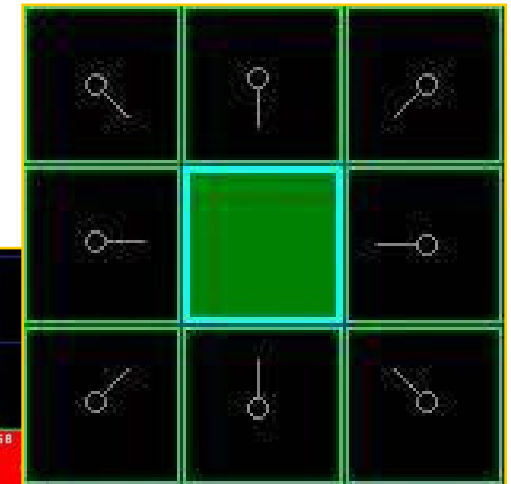
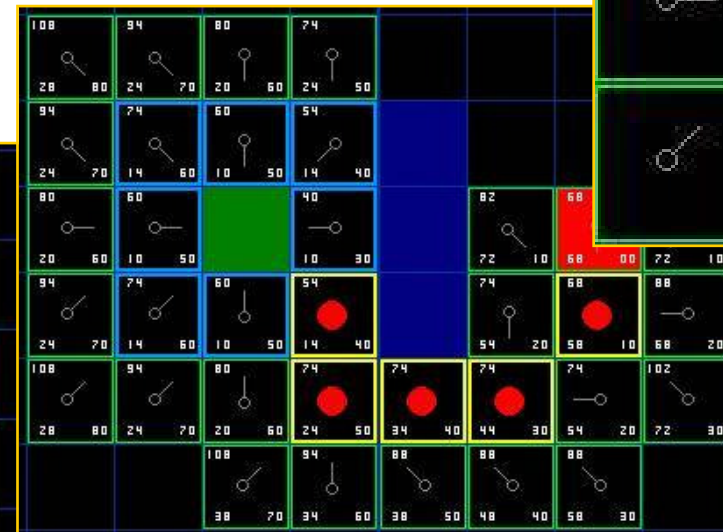
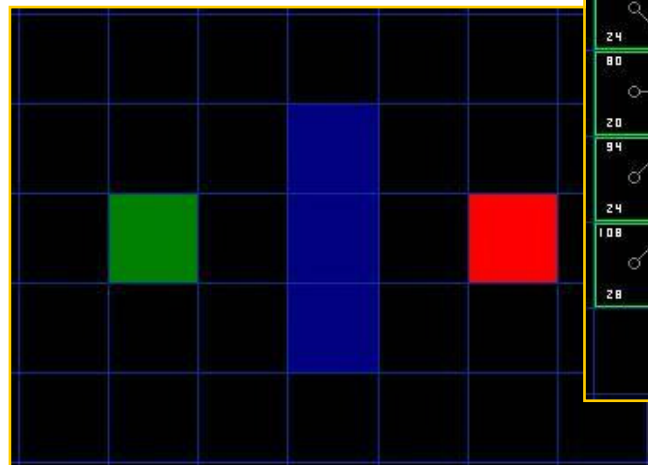


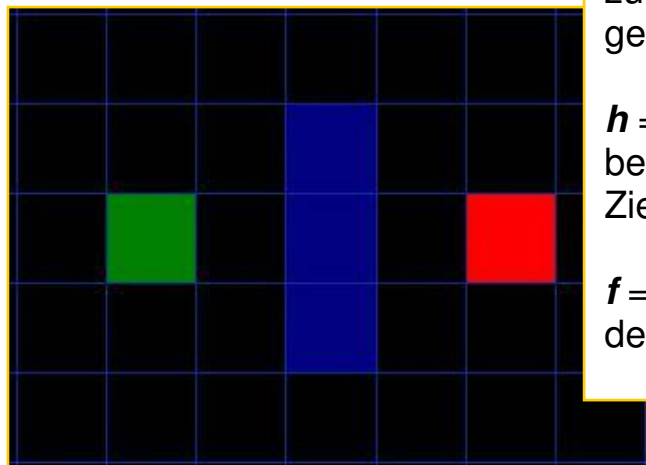
- 1 Einleitung
- 2 A* - Algorithmus
 - 2.1 Begriffserklärung
 - 2.2 Funktion und Beispiel
 - 2.3 Heuristik und Verfahren
- 3 Generic A* Machine
- 4 A* Design Architektur
- 5 Performance Optimierung
- 6 Tipps zur Optimierung
- 7 Diskussion

Erläuterung anhand
eines Beispiels ...



- 1 Einleitung
- 2 A* - Algorithmus
 - 2.1 Begriffserklärung
 - 2.2 Funktion und Beispiel
 - 2.3 Heuristik und Verfahren
- 3 Generic A* Machine
- 4 A* Design Architektur
- 5 Performance Optimierung
- 6 Tipps zur Optimierung
- 7 Diskussion

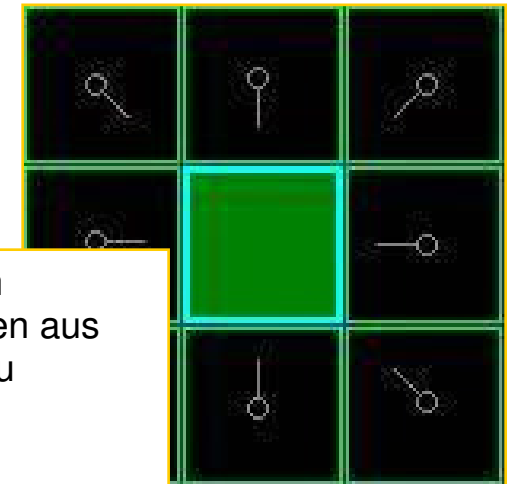
Erläuterung anhand eines Beispiels ...



g = Bewegungskosten – werden benötigt, um von dem Startknoten aus zu einem spezifischen Knoten zu gelangen

h = Geschätzte Kosten – werden benötigt, um von einem Knoten zu dem Zielknoten zu gelangen

f = Summe der geschätzten Kosten und der anfallenden Bewegungskosten



1 Einleitung

2 A* - Algorithmus

2.1 Begriffserklärung

2.2 Funktion und Beispiel

2.3 Heuristik und Verfahren

3 Generic A* Machine

4 A* Design Architektur

5 Performance Optimierung

6 Tipps zur Optimierung

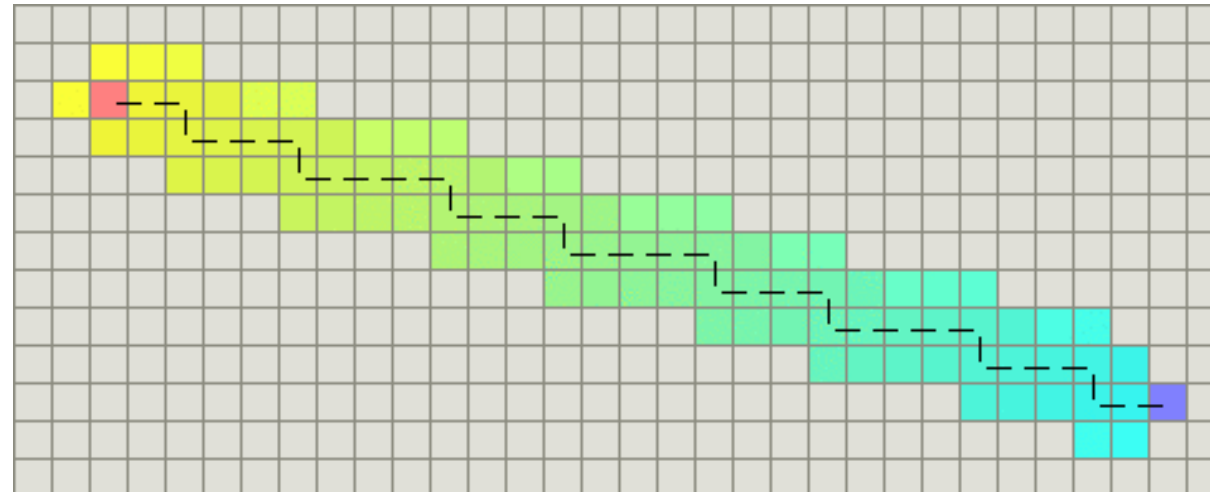
7 Diskussion

- Unterscheidung der Heuristik in zulässig und monoton
 - Zulässig – Kosten zum Ziel dürfen nicht, Kosten zwischen Knoten dürfen aber sehr wohl überschätzt werden
 - Monoton – Weder die Kosten zum Ziel, noch die Kosten zwischen den Knoten dürfen überschätzt werden

- 1 Einleitung
- 2 A* - Algorithmus
 - 2.1 Begriffserklärung
 - 2.2 Funktion und Beispiel
 - 2.3 Heuristik und Verfahren
- 3 Generic A* Machine
- 4 A* Design Architektur
- 5 Performance Optimierung
- 6 Tipps zur Optimierung
- 7 Diskussion

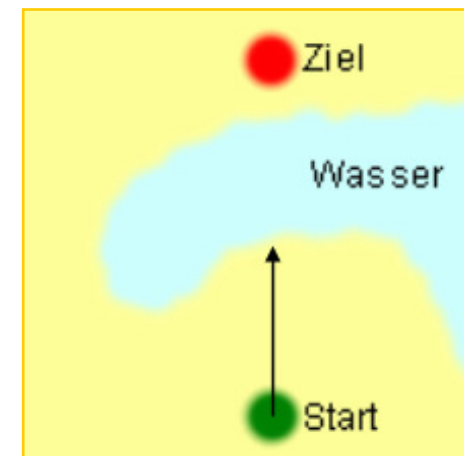
- Manhattan – Methode

$$h = 10 * (\text{abs} (\text{currentX} - \text{targetX}) + \text{abs} (\text{currentY} - \text{targetY}))$$



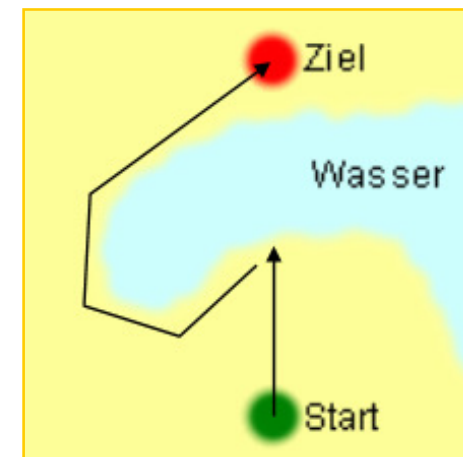
■ Quickpath

- Highspeed Pfadfinder für kurze Distanzen
- Nur bestimmte Anzahl an Revolutions für die Wegfindung
- Es wird der Punkt ausgewählt, der am nächsten am Ziel liegt
- Die Einheit bewegt sich auf diesen ermittelten Punkt als Zielpunkt zu



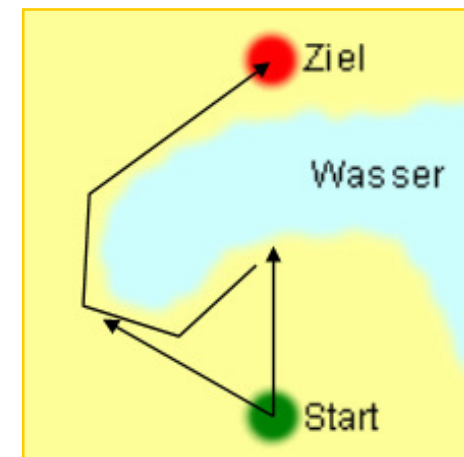
■ Fullpath

- Wird im Hintergrund durch den A* - Algorithmus ermittelt
- Startknoten ist der zuvor ermittelte Endknoten des Quickpath



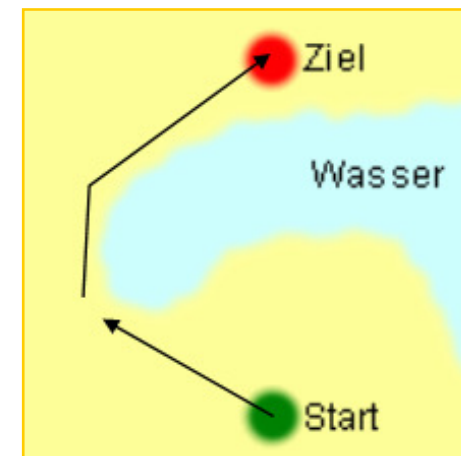
■ Splice Path

- Anpassung an realistischeres Verhalten
- Highspeed Pfadfinder für kurze Distanzen
- Startknoten ist die aktuelle Position der Einheit auf der Karte
- Punkt als Ziel für den Splice Path ergibt sich aus Erfahrungswerten



- Entfernen der Extra - Waypoints

➤ Abschließend werden die zusätzlichen Waypoints entfernt



- 1 Einleitung
- 2 A* - Algorithmus
- 3 Generic A* Machine
- 4 A* Design Architektur
- 5 Performance Optimierung
- 6 **Tipps zur Optimierung**
- 7 Diskussion

- **Reduzierung der iterativen Tiefe des A* - Algorithmus**
- Blockierte Knoten werden umgehend geschlossen
- Knoten mit zu hohen Kosten werden geschlossen
- Ermittlung eines Teilweges anstelle des vollständigen Pfades
- Cachen von fehlgeschlagenen Wegfindungen
- Begrenzung der Zeit, die in den A* - Algorithmus investiert wird
- Verwendung von Waypoints, um lange Wege zu generieren

- 1 Einleitung
- 2 A* - Algorithmus
- 3 Generic A* Machine
- 4 A* Design Architektur
- 5 Performance Optimierung
- 6 **Tipps zur Optimierung**
- 7 Diskussion

- Reduzierung der iterativen Tiefe des A* - Algorithmus
- **Blockierte Knoten werden umgehend geschlossen**
- Knoten mit zu hohen Kosten werden geschlossen
- Ermittlung eines Teilweges anstelle des vollständigen Pfades
- Cachen von fehlgeschlagenen Wegfindungen
- Begrenzung der Zeit, die in den A* - Algorithmus investiert wird
- Verwendung von Waypoints, um lange Wege zu generieren

- 1 Einleitung
- 2 A* - Algorithmus
- 3 Generic A* Machine
- 4 A* Design Architektur
- 5 Performance Optimierung
- 6 **Tipps zur Optimierung**
- 7 Diskussion

- Reduzierung der iterativen Tiefe des A* - Algorithmus
- Blockierte Knoten werden umgehend geschlossen
- **Knoten mit zu hohen Kosten werden geschlossen**
- Ermittlung eines Teilweges anstelle des vollständigen Pfades
- Cachen von fehlgeschlagenen Wegfindungen
- Begrenzung der Zeit, die in den A* - Algorithmus investiert wird
- Verwendung von Waypoints, um lange Wege zu generieren

- 1 Einleitung
- 2 A* - Algorithmus
- 3 Generic A* Machine
- 4 A* Design Architektur
- 5 Performance Optimierung
- 6 **Tipps zur Optimierung**
- 7 Diskussion

- Reduzierung der iterativen Tiefe des A* - Algorithmus
- Blockierte Knoten werden umgehend geschlossen
- Knoten mit zu hohen Kosten werden geschlossen
- **Ermittlung eines Teilweges anstelle des vollständigen Pfades**
- Cachen von fehlgeschlagenen Wegfindungen
- Begrenzung der Zeit, die in den A* - Algorithmus investiert wird
- Verwendung von Waypoints, um lange Wege zu generieren

- 1 Einleitung
- 2 A* - Algorithmus
- 3 Generic A* Machine
- 4 A* Design Architektur
- 5 Performance Optimierung
- 6 **Tipps zur Optimierung**
- 7 Diskussion

- Reduzierung der iterativen Tiefe des A* - Algorithmus
- Blockierte Knoten werden umgehend geschlossen
- Knoten mit zu hohen Kosten werden geschlossen
- Ermittlung eines Teilweges anstelle des vollständigen Pfades
- **Cachen von fehlgeschlagenen Wegfindungen**
- Begrenzung der Zeit, die in den A* - Algorithmus investiert wird
- Verwendung von Waypoints, um lange Wege zu generieren

- 1 Einleitung
- 2 A* - Algorithmus
- 3 Generic A* Machine
- 4 A* Design Architektur
- 5 Performance Optimierung
- 6 **Tipps zur Optimierung**
- 7 Diskussion

- Reduzierung der iterativen Tiefe des A* - Algorithmus
- Blockierte Knoten werden umgehend geschlossen
- Knoten mit zu hohen Kosten werden geschlossen
- Ermittlung eines Teilweges anstelle des vollständigen Pfades
- Cachen von fehlgeschlagenen Wegfindungen
- **Begrenzung der Zeit, die in den A* - Algorithmus investiert wird**
- Verwendung von Waypoints, um lange Wege zu generieren

- 1 Einleitung
- 2 A* - Algorithmus
- 3 Generic A* Machine
- 4 A* Design Architektur
- 5 Performance Optimierung
- 6 **Tipps zur Optimierung**
- 7 Diskussion

- Reduzierung der iterativen Tiefe des A* - Algorithmus
- Blockierte Knoten werden umgehend geschlossen
- Knoten mit zu hohen Kosten werden geschlossen
- Ermittlung eines Teilweges anstelle des vollständigen Pfades
- Cachen von fehlgeschlagenen Wegfindungen
- Begrenzung der Zeit, die in den A* - Algorithmus investiert wird
- **Verwendung von Waypoints, um lange Wege zu generieren**