

1. Introductie

Introductie

In dit hoofdstuk zal een korte introductie gegeven worden over het containerbedrijf Cofano en het probleem wat behandeld wordt in dit onderzoek. Vervolgens wordt het onderzoeksdoel omschreven en ten slotte worden er andere werken, die gerelateerd zijn aan dit onderzoek, besproken in het literatuuronderzoek. Na de introductie kan er gelezen worden over de opzet van dit onderzoek, hoe de literatuur is gekoppeld aan dit probleem, de dataverzameling en de methodologie. Daarna volgt het hoofdstuk met de onderzoeksresultaten, waarin de belangrijkste resultaten getoond zullen worden. Ten slotte kunt u de conclusie en discussie lezen, waarin de resultaten teruggekoppeld worden aan de doelstelling van dit onderzoek en mogelijke verbeteringen en/of punten waar tegenaan is gelopen gedurende dit onderzoek worden benoemd.

Probleemstelling

Cofano is een bedrijf dat zich bezighoudt met het optimaliseren van terminalprocessen. Deze processen houden het volgende in: er komen binnenvaartschepen aan, die containers af moeten lossen op de kade van een terminal. De containers staan daar vervolgens totdat een zeevaartschip de containers komt halen. In de volgorde waarop schepen de terminal in- en uitvaren zit een onzekerheid, met deze onzekerheid moet rekening gehouden worden. Cofano wil de tijd dat schepen aan de kade liggen minimaliseren om de kosten zo laag mogelijk te houden. Dit betekent dat de in- en aflaadprocessen geoptimaliseerd moeten worden.

Op de kade rijden stackers rond die containers kunnen verplaatsen. Hoe minder stappen een stacker nodig heeft om een container te bereiken, hoe sneller het proces verloopt. Het doel van Cofano is om deze verschillende processen te optimaliseren om zo de kosten zo laag mogelijk te houden.

Er wordt in de terminals gewerkt met reach stackers. In de rest van dit paper zal verwezen worden naar de reach stackers als stackers. Deze stackers kunnen alleen containers pakken vanaf de lange zijde en vanaf verschillende hoogtes. Als eerst moeten bovenste containers verplaatst worden, als een onderste container gepakt moet worden. Als een container tussen andere containers ligt, kan de stacker hier niet bij. Het is mogelijk maximaal vijf containers op elkaar te stapelen. Een visualisatie met een voorbeeld van een onbereikbare container kan gevonden worden in de bijlage, ([zie Bijlage, figuur 3](#)).

Het doel van dit onderzoek

In dit onderzoek is een methodologie ontwikkeld voor het automatiseren van het vinden van een optimale indeling van de yard in een terminal, zodat een stacker containers ophaalt voor verder vervoer in een minimaal aantal stappen. Dit is een complex probleem. Er zijn diverse soorten containers, en die zijn van diverse grootte en gewicht. Daarnaast zijn er vele containerterminals in de wereld, en die verschillen qua grootte, aantal containers en soorten containers. Dit maakt de zoektocht naar een algemene oplossing lastig.

Omdat het een dynamisch probleem is, is het van belang dat het model schaalbaar is. Dat wil zeggen dat het makkelijk uit te breiden is naar bijvoorbeeld een grotere yard of een situatie met meer containers en zeevaartschepen. Wel zijn er een aantal aannames gedaan waar het model mee werkt. Het model houdt geen rekening met verschillende soorten containers en gaat uit van één yard waar containers geplaatst kunnen worden, zonder rekening te houden met de verdere indeling van een terminal.

Literatuuronderzoek

Het Container Stacking Problem (CSP) is een bekend probleem, waar al meerdere onderzoeken naar zijn gedaan. In deze onderzoeken wordt er gekeken naar het optimaliseren van bepaalde terminalprocessen. Het probleem is op te delen in losse problemen, zoals bijvoorbeeld het uitladen en plaatsen van containers, het inzetten van stackers en het vervoeren van containers van de terminal naar de zeevaartschepen. Er zijn verschillende heuristieken gebruikt om zulke problemen op te lossen.

Kefi et al. (2007) vergelijkt een Uninformed en Informed Search Algorithm voor het toewijzen van een slot aan een container. Het model minimaliseert het aantal bewegingen en verplaatsingen die nodig zijn om een container te pakken. Het Informed Search Algorithm geeft de meest optimale oplossing voor het probleem. Salido et al. (2009) heeft Artificial Intelligence (AI) toegepast op het CSP. Net als in het onderzoek wat in dit paper beschreven wordt, richt dit model zich op een optimale plaatsing van containers voor het ophaalproces. Het doel is hetzelfde als in dit onderzoek, namelijk het minimaliseren van het aantal verplaatsingen dat nodig is om een container te pakken.

In een ander onderzoek (Ries et al., 2014) wordt een Fuzzy Logic Model gebruikt om binnenkomende containers toe te wijzen aan een plek in de terminal. In dit onderzoek ligt de focus op het bouwen van een flexibel model, dat rekening houdt met onzekerheden van aankomsttijden van containers bij de terminals en wat bruikbaar is voor terminals met verschillende indelingen en infrastructuren. Het is voor dit onderzoek belangrijk te werken aan een flexibel model voor het vinden van een optimale indeling, omdat in dit onderzoek de onzekerheid van aankomst van containers een rol speelt.

Daarnaast is op dit gebied ook al gewerkt met Reinforcement Learning (RL) om optimale oplossingen te vinden voor terminalprocessen. In het onderzoek van Jiang et al. (2021) ligt de focus op het optimaal stapelen van containers op basis van prioriteit van een container. Het doel is om het aantal relocaties dat moet plaatsvinden voordat alle containers op een bepaalde volgorde gepakt kunnen worden, te minimaliseren. Dit probleem wordt aangepakt met RL en de resultaten zijn net zo goed als de meest optimale oplossingen die met andere methodes gevonden zijn.

In het onderzoek van Hu et al. (2021) wordt met twee verschillende methodes, Integer Programming (IP) en RL, het vervoeren van containers naar het zeevaartschip geoptimaliseerd. Het doel is het vinden van een optimale operatievolgorde en optimale routes voor de stackers, die rijden tussen de containers en het zeevaartschip. Het RL-model scoort daarbij het beste. De resultaten van Jiang et al. (2021) en Hu et al. (2021) laten zien dat RL een geschikte methode is om soortgelijke problemen aan te pakken.

Hu et al. (2023) bekijkt het optimaliseren van het planningsproces in containerterminals. Hier wordt een systeem environment gebouwd en door middel van RL gezocht naar een optimale planning van operaties in de terminal. Hier is ondervonden dat RL efficiënt werkt en flexibeler is dan andere heuristieken. Dit is erg belangrijk voor het CSP, aangezien daar veel onzekerheden bij komen kijken en het een dynamisch probleem is.

In het onderzoek van Krishna & Sudhir (2020) worden meerdere experimenten gedaan waarbij RL-modellen worden vergeleken. Dit onderzoek kan toegepast worden voor het zo efficiënt mogelijk indelen van de containers op de kade. Uit onderzoek naar de modellen bleek dat het A2C-model voor een deel gebaseerd is op het PPO-model. Echter zijn de resultaten van het PPO-model voor de experimenten, die besproken zijn in dit onderzoek, beter.

In een video (Renotte, 2021) over RL wordt aandacht besteed aan het zelf bouwen van een environment en RL-model. Stap voor stap wordt een douche environment opgebouwd en een model die de temperatuur gedurende een uur optimaliseert. Met behulp van deze video kan er een vergelijkbaar RL-model gebouwd worden die, bijvoorbeeld voor dit onderzoek, containers op een zo optimaal mogelijke manier plaatst.

Literatuurlijst

- Hu, H., Yang, X., Xiao, S., & Wang, F. (2023). Anti-conflict AGV Path Planning in Automated Container Terminals Based on Multi-agent Reinforcement Learning. *International Journal of Production Research*, 61(1), 65-80. doi: 10.1080/00207543.2021.1998695
- Hu, X., Yang, Z., Zeng, Q. (2012) A Method Integrating Simulation and Reinforcement Learning for Operation Scheduling in Container Terminals. *Transport*, 26(4), 383-393. doi: 10.3846/16484142.2011.638022
- Jiang, T., Zeng, B., Wang, Y., & Yan, W. (2021) A New Heuristic Reinforcement Learning for Container Relocation Problem. *Journal of Physics: Conference Series*, 1873(1). 012050. doi: 10.1088/1742-6596/1873/1/012050
- Kefi, M., Korbaa, O., Ghedira, K., & Yim, P. (2007). Heuristic-based model for container stacking problem. In *19th International Conference on Production Research-ICPR* (Vol. 7).
- Krishna, V., Sudhir, Y. (2020). *Comparison of Reinforcement Learning Algorithms* [Powerpoint-slides]. Departure of Computere Science and Engeneering, University at Buffalo. Geraadpleegd op 28 november 2022, van https://cse.buffalo.edu/~avereshc/rl_fall20/
- Medium. (2021, 7 december). *Understanding PPO plots in TenserBoard*. Geraadpleegd op 21 januari 2023, van <https://medium.com/aureliantactics/understanding-ppo-plots-in-tensorboard-cbc3199b9ba2>
- Renotte, N. (2021, 6 juni). *Reinforcement Learning in 3 hours | Full Course Using Python* [Video]. YouTube. Geraadpleegd op 21 november 2022, van https://www.youtube.com/watch?v=Mut_u40Sqz4&t
- Ries, J., González-Ramírez, R. G., Miranda, P.(2014). A Fuzzy Logic Model fort he Container Stacking Problem at Container Terminals. *International Conference on Computational Logistics*, 93-111. doi: 10.1007/978-3-319-11421-7_7
- Salido, M. A., Sapena, O., & Barber, F. (2009). An artificial intelligence planning tool for the container stacking problem. *2009 IEEE Conference on Emerging Technologies & Factory Automation*, 1-4. doi: 10.1109/ETFA.2009.5347007.
- Trivedi, C. (2021, 11 december). *Proximal Policy Optimization Tutorial (Part 2/2: GAE and PPO loss)*. Medium. Geraadpleegd op 21 januari 2023, van <https://towardsdatascience.com/proximal-policy-optimization-tutorial-part-2-2-gae-and-ppo-loss-fe1b3c5549e8>