Algoritmen en Heuristieken

Wouter Vrielink

Wat zijn heuristieken?

Locale strategiën

Shortcuts

Indicatie van goede oplossingen

Geen garanties

Optimaal
Perfect
Rationeel

Heuristiek in de praktijk

Probleem:

- Reis van Limburg naar Groningen
- Geen bordjes met Groningen

Oplossingsvorm:

- Gebruik weg X om bij weg Y te komen
- Gebruik weg Y om bij weg Z te komen

Heuristiek in de praktijk

Probleem:

- Reis van Limburg naar Groningen
- Geen bordjes met Groningen

Oplossingsvorm:

- Gebruik weg X om bij weg Y te komen
- Gebruik weg Y om bij weg Z te komen

Heuristiek:

- Neem de weg die het meest naar het noorden gaat

Gebruik heuristieken om sneller te zoeken!

Wat is Algoritmen en Heuristieken?

- Project van 4 weken
- Groepen van 3
- Dagelijks van 9-17 werken aan
- 6 cases; “onoplosbare” problemen

- Assistentie
  - Mentoren
  - Dagelijks 10-14

6 Cases

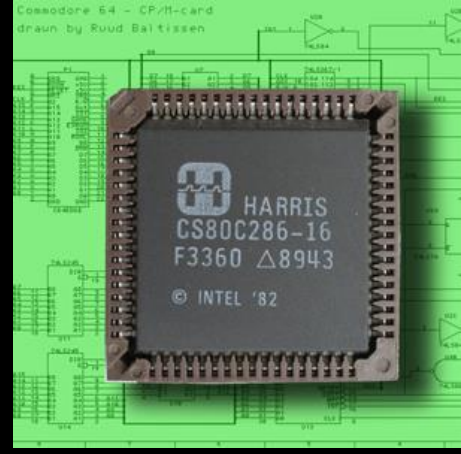
- Wat maakt een probleem ‘onoplosbaar’?
- Datastructuur/Representatie
- Oplossingen/deel-oplossingen
- De sleutel! Algoritmes

- Lectures & Lesroosters
- Chips & Circuits
- Protein Pow(d)er

- RailNL
- Rush Hour
- SmartGrid

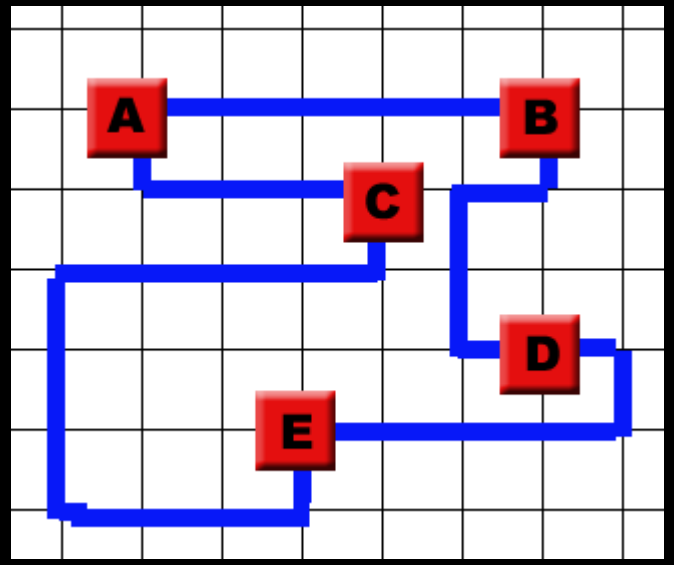
6 Cases

1 – Chips & Circuits



Netlist #1

A - B
A - C
C - E
D - B
D - E

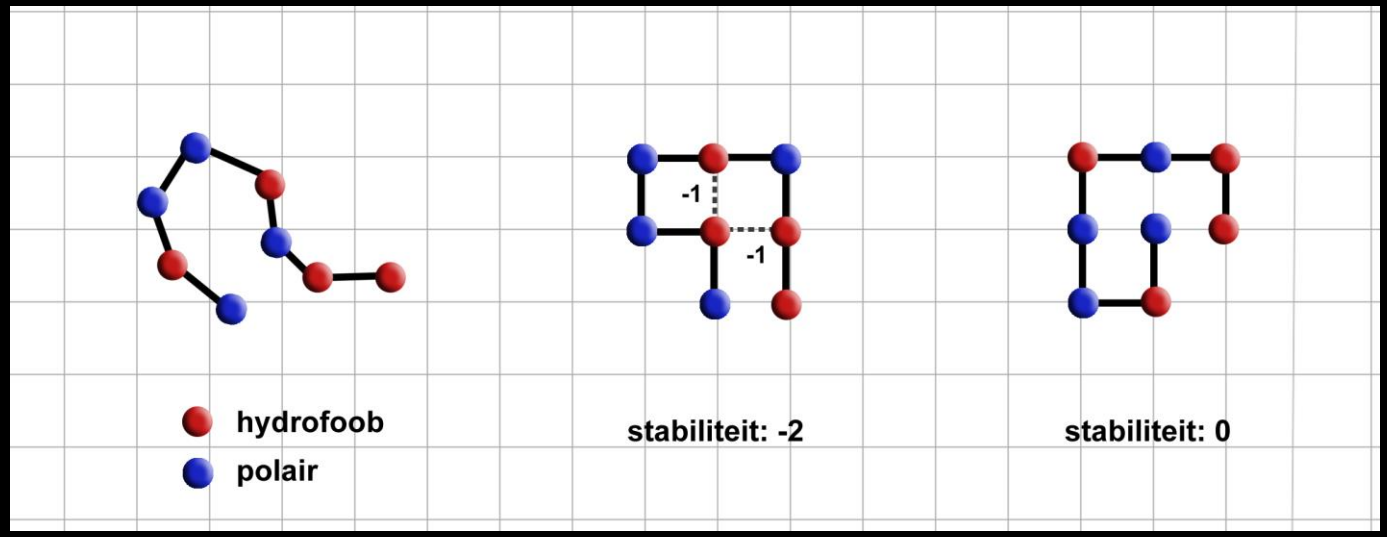


- Twee chips, met elk drie netlists
- Gates (logische poorten) verspreid over de chips
- Verbind de gates mbv de netlists
- Kortsluiting voorkomen!
- Hoe korter de kabels, des te beter de oplossing

6 Cases

2 – Protein Pow(d)er

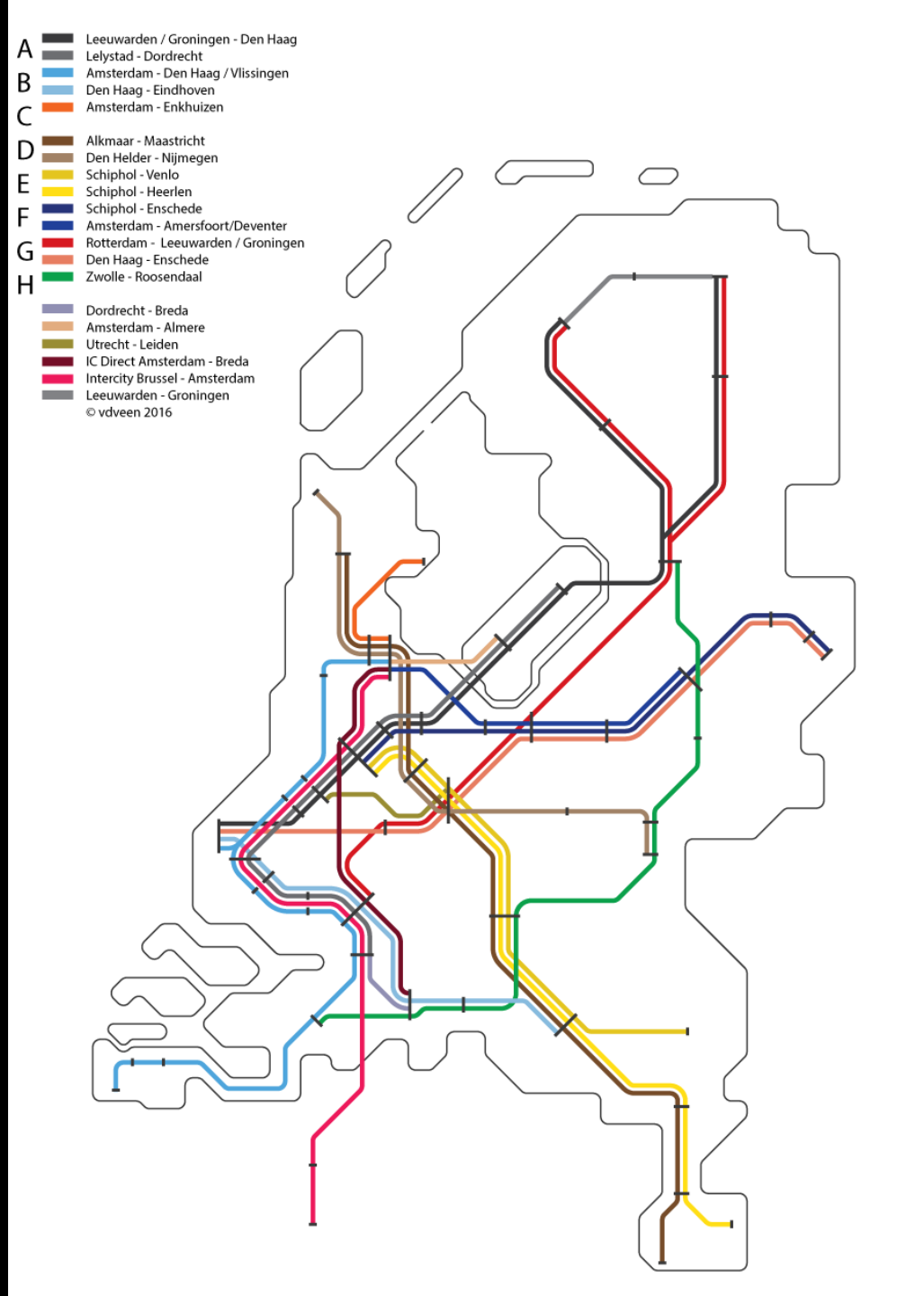
- Eiwitten vouwen: HHPHHHHPHPHHHPH
- 2D HP-Model
- Hydrofobe en Polaire aminozuren
- Stabiele vouwing wanneer hydrofobe aminozuren naast elkaar liggen



6 Cases

3 – RailNL

- Treinvoering in Holland en Nederland
- Plan trajecten in voor treinen
- Zo veel mogelijk stations passeren, in zo min mogelijk minuten
- Houdt rekening met knelpunten zoals Utrecht en doodlopende sporen zoals Den Helder.



6 Cases

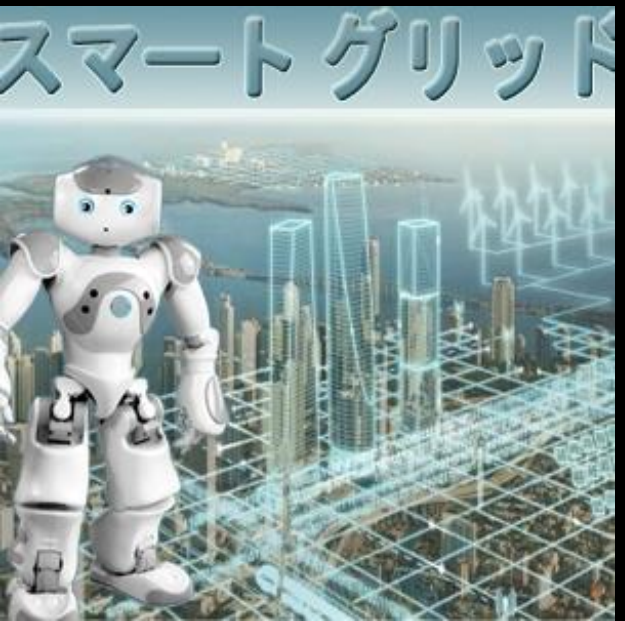
4 – Rush Hour

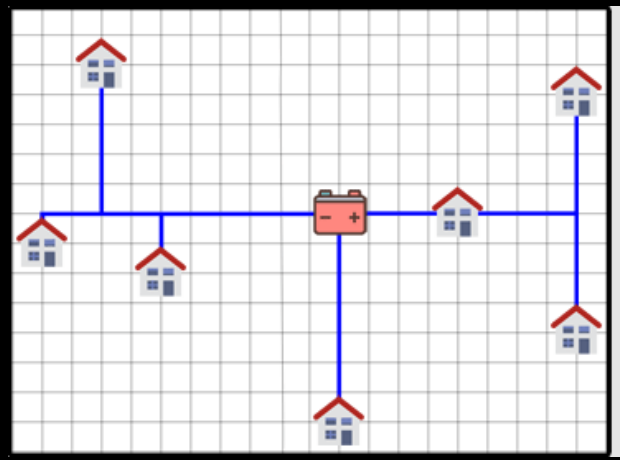
- Klassiek puzzelspel
- Rij de rode auto naar buiten
- 7 verschillende spelboorden
- Voorkom botsingen

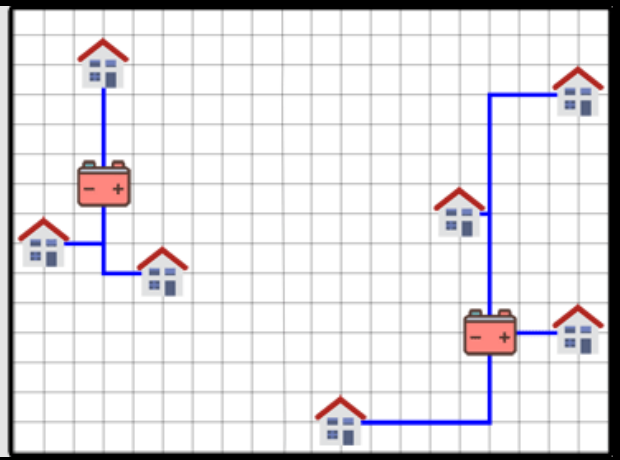
6 Cases

5 – SmartGrid

- Drie wijken; huizen en batterijen op vaste plaats
- Huizen wekken energie op
- Energie opslaan in batterijen
- Alle huizen moeten op een batterij aansluiten
- Zo min mogelijk kabel gebruiken







6 Cases

6 – Lectures & lesroosters

- Vakken, zalen, activiteiten, studenten
- Geldig rooster maken
- Maluspunten

Contacturen

- Dagelijks van 9-17

- Hoorcolleges
- Algemene techassistentie

- Met de assistent
  - Persoonlijke techassistentie
  - Presentatiesessie

Wat gebeurt er tijdens meetings?

- Éénmaal per week ~30 min
- Hulp bij (nadenken over) programmeren en je case

Maar ook

- Bugs bespreken
- Uitleg
- Afspraken maken
- Discussie over o.a. infra-/data-structuren/algoritmen
- Voortgang/milestones bespreken

Milestones

- Case kiezen (vandaag)
- State space (woensdag)
- Representatie (vrijdag)
- Baseline (volgende week donderdag
- Plan individueel onderdeel (max. dag na baseline)
- Inleveren individueel onderdeel (derde week vrijdag)
- Experiment

Pass / Fail -> **opnieuw**

Individueel onderdeel

- Idee samen, code zelf

Technische beschrijving pt. 1
- Wat zijn de functionaliteiten
- Wat moet hiervoor gemaakt worden?
- Hoe gaat dat aansluiten?

Technische beschrijving pt. 2
- Wat is anders uitgepakt?
- Links naar commits op Git

Eindcijfer

- Kwaliteit van de code en documentatie
- Eindpresentatie
- Individuele bijdrage + technische beschrijving

Voor een voldoende:

- Alle milestones behaald
- Voldoende individuele bijdrage
- Resultaten meerdere experimenten presenteren

GitHub

“GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.”

GitHub != Dropbox

Beoordeling code

Minimaal aanwezig

- README (+ inhoud)
- Duidelijke en inzichtelijke repo structuur
- Geen crashes
- requirements.txt of equivalent
- Presentatie

Git README.md

- Inleiding
- Installatie -> requirements.txt
- Waar kan ik wat vinden? Zet alles netjes in mappen.
- Tests / hoe run ik dit programma
- Auteurs

Zie voor een voorbeeld:

- https://github.com/minprog/voorbeeld-repo
- https://github.com/minprog/radio_russia_demo

Beoordeling code

Reproducibility
- README
- Genereren van oplossingen/metrieken
- Methode van presenteren oplossingen/metrieken

Code
- Modulariteit
- Abstractie
- Duplicate/redundant code
- Style
- Documentatie
- etc

Week		Onderdeel
1	Interpretatie van de case	Maandag 11 uur: Openingscollege
		Dinsdag 15 uur: College over problemen
		Donderdag 15 uur: Live coding college
2	Een baseline zetten	Maandag 15 uur: College over zoekalgoritmes
		Dinsdag 15 uur: College over optimalisatie-algoritmes
		Donderdag 15 uur: Live coding college
3	Algoritmes & resultaten	Maandag 15 uur: College over experimenteren
		Woensdag: Oefenpresentaties
4	Vergelijken & presenteren	Donderdag & Vrijdag: Eindpresentaties

Door de weken heen

Week 1 - Interpretatie van de case
- Wat is mijn case?
- Welk (sub)probleem moet ik oplossen? Wat is een oplossing en hoe ziet die er uit?
- Hoe kan ik die (digitaal) representeren? -> datastructuur
- Hoe genereer ik random een oplossing?

Week 2 - Een baseline zetten
- Random is af. Kan resultaten laten zien.
- Hoe genereer ik betere oplossingen dan random?
- Waarom zou dit beter zijn?

Week 3 - Algoritmes & resultaten
- Maken nieuwe algoritmes

Week 4 – Vergelijken & presenteren

Zometeen