Digital Twin for Emission Control in Inland Water Transport

P2Z-WP 2 (15.05.2024)

Dhiraj Kumar

PhD Candidate

Section of Rivers and Ports

Department of Hydraulics Engineering

Faculty of Civil Engineering and Geosciences

Delft University of technology

Supervisor: Dr Alex Kirichek

Promotor: Prof Dr Mark van Koningsveld

TU Delft

Research objective

Research objective:
to develop and implement a digital twin to support energy transition decision making for inland shipping

Hypothesis:
There are *decision problems* in IWT that are so *complex* that stakeholders need a *digital twin* to decide

All of these problems involve: water depths, currents, locks and bridges, engine age, transport performance, etc.

- Finding consensus among stakeholders regarding the most effective emission reduction measures
- Supporting ship owners to pick the most effective energy carrier – energy converter combination
- Quantifying the potential of emission reduction measures for a specific network

What is a decision problem?

- System scale behavior is governed by non-linear relations and cascading effects.
- Same system is valued differently by different stakeholders
- Problem involves a very large number of agents
- Results require data that is not easily available

When is it considered complex?

- A Digital Twin can resolve complex relations, so they are included in decision making.
- Interaction may help stakeholders to understand trade-offs between interests
- Smart data structures may allow zooming in and out
- The potential impact of uncertainty can be shown

How can a digital twin help?

TU Delft

Research questions

1. How can we evaluate if a decision problem’s complexity makes it suitable for digital twin implementation?

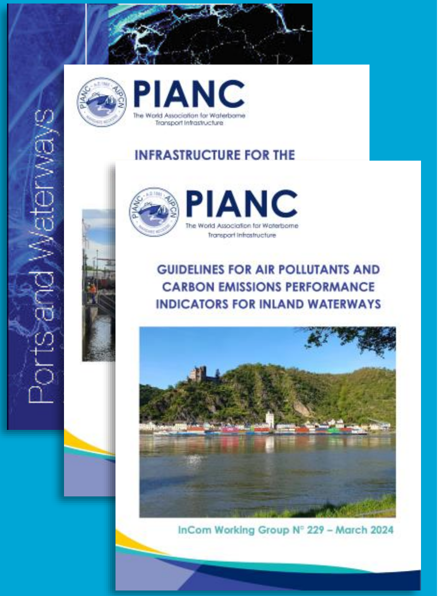
2. How can we design a simulation framework (and validate it) to quantify the identified complex decision problem?

3. How can we integrate the identified simulation framework into a digital twin that is interactive?

4. How can we confirm that stakeholder’s indeed make better decisions using a digital twin?

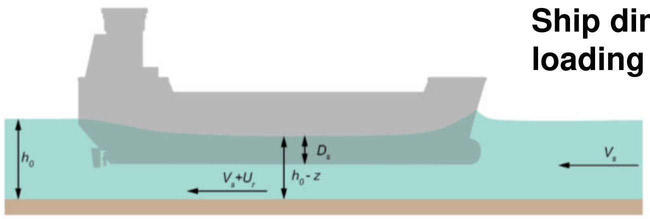
TUDelft

Research question 2 – the framework

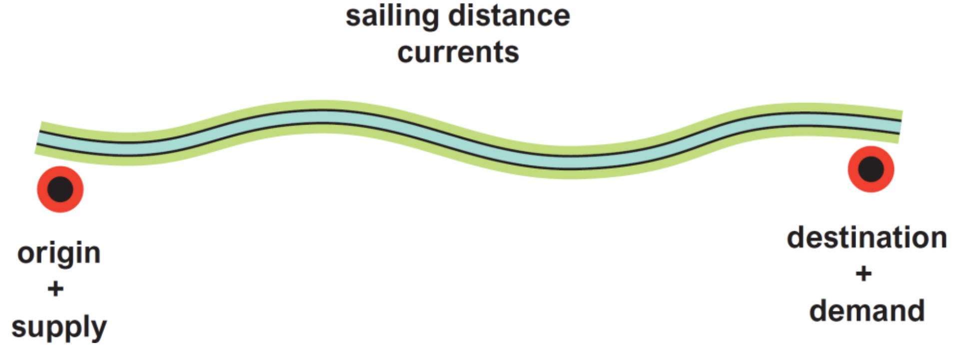


- a simulation framework … to quantify … complex decision problems

We propose **meso-scale discrete event modelling** to generate **bottom-up corridor scale insights** with **fairway section scale detail**

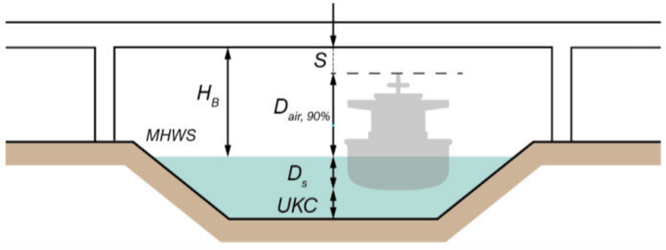


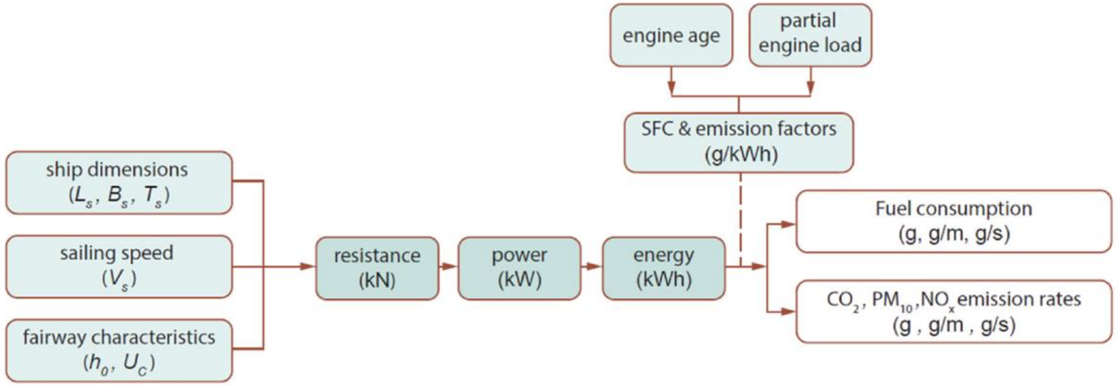
Ship dimensions, loading capacity, fleet



Sailing speed

Currents, channels, bridges, locks, etc.





This approach can be used to estimate **fuel use and emissions**, but also to investigate the **effects of discharge extremes!**

TU Delft

Research question 2 – the framework

- OpenTNSim is a useful toolbox available in Python

By utilizing the NetworkX package we can use advanced graph processing tools

- From very simple one-edge graphs
- Via more complex, but still highly simplified, multi-edged graphs
- To the full-scale Fairway Information System graph maintained by Rijkswaterstaat

All coordinate projections available
Detailed date-time control available
Full discrete-event simulation via Simpy

TU Delft

Research question 2 – the framework

- OpenTNSim is a useful toolbox available in Python

A basic simulation involves:

1. Specification of a graph
2. Specification of a vessel object
3. Simulation of the vessel navigating the graph
4. Inspection of the log results

Logs are useful when seeking validation against actual vessel logs

1

2

```
Vessel = type('Vessel',(energy_module.ConsumesEnergy,   # The vessel consumes energy
                        vessel_module.IsVessel,         # Basic information of the vessel
                       ),{})
```

```
# instantiate vessel object with the following inputs
vessel = Vessel(**{ "env": env,
                    "name": 'Vessel',           # you can give the vessel an arbitratry name
                    "origin": '0',              # start node of the route to sail
                    "destination": '1',         # stop node of the route to sail
                    "type": 'Va/M9 - Verl. Groot Rijnschip', # This indicates the vessel class. This info is mainly informative.
                    "L": 135,                   # m
                    "B": 11.45,                 # m
                    "T": 2.75,                  # m
                    "v": 5,                     # m/s If None: this value is calculated based on P_tot_given
                    "safety_margin": 0.2,       # for tanker vessel with sandy bed the safety margin is recommended as 0.2 m
                    "h_squat": False,           # if the ship should squat while moving, set to True, otherwise set to False
                    "P_installed": 1750.0,      # kW
                    "P_tot_given": None,        # kW If None: this value is calculated value based on speed
                    "bulbous_bow": False,       # if a vessel has no bulbous_bow, set to False; otherwise set to True.
                    "P_hotel_perc": 0.05,       # 0: all power goes to propulsion
                    "P_hotel": None,            # None: calculate P_hotel from percentage
                    "x": 2,                     # number of propellers
                    "L_w": 3.0 ,
                    "C_B": 0.85,                # block coefficient
                    "C_year": 1990,             # engine build year
                    "arrival_time": datetime.datetime(2024, 1, 1, 0, 0, 0),
                  }
               )

env.process(vessel.move())
```

3

```
env.run()
```

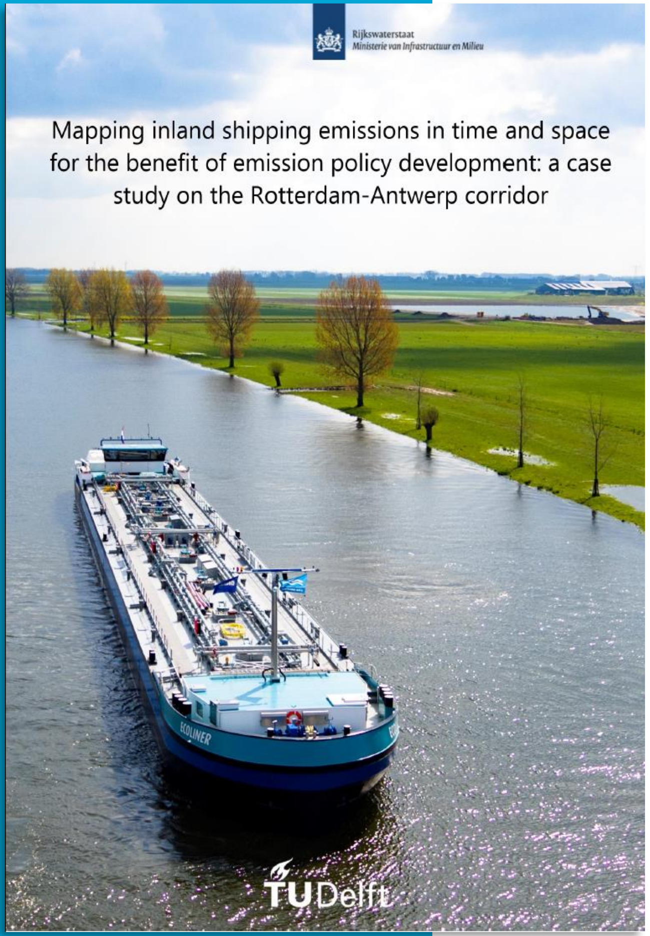
4

```
# the logging information is found in the logbook
pd.DataFrame.from_dict(vessel.logbook)
```

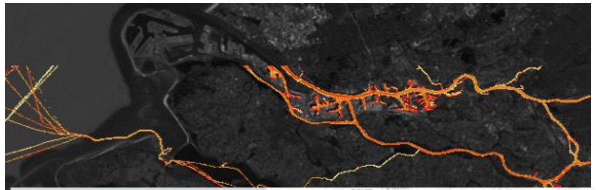
	Message	Timestamp	Value	Geometry
0	Sailing from node 0 to node 1 start	2024-01-01 00:00:00	{'origin': '0', 'destination': '1', 'route': [...	POINT (0 0)
1	Sailing from node 0 to node 1 stop	2024-01-01 05:33:20	{'origin': '0', 'destination': '1', 'route': [...	POINT (0.8983152841195216 0)

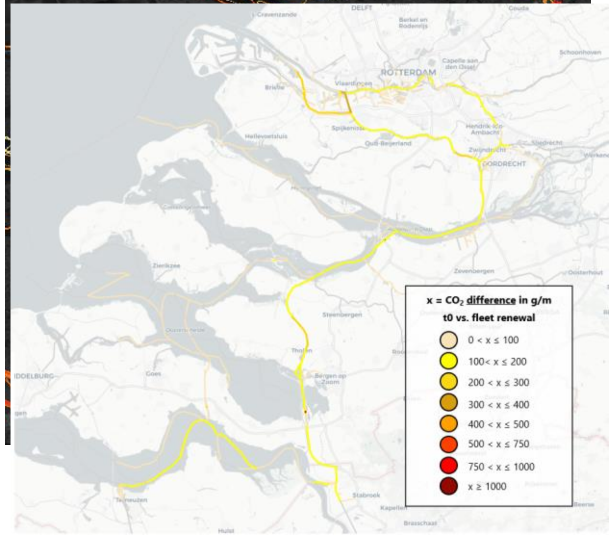
Evaluating effective emission measures

Use simulation to test 'Fleet renewal' (OpenTNSim)
Use AIS to determine 'T0 emission scenario'



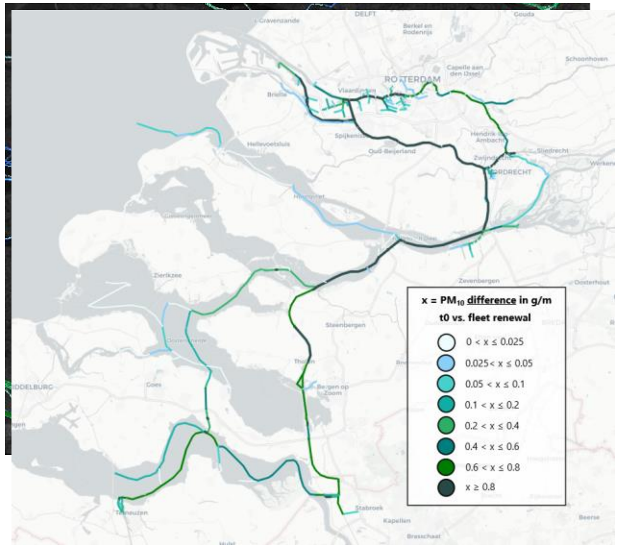
CO_2 emissions

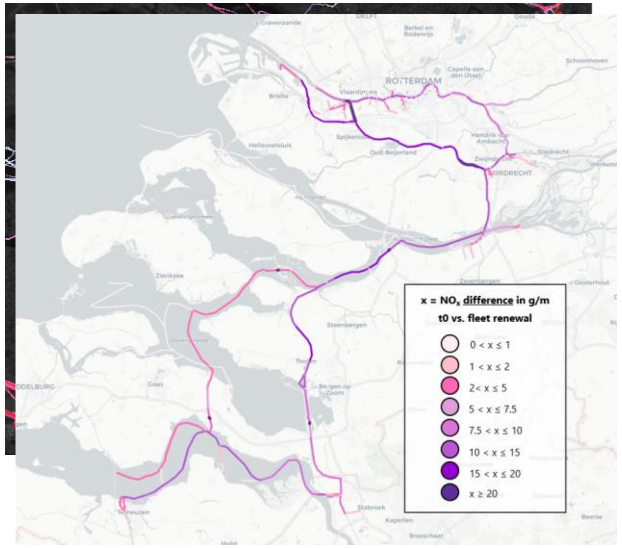




	Model simulation t0	Model simulation fleet renewal	Reduction
Average fuel consumption [L/h]	119	113	-5%
Average required power [kW]	482	482	0%
Total fuel consumption [L]	530730	504898	-5%
Total CO2 emission [kg]	1547533	1471233	-5%
Total PM10 emission [kg]	685,86	229,37	-67%
Total NOx emission [kg]	19546	13182	-33%

Table 8.3: Output of model simulation of one day (2 Sept 2019): simulated 't0 emission scenario' versus fleet renewal scenario (all engines from the year 2007 or older are replaced by new Stage V engines). Effect on average fuel consumption and power, and on total fuel consumption and potential emissions.

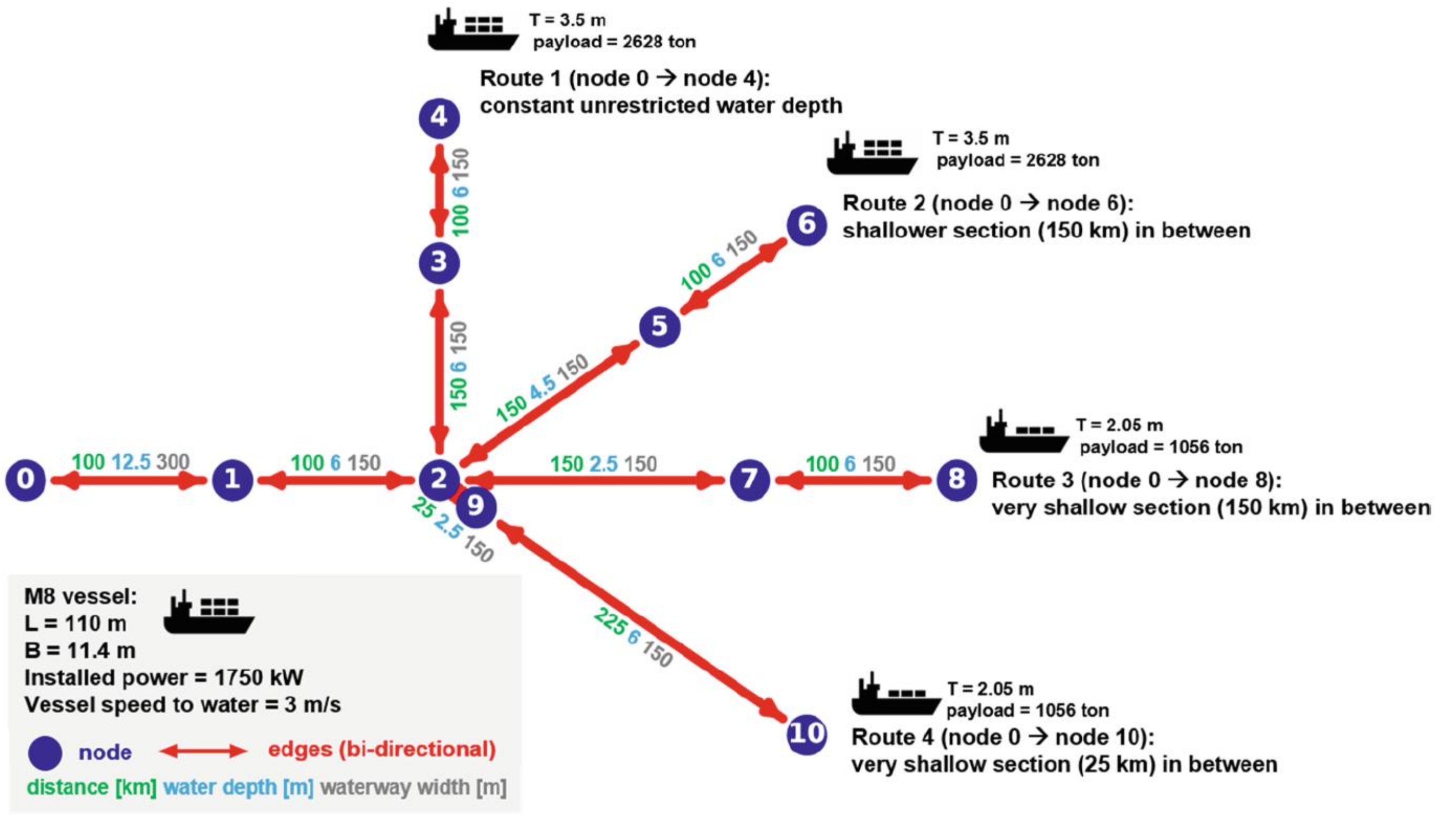




Selecting appropriate alternative fuels

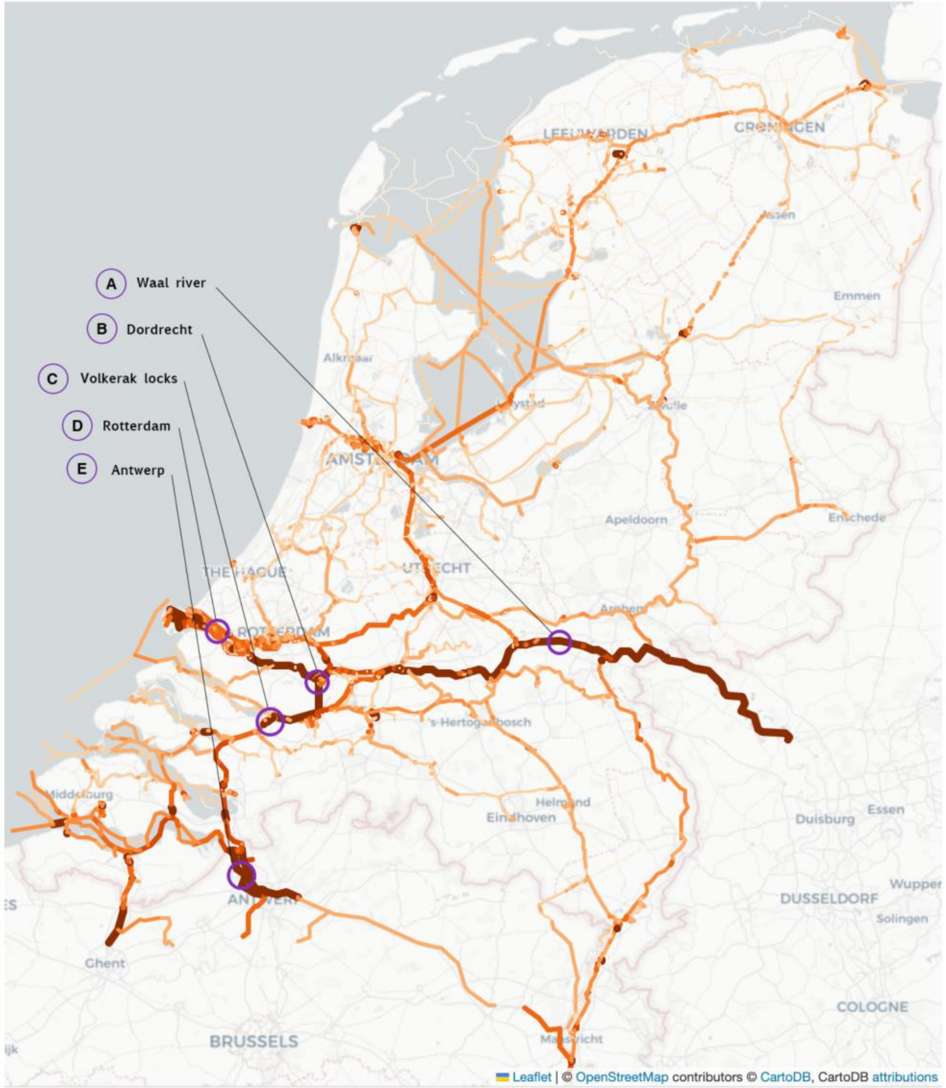
Corridor Scale Planning of Bunker Infrastructure for Zero-Emission Energy Sources in Inland Waterway Transport

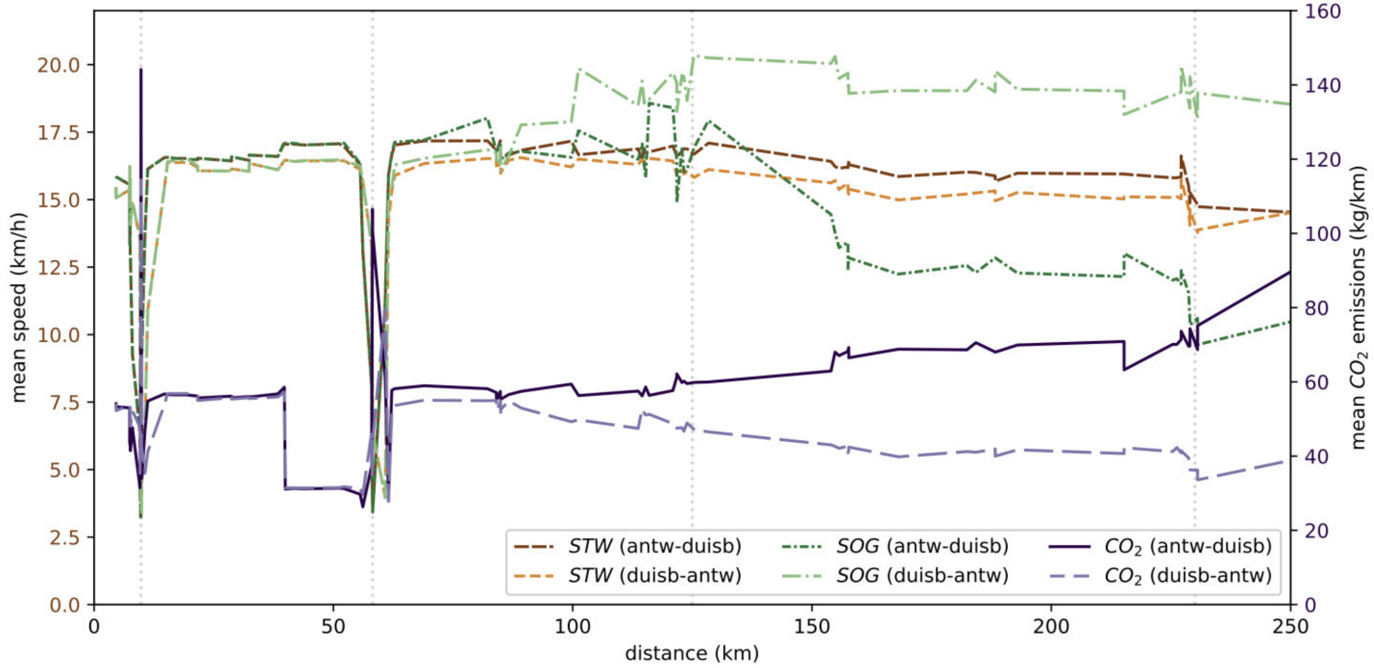
Man Jiang[1(✉)], Fedor Baart[1,2], Klaas Visser[1], Robert Hekkenberg[1], and Mark Van Koningsveld[1,3]

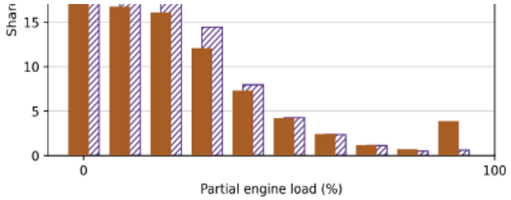


	Required energy amount (MWh)	Required amount of zero emission energy sources (for fuel only, excludes storage system)						
			Hydrogen (liquid, -253℃)	E–NH3 (liquid, -34℃)	E–methanol (liquid)	E–LNG (liquid)	Battery 2 MWh (20ft Containers)	Diesel
Route 1: constant unrestricted water depth	13.12	Mass (ton)	1.02	6.55	6.13	2.35		3.48
Payload: 2628 ton		Volume (m^3)	13.26	8.65	7.72	5.84	10.4 containers	4.77
Route 2: with	13.35	Mass (ton)	1.03	6.66	6.22	2.39		3.53
shallower section (150km) **Payload:** 2628 ton		Volume (m^3)	13.47	8.79	7.84	5.93	10.6 containers	4.85
Route 3: with very shallow section (150km)	13.18	Mass (ton)	1.02	6.57	6.14	2.36		3.49
Payload: 1056 ton		Volume (m^3)	13.28	8.67	7.74	5.85	10.5 containers	4.78
Route 4: with very shallow section	11.36	Mass (ton)	0.89	5.74	5.37	2.06		3.05
(25 km) **Payload:** 1056 ton		Volume (m^3)	11.61	7.58	6.76	5.12	9.2 containers	4.18

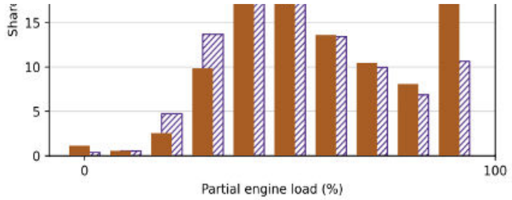
Quantifying hotspot causes







Partial engine load (Volkerak)



Partial engine load (Waal)

OpenCLSim

OpenTNSim

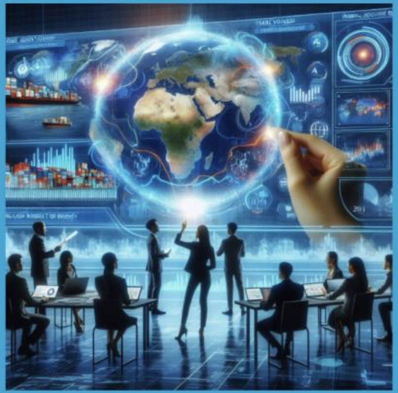
OpenQTSim

OpenTISim

TU Delft

Lesson overview

Cargo transport volumes
CBS, DESTATIS, EUROSTAT



OpenTISim
Cashflow analysis

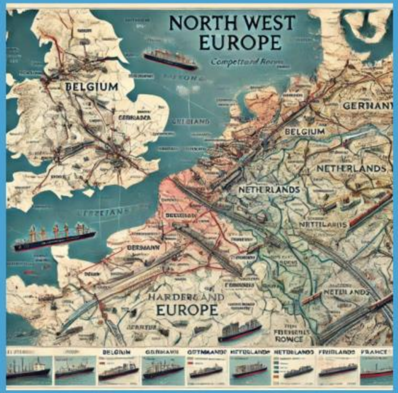
OpenTNSim
Energy-module

OpenTNSim
Locking-module

OpenTNSim
Port accessibility

OpenCLSim
Prolonged drought and IWT

Port competition
Contested hinterland areas



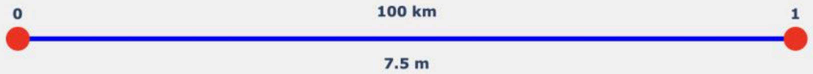
Nautical safety
Allision risk to wind parks

Digital Twin Dashboard

OpenTNSim
Energy-module

Lesson overview

Intro | Step 1 | Step 2 | Step 3



Origin: 0
Destination: 1
Start time: 2024-01-01 00:00:00
Vessel Type: Va/M9 - Verl. Groot Rijnschip
Length (L) [m]: 135
Beam (B) [m]: 11.45
Draught (T) [m]: 2.75
P_installed [kW]: 1750
C_year [-]: 1990
P_hotel_perc [-]: 0.05
Power demanded [kW]:
0 100 200 300 400 500 600 700 800 900 1000 1100 1200 1300 1400 1500 1600 1700 1800 1900 2000

Run Simulation

Resistance components as a function of power demand

The previous step addressed the resistance a vessel experiences as a function of its velocity. An additional aspect that will affect a vessels performance is the power it has available on board. As the vessel sails faster its resistance will increase, and so does the amount of power that the engine needs to deliver to overcome this resistance. When the required power approaches the available power the vessel should not be able to sail any faster.

Task - Change the range of power demand settings with the range slider, run the simulation and explore how the resistance components behave.

Q - what change do you observe in the resistance component graph?

Q - can you find out what limits the installed power provides to the vessel performance?

Q - how does the hotel power (as fraction of the installed power) affect the vessels performance?

Q - how does the installed power affect the vessel speed you observed under Step 1?

Simulation Results

Resistance components as a as a function of power demand

Total power delivered as a function of power demand

TU Delft

OpenTNSim
Locking-module

Lesson overview

Intro | Step 1 | Step 2

Origin vessel1: 0
Destination vessel1: 1
Start time vessel1: 2024-01-01 00:00:00

Origin vessel2: 1
Destination vessel2: 0
Start time vessel2: 2024-01-01 00:00:00

Vessel Type: Va/M8 - Groot Rijnschip
Length (L) [m]: 110
Beam (B) [m]: 11.4
Draught (T) [m]: 2.75

Speed vessel1 (v) [m/s]:
0.5 1.5 2.5 3.5 4.5 5.5 6.5 7.5 8.5 9.5

Speed vessel2 (v) [m/s]:
0.5 1.5 2.5 3.5 4.5 5.5 6.5 7.5 8.5 9.5

Run Simulation

Two vessel passing a lock in opposite direction

Interesting dependencies occur when two vessels pass a lock in opposite direction. The second vessel must wait until the first vessel resumes its journey after passing the lock.

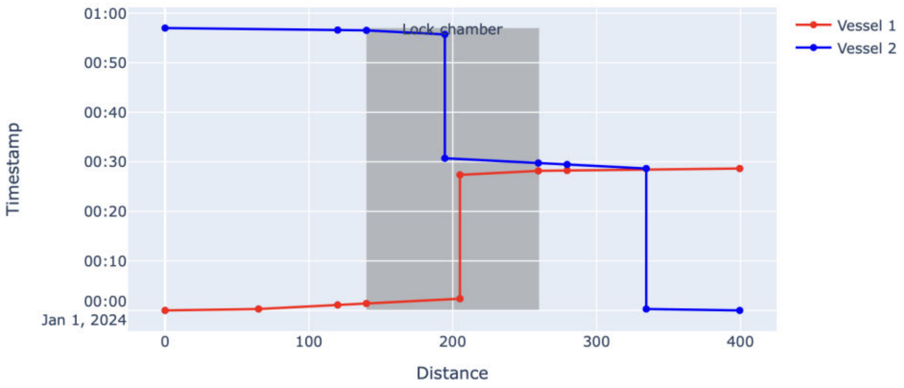
Task - Run the simulation and explore the steps each vessel takes passing the lock.

Q - Modify the start time of vessel2, and run the simulation again. Can you understand what happens and why?

Q - Modify the velocities of vessel1 and vessel2, and run the simulation again. Can you understand what happens and why?

Q - Compare this figure with the figure from the lecture notes that is shown under the Intro button. Can you identify similarities? What about differences?

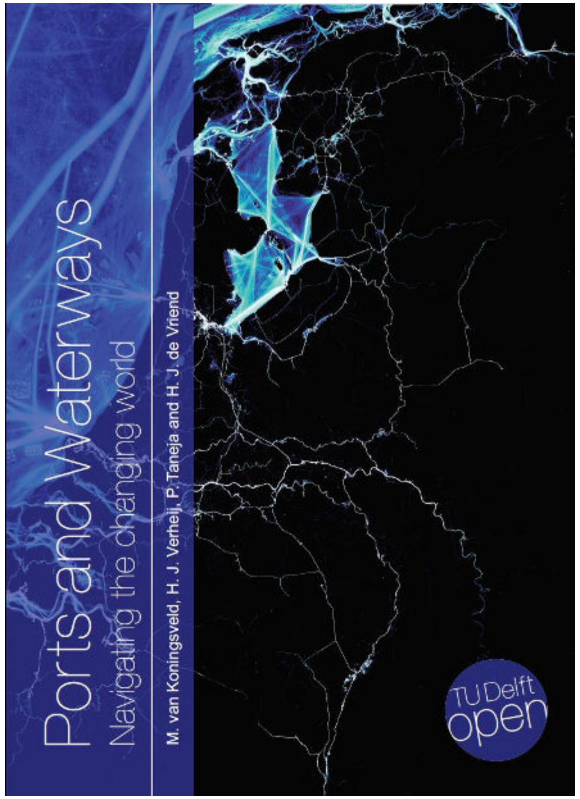
Simulation Results



Log file vessel1

Message	Timestamp	Distance
Sailing from node 0 to node 1 start	2024-01-01 00:00:00.000000	0
Sailing to position in line-up area start	2024-01-01 00:00:00.000000	0
Sailing to position in line-up area stop	2024-01-01 00:00:17.333333	65
Sailing to end of line-up area start	2024-01-01 00:00:17.333333	65
Sailing to end of line-up area stop	2024-01-01 00:01:06.222222	120
Sailing to first set of lock doors start	2024-01-01 00:01:06.222222	120
Sailing to first set of lock doors stop	2024-01-01 00:01:24.000000	140
Sailing to assigned location in lock start	2024-01-01 00:01:24.000000	140
Sailing to assigned location in lock stop	2024-01-01 00:02:21.777777	205
Passing lock start	2024-01-01 00:02:21.777777	205
Passing lock stop	2024-01-01 00:27:21.777777	205

TU Delft



OpenCLSim

Doi: [10.5281/zenodo.3251545](https://doi.org/10.5281/zenodo.3251545)

OpenTNSim

Doi: [10.5281/zenodo.3341516](https://doi.org/10.5281/zenodo.3341516)

Reading knowledge → **Coding** developing → **Playing** understanding

Questions

- We are considering suitable ‘complex’ problems to include in the research proposed here:
  - We would love to hear about ‘complex’ problems that would be of particular interest to you
  - What kind of information do you think would help to improve decision making?
  - What are your thought on Digital Twin functionality?
    - Education – inform stakeholders on key complexity
    - Negotiation – provide insight into key trade-offs
    - Optimization – help users to find the best solution
    - Other?

TUDelft