Report 1: Shared Python tool for modular computation and comparative of the TCO

Eco4Impact Team:
TORRES Lizbeth
AQUINO Luis
EZZAHRAOUI Hajar
Houcine
Salah

EcoBoatTwin Team:
LAAMOURI Karim
EZZEDINE Rania
RUIZ PATIÑO Ángel
BERMUDEZ Martin
BEN SALAH Mariem

Tuteur école : PELOV Alexander
Client : BERTHEBAUD Sebastien

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Vectura System

ARCHITECTURE FOR THE SHARED PYTHON TOOL FOR MODULAR COMPUTATION AND COMPARATIVE OF THE TCO

1.INTRODUCTION

This document presents the architecture of the module for computation of the TCO in detail. In first place, there is shown the format that should be followed in the tables for the database in order to keep the parameters needed for the computation of each sub-module value as well as a table for testing purposes called “DUMMY_INPUTS”.

In second place, there is presented the diagram block, the equation and the description for inputs (from the user/digital twin simulation), parameters (from the database) and outputs (from each module) for the computation of the CAPEX, OPEX and Residual Value (RV) for both land logistics and vessels cases.

Finally, the general diagram block for the computation of the Total Cost of Ownership (TCO) is presented in order to give a more general point of view.

Database Architecture

In this section, there is presented the specifications for the database architecture for the parameters needed in order to compute the CAPEX, OPEX and RV values.

2. GENERAL EXPLANATION OF ARCHITECTURE

The database contains two main tables: COUNTRY and VEHICLE_CHARACTERISTICS, and each table contains categories that specify the values which will provide the initial parameters for each sub-module.

2.1. COUNTRIES

In Table 2.1. are contained the parameters for the category of countries regarding fields such as taxes, costs of operations, wages, subsidies, prices of energy, among others.

COUNTRIES																						
COUNTRY	CAPEX TAX_1	CAPEX TAX_2	...	OPEX TAX_1	OPEX TAX_2	...	TOLLS	PORTS	WAGE OF DRIVERS	WAGE OF CREW_1	WAGE OF CREW_2	...	FEES	SUBSIDIES	INSURANCE	PRICE DIESEL_X	PRICE DIESEL_Y	...	GAS	HYDROGEN FUEL-CELL	HYDROGEN ICE H2	ELECTRICITY PRICE
Country 1																						
Country 2																						
...																						

Table 2.1. Structure of the table “COUNTRIES” of the database

The description for each field on the table 1 is given below, all fields described contains numerical values except for the COUNTRY field:

- *COUNTRY:* This field contains the name of the possible countries to be selected by the user.
- *CAPEX_TAX_**N**:* Contains all the parameters in order to compute the CAPEX’s sub-modules. The number of fields **N** depends on the country with the most taxes on CAPEX computations. The details in the number sequence is to be given in the final documentation since each country may share (or not) the same taxes.
- *OPEX_TAX_**N**:* Contains all the parameters in order to compute the OPEX’s sub-modules. The number of fields **N** depends on the country with the most taxes on

OPEX computations. The details in the number sequence is to be given in the final documentation since each country may share (or not) the same taxes.

- *TOLLS:* Contains the parameters for computing the tolls cost in terms of price per kilometer for each country.
- *PORTS:* Contains the parameters for computing the ports cost in terms of price per day in the sea for each country.
- *WAGE_OF_DRIVER:* Contains the parameters for computing the wages for truck drivers.
- *WAGE_OF_CREW_**N**:* Contains the parameters for computing the wages for vessel crew per rank according to applicable laws for each country. The total number of ranks will depend on the vessel with the largest rank. Besides, the specification in the sequence of numbers will be described in the documentation, since each activity (fishing or transport) may require similar (or not) ranks.
- *FEES:* Parameters for computing the fees for each country.
- *SUBSIDIES:* Parameters for computing the subsidies for each country.
- *INSURANCE:* Parameters for computing the insurance costs for each country.
- *PRICE_DIESEL_**A**:* Price of diesel of type **A** per country. **A** contains the name of the type of diesel used in vehicles.
- *GAS:* Price of gas per country.
- *HYDROGEN_FUEL_CELL:* Price of Hydrogen for fuel-cell, parallel hybrid technology per country.
- *HYDROGEN_H2:* Price of Hydrogen for 100% H_2 Internal Combustion Engine (ICE) per country.
- *ELECTRICITY_PRICE:* Price of Electricity per country.

2.2. CHARACTERISTICS OF THE VEHICLE

In Table 2.2. are contained the parameters for the category of vehicles regarding the type of vehicle used.

VEHICLE CHARACTERISTICS											
VEHICLE	VEHICLE CAT_1	VEHICLE CAT_2	...	DIESEL_X	DIESEL_Y	...	GAS	HYDROGEN FUEL-CELL	HYDROGEN ICE H2	ELECTRIC	HYBRID
Truck											
Boats											

Table 2.2. Structure of the table "VEHICLE_CHARACTERISTICS" of the database

The description for each field on the table 2 is given below, all fields described contains numerical values except for the VEHICLE field:

- *VEHICLE:* This field contains the two main categories of vehicles selected by the user.
- *VEHICLE_CAT_**N**:* Vehicle category **N** per type of vehicle. **N** contains the categories according to the weight (for trucks) and activity/size (for ships). The number of categories is the greatest amount of subdivisions between trucks and boats and its

definition is to be in the final documentation, since it changes depending on the vehicle.

- *DIESEL_**A**:* Parameter that defines that the type of diesel of type **A** is used by the vehicle.
- *GAS:* Parameter that defines the type of fuel on the vehicle is gas
- *HYDROGEN_FUEL_CELL:* Parameter that defines that the type of fuel on the vehicle is the hydrogen fuel-cell parallel hybrid technology.
- *HYDROGEN_H2:* Parameter that defines that the type of fuel on the vehicle is the 100% H_2 Internal Combustion Engine (ICE) technology.
- *ELECTRIC:* Parameter that defines that the vehicle is an electric vehicle.
- *HYBRID:* Parameter that defines that the vehicle is an hybrid vehicle.

2.3. DUMMY INPUTS

This table is not shown since this table will be used for testing purposes for the module. However, it will store data as inputs such as purchase value, maintenance, country, type of vehicle with its specifications, members of the crew (in case of a vessel), among other input values.

3. CAPEX

CAPEX FORMULA

CAPEX = VEHICLE_COST + INFRASTRUCTURE_COST + TAXES + FINANCING - SUBSIDIES

3.1.VEHICLE COST :

Three Purchase Scenarios :

Scenario 1: Buy NEW vehicle

vehicle_cost = purchase_price_new

Scenario 2: Buy USED vehicle + Convert/Retrofit

vehicle_cost = purchase_price_used + conversion_cost + certification_cost

Scenario 3: Convert YOUR OWN existing vehicle

vehicle_cost = conversion_cost + certification_cost

3.2. INFRASTRUCTURE COST

infrastructure_cost = hardware + installation + grid + software + site + safety + licensing

3.2.1 Hardware Cost

BEV / PHEV

Inputs :

- **E_t** : Total annual energy of vehicle t (kWh/year)
- **S_t** : Percentage of energy via slow chargers (%)
- **F_t** : Percentage of energy via fast chargers (%)
- **U_t** : Percentage of energy via ultra-fast chargers (%)
- **P_t** : Percentage of energy via public charging (%)

Constraint: $S_t + F_t + U_t + P_t = 1$ for each vehicle t

ENERGY DEMAND BY CHARGER TYPE

Per vehicle

$E_t^{slow} = E_t \times S_t$

$E_t^{fast} = E_t \times F_t$

$E_t^{ultra} = E_t \times U_t$

For the fleet

$E_{total}^{slow} = \Sigma_t E_t^{slow}$

$E_{total}^{fast} = \Sigma_t E_t^{fast}$

$E_{total}^{ultra} = \Sigma_t E_t^{ultra}$

CHARGING HOURS REQUIRED

$H_{demand}^{type} = E_{total}^{type} / (P_{type} \times \eta)$

Where:

- **P_{type}** : Charger power (kW)
  - $P_{slow} = 11$ kW
  - $P_{fast} = 50$ kW
  - $P_{ultra} = 150$ kW
- **η** : Charging efficiency = 0.95

Annual available capacity

$H_{cap}^{type} = n_{type} \times H_{op} \times D \times u_{target}$

Parameters:

- **n_{type}** : Number of chargers of the type
- **H_{op}** : Operating hours per day (16-24h)
- **D** : Operating days per year (365)
- **u_{target}** : Target utilization rate (0.70)

Required number of chargers

$n_{slow} = \lceil H_{demand}^{slow} / (H_{op} \times D \times u_{target}) \rceil$

$n_{fast} = \lceil H_{demand}^{fast} / (H_{op} \times D \times u_{target}) \rceil$

$n\_ultra = \lceil H\_demand^{ultra} / (H\_op \times D \times u\_target) \rceil$

Note: ⌈ ⌉ represents rounding up to the nearest integer

Annual cost per charger type

$cost\_infra^{slow} = (n\_slow \times price\_slow)$

$cost\_infra^{fast} = (n\_fast \times price\_fast)$

$cost\_infra^{ultra} = (n\_ultra \times price\_ultra)$

Typical parameters:

- **price_slow** = 5,000 €
- **price_fast** = 40,000 €
- **price_ultra** = 100,000 €

COST ALLOCATION PER VEHICLE
Vehicle share in each type

$share\_t^{slow} = cost\_infra^{slow} / E\_total^{slow}$

$share\_t^{fast} = cost\_infra^{fast} / E\_total^{fast}$

$share\_t^{ultra} = cost\_infra^{ultra} / E\_total^{ultra}$

Hardware cost allocated to vehicle t

$$hardware\_cost\_t = cost\_infra^{slow} \times share\_t^{slow} + cost\_infra^{fast} \times share\_t^{fast} + cost\_infra^{ultra} \times share\_t^{ultra}$$

Diesel / Biodiesel / HVO / E-Diesel / HEV /FCET / HICE / CNG / LNG

- **P_t** : Percentage of energy via public fuellilng (%)
- **D_t** : Percentage of energy via private charging (%)
- $E\_t^{private} = E\_t \times D\_t$
- $E\_total^{private} = \Sigma\_t\ E\_t^{private}$
- $share\_t^{private} = E\_t^{private} / E\_total^{slow}$

$hardware\_cost = share\_t^{private} * n\_stations * hardware\_cost\_station$

3.2.2 **Software Cost**

```
# BET / PHEV

software_cost_bet = "base software"

if smart_charging_enabled:

    software_cost_bet += " + load management"

# FCET / HICE

software_cost_h2 = "H2 monitoring and control"

# CNG / LNG
```

software_cost_gas = "optional gas monitoring"

Diesel / Biodiesel / HVO / E-Diesel / HEV / PHEV

software_cost_diesel = 0

software cost allocated to vehicle t

software_cost = softwaere_cost /number_vehicles_fleet

3.2.3 Grid Cost

BET / PHEV

total_power_kw = n_slow_chargers * "slow charger kW" + n_fast_chargers * "fast charger kW" + n_ultra_fast_chargers * "ultra-fast charger kW"

contribution ratio to grid cost = (n_slow_chargers * "slow charger kW" *share_t^slow + n_fast_chargers * "fast charger kW"*share_t^fast + n_ultra_fast_chargers * "ultra-fast charger kW"*share_t^ultra) / total_power_kw

grid_cost_per_truck = grid_cost *contribution ratio to grid cost

```
total_power_kw = n_slow * 11 + n_fast * 100 + n_ultra_fast * 350
# n_slow, n_fast, n_ultra_fast = number of chargers of each type
#11 kW for slow, 100 kW for fast, 350 kW for ultra-fast

if total_power_kw < 50:
   grid_cost = 5,000
elif total_power_kw < 500:
   grid_cost = 50,000
elif total_power_kw < 2000:
   grid_cost = 200,000
else:
   grid_cost = 500,000
```

FCET / HICE

grid_cost = n_stations * electrolyzer_grid_connection_cost

CNG / LNG

if gas_type == "LNG":

grid_cost = n_stations * lng_electricity_connection_cost/number_vehicles_fleet # liquefaction or pumping

else: # CNG

grid_cost = n_stations * cng_electricity_connection_cost /number_vehicles_fleet # compressors only

Diesel / Biodiesel / HVO / E-Diesel / HEV

grid_cost_diesel = 0

3.2.4 Installation Cost

installation_cost_bet = n_slow_chargers * "slow charger installation" *share_t^slow+ n_fast_chargers * "fast charger installation"*share_t^fast + n_ultra_fast_chargers * "ultra-fast charger installation"*share_t^ufast

installation_cost_h2 = n_stations * "H2 station installation"/number_vehicles_fleet

installation_cost_gas = n_stations * "compressor or LNG plant setup"/number_vehicles_fleet

installation_cost_diesel = n_pumps * "fuel pump/tank installation"/number_vehicles_fleet

3.2.5. Safety Cost

safety_cost= n_stations * safety_cost_unit

Safety_cost_unit=f(powertrain_type)

3.2.6. Licensing / Permits Cost

licensing_cost = f(powertrain type)

3.3 SUBSIDIES

Vehicle_Subsidy=f(powertrain type,vehicle weight/class,purchase cost,country,year)

Infrastructure_Subsidy=f(infrastructure cost,powertrain type,country,year)

3.4 TAXES

taxes=f(powertrain type,vehicle weight/class,purchase cost,country,year)

3.5 FINANCING :

CRF = Capital Recovery Factor

r = annual interest rate **after ESG adjustment**

n = loan term in years

$$\mathrm{CRF} = \frac{r \cdot (1+r)^n}{(1+r)^n - 1}$$

4. OPEX

This sections presents each sub-module used to compute the OPEX value, which is the aggregate of the results, for a selected vehicle within a specific country.

4.1. TAXES

The module calculates the annual operating taxes and environmental fees associated with a vessel.
It integrates all national and international tax components such as registration, ownership, circulation, energy, and emission-based charges depending on the vessel's characteristics and operational profile.

4.1.1 Compute taxes ships()

$$\begin{aligned} O_{Taxes(ship)} = {} & tax\_reg(c,k,L) + tax\_annual(c,k,L) + tax\_initial(c) \cdot P \\ & + D \cdot [\, I\_e(c,k,e,L) \cdot (\text{tax\_}energy(c,e) + EF\_CO2(e) \\ & \cdot (\text{tax\_}CO2(c,e) + p\_EUA(c)) + r\_NOxSOx(e) \\ & \cdot \text{tax\_}NOxSOx(c,k,e))\,] - discount\_env(c,k,e) \end{aligned}$$

- tax_reg(c, k, L): Registration or circulation tax **DB**
- tax_annual(c, k, L): Annual ownership tax **DB**
- tax_initial(c) · P: Purchase value tax **(P from User)**
- D: Annual distance traveled (in nautical miles or km) **User**
- I_energy(c, k, e, L): Vessel energy consumption (MWh/km or tons of fuel/km) **from User**
- tax_energy(c, e): Energy tax on fuel **DB**
- EF_CO2(e): CO_6 emission factor by energy type **from User**
- tax_CO2(c, e): National CO_6 tax **DB**
- NOxSOx(e): NOx/SOx pollutant emission rate **from User**
- tax_NOxSOx(c, k, e): NOx/SOx levy **DB**
- discount_env(c, k, e): Environmental bonuses or discounts **DB**

4.1.1.1 Mean Variables

- c = country of registration
- k = ship class
- e = type of energy (diesel, hybrid, electric, etc.)
- L = vessel size/length

- P = purchase cost
- D = annual distance travelled
- r = route (for ETS coverage)

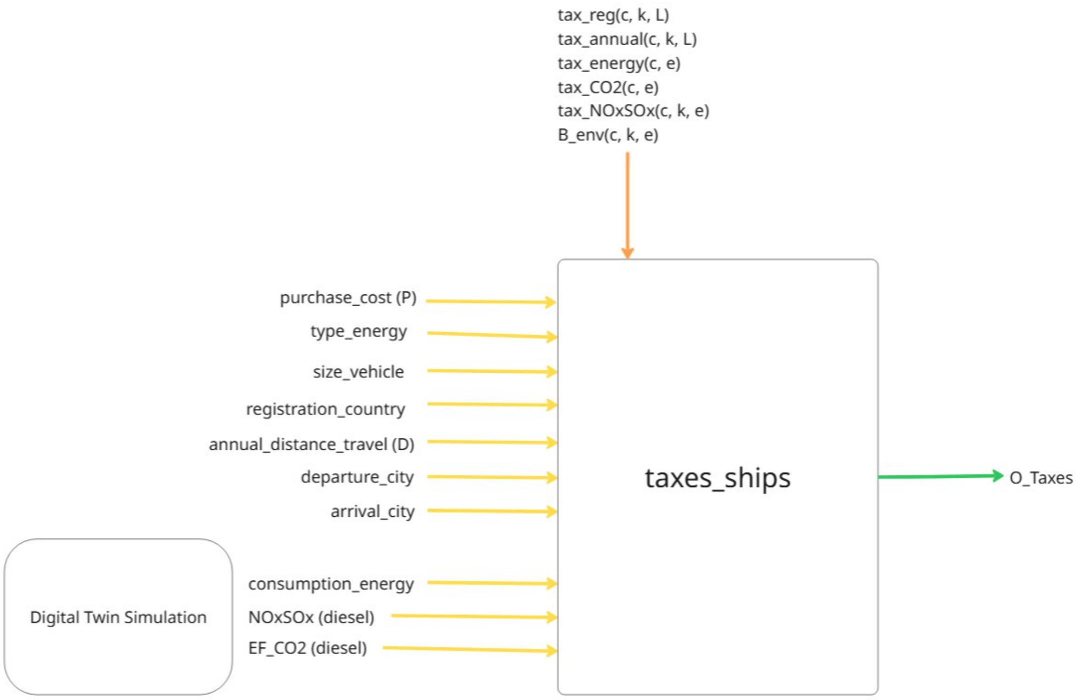

Fig. 4.1. Opex Taxes Ships

4.1.2 Compute taxes trucks ()

$$O_{Taxes\_trucks}(c, w, f, e, r) = \text{Energy\_consumption} \cdot \text{Base\_Tax(c,w)} \cdot \text{Fuel\_Multiplier(f)} \cdot \text{Emission\_Factor(e)} \cdot \text{Regional\_Coefficient}$$

4.1.2.1 Classification Dimensions:

- **c = Truck Class (EU vehicle categories)** N1: Light commercial vehicles (≤ 3.5t GVW) N2: Medium trucks (3.5t < GVW ≤ 12t) N3: Heavy trucks (GVW > 12t)
- **w = GVW Band (Gross Vehicle Weight in tonnes)**
- Bands: [0-3.5], [3.5-7.5], [7.5-12], [12-18], [18-26], [26-32], [32-40], [40+]
- **f = Fuel Type**
- **e = Euro Emission Standard**
- Euro 0, I, II, III, IV, V, VI, VI-d (increasingly stringent)
- **Base_Tax:** Base tax rate depending on country and possibly vehicle weight/power
- **Fuel_multiplier:** Factor depending on fuel type (Diesel, Electric, Hydrogen, etc.)
- **Emission_Factor:** Scales cost by CO_6 emissions (g/km or g/kWh equivalent)
- **Regional_coefficient:** Adjusts based on local taxation policies or environmental zones

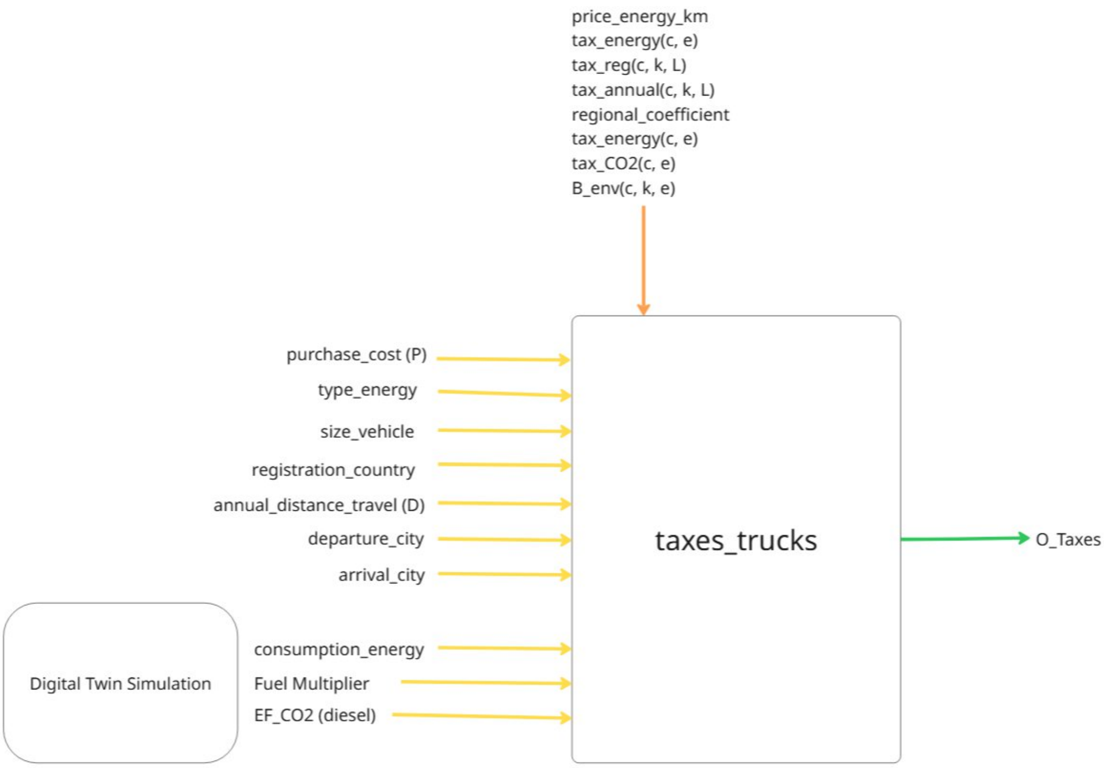

Fig. 4.2. Opex Taxes Trucks

4.2 TOLLS/PORTS

4.2.1 Compute tolls truck()

For computing the costs for tolls (in the case of trucks) and for ports (in the case of boats) costs, defined as “Access Costs”, there are two different formulas shown below.

In order to compute the access cost for trucks, this is the cost of using tolls:

$$O_{access\_costs} = price\_per\_km \times distance$$

This cost is the price for the trip multiplied by the number of kilometres travelled.

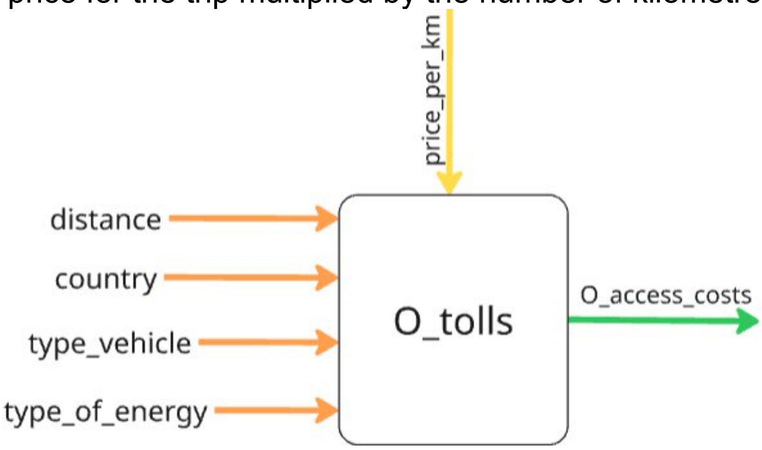

Fig. 4.3. Opex Tolls Trucks

In the previous diagram, there are the inputs (in orange) for the distance travelled, the country where the vehicle is travelling, the type of vehicle and its type of energy, the parameters from the DB (in yellow) for the price per kilometer and the output (in green) for the total cost of Opex Access Costs.

4.2.2 Compute ports ships()

In order to compute the access cost for boats, this is the cost of using ports:

$$O_{access\_costs} = price\ per\ day\ \times\ \#\ days\ of\ trip$$

This cost is the price of the trip multiplied by the number of days of the trip.

Fig. 4.4. Opex Ports Ships

4.3 INSURANCE

4.3.1 Compute Insurance ships/trucks()

$$O_{Insurance} = \%\ Insurance\ per\ energy\ \times (Purchase\ price\ -\ RV)$$

Represents the annual cost of insurance multiplied by purchase price, involving the residual value.

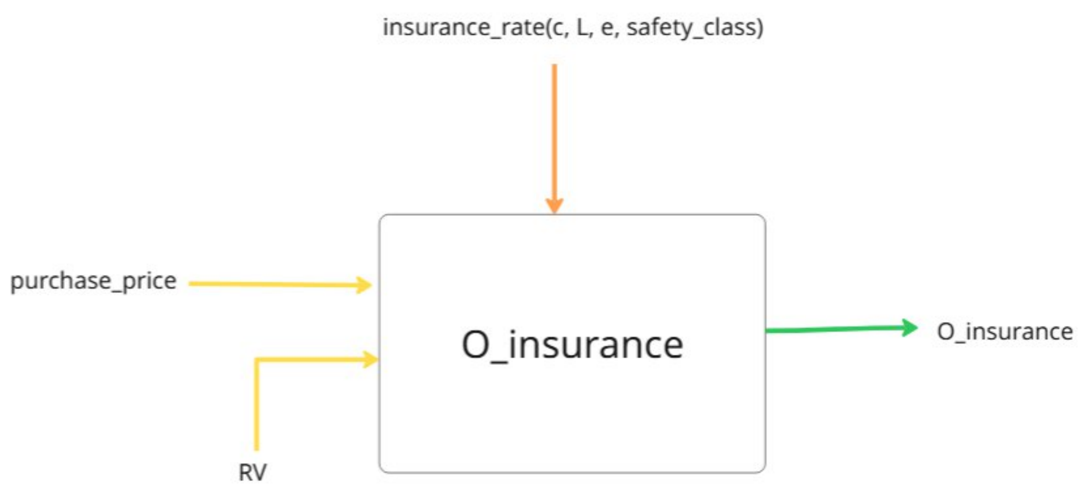

Fig. 4.5. Opex Insurance

4.3.2 Mean Variables database

c = country of registration
e = type of energy (diesel, hybrid, electric, etc.)
L = vessel size/length
safety_class = type of membership or secure

4.4 CREW

In the case where the user decides to choose a terrestrial route or a maritime route in either fishing, transport or crew transport vessel and each crew has a salary that depends on the rank, the sub-module to apply will follow the next equation:

$$O_{crew} = O_{wage/year} \times N$$

$$O_{wage/year} = \sum_{team_i}^{total\_ranks} wage\_rank_{g,a} \times team_i$$

Where *O_crew* is the total wage for the crew that depends on the O_wage/year and the number N of years. The first value depends on the sum of $wage\_rank_{g,a}$ that represents the wage for a country g and attribute a for an specific rank $team_i$ is the number of members of the same rank and $total\_ranks$ is the number or ranks in the crew (for trucks this value is 1, i.e, the driver).

In the following figure is shown graphically the inputs given by the user (orange), the inputs from the database (yellow) and the (output) given by applying the *O_crew* equation.

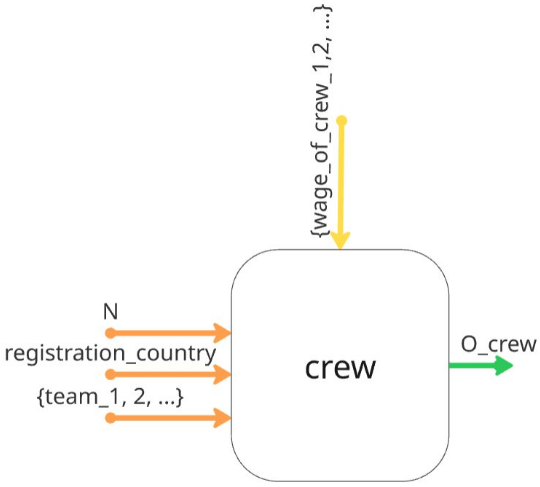

Fig. 4.6. Opex Crew

4.5 MAINTENANCE

The maintenance value is given directly from the user as an input whose value is aggregated in the OPEX calculation independently of the selected vehicle and country.

Fix values or parameters for each kind of energy and value.

We may use some standards values in the future in order to test the code and compare the results with our expectations.

4.6 ENERGY

The details for the energy consumption for trucks :

- **Inputs:**
  - type_energy
  - type_vehicle
  - size_vehicle
- **From Database:**
  - distance_km (annual distance traveled)
  - price_energy_per_km (per unit, filtered by type_energy and possibly country)
  - energy_consumption (consumption rate per km, filtered by type_vehicle, size_vehicle, and type_energy)
- **Output**
  - total_energy_cost (annual energy cost in €)

The formula to compute the energy cost O_{Energy} is:

$$O_{Energy} = distance\_km \times energy\_consumption \times price\_energy\_per\_km$$

In the case of boats, the computation of the energy cost value, the following equation is applied :

$$O_{Energy} = O_{kWh} \times O_{Energy\_price}$$

The cost of energy O_{energy} depends on the consumption of the energy O_{kWh} in the vehicle, determined with the values given by the digital twin simulation and the price of energy O_{Energy_price} within a country.

In the following figure is shown graphically the inputs given by the user (orange), the inputs from the database (yellow), the values which are retrieved from the digital twin simulation (red) and the (output) given by applying the *O_energy* equation.

Add the registration country and the country where the vehicle is running in eco4impact

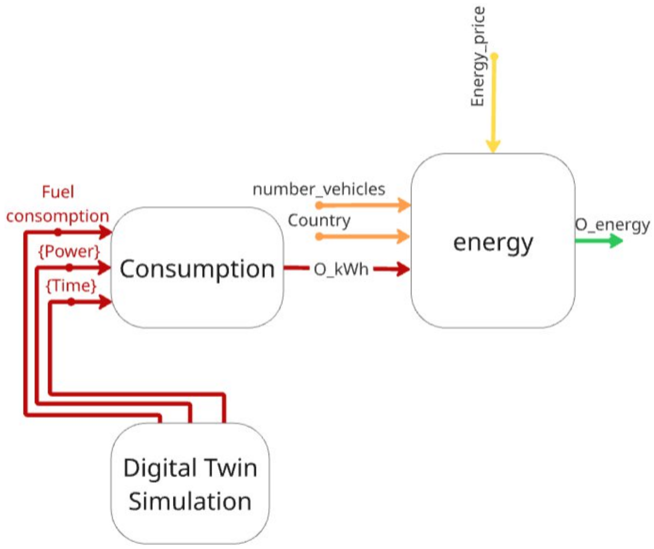

Fig. 4.7. Opex Energy - Diagram block of the "energy" sub-module

5. RESIDUAL VALUE (RV)

This sections presents each sub-module used to compute the RV value.

$$RV = \frac{DEPRECIATION}{\mathrm{IMPACT\_OF\_HEALTH\_POWERTRAIN}} + TAXES$$

5.1. DEPRECIATION

Inputs:

- **purchase_cost:** Initial acquisition cost of the vehicle.
- **age_vehicle:** Operational age of the vehicle at the evaluation time.
- **travel_measure:** Cumulative distance (km) or hours of operation (h).
- **maintenance_cost:** Cumulative maintenance expenditure up evaluation.
- **type_vehicle:** Truck or Ship
- **type_energy:** Type of energy (Diesel, Gas, BEV, Hydrogen, Hybrid)
- **country**

Parameters of database:

- **depreciation_rate_per_year:** Annual depreciation coefficient based on technological aging and energy
- **depreciation_rate_by_usage:** Depreciation coefficient per unit of use (**e.g., km or hour**), adjusted for energy type. Vehicles with certain energy (e.g., Electric or Hydrogen) experience higher depreciation per unit of use due to battery wear.

– **coef_depreciation_maintenance:** Coefficient that specifies how cumulative maintenance expenditures affect the depreciation of a vehicle. Higher values indicate that maintenance expenditures have a greater impact, reflecting the sensitivity of each energy type (Diesel, Electric, Hydrogen, etc.) to wear, repairs, or battery/tech degradation.

Equation:

$$\begin{aligned} &depreciation \\ &= purchase\_cost - depreciation\_per\_year() - depreciation\_by\_usage() \\ &- depreciation\_due\_to\_maintenance\_cost() \end{aligned}$$

Where:

$$depreciation\_per\_year() = depreciation\_rate\_per\_year * age\_vehicle$$

$$depreciation\_by\_usage() = depreciation\_rate\_by\_usage * travel\_measure$$

$$depreciation\_due\_to\_maintenance() = coef\_depreciation\_maintenance * maintenance$$

In the following figure is shown graphically the inputs given by the user (orange), the inputs from the database (yellow), and the output (green).

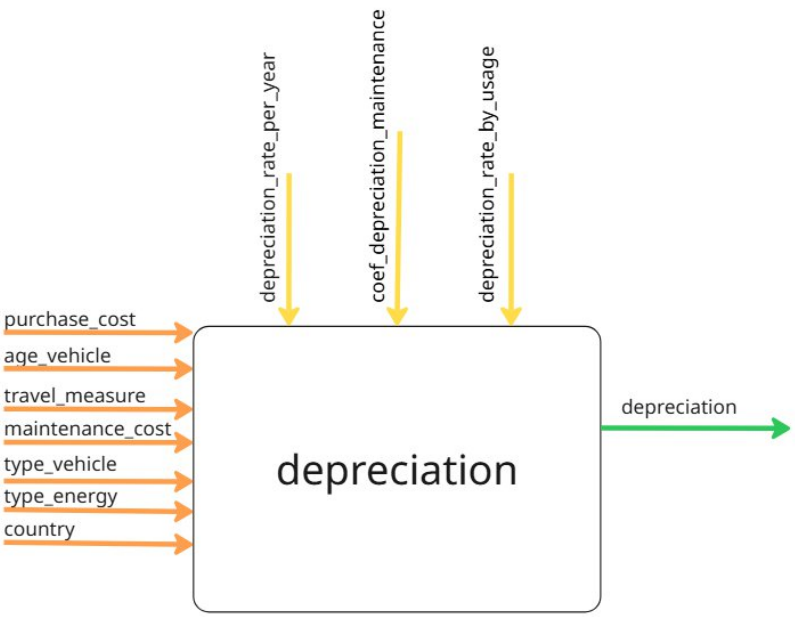

Fig.5.1. Diagram block of the “depreciation” sub-module

5.2. IMPACT OF HEALTH POWERTRAIN

5.2.1. eficiency()

Inputs:

- type_vehicle (truck or ship)
- type_energy (to select the correct heating value Q_{HV})
- minimum_fuel_consumption (SFC, from engine datasheet, in g/kWh)
- consumption_real: Actual or certified real-world consumption (BEV, FCET)
- utility_factor: The ratio between what an vehicle can drive electrically and what it can run in total.

Parameters of database:

- consumption_benchmark: Typical consumption for each type of energy (BEV, FCET)
- heating_value : Q_HV, Lower Heating value for the reference fuel
- η_{EV} : Efficiency of the electric propulsion system
- η_{ICE} : Efficiency of the internal combustion engine

A) Diesel, H2-ICE, Natural gas:
In practice, using SFC in g/kWh and Q_{HV} in MJ/kg, we use:

$$\eta_f = \frac{3600}{\text{SFC [g/kWh]} \cdot Q_{HV} \text{ [MJ/kg]}}$$

The function tech() computes the fuel-conversion efficiency of the engine, denoted by η_f. It links the specific fuel consumption (SFC) of the engine and the lower heating value of the fuel Q_{HV}. A low SFC and a high heating value result in a higher efficiency. This expression is valid for any fuel type (diesel, e-diesel, HVO, etc.); only the value of Q_{HV} changes with the energy type.

B) BET, FCET:
In practice, using SFC in g/kWh and Q_{HV} in MJ/kg, we use:

$$\eta_{sys} = \frac{consumption_{benchmark}}{consumption_{real}}$$

- BEV : kWh/km
- FCET: Kg H2/100km

C) HEV/PHEV:
In practice, using SFC in g/kWh and Q_{HV} in MJ/kg, we use:

$$\eta_{hybrid} = \frac{1}{[\alpha/\eta_{EV}] + [(1-\alpha)/\eta_{ICE}]}$$

- Utility Factor (α): is the electric fraction
- $1 - \alpha$: is the ICE (Internal Combustion Engine) fraction

General Equation :

$$tech() = (1 - \eta) * 100\%$$

5.2.2. obsolence() – Powertrain Model Obsolescence Factor (DM)

- powertrain_model_year (year of the powertrain model)
- type_vehicle (vehicle type: *truck or ships*)
- type_energy (energy type: diesel, H2, BET, etc.)
- distance_travel (depending on vehicle type)
- country

If obsolescence depends on time (trucks):

$$DM = e^{-\lambda_y (y_{now} - y_{motor})}$$

If obsolescence depends on distance (ships):

$$DM = e^{-\lambda_d (D_{now} - D_{motor})}$$

where:

- y_{now}: current year
- y_{motor}: powertrain model year
- D_{now}: total distance travelled (in km or nautical miles)
- D_{motor}: initial distance (usually 0)
- λ_y: yearly obsolescence rate (1/year)
- λ_d: distance-based obsolescence rate (1/km or 1/nm)

The function power_model() calculates the DM factor, representing the technological obsolescence of the powertrain system. The model uses a single ageing variable — either time or distance, depending on the vehicle category:

- For trucks, degradation mainly depends on the age of the model → the time-based formulation is used.
- For ships and some intensive transport trucks, degradation is linked to the distance travelled → the distance-based formulation is used.

A value of $DM = 1$ indicates a recent or up-to-date powertrain model, while values approaching 0 represent an outdated or obsolete technology. The mathematical structure is valid for any fuel type; only the parameters λ_y or λ_d change depending on the vehicle and energy type.

5.2.3. warranty() – Warranty factor DW

- warranty: total warranty duration (in years or km)
- type_warranty: type of warranty (for example, time-based or distance-based)

$$DW = 1 - \frac{\text{years elapsed since purchase}}{\text{years of warranty}}$$

(limited to the range [0, 1])

The function warranty() calculates the Warranty factor (DW), which represents the remaining share of the manufacturer's warranty. A newly purchased vehicle has DW ≈ 1, meaning it is fully covered by the warranty. As time passes or mileage increases depending on the warranty type, DW decreases toward 0, indicating that the warranty coverage is expiring. This factor mainly reflects the financial and contractual protection associated with the powertrain or main components. Its structure does not depend on the fuel or energy type, therefore it is valid for all powertrains such as diesel, electric, hybrid, or hydrogen.

Penalization :

$$RISK_W = 1 - DW$$

5.2.4. charging()

The charging() function quantifies technological degradation associated with the vehicle's charging behavior. It returns a normalized health factor between 0 and 1, where 1 implies ideal charging conditions and lower values indicate accelerated degradation due to frequent fast or ultra-fast charging.

Inputs:

- **E_annual_kwh:** Annual energy consumption of the vehicle. Represents the total energy charged per year.
- **C_bat_kwh:** Nominal battery capacity, as specified by the manufacturer or diagnostic system.
- **DoD:** Average depth of discharge (typical percentage of battery used per cycle)
- **S_slow, S_fast, S_ultra:** Proportion of energy charged via slow, fast, and ultra-fast methods, respectively, ensuring their sum equals one.

$$S\_slow + S\_fast + S\_ultra = 1$$

Parameters of database :

- d_slow, d_fast, d_ultra : Relative degradation factors for slow, fast, and ultra-fast charging to quantify cycle-specific impact.
- k_d : Global scaling coefficient, controlling the sensitivity and exponential decay of battery health under accumulated degradation.

Average degradation per cycle:

$$degradation\_per\_cycle = S_{slow}\ x\ d_{slow} + S_{fast}\ x\ d_{fast} + S_{ultra}\ x\ d_{ultra}$$

This expression gives a weighted average degradation per cycle, based on the share of energy charged at each power level

Equivalent full cycles per year :

$$cycles = \frac{E_{annual\_kwh}}{C_{bat\_kwh}\ x\ DoD}$$

Calculates how many "full equivalent cycles" the battery undergoes annually, based on total energy throughput and typical operational usag

Total annual degradation:

$$D = cycles\ x\ degradation\_per\_cycles$$

Aggregates the accumulated stress due to charging, combining cycle count and the weighted degradation factor. D increases with more annual cycles and greater use of rapid/ultra-rapid charging.

Charging health factor:

$$health\_charging = e^{-k_d \cdot D}$$

Translates the accumulated damage (D) into a normalized factor between 0 and 1. The exponential function reflects battery aging.

Penalization:

$$charging = 1 - e^{-k_d \cdot D}$$

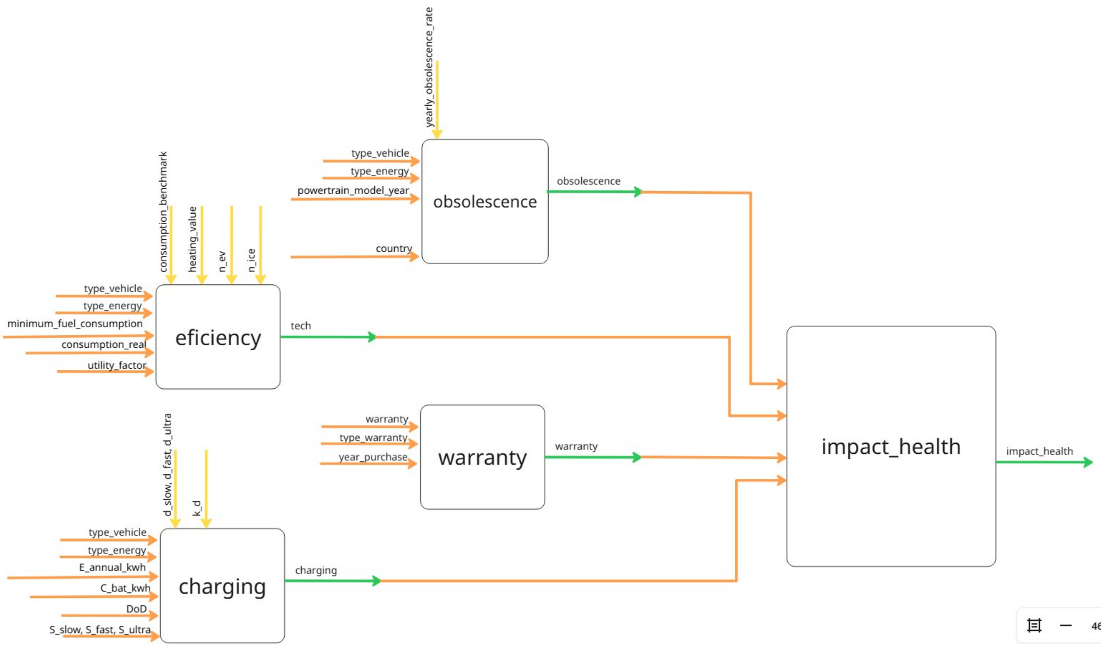

Fig.5.2. Diagram block of the "impact_health" sub-module

5.3. EXTERNAL FACTORS

5.3.1. ENERGY PRICE

On utilise le même diagramme qu'Opex pour ce calcul:

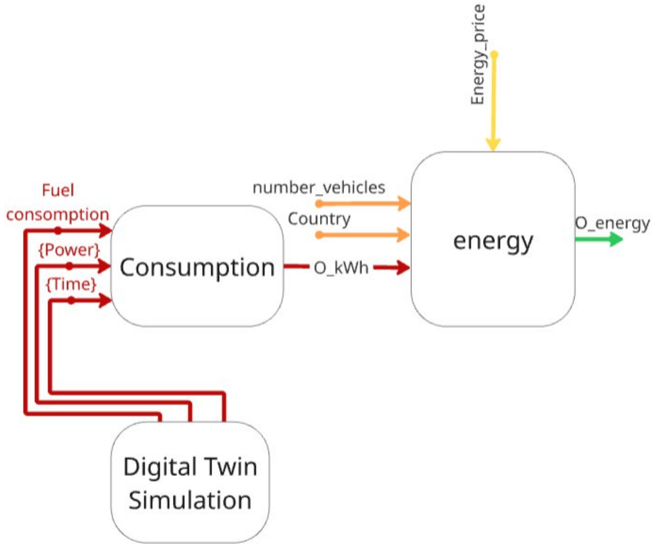

Fig. 5.3. Diagram block of the "energy" sub-module

La différence c'est qu'on utilise les prix futurs pour energy_price.

5.3.2. CO2 regulations, taxes, and emission standards

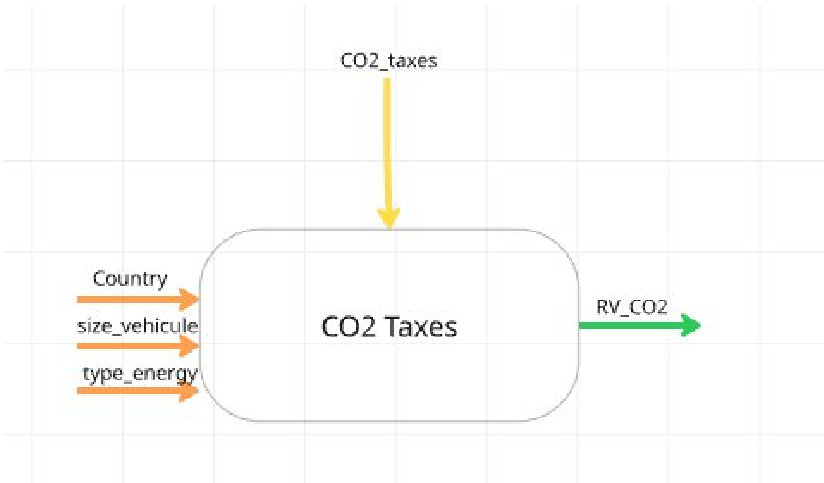

Fig. 5.4. Diagram block of the "RV_CO2"

5.3.3. SUBSIDIES

get_SS(): only depend on **Country**

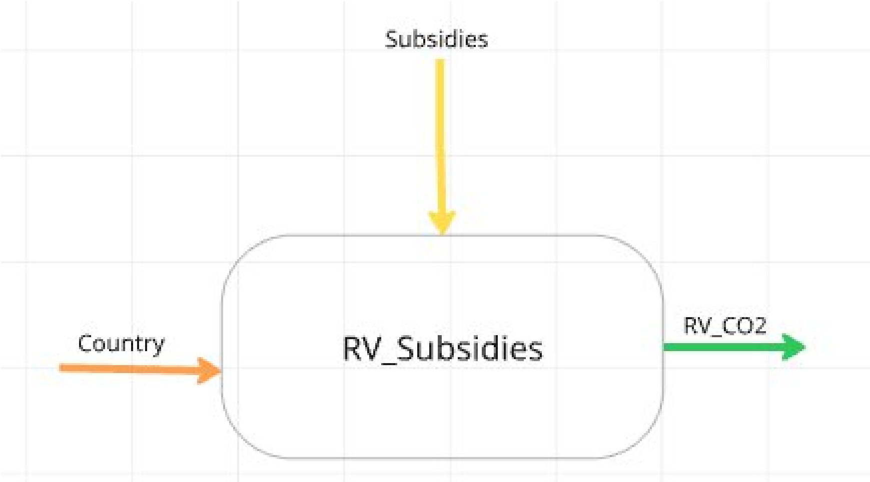

Fig. 5.5. Diagram block of the “RV_Subsidies”

5.3.4 General schema for external factors

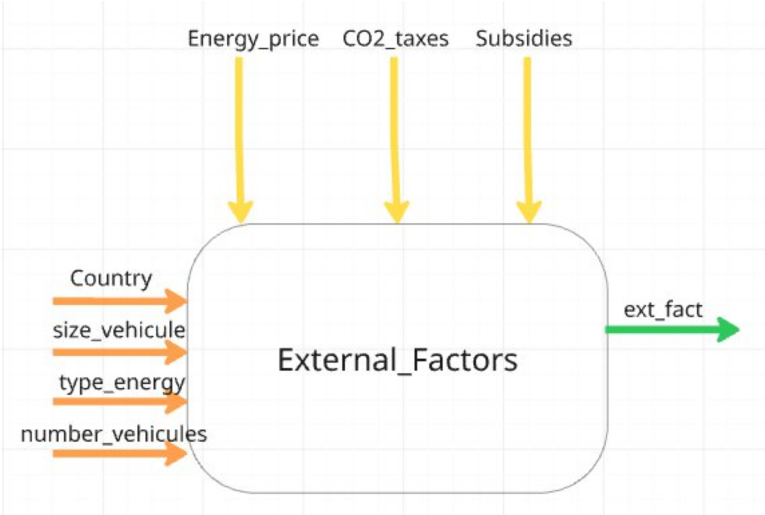

Fig. 5.6. Diagram block of the “RV_External_Factors”

$$\mathrm{RV}_{\mathrm{EXT}} = \beta_0 + \beta_1(\mathrm{Energy\_price}) + \beta_2(\mathrm{CO2\_taxes}) + \beta_3(\mathrm{Subsidies})$$

Where :

- **beta_0 (Baseline Constant):** 0
- **beta_1 (Energy_price Factor):** -12,000€
- **beta_2 (CO2_taxes Factor):** -3.0€
- **beta_3 (Subsidies Factor) :** +0.80€

Example : **Boat Type:** Diesel **Age:** 5 years

Where :

Energy_price = 1.7€/Litre
CO2_taxes = 100€ * 5 years = 500€
Subsidies = 0

$$\mathrm{RV}_{\mathrm{EXT}} = 0 - 12,000(1.70) - 3.0(500) + 0.80(0)$$

$$\mathrm{RV}_{\mathrm{EXT}} = 0 - 20,400 - 1,500 + 0$$

$$\mathrm{RV}_{\mathrm{EXT}} = -21,900€$$