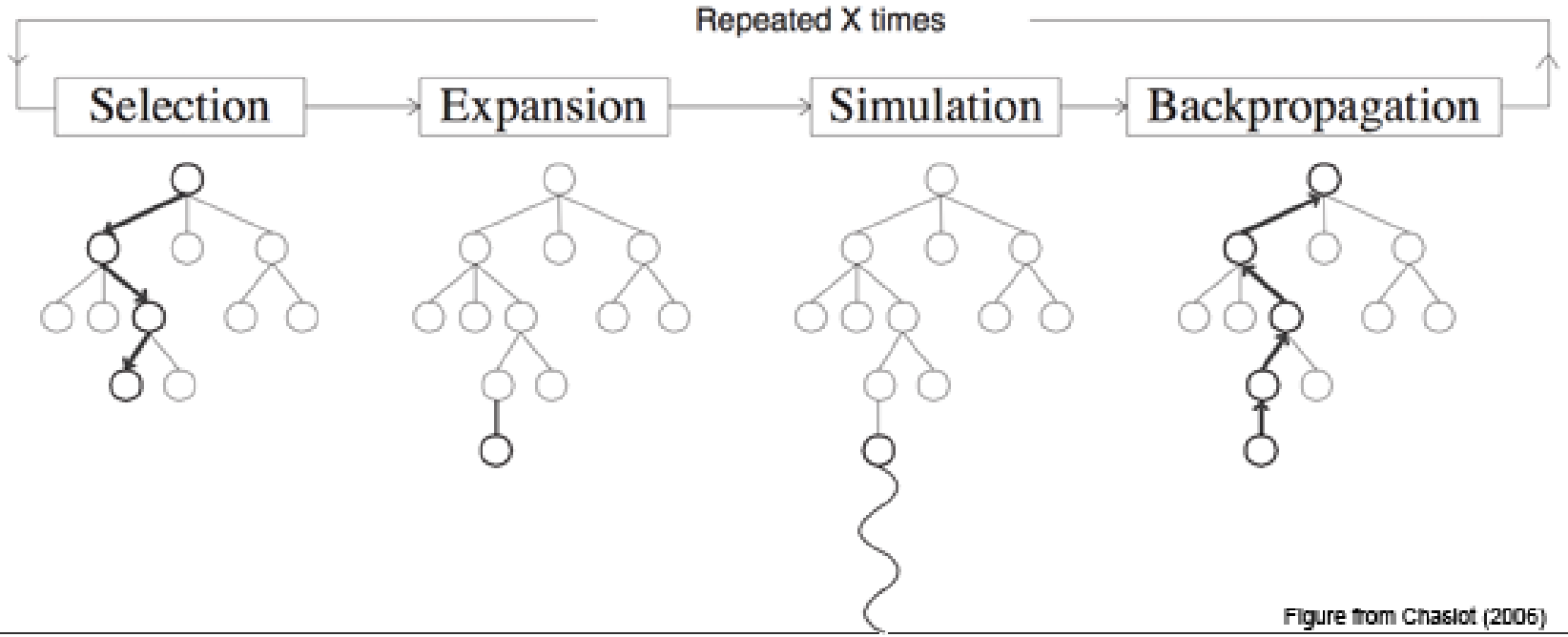


Figure from Chaslot (2006)

Lösungsansätze II (Stochastic Tree Search)

TECHNISCHE UNIVERSITÄT DARMSTADT

Monte Carlo Tree Search

2. Expansion:

   Erstelle neuen Kindknoten.

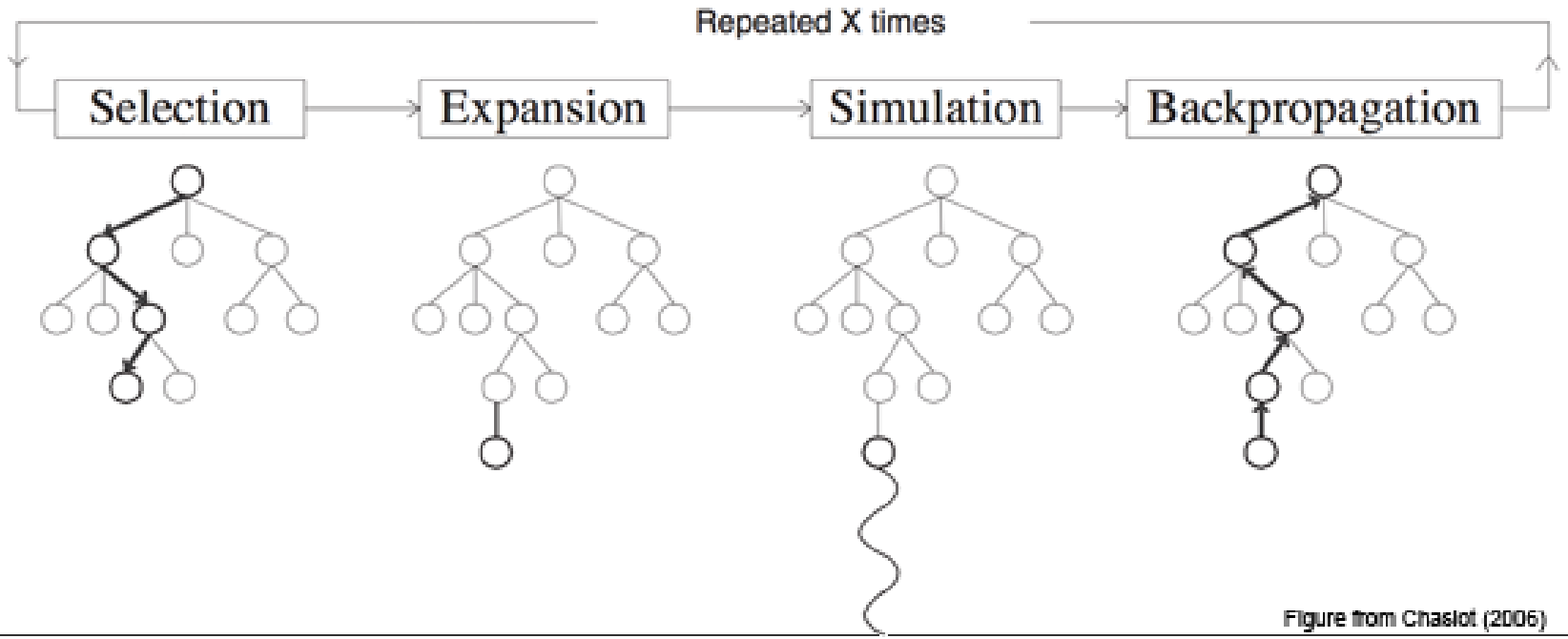


Figure from Chaslot (2006)

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.

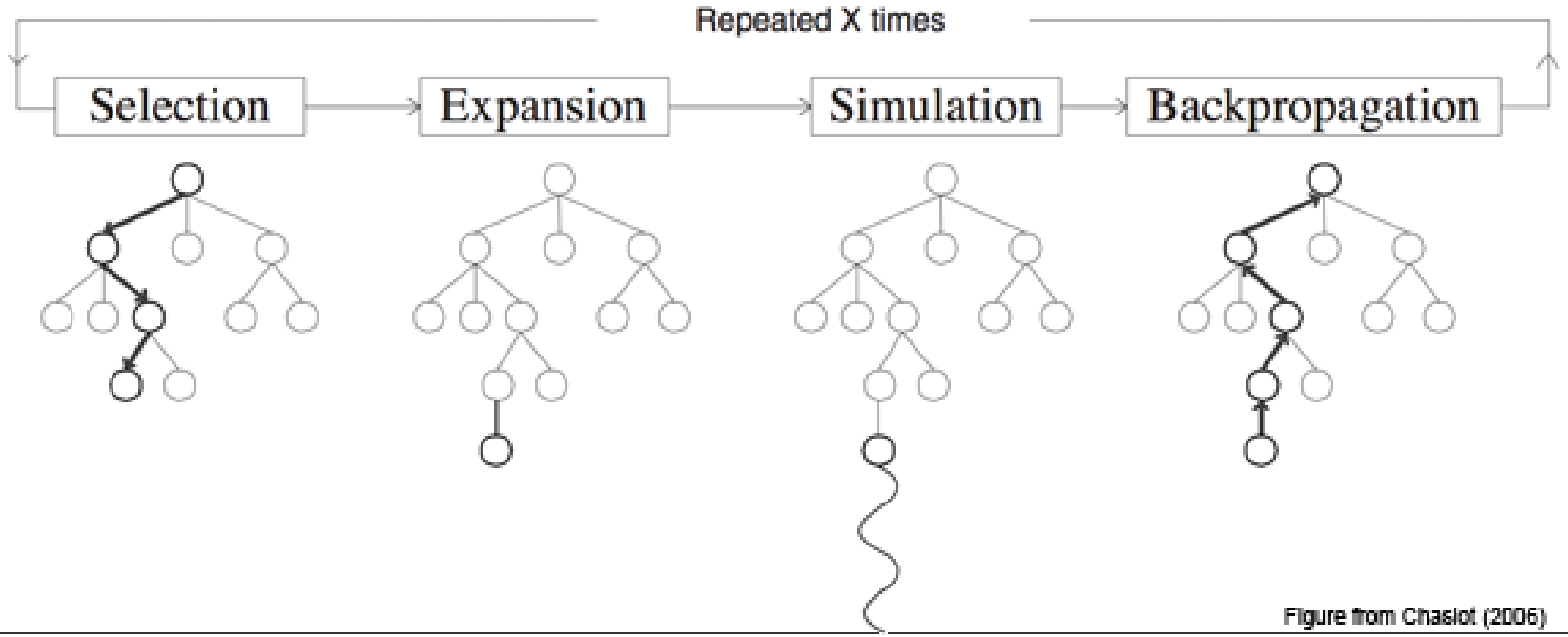


Figure from Chaslot (2006)

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

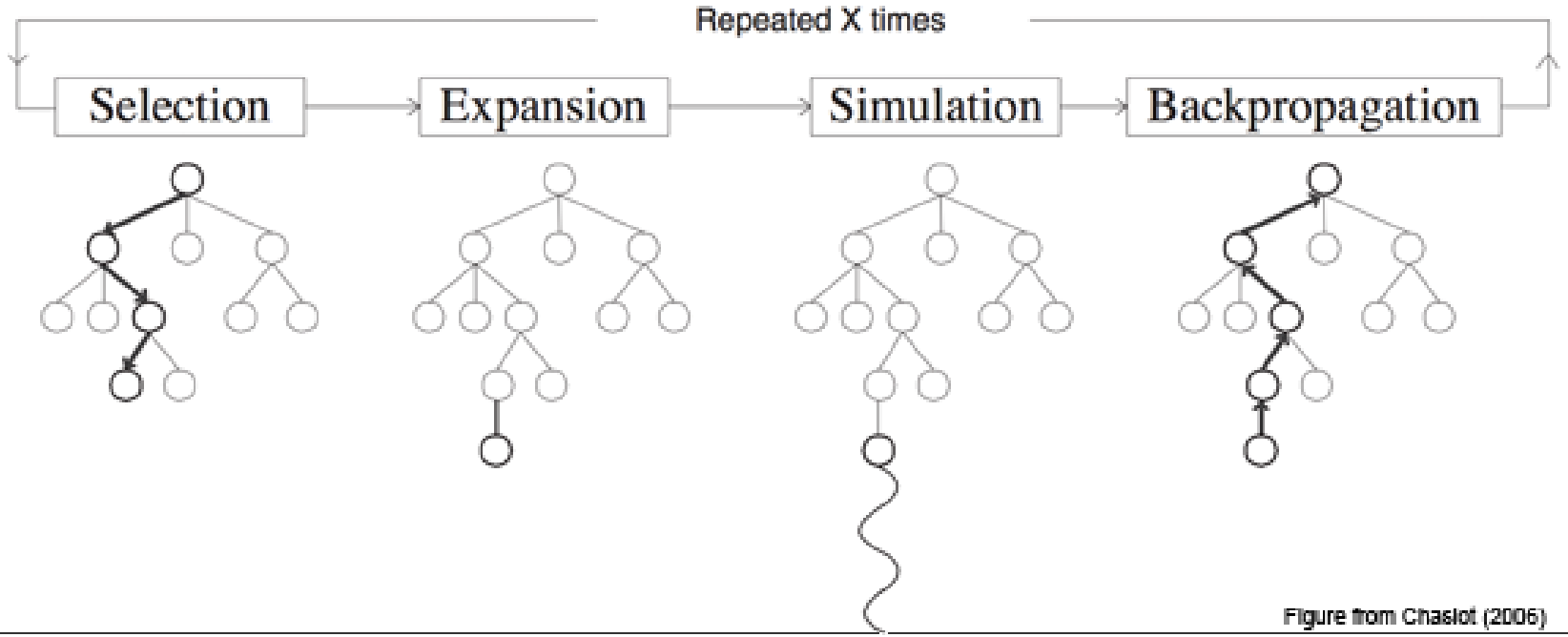


Figure from Chaslot (2006)

Lösungsansätze II (Stochastic Tree Search)

TECHNISCHE UNIVERSITÄT DARMSTADT

Monte Carlo Tree Search

- Wiederhole, bis Zeit abgelaufen

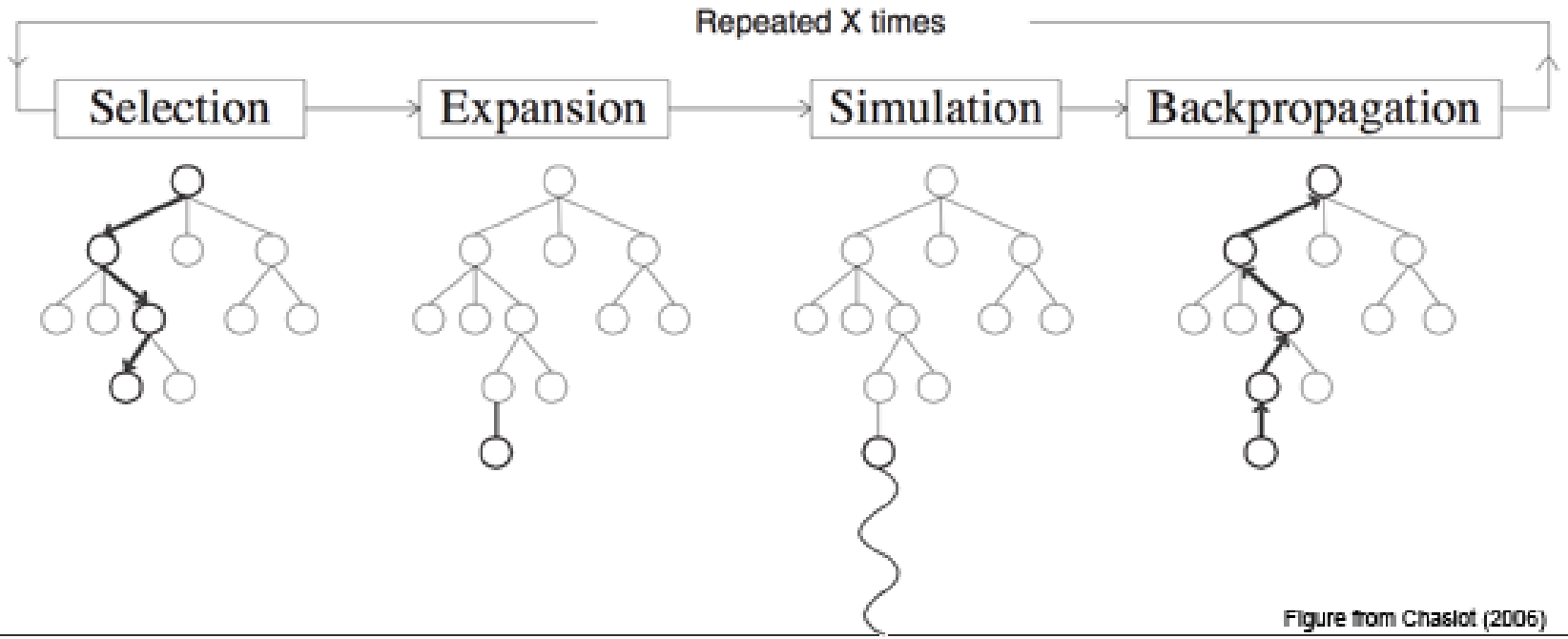
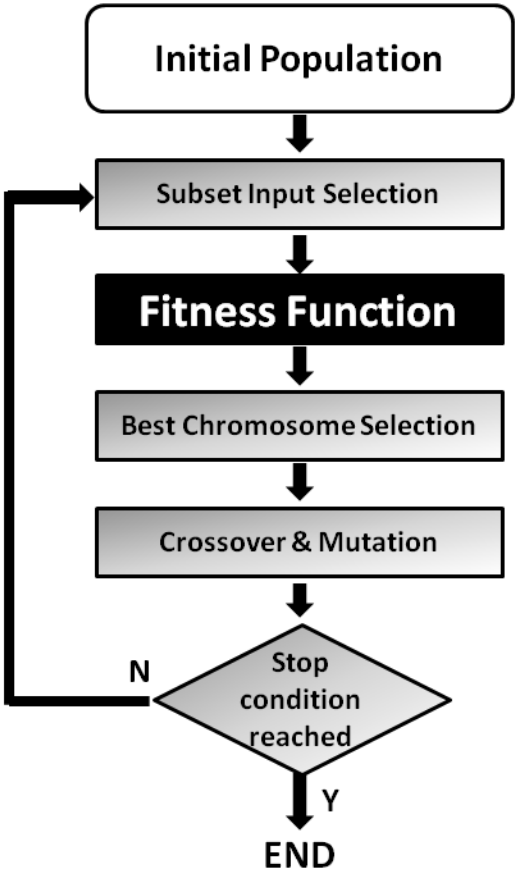


Figure from Chaslot (2006)

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

GVGAI 2017 Competition

TECHNISCHE UNIVERSITÄT DARMSTADT

- Single Player Planning Track:
  - Weiterführung des letztjährigen Wettbewerbs
  - Neu: Kontinuierliche Physik
  - GECCO 2017 – Berlin (15^{th} – 19^{th} July)
  - Voraussichtiliche Abgabefrist: 15. Juni

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

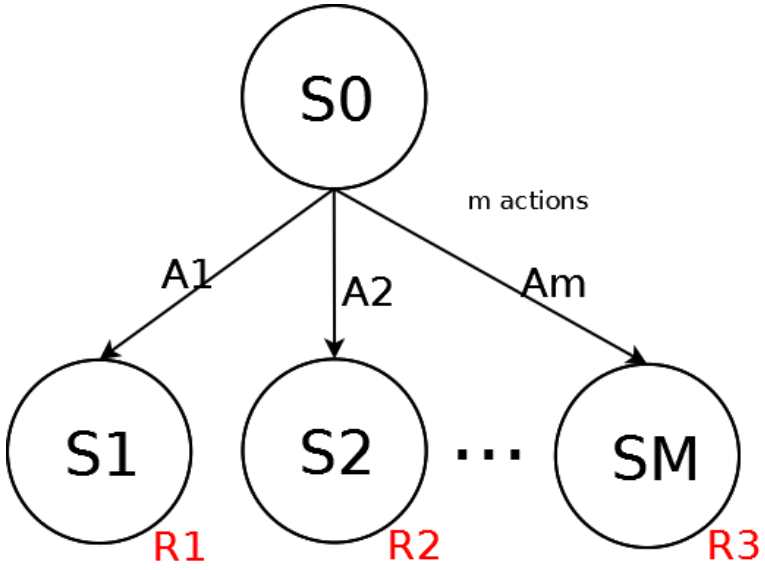
    [12,10,8,6,4,2,1,0,0,0,0,…]

Beispielagent (One Step Look Ahead)

TECHNISCHE UNIVERSITÄT DARMSTADT

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;
}
```

Knowledge Base

TECHNISCHE UNIVERSITÄT DARMSTADT

Wir stellen die Knowledge Base eines unserer Agenten zur Verfügung:

- Lernt Kollisionen des Agenten → diverse Vorhersagen möglich
- Lernt primitive Gegnerbewegung

→ Kann ich sterben, wenn ich Aktion X ausführe?

→ Kann es sinnvoll sein, wenn ich X ausführe? (z.B. nicht gegen Wand laufen)

→ Welche Felder kann ich aktuell erreichen?

→ Was passiert, wenn ich das Objekt Y berühre / benutze?

- Siehe unsere Webseite für Download und Anwendungsbeispiel

Organisatorisches

TECHNISCHE UNIVERSITÄT DARMSTADT

- Wöchentliches Treffen mit Besprechung des Fortschritts
  - Regelmäßig: Montags 15:20 hier
  - Unregelmäßig: Mittwochs 13:10 hier

  → Aktuelle Termine stehen immer im KE-Kalender
  http://www.ke.tu-darmstadt.de/termine

- Arbeitsbeginn: Jetzt!
  - Meiste Arbeit fällt ab sofort an, nicht zum Ende des Semesters

Arbeitsaufteilung

TECHNISCHE UNIVERSITÄT DARMSTADT

Einzelne Gruppen können sich spezialisieren in:

- Kompletten Agenten entwickeln
- GA basic implementation
- YoloKnowledge Erweiterungen
- YoloKnowledge Kollisionsklassifizierer verbessern
- BFS mit KB verwenden
- Kontiunierliche Physik

Seid ihr interessiert?