

Slutrapportering Östgötatrafiken

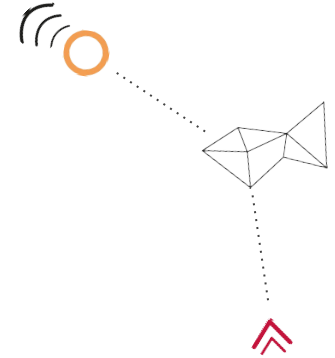
Automatisk (rutt)planering

Attentec



Agenda

- Presentation av deltagare
- Bakgrund
- Automatisk planering
- Automatisk ruttplanering
- Demo
- Resultat
- Slutsatser - Vad har vi lärt oss?
- Fortsatt arbete
- Diskussion





Rebecka Håkansson

Konsultchef

+46 73-937 06 75
rebecka.hakansson@attentec.se



Christoffer Johansson

Konsult & ML-expert

+46 73-066 34 99
christoffer.johansson@attentec.se

Bakgrund

Varför genomförde vi detta?

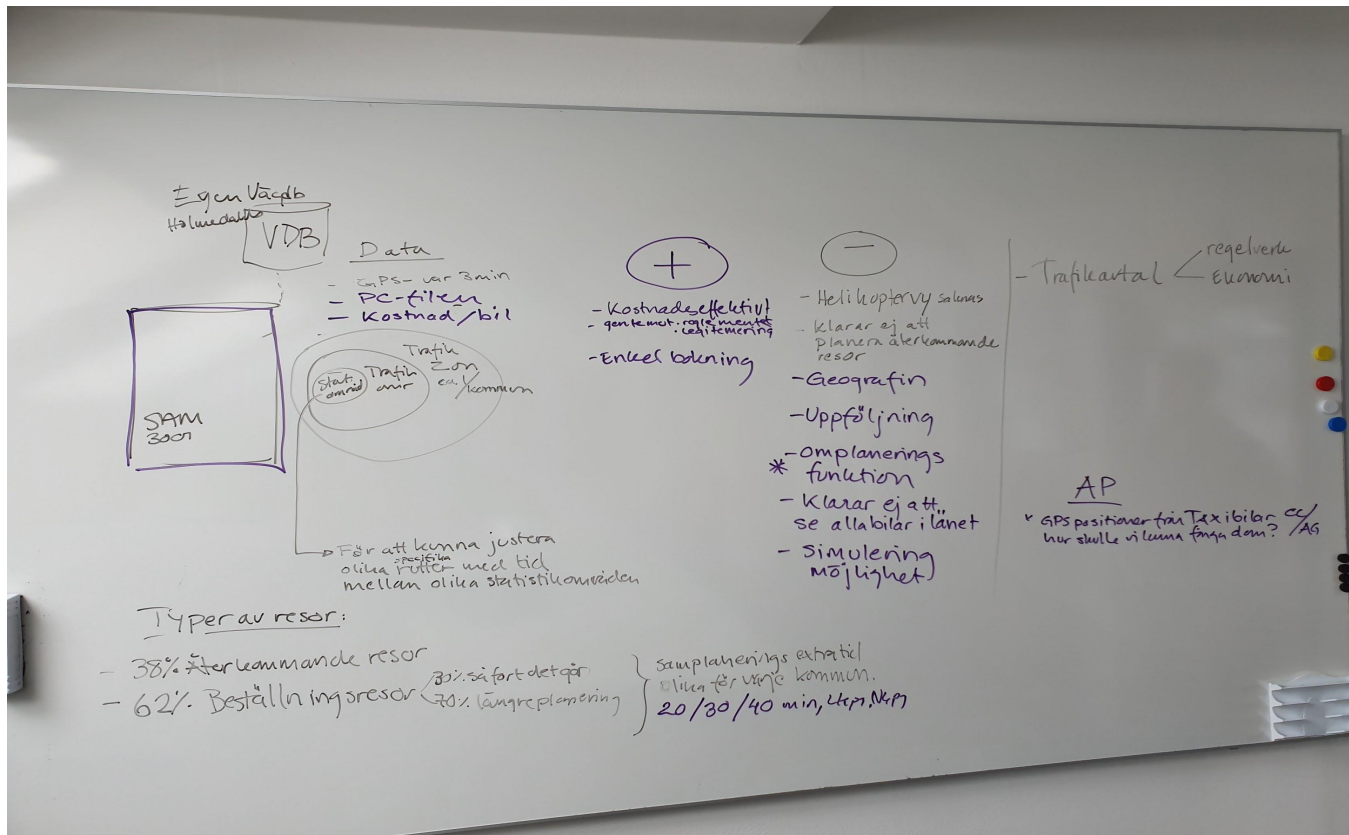
“De tekniska stöd som idag finns inom Östgötatrafiken är enligt vår bedömning, undermåliga när det gäller optimering av ruttplanering.”

Mål:

- Få kunskap om hur bra ruttplaneringen är i dagens system och vad som kan förbättras
- Öka förståelsen för lämplig arkitektur för boknings- och ruttplaneringssystem
- Kunna användas vid framtida kravställning för nästa version av ett sådant system
- Utredda om AI/Machine Learning kan användas

Initial workshop

Bakgrund



Artificiell intelligens

-Bör vi använda det?-

JA!

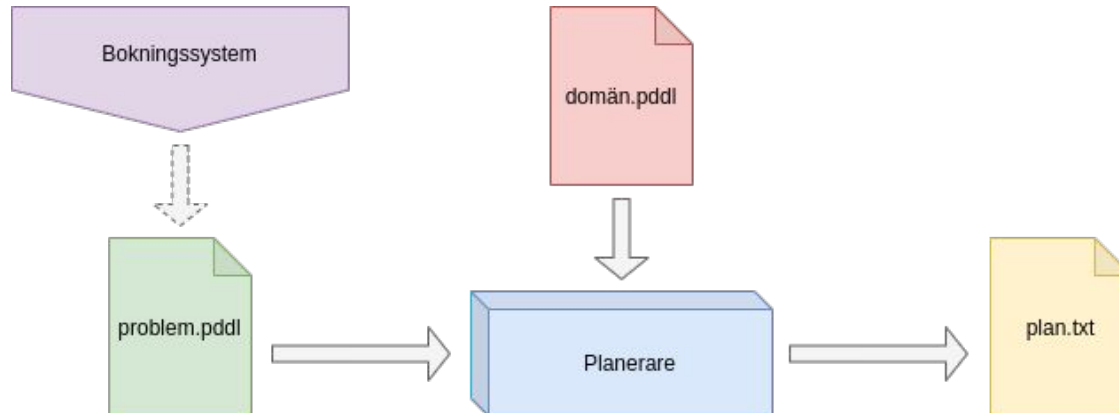
Attentec



Automatisk planering

Exempel: Gör handlingar i en ordning som uppfyller vårt mål

- Beskriv initialt tillstånd
- Beskriv önskad tillstånd (måltillstånd)



Demo: Modellering av domän och problem



Automatisk planering

Planerare

- Pågående forskningsområde
 - [IPC 2018](#)
- Olika 'taktiker'
 - Optimala planer
 - Beviskrav
 - Tillfredsställande planer
- Oberoende handlingar
 - Ställer krav på planeraren

PLANNERS

4 planners + Optic as a base line participated in the temporal satisficing track. Planner abstracts for each planner are available as a [planner abstract booklet](#) or as individual files linked below.

The source code of all entries is publicly available on bitbucket. To build the planners, follow the instructions in the [FAQ](#). Note that some planners require the LP solver CPLEX to compile. To build such a planner, acquire a CPLEX license, download the installation files to the directory `./third-party` or change that path in the Singularity file. Since building a singularity image requires root privileges, we recommend building the image inside a virtual machine. For example, you may use vagrant [Vagrant](#) with this [Vagrantfile](#). It expects a directory `./input` where the planner is located and an empty directory `./output` where the image will be created.

Some planners had minor bug fixes after the competition. We have updated the last version of each planner.

SATISFICING TRACK

- **CP4TP** ([code](#))
by Daniel Furelos-Blanco and Anders Jonsson
([planner abstract](#))
- **TFLAP** ([code](#))
by Oscar Sapena Vercher and Eva Onalinda de la Rivaherrera
([planner abstract](#))
- **TemPorAl** ([code](#))
by Isabel Cenamor, Mauro Vallati, Lukas Chrpá, Tomás de la Rosa and Fernando Fernández
([planner abstract](#))
- **PopCorm** ([code](#))
by Emre Savas and Maria Fox and Derek Long
([planner abstract](#))
- **OPTIC** ([code](#))
by Andrew Coles and Amanda Coles

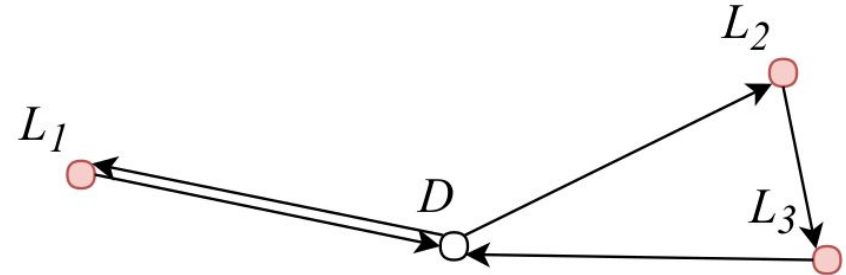
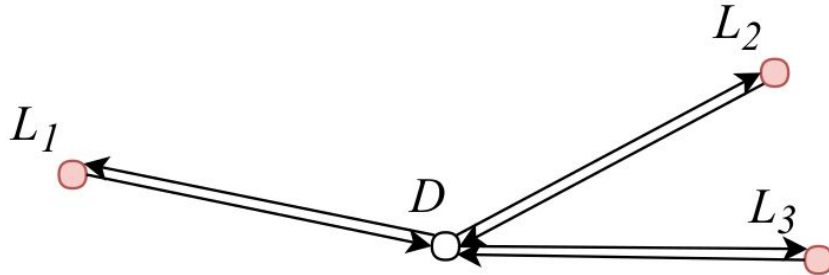
Automatisk planering

Exempel: Gör handlingar i en ordning som uppfyller vårt mål



Automatisk planering

Exempel: Gör handlingar i en ordning som uppfyller vårt mål





Automatisk ruttplanering

Vi behöver domänkunskap!

- Geografer
- Regelverk
- Trafikavtal
- Domän
- Problem

Resenär	Upphåtnings-adress	Destination	...
Resenär A	Upphåtningsgatan 1	Vårdcentral 1	...
Resenär B	Upphåtningsgatan 2	Vårdcentral 2	...
Resenär C	US Linköping Norra entrén	Destinationsgatan 3	...
...	...	...	...

RT-90 -> Longitud, Latitud

RoutingProgram.py

Generera PDDL med sträckor och tid

Körsträcka
Uppskattad körtid



```

124 ; Connect all roads and set their driving distance/estimated time values
125 (road l1 l10)
126 (= (drive-time-in-s l1 l10) 23162.5)
127 (= (drive-distance-in-km l1 l10) 339.146)
128 (road l1 l4)
129 (= (drive-time-in-s l1 l4) 282.4)
130 (= (drive-distance-in-km l1 l4) 1.839)
131 (road l1 l22)
132 (= (drive-time-in-s l1 l22) 335.6)
133 (= (drive-distance-in-km l1 l22) 3.664)
134 (road l1 l9)
135 (= (drive-time-in-s l1 l9) 463.8)
136 (= (drive-distance-in-km l1 l9) 5.37)
137 (road l1 l18)
138 (= (drive-time-in-s l1 l18) 396.4)
139 (= (drive-distance-in-km l1 l18) 4.188)

```



Automatisk ruttplanering

Regelverk

- Regler att ta hänsyn till när vi modellerar domänen
- Förskjutningstiden är önskad tid +/- 30 minuter
- Direktresetiden (restid + hanteringstid) + 30 minuter
 - Maximal samplanerad resa
- 2 minuter hanteringstid per resenär vid hämtning och lämning
 - + 3 minuter för rullstol eller permobil
 - + 2 minuter om extra hjälp finns med
 - Ingen extra hanteringstid om hjälpmedel rollator finns med
- En rullstol tar upp 1 rullstolsplats i SPEC-buss
- En permobil tar upp 2 rullstolsplatser i SPEC-buss
- Rollator har ingen påverkan på utrymmet i fordonet
- En resenär per sittplats



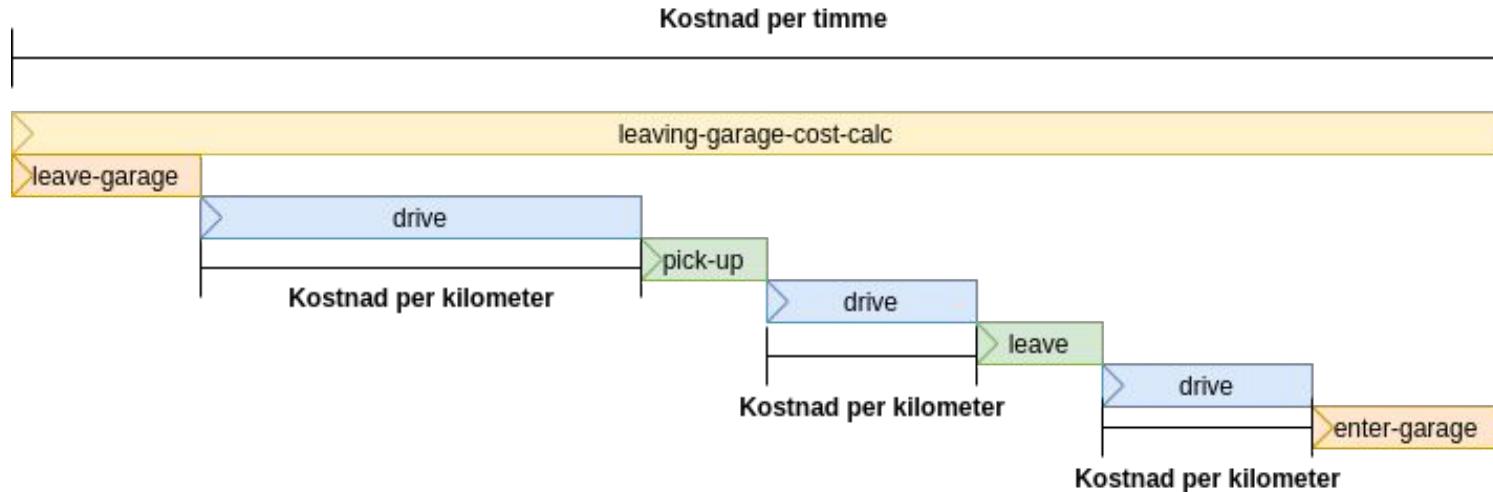
Automatisk ruttplanering

Trafikavtal

- Tre avtal med olika priser och garage
- Avtal 1:
 - 34 SPEC-bussar (5 sittande + 2 rullstolsplatser)
 - 222 kr/h
 - 4.19 kr/km
 - Garage: Östra malmskogen i Linköping
- Avtal 2:
 - 34 SPEC-bussar (5 sittande + 2 rullstolsplatser)
 - 250 kr/h
 - 4.00 kr/km
 - Garage: Roxengatan i Linköping
- Avtal 3:
 - 20 taxibilar (4 sittplatser)
 - 250 kr/h
 - 3.50 kr/km
 - Garage: Roxengatan i Linköping

Automatisk ruttplanering

Modellera en domän som optimerar på sträcka och tid



Demo: Problem A

Resultat

Sammanfattning

```
; Plan found with metric 697.012
; Theoretical reachable cost 697.012
; States evaluated so far: 124
; States pruned based on pre-heuristic cost lower bound: 0
; Time 0.44
43019.991: (leaving-garage-cost-calc v1) [3600.012]
43019.991: (leave-garage v1 g1) [0.010]
43020.001: (drive v1 g1 l1) [1080.000]
44100.001: (pick-up-passenger v1 l1 p2) [120.000]
44220.001: (drive v1 l1 l2) [1140.000]
44279.991: (leaving-garage-cost-calc v10) [3240.012]
44279.991: (leave-garage v10 g1) [0.010]
44280.001: (drive v10 g1 l3) [720.000]
45000.001: (pick-up-passenger v10 l3 p1) [120.000]
45120.001: (drive v10 l3 l4) [1140.000]
45360.001: (leave-passenger v1 l2 p2) [120.000]
45480.001: (drive v1 l2 g1) [1140.000]
46260.001: (leave-passenger v10 l4 p1) [120.000]
46380.001: (drive v10 l4 g1) [1140.000]
46620.001: (enter-garage v1 g1) [0.010]
47520.001: (enter-garage v10 g1) [0.010]
```

Resultat

Sammanfattning

- Vi hittar en plan för problem A på ~ 0.44 sekunder!
- Problem B utökar antalet fordon och passagerare
 - Närmar oss hur det ser ut i verkligheten
 - Hittar inte tillfredsställande planer inom en rimlig tid
 - Endast testat med OPTIC

Slutsatser

Vad har vi lärt oss?

- Vi kan modellera en domän på ett meningsfullt sätt!
 - Det finns andra/nya planerare att studera
- Extremt viktigt att försöka hålla nere komplexiteten
- Exempelvis genom att:
 - Minska antalet vägförbindelser på ett smart sätt
 - Behåll de k närmaste grannarna, garagen och destinationen
- Vi separerar bokningar och ruttplanering på ett föredömligt sätt
- Vi tar hänsyn till snabba förändringar
- Samplanering är otroligt viktigt för kostnadsbesparingar



Future Work

Intressanta trådar att jobba vidare med

- Utvärdera hur andra planerare hanterar tidsfönster
- Applicera tekniker som minskar komplexiteten
- Bygga stöd för att skapa problem/tolka lösningar (planer)
 - Räkna ut mätvärden (t.ex. % samplanering) automatiskt

Frågor



Tack för oss!