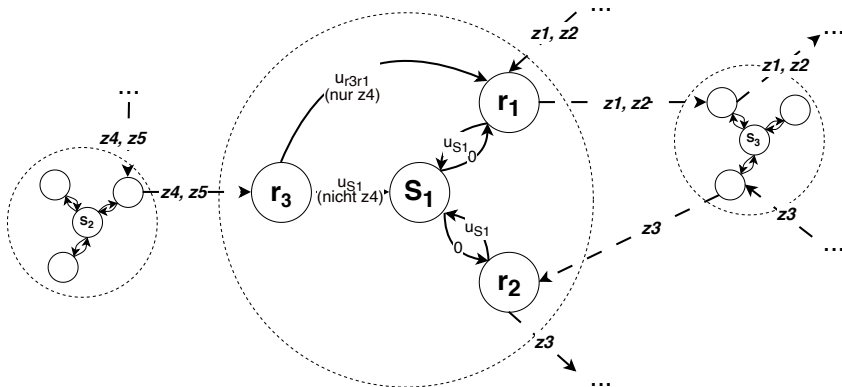


Seminar Algorithmen zum Fahrplanentwurf



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Graphenmodelle



Analoge Fahrpläne

Wie man sie kennt und nie benutzt...



3/4

Darmstadt → Langen → Frankfurt → Kronberg / Bad Soden

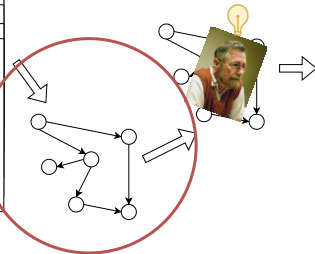
Kundendialog DB Regio Hessen, Tel.: (01806) 99 66 33 (20 ct/Anruf aus dem dt. Festnetz, Mobilfunkpreise anbieterabhängig, max. 60 ct/Anruf)













Montag - Freitag									
Darmstadt Hbf	4.35	5.05				21.35	22.05	22.35	23.35
Egelsbach	4.46	5.16			30	21.46	22.16	22.46	23.46
Langen (Hess)	4.49	5.04	5.19	5.34		21.34	21.49	22.19	22.49
Langen-Flugsicherung	4.51	5.06	5.21	5.36		21.36	21.51	22.21	22.51
Dreieich-Buchschlag	4.53	5.08	5.23	5.38		21.38	21.53	22.23	22.53
Neu-Isenburg	4.56	5.11	5.26	5.41	alle	21.41	21.56	22.26	22.56
Frankfurt (M) Süd	5.03	5.18	5.33	5.48	15	21.48	22.03	22.33	23.03
Frankfurt (M) Ostendstraße	5.07	5.22	5.37	5.52	Min	21.52	22.07	22.37	23.07
Frankfurt (M) Hauptwache	5.10	5.25	5.40	5.55		21.55	22.10	22.40	23.10
Frankfurt (M) Hbf (tief)	5.14	5.29	5.44	5.59		21.59	22.14	22.44	23.14
Frankfurt (M) West	5.20	5.35	5.50	6.05		22.05	22.20	22.50	23.20
Niederhöchstadt	5.31	5.46	6.01	6.16		22.16	22.31	23.01	23.31
Kronberg (Taunus)		5.51	6.21		30	22.21		23.06	0.06
Bad Soden (Taunus)	5.39	6.09			30		22.39		23.39
Samstag									
Darmstadt Hbf	4.35	5.05				18.35		21.35	22.35
Egelsbach	4.46	5.16			30	18.46		21.46	22.46
Langen (Hess)	4.49	5.04	5.19	5.34		18.34	18.49	21.49	22.19
Langen-Flugsicherung	4.51	5.06	5.21	5.36		18.36	18.51	21.51	22.21
Dreieich-Buchschlag	4.53	5.08	5.23	5.38		18.38	18.53	21.53	22.23
Neu-Isenburg	4.56	5.11	5.26	5.41	alle	18.41	18.56	21.56	22.26
Frankfurt (M) Süd	5.03	5.18	5.33	5.48	15	18.48	19.03	22.03	22.33
Frankfurt (M) Ostendstraße	5.07	5.22	5.37	5.52	Min	18.52	19.07	22.07	22.37
Frankfurt (M) Hauptwache	5.10	5.25	5.40	5.55		18.55	19.10	22.10	22.40
Frankfurt (M) Hbf (tief)	5.14	5.29	5.44	5.59		18.59	19.14	22.14	22.44
Frankfurt (M) West	5.20	5.35	5.50	6.05		19.05	19.20	22.20	22.50
Niederhöchstadt	5.31	5.46	6.01	6.16		19.16	19.31	22.31	23.01
Kronberg (Taunus)		5.51	6.21		30	19.21			23.06
Bad Soden (Taunus)	5.39	6.09			30	19.39		22.39	23.39

The diagram shows a sequence of three network states. The first state is a simple graph with 5 nodes and 4 edges. A large arrow points to the second state, which is a more complex graph with 7 nodes and 10 edges. Another large arrow points to the third state, which features a photograph of a man (likely a historical figure) as a central node, with a lightbulb icon above it, signifying a new idea or insight. This visualizes how a network can evolve to produce novel concepts.



S	3/4	Darmstadt → Langen → Frankfurt → Kirsberg / Bad Soden
<small> Darmstadt Hbf (S) - 10 min - Langen (S) - 10 min - Frankfurt (S) - 10 min - Kirsberg (S) - 10 min - Bad Soden (S) Darmstadt Hbf (S) - 10 min - Langen (S) - 10 min - Frankfurt (S) - 10 min - Kirsberg (S) - 10 min - Bad Soden (S) </small>		
S	3/4	Darmstadt → Langen → Frankfurt → Kirsberg / Bad Soden
<small> Darmstadt Hbf (S) - 10 min - Langen (S) - 10 min - Frankfurt (S) - 10 min - Kirsberg (S) - 10 min - Bad Soden (S) Darmstadt Hbf (S) - 10 min - Langen (S) - 10 min - Frankfurt (S) - 10 min - Kirsberg (S) - 10 min - Bad Soden (S) </small>		
S	3/4	Darmstadt → Langen → Frankfurt → Kirsberg / Bad Soden
<small> Darmstadt Hbf (S) - 10 min - Langen (S) - 10 min - Frankfurt (S) - 10 min - Kirsberg (S) - 10 min - Bad Soden (S) Darmstadt Hbf (S) - 10 min - Langen (S) - 10 min - Frankfurt (S) - 10 min - Kirsberg (S) - 10 min - Bad Soden (S) </small>		
S	3/4	Darmstadt → Langen → Frankfurt → Kirsberg / Bad Soden
<small> Darmstadt Hbf (S) - 10 min - Langen (S) - 10 min - Frankfurt (S) - 10 min - Kirsberg (S) - 10 min - Bad Soden (S) Darmstadt Hbf (S) - 10 min - Langen (S) - 10 min - Frankfurt (S) - 10 min - Kirsberg (S) - 10 min - Bad Soden (S) </small>		



19:24 - 19:40 16min		15,30 €	
20:00	20:28	16:11	
Darmstadt-Hbf		Frankfurt/Main-Hbf	
 1x Tages-Mehrfahrten v.a.s.		Details 	Kundenkonto einloggen
20:07 - 20:24 16min			
20:11	20:27	16:08	
Darmstadt-Hbf		Frankfurt/Main-Hbf	
 1x Tages-Mehrfahrten v.a.s.		Details 	Kundenkonto einloggen
20:30 - 20:48 17min			
20:30	20:49	16:07	
Darmstadt-Hbf		Frankfurt/Main-Hbf	
 1x Tages-Mehrfahrten v.a.s.		Details 	Kundenkonto einloggen
20:35 - 21:03 27min			
20:35	21:02	15:1	
Darmstadt-Hbf		Frankfurt (Hbf) (Sbf)	
 1x Tages-Mehrfahrten v.a.s.		Details 	Kundenkonto einloggen

Frage 1

Wie wandle ich Fahrpläne zu Graphen um?

Frage 2

Wie modelliere ich meinen Graphen, um das Earliest-Arrival Problem zu lösen?

Überblick

- Grundlagen
- Das 1. Graphenmodell
 - ▣ Verfeinerungen
- Das 2. Graphenmodell
 - ▣ Verfeinerungen
- Vergleiche
- Zusammenfassung

1. Griff in die Terminologie-Kiste

Züge und Stationen

Was ist ein Zug?



1. Griff in die Terminologie-Kiste

Züge und Stationen

Was ist ein Zug?



1. Griff in die Terminologie-Kiste

Züge und Stationen

Was ist ein Zug?



1. Griff in die Terminologie-Kiste

Züge und Stationen

Was ist ein Zug?



1. Griff in die Terminologie-Kiste

Züge und Stationen

Was ist eine Station?



1. Griff in die Terminologie-Kiste

Züge und Stationen

Was ist eine Station?



Größenordnungen

- Ein paar Zahlen im Jahr 2008 (**nur** Züge):

Anzahl	
Stationen	etwa™ 5000
Züge	68073
Fußwege	425

Wie modelliere ich einen Fahrplan als Graphen?

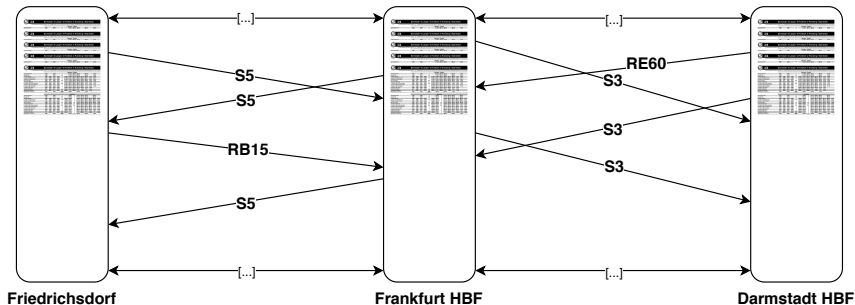
- Simple Idee: Knoten sind Stationen, Kanten sind Verbindungen



Was ist das Problem hier?

Wie modelliere ich einen Fahrplan als Graphen?

- Wir müssen irgendwie Zeitverhältnisse innerhalb des Graphen darstellen!



Was ist das Problem hier?

Probleme

Die es zu lösen gilt...

- Zeitverhältnisse im Graphen

Probleme

Die es zu lösen gilt...

- Zeitverhältnisse im Graphen
- **Takt**fahrpläne

Probleme

Die es zu lösen gilt...

- Zeitverhältnisse im Graphen
- **Takt**fahrpläne
- Umstiege bzw. Umsteigezeiten

Probleme

Die es zu lösen gilt...

- Zeitverhältnisse im Graphen
- **Takt**fahrpläne
- Umstiege bzw. Umsteigezeiten
- Fußwege zwischen Stationen

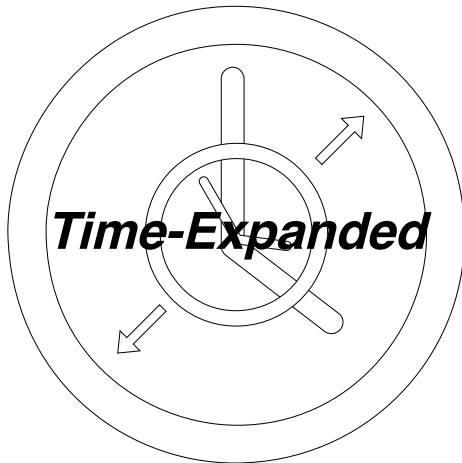
Probleme

Die es zu lösen gilt...

- Zeitverhältnisse im Graphen
- **Takt**fahrpläne
- Umstiege bzw. Umsteigezeiten
- Fußwege zwischen Stationen
- Intermodalität

Zwei gängige Modelle

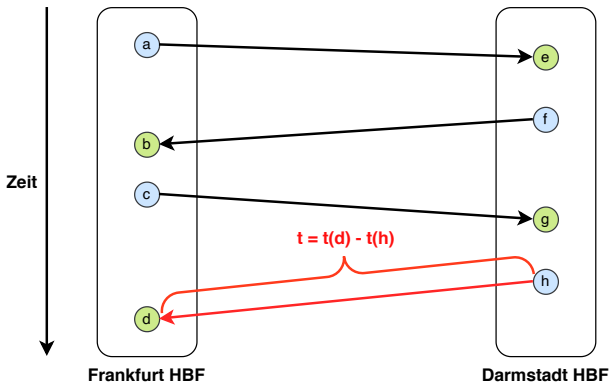
Time-Expanded vs. Time-Dependent



Das Time-Expanded Modell

Train-Edges

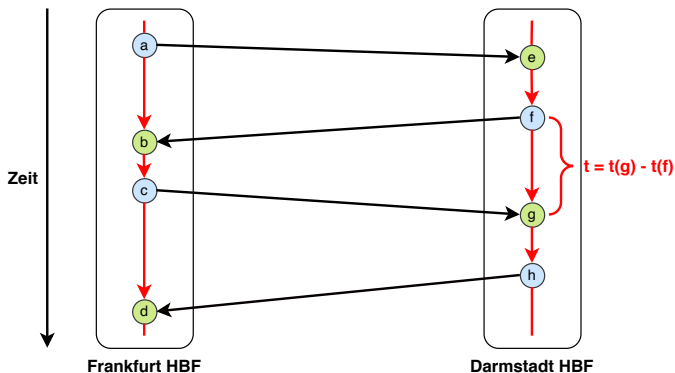
- Modelliere jedes "Event" innerhalb einer Station als eigenen Knoten.



Das Time-Expanded Modell

Waiting-Edge

- Erstelle eine Stationsinterne Kante, die **Exchange-Edge**.



Das Time-Expanded Modell

Was bringt uns das Ganze jetzt?

Kanten innerhalb einer Station \approx "Umstiege"

16:30 - 17:17 | 1 Umstieg

RE 60

RE 16

Darmstadt Hbf

Friedrichsdorf(Taunus)

Es liegen Meldungen vor.

Details ^

16:30 Darmstadt Hbf >

Gl. 7



21min

RE 60

nach Frankfurt(Main)Hbf

Beförderer

Fahrradmitnahme

Weitere Informationen

Zuginformationen

1 Haltestelle v

16:51 Frankfurt(Main)Hbf >

Gl. 10

0min

Umstieg

Gl. 14

16:51 Frankfurt(Main)Hbf >

26min

RE 16

nach Friedberg(Hess)

Beförderer

Fahrradmitnahme

Weitere Informationen

Zuginformationen

2 Haltestellen v

17:17 Friedrichsdorf(Taunus) >

Gl. 5

16:30 - 17:17 | 1 Umstieg

RE 60

RE 16

Darmstadt Hbf

Friedrichsdorf(Taunus)

Es liegen Meldungen vor.

Details ^

16:30 Darmstadt Hbf >

Gl. 7



21min

RE 60

nach Frankfurt(Main)Hbf

Beförderer

Fahrradmitnahme

Weitere Informationen

Zuginformationen

1 Haltestelle v

16:51 Frankfurt(Main)Hbf >

0min

Umstieg

16:51 Frankfurt(Main)Hbf >

26min

RE 16

nach Friedberg(Hess)

Beförderer

Fahrradmitnahme

Weitere Informationen

Zuginformationen

2 Haltestellen v

17:17 Friedrichsdorf(Taunus) >

Gl. 10

Gl. 14

Gl. 5

Frage 3

Wie modelliere ich Umstiegszeiten mit "Puffer"?

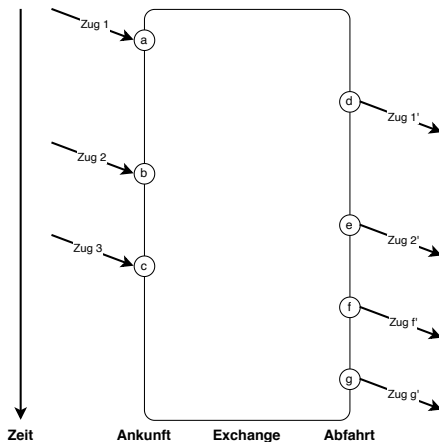
2. Griff in die Terminologie-Kiste

Realistische Umstiegsregeln

- Definiere **Konstante** und **Variable** Umstiegsregeln:
 1. **Standard-Umstiegszeit für alle Züge**
 2. **Regeln basierend auf Transferklassen & Zuglinien**
 3. **Regeln zwischen einzelnen Zügen**

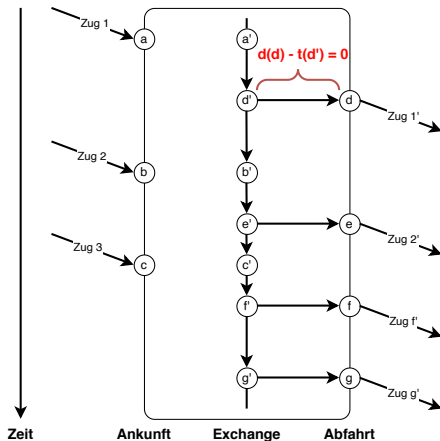
Konstante Umstiegszeiten

- Spalte Exchange-Edge in Ankunfts- und Abfahrtsknoten



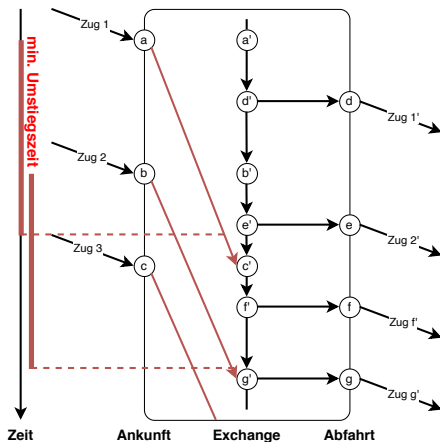
Konstante Umstiegszeiten

■ Erstelle neue Exchange Edge



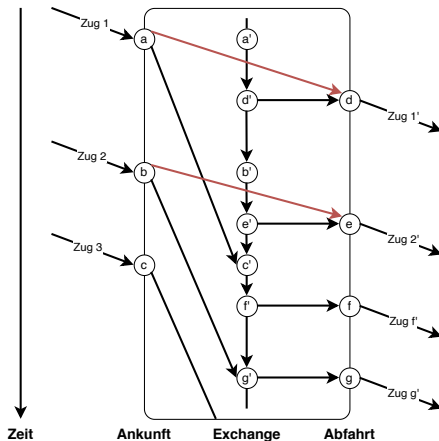
Konstante Umstiegszeiten

- Kanten von Ankunft → Exchange-Edge basierend auf min. Umstiegszeit



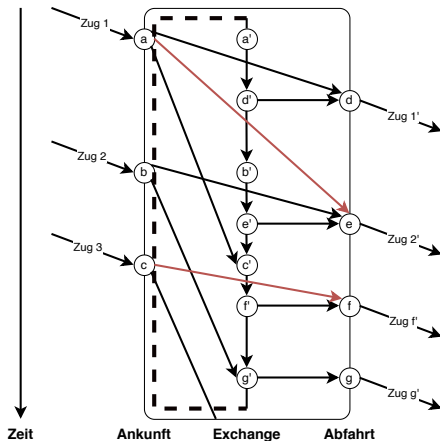
Variable Umstiegszeiten

- Erstelle Kanten von Ankunft → Abfahrt des selben Zuges



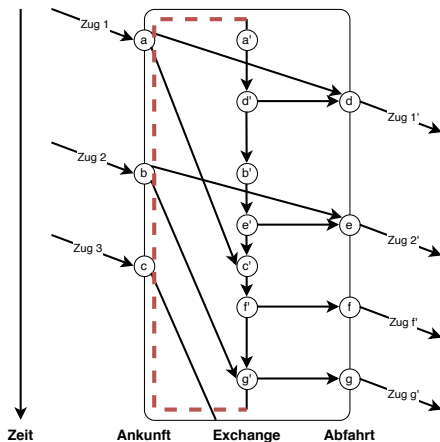
Variable Umstiegszeiten

- Weitere variable Umstiegszeiten als zzgl. Kanten



Taktfahrplanmodellierung

■ Taktfahrplanmodellierung via Exchange-Edge-Schleife



Verfeinerung 1: Verkehrstage

Problem: Der Takt unseres Fahrplans ist nicht nur ein Tag!

- Wie würde unser Graph wachsen, wenn wir N Tage modellieren würden?
 - ▣ Wir hätten etwa N -mal so viele Knoten und Kanten!

Verfeinerung 1: Verkehrstage

- Versehe Knoten mit absoluter Zeit $[0, N \cdot 1440]$
- Versehe Kanten mit einer Liste an Verkehrstagen: $[d], d \in N$
- Beim SP-Durchlauf:

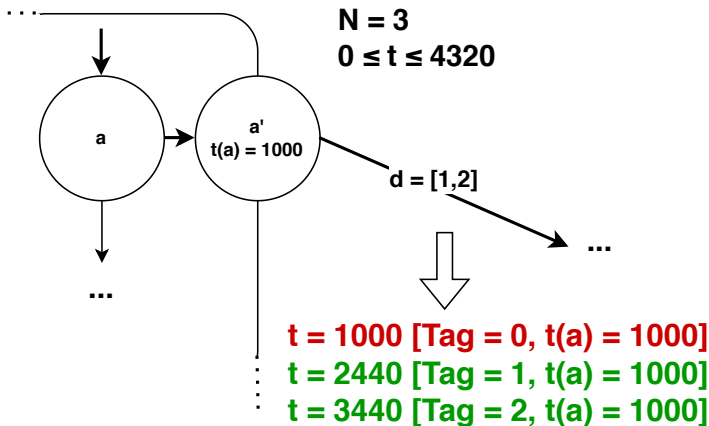
$$d_i = \text{floor}\left(\frac{t_{\text{Anfrage}}}{1440}\right)$$

- Und wenn der Zug an Tag d_i fährt:

$$\text{length}(E) = (t_{\text{Ankunft}} - t_{\text{Abfahrt}}) \bmod 1440$$

Verfeinerung 1: Verkehrstage

Das ganze als Visuelles Beispiel



Verfeinerung 1: Verkehrstage

Problem: Wie behandeln wir besuchte Knoten beim SP-Durchlauf, die invalide sind?

¹Pyrga et. al.: Efficient Models for Timetable Information in Public Transportation Systems

Verfeinerung 1: Verkehrstage

Problem: Wie behandeln wir besuchte Knoten beim SP-Durchlauf, die invalide sind?

- Als Beispiel am Dijkstra:
 - ▣ $dist \rightarrow \infty$
 - ▣ Füge Knoten wieder am Ende des Sets ein ("Nächster" Tag).

Weitere Verfeinerungen sind möglich¹...

¹Pyrga et. al.: Efficient Models for Timetable Information in Public Transportation Systems

Verfeinerung 2: Fußwege

Die Rückkehr der Terminologie-Kiste

Was macht Fußwege so besonders?

- Fußwege können zu jeder Zeit genutzt werden
- Man könnte sagen sie sind... *Zeitabhängig* (Time-Dependent)...

Foreshadow

verb

A literary device used to hint at events yet to come – and to keep readers guessing.

Verfeinerung 2: Fußwege

- Alle Fußwege im Graphen speichern ist nicht sinnvoll.

Verfeinerung 2: Fußwege

- Alle Fußwege im Graphen speichern ist nicht sinnvoll.
- Speichere Wege in jeder Station, maskiere Wege in Suchanfragen als Kanten zwischen Stationen
 - ▣ Durchsuche Fußwege, wenn eine Arrival-Node bearbeitet wird
- Beachte Fußwege beim Start einer Reise!

Größenordnungen

- Ein paar Zahlen im Jahr 2008 (**nur** Züge):

Anzahl an Knoten	
Ankunft	801.8 Tsd.
Abfahrt	801.8 Tsd.
<i>Change-Departure</i>	556.6 Tsd.
Gesamt	2160.2 Tsd.

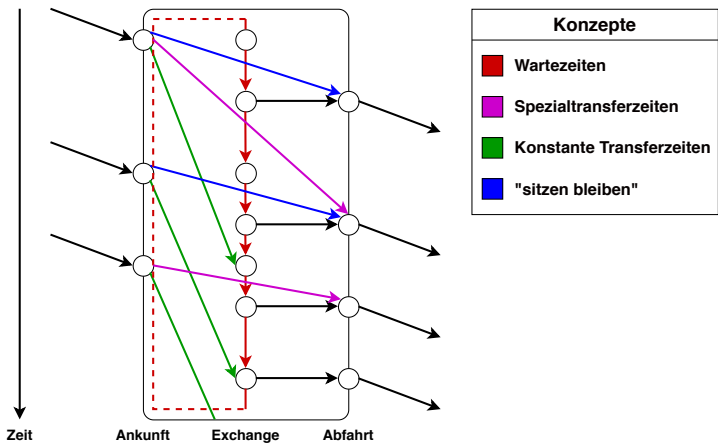
Größenordnungen

- Ein paar Zahlen im Jahr 2008 (**nur** Züge):

Anzahl an Kanten	
Zug	801.8 Tsd.
Weiterfahrt	733.7 Tsd.
Ankunft	796.7 Tsd.
Abfahrt	796.7 Tsd.
Warten (auf Exchange-Edge)	556.6 Tsd.
Besondere Umstiege	20.2 Tsd.
Gesamtanzahl	3705.7 Tsd.

Time-Expanded

Alle Konzepte auf einen Blick

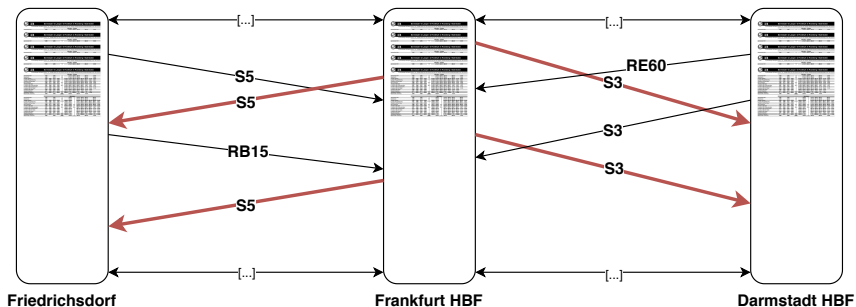




Time-Dependent

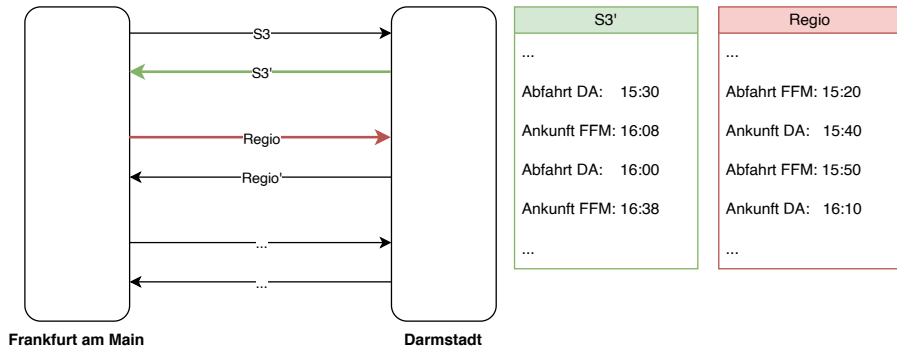
Das Time-Dependent Modell

- Denken wir zurück an das erste Beispiel...



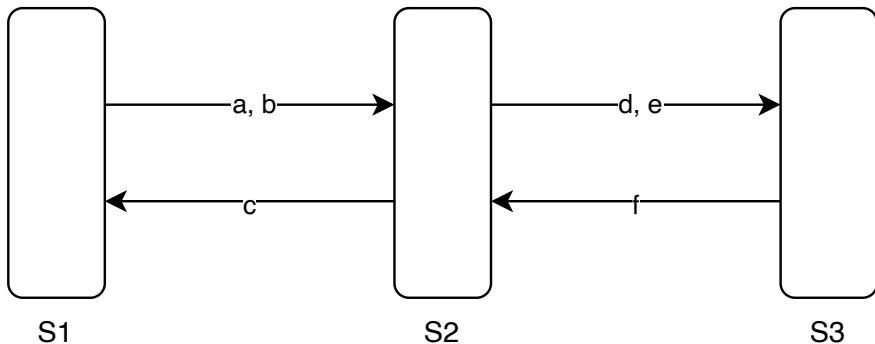
Das Time-Dependent Modell

- Jetzt bauen wir das ganze etwas um...



Das Time-Dependent Modell

- Oder etwas formeller:



Das Time-Dependent Modell

- Wir definieren für jede Kante (u, v) eine Funktion $f(t) : \mathcal{T} \rightarrow \mathcal{T}$.
 - ▣ $t \in \mathcal{T}, \mathcal{T} \triangleq \text{Zeit}$.

Das Time-Dependent Modell

- Wir definieren für jede Kante (u, v) eine Funktion $f(t) : \mathcal{T} \rightarrow \mathcal{T}$.
 - ▣ $t \in \mathcal{T}, \mathcal{T} \triangleq \text{Zeit}$.
- Dann ist unser Kantengewicht (*Reisezeit*):

$$\text{travel_time}(t) = f_{(u,v)}(t) - t$$

Wie sieht so eine Funktion in der Realität aus?

Das Time-Dependent Modell

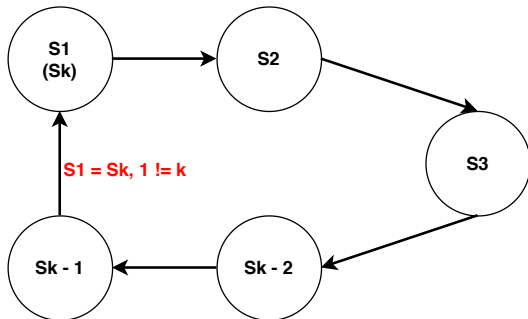
Konstante Umstiegszeiten

- Für konstante Umstiegszeiten definieren wir Zugrouten:
 - ▣ Sei R eine Zugroute $s_0, s_1, \dots, s_{k-1}, s_k$ für $k > 0$

Das Time-Dependent Modell

Konstante Umstiegszeiten

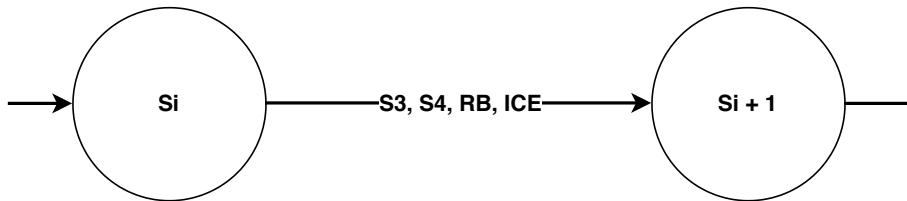
- Für konstante Umstiegszeiten definieren wir Zugrouten:
 - ▣ Sei R eine Zugroute $S_0, S_1, \dots, S_{k-1}, S_k$ für $k > 0$
 - ▣ Erlaubt: $S_i = S_j, i, j \in k$, für $i \neq j$ (Schleifen)!



Das Time-Dependent Modell

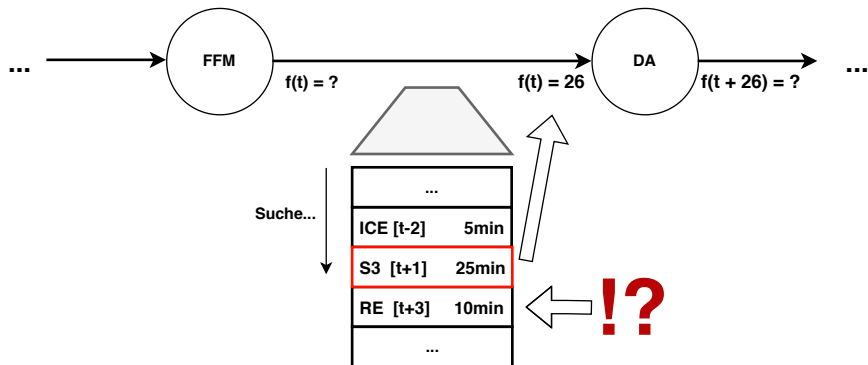
Konstante Umstiegszeiten

- Jetzt gruppieren wir alle Züge, die die gleiche Strecke fahren, in eine Zugroute
- Was ist hier das Problem?



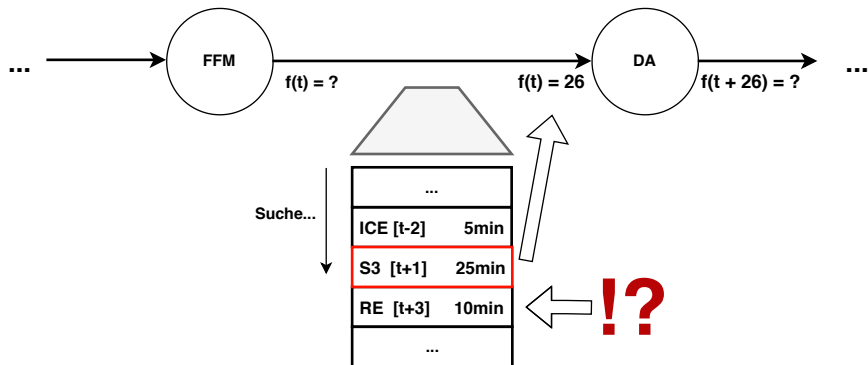
Das Time-Dependent Modell

Konstante Umstiegszeiten



Das Time-Dependent Modell

Konstante Umstiegszeiten

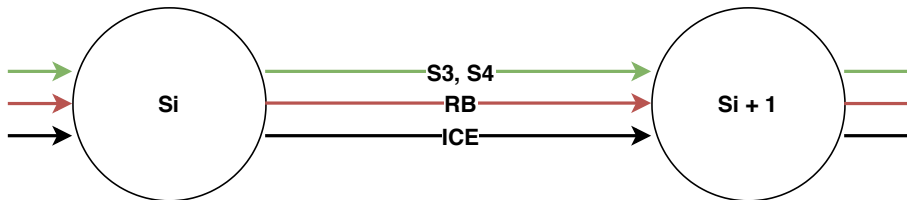


In welchen Fällen aber wäre es doch besser, die S3 zu nehmen?

Das Time-Dependent Modell

Konstante Umstiegszeiten

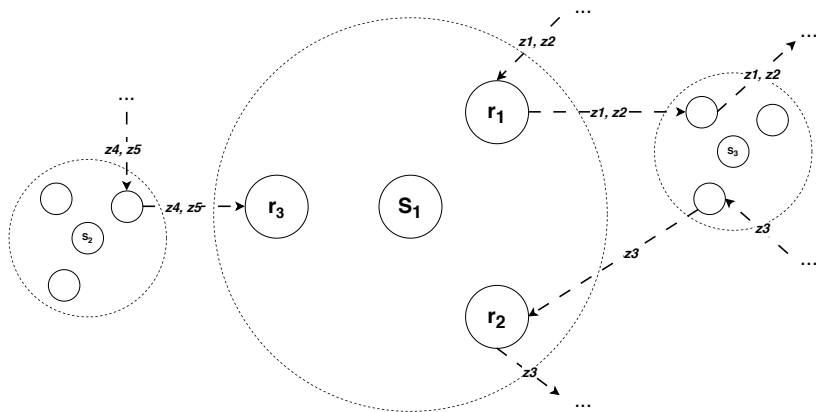
- Keine Züge z_1, z_2 dürfen S_i um t_1, t_2 ($t_1 \leq t_2$) verlassen, und z_2 vor z_1 S_{i+1} erreichen!
- In diesem Fall spalten wir in einzelne Routen:



Das Time-Dependent Modell

Konstante Umstiegszeiten

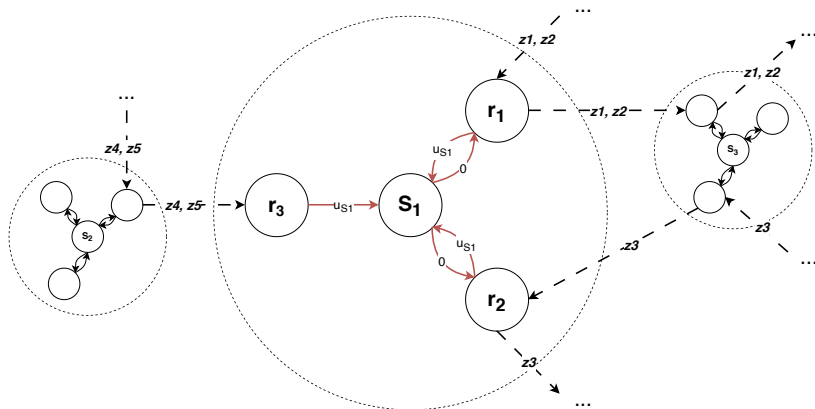
- Wir teilen eine Station in einen Stationsknoten S und Routenknoten r_i



Das Time-Dependent Modell

Konstante Umstiegszeiten

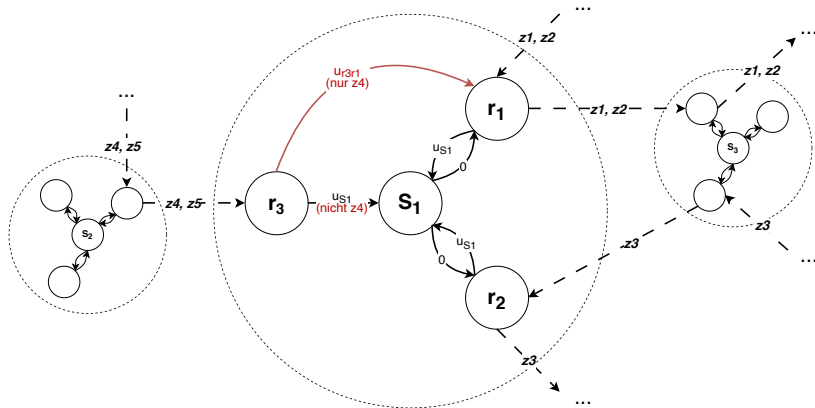
- Wir erstellen Transferkanten mit *ausgehenden* Transferkosten



Das Time-Dependent Modell

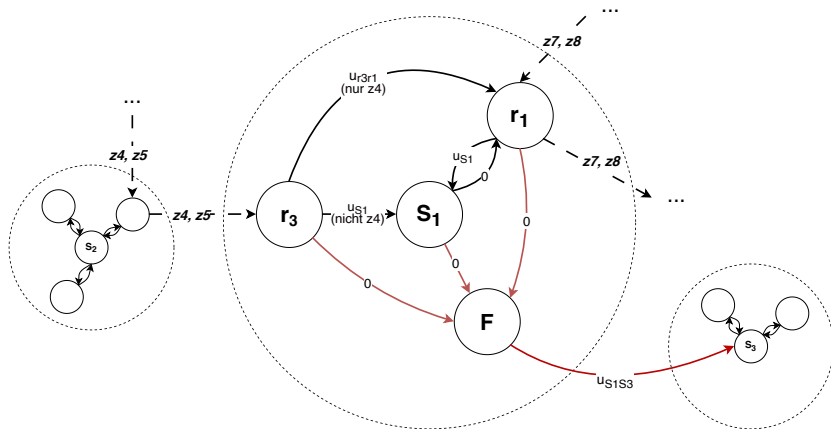
Variable Umstiegszeiten

- Wir erstellen spezielle Kanten und Regeln zwischen Knoten



Fußwege

- Fußwege sind immer abhängig von der Zeit, also einfach zu modellieren



Größenordnungen

- DB Fahrplan 1996/97 (**nur** Züge):

	Time Expanded	Time Dependent
Knoten	931746	6961
Kanten	1397619	18664

	Time Expanded	Time Dependent
Algorithmus	Dijkstra	Dijkstra + Binärsuche
∅ Laufzeit	44.17ms	5.61ms
Besuchte Knoten	33653	1515

(Pyrga et al. *Time-Expanded vs Time-Dependent Models for Timetable Information*)

Performanz

- Aber: Komplexität von TD wächst schnell mit mehr Kriterien und Regeln
- Also in realistischen Szenarien!
 - ▣ TD nur 58% schneller (in CPU-Zeit) als TE

(Pyrga et al.: *Experimental Comparison of Shortest Path Approaches for Timetable Information*, 2004)

Implementation in MOTIS

Graphenmodell

- Time-Expanded mit Verfeinerungen
 - ▣ Konstante und Variable Umstiegszeiten
 - ▣ Verkehrstage
 - ▣ Fußwege

Implementation in MOTIS

Kantengewichtungen

- Reisezeit ($t_{Ankunft} - t_{Abfahrt} \bmod 1440$)
- Anzahl der Umstiege (Ankunfts- und Spezialkanten)
- Ticketpreise → Vortrag Spezialangebote

Implementation in MOTIS

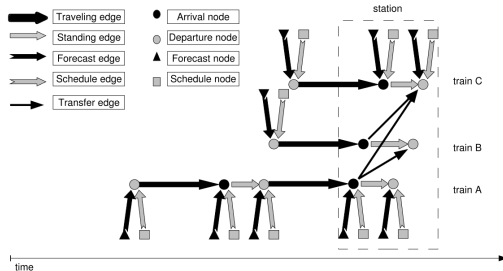
Algorithmus

- Viele Algorithmusverfeinerungen und Spezialattribute
 - ▣ Thema des nächsten Vortrags!

Das dritte Graphenmodell

Es ist das dritte Graphenmodell mit einem Klappstuhl?!?!!

- In MOTIS gibt es noch ein drittes großes Graphenmodell: Den **Abhängigkeitsgraphen**
- Abhängigkeiten zwischen Zügen und Stationen
- Wichtig für **Verspätungen!**



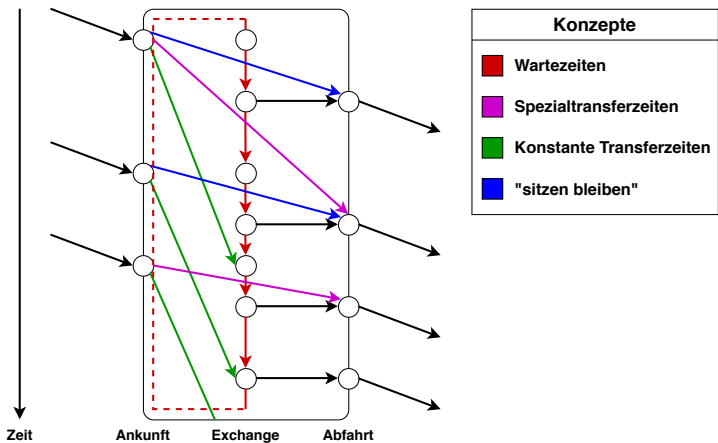
Zusammenfassung

Sachen zum Mitnehmen für die späteren Vorträge...

- Time-Expanded vs. Time-Dependent
 - ▣ Umstiegszeiten, Fußwege, Takt
 - ▣ In MOTIS: Time-Expanded
- Größenordnungen der Graphen unterscheiden sich stark
- Für realistische Anwendung: Jeweils unterschiedliche Verfeinerungen und Anpassungen notwendig

Time-Expanded

Nochmal zum Mitnehmen



Literaturquellen

- Schnee, Matthias, *Fully Realistic Multi-Criteria Timetable Information Systems*, 2009, Technische Universität Darmstadt
- Evangelia Pyrga, Frank Schulz, Dorothea Wagner, Christos Zaroliagis, *Towards Realistic Modeling of Time-Table Information through the Time-Dependent Approach*, Electronic Notes in Theoretical Computer Science, Volume 92, 2004, Pages 85-103
- Evangelia Pyrga, Frank Schulz, Dorothea Wagner, Christos Zaroliagis *Efficient models for timetable information in public transportation systems*. ACM Journal of Experimental Algorithmics, Volume 12, 2007
- Evangelia Pyrga, Frank Schulz, Dorothea Wagner, Christos Zaroliagis *Two Approaches for Time-Table Information: a Comparison of Models and Performance.*, Konstanzer Schriften in Mathematik und Informatik, Volume 190, 2003

Weitere Bildquellen nach Folie

- Verkehrsmittel (8): Wikimedia
- Portrait von Edsger W. Dijkstra (3, 4):
https://de.wikipedia.org/wiki/Edsger_W._Dijkstra (06.01.23)
- Fahrplanauszüge (3, 4, 18, 19): www.rmv.de und www.bahn.de (07.01.23)
- Abhängigkeitsgraph (55), (sowie *alle weiteren Diagramme in Eigenanfertigung nach*) Schnee, Matthias, *Fully Realistic Multi-Criteria Timetable Information Systems*, 2009, Technische Universität Darmstadt