Exact and heuristic approaches for the ship-to-shore problem

The authors of this papers presented an exact and heuristic approach. This paper is in the field of OR. It is about transporting resources in hard to reach places with multiple landing places. Defs:

- connectors: small boats, helicopters, etc that pick up resources.
- sea base: Have multiple type of loading spots and each spot has a loading and unloading capacity.A
- loading: Every connector has a specification what it can take

The authors wanted to have an exact method to solve this problem and compare it to heuristics, using artificial data and Dutch navy data.

The algorithm is NP-hard.

In the paper priority levels are introduced for different kind of items and groups of items which should be transported together.

To model this problem we use a time-space network. Every node has a location in time and space. All the loading and unloading is done in a single timeslot.

Related problems are:

- Evacuation: Resource constraints are not included, but some paths are more dangeruous than others
- windfarms:Resource constraints are included, because there is an order in which the materials need to be delivered.

The authors made an ILP-formulation. A better option is to use a route. Every connector has a route. This makes it possible to disregard certain routes, which makes it possibl

- objective = minimize the makespan
- s.t. all resources are transported
- every connectors uses a route
- respecting (un)loading capacities
- respect the order of the priority levels
- one delivery wave for every resources set.

Branch and bound algorithm (example)

There are tasks and agents who perform the tasks. You can make a table out of this with cells of the table containing the cost for the given task and agent. The total cost will be the sum of the costs of the tasks being performed by the given agent. We can make a tree out of this by having the an agent on every depth level of the tree. This agent chooses its task on every level. This way we do not need to go through all the possibilities.

Column generation

If we have a set of routes $R$ we will look at a subset $R'\subset R$. We are only adding routes to this subset that will improve the objective.

Combine both idead (Branch-and-price

It uses a subproblem like branch and bound. And for every subproblem we use column generation.

Instances

They constructed instances from

- Demand sets: Resources that should be delivered
- supply sets: Connectors and loading and landing capacity
- Operational constraints: Priority constraints and resource level constraints.

Results

- Greedy heuristic
    - 2/3 of instances where optimal
    - but performs worse with priority constraints
- Branch-and-price algorithm
    - 85% of instances solved within an hour
- Greedy heuristic vs Branch-and-price algorithm
    - Instances without resource set constraints: 40% of the instances improved by btranch-and-bound average improvement of 40%.
- Artificial instances:
    - Quite similar

Questions

How does the model represent refueling?: They stated that the refueling would take less time than the reloading of the resources, so it could be done simultaneously.He thinks