

Wiki zum Projekt Ampelsteuerung

Projekt Information

Das Projekt “Ampelsteuerung” ist die Praktische Umsetzung des Erlernten aus der Vorlesung Wissensverarbeitung im 7. Semester. Die Aufgabenstellung zu dem Projekt lautet wie folgt: >**Ampelsteuerung**> >Modellieren Sie die Verkehrssituation an der komplexen Kreuzung am Bahnhof in Stralsund (Ampeln, Schranken, Sensoren), von der Barther Straße bis zu den beiden Teichen in Richtung Stadtmitte. (Das Verkehrsaufkommen soll zufällig variieren, der Verkehrsfluss simuliert werden.) > >Lässt sich aufteilen in einen Prolog-Teil für die Berechnung und einen anders implementierten Teil für die grafische Darstellung.

Das Projekt wurde im Zeitraum von Anfang Oktober bis zum 25.11.2016 bearbeitet.

Starten des Projekts

Bauamt

- [Ampelschaltung Bauamt HST](#)
- [Ampelschaltung Interpretation](#)

Prolog und Abfragen aus C

- [Anfragen an Prolog und deren Verarbeitung](#)
- [Funktionsweise der Ereignis-Queue in Prolog](#)

Scripts

Szene

- [Steuerung und UI](#)
- [Modelle](#)

Treffen mit dem Professor

- [09.11.2016](#) # Anfragen an Prolog und deren Verarbeitung **alle Bezeichnungen beziehen sich auf das PDF vom Bauamt**

Allgemein Anfragen an Prolog

Prolog mitteilen, dass ein neues Ereignis eingetreten ist:

```
neuesEreignis(Ampelkreuzung, Ausloeser).
```

Beispiel:

```
?-neuesEreignis(a, k10).  
true
```

Die nächste Phase von Prolog anfordern:

```
getnextPhase(Kreuzung, GrueneAmpelnListe).
```

liefert:

```
GG=[[Liste_der_gruenen_Ampeln], phase, Zeit_in_Sekunden].
```

Beispiel:

```
?-getnextPhase(a, GG).  
GG=[[k9, k13, b4, h6, fg9], phase11, 35].
```

Kreuzung A, der kleine Phasenkreis (4 Phasen)

Auslöser	Folgephase	Phasendauer in sec
<i>(keine Aktion)</i>	Phase 11	35
b3	Phase 12	10
k10	Phase 13	20
fa10	Phase 14	18
k12	Phase 14	18

Kreuzung B, der große Phasenkreis (6 Phasen)

Auslöser	Folgephase	Phasendauer in sec
<i>(keine Aktion)</i>	Phase 1	35
fa4	Phase 3	9
fa1	Phase 2	10
k8	Phase 2	10

Auslöser	Folgephase	Phasendauer in sec
k6	Phase 2	10
k4	Phase 4	5
b1	Phase 5	5
boomgate	Phase 6	15

Suche nach dem besten Pfad

Das Übergeben bzw. Einlesen eines Graphen an/in Prolog

`setNewGraph(Liste)`

Das Prädikat `arc` zum Verbinden von Knoten innerhalb eines Graphen nutzen

```
arc(e, f, 5, 7)
e:=Knoten A
f:=KnotenB
5:= Pfadkosten
7:=Heuristischer Wert

7           5
(e)-----(f)
```

Beispiel:

```
liste=[arc(e, f, 5, 7), arc(s, e, 2, 6), arc(s, a, 2, 6), arc(a, b, 2, 5), arc(b, c, 2, 4),
?-setNewGraph(liste).
true
```

Erhalten des konkreten Pfades

`getPath(Start,Goal,Path)`

Beispiel:

```
?-getPath(s,d,Path)
Path = [d, c, b, a, s].
```

Darstellung der Funktionsweise Queue

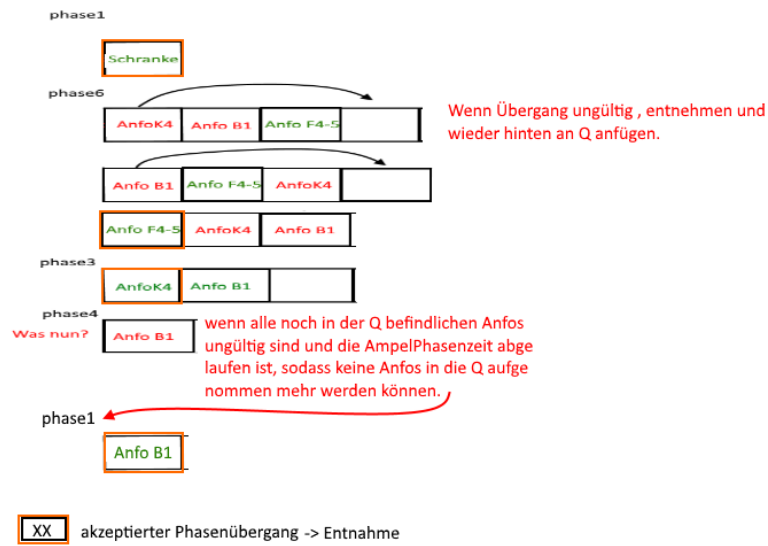


Figure 1: FunktionsweiseQueue

Steuerung und UI

Steuerung

Für die Steuerung in der Szene sind folgende Keybindings festgelegt

Taste	Aktion
w	Kamera noch vorne bewegen
a	Kamera noch links bewegen
s	Kamera noch rechts bewegen
d	Kamera zurück bewegen
Shift	Kamera schneller bewegen
Mausrad hoch	Zoom in
Mausrad runter	Zoom out

Eine weitere Eingabemöglichkeit wäre ein Spiele-Controller. Dabei kann es jedoch sein, dass **nicht** jede Funktion vom Spiele-Controller unterstützt wird

UI

Über das UI kann der Benutzer das geschehen in der Szene besser verfolgen, da hier die Daten übersichtlich zusammen gefasst werden. Das UI ist in zwei Bereiche aufgeteilt. Zum einen in den Bereich “Konsole” und zum Anderen in “Übersicht der Zustände, Kamerapositionen, Geschwindigkeitsregler und der Autospawner”.

Konsole In der Konsole werden alle Anfragen, die an Prolog aus C# gestellt werden, aufgezeichnet. Ebenfalls werden die Antworten auf die Abfragen die von Prolog an C# gehen aufgezeichnet. Die neusten Einträge stehen jeweils am unteren Rand.

Übersicht der Zustände, Kamerapositionen, Geschwindigkeitsregler und der Autospawner

Übersicht der Zustände

Kamerapositionen Unter dem Punkt `Camera positions` befinden sich 4 Buttons mit der Beschriftung 1-4. Beim Auswählen einen dieser Buttons wird die Kamera der Szene zu einer vordefinierten Stelle transformiert um eine schnellere Bewegung in der Szene zu ermöglichen

Button	Position in der Szene
1	Startposition in der Szene, Übersicht über ganze Szene
2	Kreuzung b, Bahnhofskreuzung
3	Kreuzung a, Kreuzung Richtung Altstadt Stralsund
4	Bahnübergang

Geschwindigkeitsregler Der Geschwindigkeitsregler ändert die Geschwindigkeit in der Szene. Hinter dem Regler befindet sich ein Inputfield, über das die aktuelle Geschwindigkeit abgelesen und auch abgeändert werden kann. Wird der Regler nach links verschoben wird der Geschwindigkeitswert herunter gesetzt, nach rechts hochgesetzt.

Geschwindigkeitswert	Auswirkung
1	Normal Geschwindigkeit
<1	Spielgeschwindigkeit wird verlangsamt
>1	Spielgeschwindigkeit wird erhöht

Autospawner

Mitschriften vom Vorstellen 09.11.2016

- [x] bool umbenennen
- [] Doku erklären was Phase ist usw.
- [] checkifzulaessing kein wahrheitswert
- [x] Cut bei Q bei checkQueueIsEmpty Q kann [] obere
- [x] Cut kann weg bei checkQueueIsEmpty unten
- [] Sinnvolle Variablennamen
- [x] Zeile 278 queue(Kreuzung, Queue) ist überflüssig
- [] checkalreadyAccepted verodern?
- [x] mehrere getNextPhase -> mehrere oder Fälle -> Refactoring
- [x] 184 und 200 wer hat was an Q geändert -> doCheck !!

- [] Ablauf für Autos ... wann hat er den Punkt erreicht
- [] Zeit anhalten oder verlangsamt für die Autos usw.
- [] Doku => Handbuch
- [x] Cut Klausel bricht ab + Prädikat

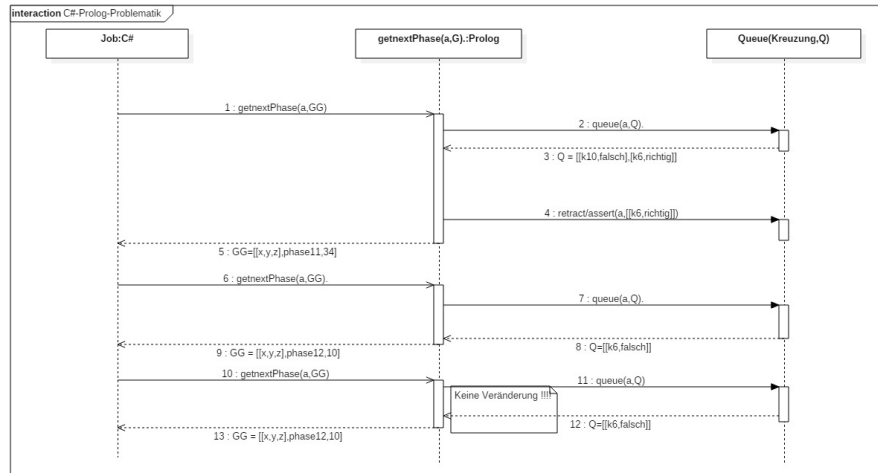


Figure 2: C-Prolog-Problematic

Wie dem Dokument des Bauamtes Stralsund zu entnehmen ist, gehen die Konstrukteure von 2 Kreuzungsanlagen bzw Ampelanlagen aus. Dies ist zum einem die Kreuzung a, welche sich am unteren Ende des Bahnhofkomplexes befindet und zum anderen die Kreuzung b, welche sich direkt am Bahnhof bzw. Bahnhofsvorplatz befindet. Jede Kreuzung besitzt eine Anzahl n von verschiedenen Phasen p(Zuständen). Jede Kreuzung besitzt eine Standardphase. Jede Kreuzung befindet sich für einen gewissen Zeitraum in genau einer Phase. Jede Ampelanlage besitzt n Ampeln. Eine Ampel kann dabei in 3 verschiedenen Farben aufleuchten laut StVo grün , rot ,gelb Jeder Zustand wird durch die Anzahl der in ihm grün leuchtenden Ampeln beschrieben. Jeder Phasenwechsel wird durch einen Impuls bzw Auslöser hervorgerufen. Nicht Jeder Phasenwechsel ist zulässig sondern gilt nur unter bestimmten Bedingungen. Wenn keine Impulse über einen gewissen Zeitraum wahrgenommen werden, wird wieder in die Standardphase gewechselt.

Abkürzung	Bedeutung
An	wenn nicht rot, leer wenn am
Ab	
TF	TU - An oder auch gesagt Wie lange die Grünphase ist, Effektive Fahrtfreigabephase

Abkürzung	Bedeutung
TU	Zeitraum für Phasenübergang
H	Busampeln
K	Strassenampeln für Autos
FG	Fussgängerampeln
ANFO	Fussgängerampeln
P2_TAG Ph 125	besteht aus den Phasen 1,2,5 die leicht verändert sind

- kleiner null bedeutet Vorzustand