



HUMAN MACHINE TEACHERS

JESSE, JOANNE, ERIC,
MARTTI, SEFA & AYRTON

VOORTGANG CONTAINER PROJECT



1. **Onderzoeksopzet**
2. **Simulatie Environments**
3. **Restricties**
4. **Reinforcement Learning**
5. **Resultaten**
6. **Plannen voor volgende 4 weken**

Onderzoeksofzet

Hoofdvraag

Met welke methode(s) kunnen we het uitladingsdeel van het container stacking probleem optimaal oplossen?

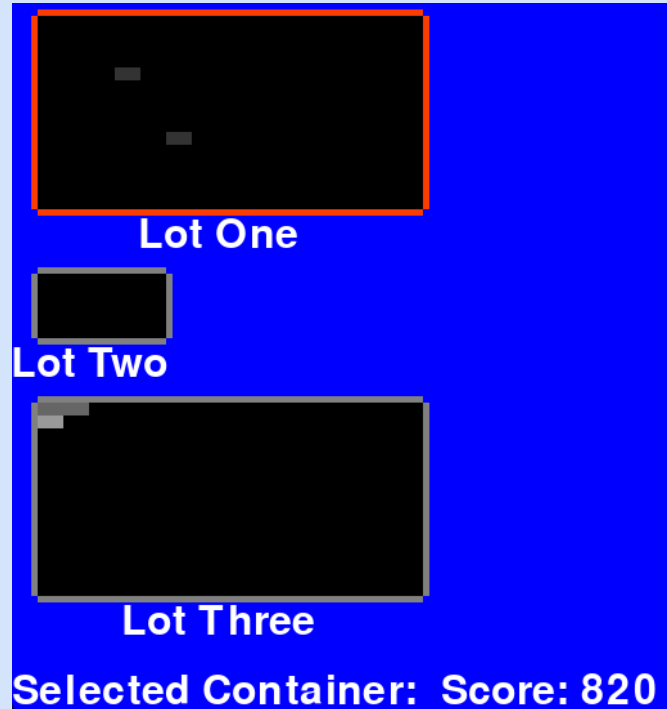
Deelvragen

1. Welke methoden (heuristieken) zijn mogelijk bij het container stacking probleem?
2. Wat is een move en wat zijn de restricties?
3. Welke containers zijn er en welke gaan we gebruiken?
4. Hoe is de haven ingericht en wat zijn de restricties?
5. Hoe kunnen we de container data simuleren?



Simulation Environments: Eerste prototype

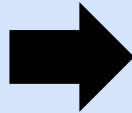
- Environment
- Reachstacker
- PyGame (GUI)
- Niet in Jupyter Notebook
- Meerdere lots
- Complex



Simulatie Environments: Tweede Prototype

Bij onze methode wordt onder andere de environment als input gebruikt. Deze wordt omgezet naar een array die rekening houdt met het 3-dimensionale probleem

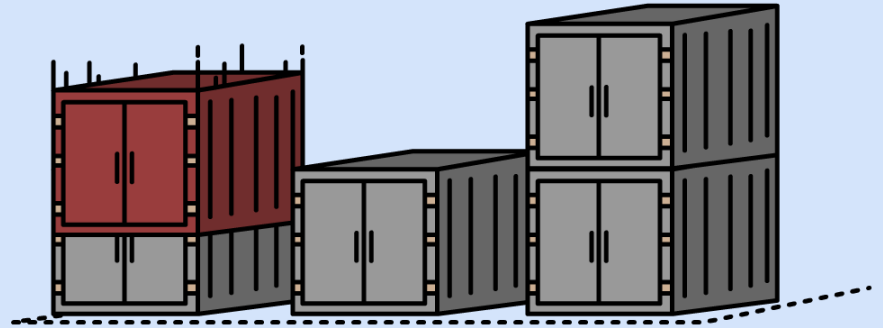
H	0	L	0
0	0	H	0
0	L	0	0
H	0	0	L



```
speelveld
array([[[['H'],
         ['0'],
         ['0'],
         ['H']],
        [['0'],
         ['0'],
         ['L'],
         ['0']],
        [['L'],
         ['H'],
         ['0'],
         ['0']],
        [['0'],
         ['0'],
         ['0'],
         ['L']]]], dtype=object)
```

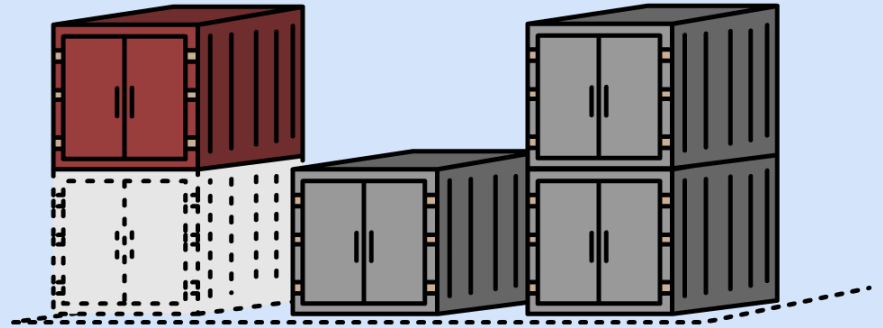
Restricties

Restrictie 1: Container niet plaatsen waar al een container staat



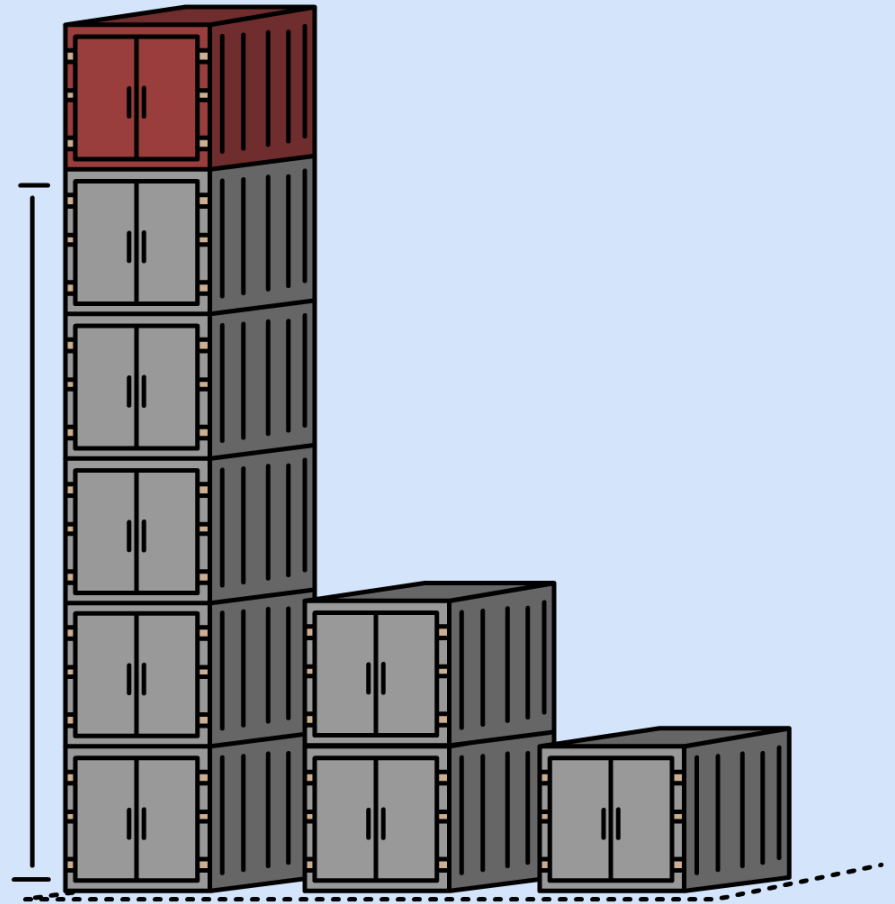
Restricties

Restrictie 2: Container niet plaatsen waar geen container onder zit



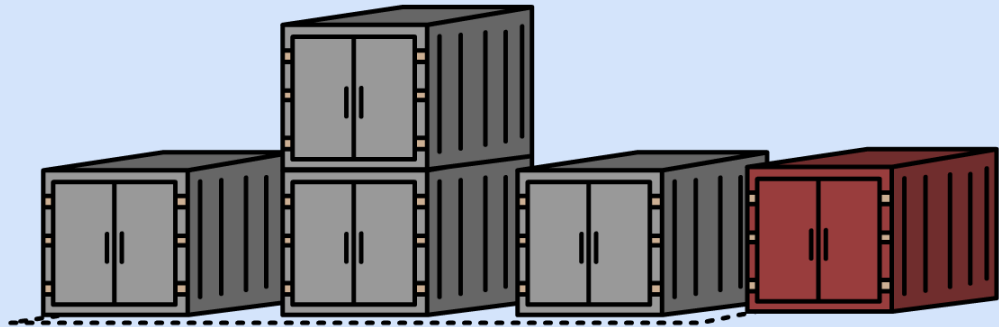
Restricties

Restrictie 3: Container niet plaatsen op een stapel die de max hoogte heeft bereikt



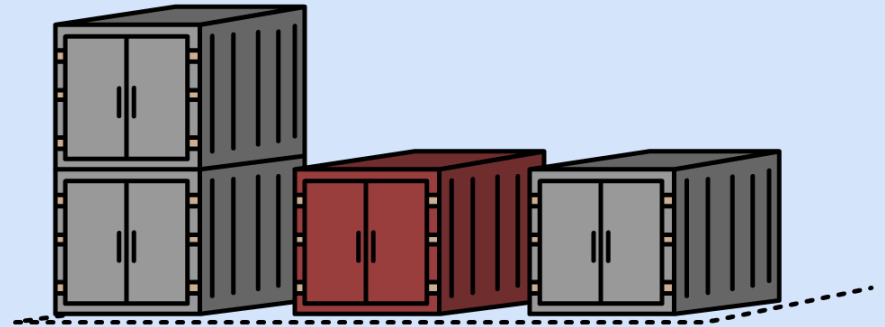
Restricties

Restrictie 4: Container niet plaatsen buiten het lot



Restricties

Restrictie 5: Container niet plaatsen tussen twee stapels
(reachstacker kan namelijk alleen aan lange kant pakken)



Reinforcement Learning: Agent

Reinforcement learning: Gedrag van model belonen of straffen

Agent neemt lotindeling en huidige container als input en geeft een action als output

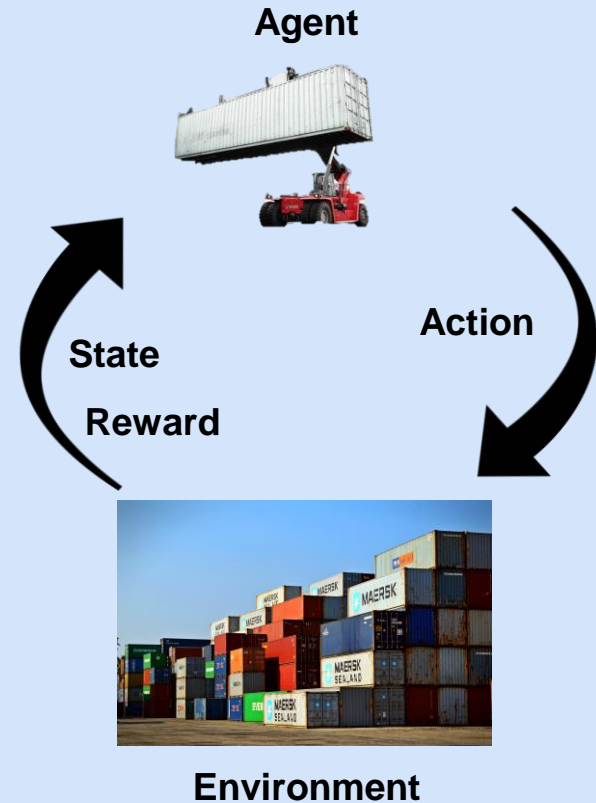
- **Action:** Het verplaatsen van de container op een plek in het lot

Environment geeft state en reward terug

- **State:** De containerverdeling over het lot na de vorige gekozen action
- **Reward:** score die is ontvangen voor de gekozen action

Rewards/punishments:

- Container plaatsen op een plek waar die kan staan -> score + 3
- Container plaatsen waar dat niet kan -> score - 5



Reinforcement Learning: Neuraal Netwerk

Agent stopt prioriteit van eerstvolgende uit te laden container en de huidige staat van het lot in het neurale netwerk.

Input:

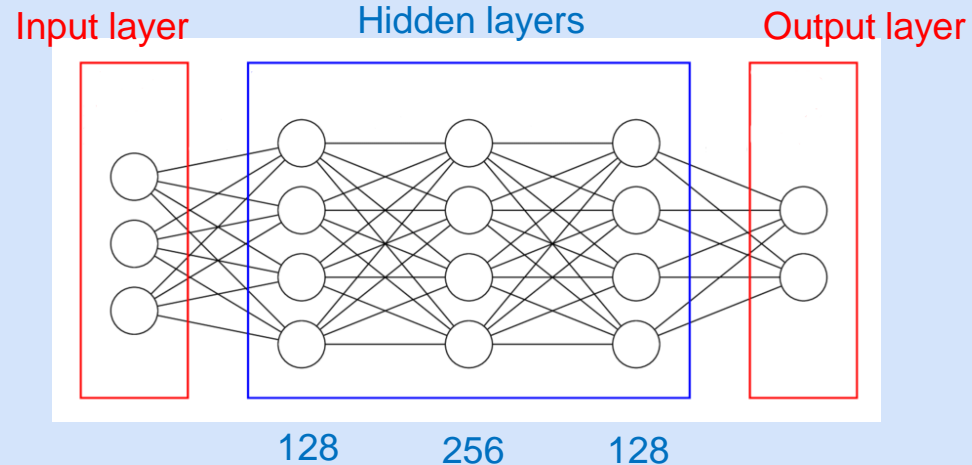
- lotsindeling array
- Prioriteitsklasse ('L' of 'H')

Optimizer:

- Adam

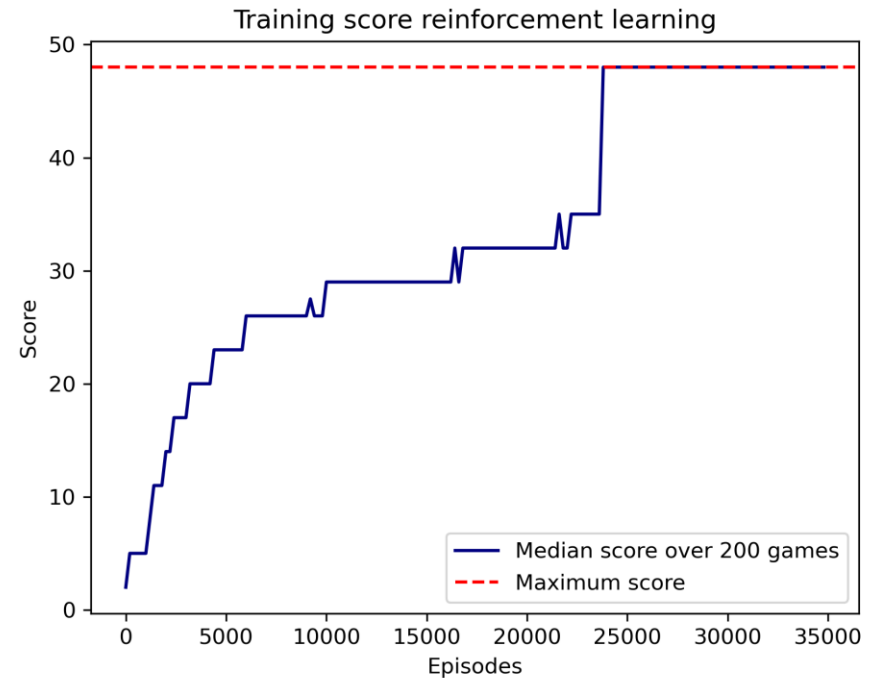
Output:

- 'L' of 'H' in array zetten (action)



Resultaten

- Model traint over 100.000 episodes
- Vindt na 25.000 episodes consistent de max score
- Leert na eerste 5000 episodes score boven 25 te behalen



Plannen voor volgende 4 weken

- Bestaand model verbeteren
- Gebruik van convolutional neural networks
- Scoren op verdeling van de containers
- Scoren op de route die reachstacker rijdt
- Environment vergroten (stapelhoogte meenemen)

Vragen?