Integrating operational and tactical decision-making in spare part inventory management

Using Deep Reinforcement Learning to determine (tactical) base stock levels for repairable spare parts while considering day-to-day (operational) flexibility enabled by interventions.

UNIVERSITY OF TWENTE.

1. RESEARCH GOAL

On a tactical level, base stock levels are determined based on aggregated product and demand data. Interventions (e.g. expediting repair jobs, lateral transshipments) are performed on a day-to-day basis to prevent and solve short term inventory issues. Currently, this operational flexibility enabled by interventions is not considered when determining base stock levels. This lead us to the following research question:

How can the spare part inventory planning on a tactical level and on an operational level be integrated?

2. LITERATURE

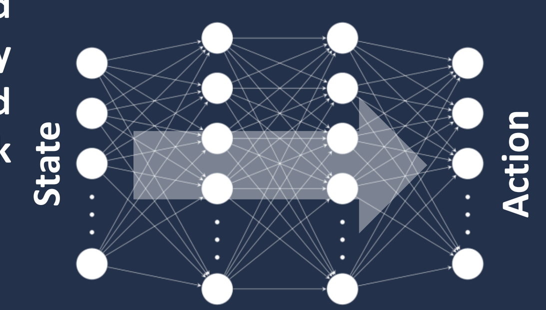
We found the following from currently available literature on the subject of integration of the operational and tactical decision-making:

- Often, only one (at most two) intervention is considered.
- Most focus is put on lateral transshipments.
- Methods are difficult to generalize.
- Only one product type is considered.
- Only one paper found on repairable spare parts.

These observations together with the four directions for future research on integrated supply chain planning (1. include uncertainty, 2. bigger problems, 3. complex systems, 4. long-term impact of daily actions) bring us to Deep Reinforcement Learning as our solution approach.

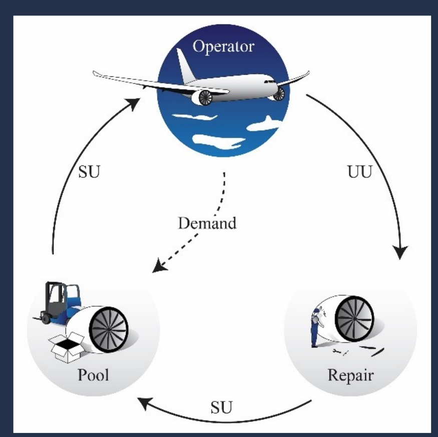
3. DEEP REINFORCEMENT LEARNING

Deep Reinforcement Learning (DRL) is a type of machine learning that combines reinforcement learning (RL) with deep learning. DRL algorithms use neural networks to approximate the value function or policy for an RL agent. The input of the neural network are state features (e.g. all relevant information on the inventory status) and the output denotes the action (e.g. intervention) that should be taken. The agent interacts with its environment by taking actions, receiving rewards (often generated through simulation), and adjusting its neural network parameters accordingly.



4. GENERAL MODEL DESCRIPTION

We consider a single-site, single-echelon network where spare parts are delivered to customers directly. Upon failure of a component, the customer requests a replacement item from the spare part pool. When a replacement is received, the failed component (Unserviceable Unit; UU) is sent to the repair shop. After repair, the now Serviceable Unit (SU) is sent to the spare part pool.



We use the Deep Q Learning algorithm to train the neural network. We create two DRL models: one optimizing the timing and order of interventions (operational) and one determining the base stock levels minimizing total integrated cost (tactical/integrated). We use Discrete Event Simulation to determine the rewards (negative costs) for taking certain actions in given states. At first, we consider expediting repair jobs as sole intervention. Later, we demonstrate the possibility for model extensions by introducing temporarily hiring components from external parties

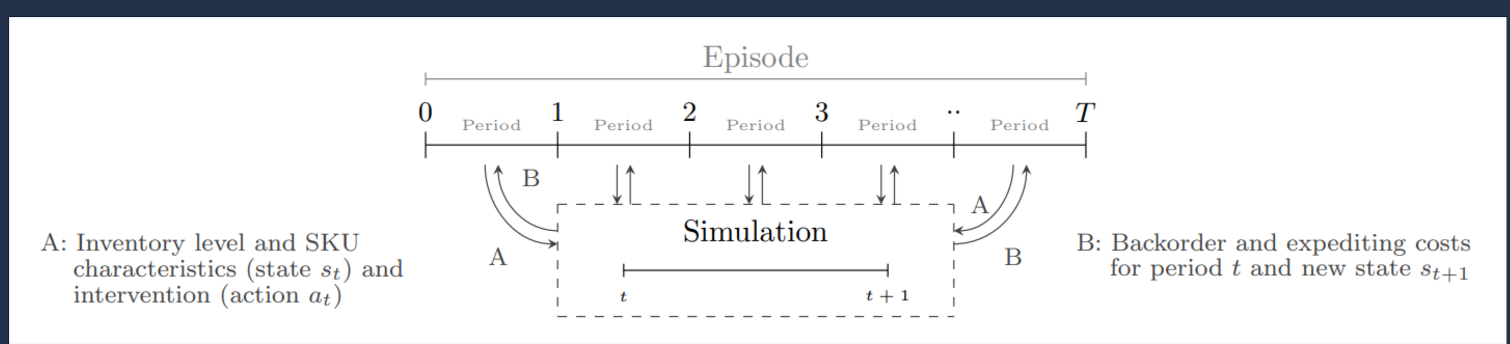
5. OPERATIONAL LEVEL

Model

First, we optimize interventions for a wide range of base stock levels.

State: stock on hand, backorders, repair pipeline, demand rate, acquisition costs, repair lead time, expediting success probability
Action: the number of repair jobs expedited
Cost: Backorder and expediting costs

By inserting SKU charactersitics (e.g. demand rate, acquisition cost) as state features, our model learns to take different actions for different products and is therefore very scalabe.



Experiments

We find that our model is successful in reducing costs while improving the fill rate. Additionally, the scalable approach of training one model to take actions for different SKUs works as intended. Below we explain our model's behaviour.

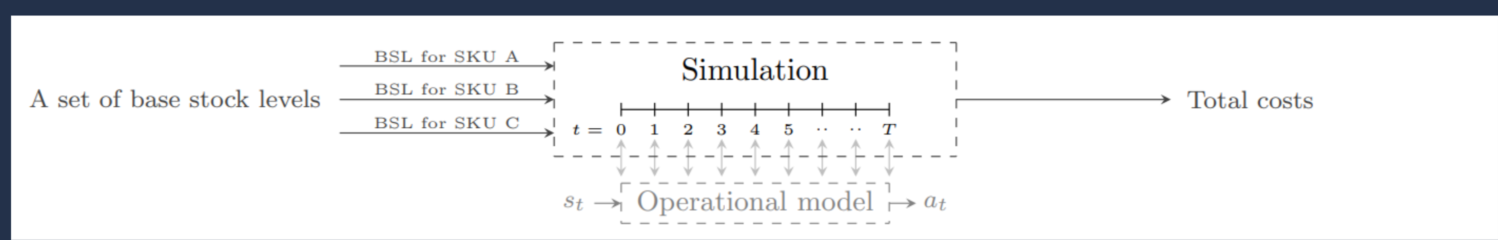
We find that more repair jobs are expedited for:

- high acquisition costs
- high demand rates
- high repair lead times
- low expediting success probabilities

6. TACTICAL/INTEGRATED LEVEL

Model

On a tactical/integrated level, we optimize the base stock levels (BSLs) such that the sum of the operational costs (i.e. backorder and intervention costs) and tactical costs (i.e. holding costs and penalty costs for not meeting service level agreements) are minimized. Operational decisions (interventions) are now fixed and given by the operational model, but the costs are dependent on the base stock levels (determined tactically).



The state of our integrated model does not change. Therefore, we do not have to link different states to different actions. Other optimization techniques can be used to solve the formulated model. We implement Simulated Annealing to compare our method's performance to.

Experiments

We find comparable results for both our DQL and Simulated Annealing solving approach. Both models find base stock levels that enable a 1 to 2% total cost reduction compared to our benchmark. We find that in general, performing interventions allows for lower base stock levels. How much the base stock levels can be reduced are dependent on the SKU's characteristics. Our model reduces base stock levels for:

- low acquisition costs: backorders costs do not outweigh holding costs,
- low demand rates: for lower demand rates, it is more cost efficient to expedite occasionally than to hold more stock,
- high repair lead times: expediting repair jobs with high lead times has more effect,
- high expediting success probabilities: the model learns to hold less stock for SKUs for which interventions have more effect.

Our model's behavior is found to be logical. As a model extension, we add the intervention of hiring components from third parties to quickly solve backorders. A 3.5% total cost reduction was found in the experiment run for this model extension.

7. CONCLUSION & RECOMMENDATIONS

Practical implications

Determining base stock levels in an integrated way should lead to better synchronization between the planning levels. Possible improvements:

- Lower total costs
- Improved service level
- Less interventions required
- Lower inventory investment costs

Contribution to literature

- Using Deep Reinforcement Learning for integration of operational and tactical decision-making
- Training one model to take decisions for different SKUs
- Relatively easy addition of interventions due to Markov Decision Problem formulation

Limitations

- Stability issues for Deep Q Learning algorithm
- Period length of two weeks does not allow for daily actions

Recommendations

- Choose a more stable algorithm (using stochastic policies)
- Reformulate repair pipeline to limit size of state vector and allow for stochastic lead times
- Include more than two interventions
- Explore possibility of building one neural network