University of Stavanger

FACULTY OF SCIENCE AND TECHNOLOGY

MASTER'S THESIS

Study program/specialization: Petroleum Engineering. Natural Gas Technology.	**Spring semester, 2021** **Open**
Author: Andrey Grodzinskiy	
Program coordinator: Mohsen Assadi	
Title of master's thesis: Techno-economic assessment of low-carbon ammonia production	
Credits: 30	
Keywords: Hydrogen Green Ammonia Power-to-X Industrial decarbonization	Number of pages: 82 + supplemental material/other: 0 Stavanger, 19.05.2021

Acknowledgements

I am grateful to the people from the University of Stavanger that have helped me with the Master's Thesis. It would not be possible to do this alone. I thank Øystein Arild – the head of the Department of Energy and Petroleum Engineering at the University of Stavanger and Aly Anis Hamouda as they supported my decision to research green ammonia production feasibility in Norway.

I would like to thank my supervisor Mohsen Assadi and co-supervisor Homam Nikpey for the guidance and help in the process of writing this work.

I would like to thank Abhinav Baskar and Rockey Abhishek for the help with the Python programing language since it was quite hard to develop the code for the thesis without a professional recommendation.

I would like to thank Reidar Brumer Bratvold from the University of Stavanger for his course "Decision making and risk analysis". During this course, I have obtained invaluable knowledge needed for the economic evaluation which was successfully implemented in my current work.

I am so grateful to the Norwegian government which allows for a thousand students each year to come to Norway and study at any University for free. The time I had been living in the Kingdom of Norway will forever alive in my memory.

I would like to thank the Jernalderveien 57 dormitory and people with whom I spent the Covid2019' quarantine for the best gift which I could not imagine receiving: I have started to be a writer being there, and the first book of mine written in the Russian language "Patient № 402" has already been shortlisted in London, Great Britain.

I thank my family for endless support and faith that I can finish the thesis. I would like to thank all the generations of my family for a long and honorable history started in 1432.

Abstract

Decarbonization of the ammonia production industry would require the use of innovative low-carbon production technologies. The use of electrolyser-based ammonia production instead of steam methane reforming may potentially reduce emissions from ammonia production significantly. This paper presents the results from the techno-economic assessment of the green ammonia production plant connected to the Norwegian electricity grid.

The desalination system and alkaline electrolyser (hydrogen production), air separation unit (nitrogen production), ammonia reactor, and ammonia storage unit were considered as the main steps in the scheme of the green ammonia production plant.

The system used 0.17 tons of hydrogen and 0.83 of nitrogen to form 1 ton of green ammonia. The specific energy consumption for the green ammonia production plant was found to be 10.02 MWh/ ton NH_3 The water electrolysis system with a respective electrolyser efficiency at stack 4.4 kWh/Nm^3 consumed 53.13 MWh/ ton H_2 The main source of emission was considered to be only the carbon intensity factor of the grid at the level of 25 kg CO_2 /MWh. Uncertainty and sensitivity analysis were conducted for the NPV of the base case scenario, scenario 1 and scenario 2. Breakeven study was conducted for scenario 1 and scenario 2.

List of content

List of figures

List of tables

Abbreviations

Following abbreviations were used in the work:

Mt — Megaton

GHG — Greenhouse gas emission

$kgCO_2$ — Kilogram of carbon dioxide

$GtCO_2$ — Gigaton of carbon dioxide

CO_2 — Carbon dioxide

H_2 — Hydrogen

N_2 — Nitrogen

H_2O — Water

NH_3 — Ammonia

TRL — Technology readiness level

MVC — Mechanical vapor compression

ASU — Air separation unit

CAS — Cryogenic air separation

SMR — Steam methane reforming

CAGR — Compound annual growth rate

CAPEX — Capital expenditure

$ — US dollar

€ — Euro

A — Annuity coefficient

n — Project lifetime

NPV — Net present value

CF — Cash flow

CI — Carbon intensity factor

MW — Megawatt

MWh — Megawatt hour

kWh — Kilowatt hour

kW — Kilowatt

kJ — Kilojoule

ºC — Celsius

K — Kelvin

kg — Kilogram

LCOA — Levelized cost of ammonia

LCOE — Levelized cost of electricity

SEC — Specific energy consumption

1. Introduction

Anthropogenic climate change caused by the release of greenhouse gas emission is one of the biggest challenges faced by humanity. Complete decarbonization of all sectors of the economy is essential to reach the goal of limiting global temperature rise to 1.5 ºC [1], as stated in the Paris climate agreement. CO_2 emissions have become a serious issue as shown on Figure 1 [2]. While the transition from fossil fuels to renewable energy generators for energy supply is gaining traction across the globe, there has been far less attention on the industrial sector till now. The industrial sector is responsible for approximately 25% of the global GHG emissions. Within the industrial sector, the chemical and petrochemical industry have a substantial carbon footprint at the level of 3.6 % of the global GHG emissions [3].

One of the major contributors to GHG emissions, within the chemical industry is the production of ammonia using hydrogen produced from natural gas-based steam methane reforming. It contributes to 1.44 % of the total energy related GHG emissions globally [4]. The primary demand for ammonia is in the agricultural sector, where it is used for the production of urea and other fertilizers. Annually, 180 Mt of ammonia is produced at present, with a respective market value of €80 billion [5]. Although new agricultural practices, which require less artificial fertilizers are being explored, the demand for ammonia is stated to increase in the short and medium term to ensure food security for the increasing world population [6]. The role of ammonia in the shipping industry and for storing large quantities of hydrogen economically is being explored, and could lead to higher demand for ammonia in the future [7], [8], [9].

At present, Ammonia is produced by combining hydrogen and nitrogen in the Haber-Bosch process. The ammonia production process can be divided into three stages. The first step involves production of reagents, i.e. nitrogen and hydrogen. The reagents react at high temperature and pressure (450 degree Celsius and 150 bar) inside a catalytic chemical reactor producing ammonia in the second step, and finally, the ammonia is liquefied/pressurized and stored in the third step. In natural gas-based ammonia production, hydrogen is produced by steam methane reforming of natural gas, which results in the release of 7 kg of CO_2/ton of ammonia [10].

Green ammonia production differs from the baseline process in the hydrogen production step. Hydrogen is produced from water electrolysis using renewable electricity instead of SMR. The underlying exothermic reaction, as shown in Equation 1 remains the same in both cases.

$$N_2 + 3H_2 \leftrightharpoons 2\ NH_3 \quad -92.4\ \text{kJ/mol} \quad (1)$$

In the next sections, these steps along with the sub-components selected for carrying out the different processes are described briefly.

Advantage of green ammonia over hydrogen in shipping industry is quite clearly presented following the fact that ammonia is easier to store and can be used without preparation step in internal combustion engines [11]. This directly leads to cheaper prices for ammonia-based fuel.

Considering European intention to reduce CO_2 emissions by 2030, the green ammonia concept seems to be a foreseeable alternative to replace ammonia production through conventional processes [12]. The main purpose of the current master thesis work is to perform a techno-economical evaluation of a green ammonia plant located in Norway. This paper identifies scenarios in which a green ammonia in Norway could be economically feasible in addition to leading to lower CO_2 emissions.

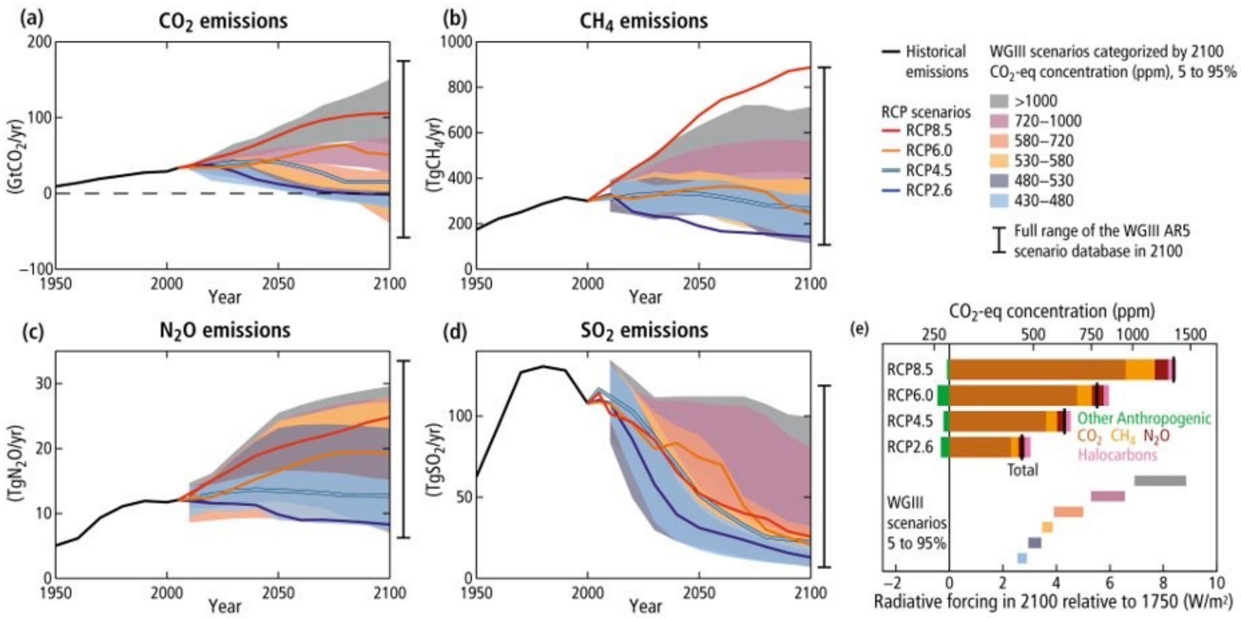

Figure 1. Industrial Decarbonization [2].

2. Literature review

H. Zhang et al. performed a comparative study of two possible alternatives for green ammonia production, where the required amount of hydrogen was either produced from the biomass or using an electrolyzer fed by renewable electricity [13]. The most desirable option in terms of the lowest production cost was a methane-to-ammonia system with a respective production cost of 400 $/ton. The biomass-to-ammonia case was presented with the production cost of 450 $/ton, while the power-to-ammonia system was not economically feasible and could only become viable if the price at stack would be reduced to 470 $/stack. C. Fúnez Guerra et al. conducted a techno-economic analysis of a green ammonia production plant in Chile with its transport to Japan [14]. The main parameter to reach a pay-back period of 5 years was the sales price of green ammonia determined at a value lower than 400 €/ton. Performing sensitivity analysis, authors presented electricity price as the key parameter to regulate the economic feasibility of the project and it explained almost 97% of the NPV variance. (electricity price was determined to be 26 €/MWh or lower to obtain positive value for NPV). However, there were certain limitations of the study, related to the special power purchase agreement and not taking under consideration the ammonia market volume and the possibility for green ammonia produced in the plant to enter the market. Richard Michael Nayak-Luke et al. conducted an analysis on techno-economic viability of islanded green ammonia as a carbon-free energy vector and as a substitute for conventional production [15]. In this research, authors chose 534 locations in 70 countries as a subject to study. For each location, 2 scenarios were introduced: present production possibility and forecast for 2030. The main ways of generating electrical power for the plant of electricity were wind and solar energy. According to the paper, the best geographical locations with associated lowest LCOA were Southern and Western parts of Asia with a respective Levelized cost below 700 $/ton. Also, there were few parts of Europe where the LCOA was at about 550 $/ton. For the 2030 viability forecast, the authors demonstrated even lower numbers for LCOA, considering predicted reduction in the LCOE by 2030 for solar and wind (375 $/ton both for European and Asian regions).

3. Methodology

Green chemical technologies at low TRL approach was used in this work for the techno-economic evaluation [16]. As the first step, market demand at present, as well as future forecasts for the green ammonia production were evaluated. The second step involved development of the conceptual process models, material and energy balance models to obtain a preliminary estimation (30 % accuracy) of specific energy consumption, associated emissions and component sizes of the green ammonia production plant. The economic analysis (calculation of NPV) from the investor's perspective was performed as the third step. Finally, a sensitivity analysis of NPV being target value for study was conducted.

3.1. Market analysis and demand assessment

The Global Green Ammonia market is expected growing at a CAGR of 52.9 % by 2027 (for a study period from 2019 to 2027) [17]. Currently, the green ammonia production cost is not competitive compared to the conventional green ammonia market price [18]. In future, a key role will play incentives provided and regulations enforcing the use of green ammonia in a form of CO_2 taxation and credits for the emission-saving projects [18].

3.2. Conceptual process model

A conceptual process model was developed in Python with certain subcomponents (all the necessary codes could be found in the Appendix of the current work). Assuming capacity of the green ammonia production plant to be 400 ton per day, material and energy balance calculations were performed to estimate the amount of energy required per 1 ton of green NH_3 and associated emissions from the green ammonia plant.

The green ammonia production process can be divided into three subsystems i.e. hydrogen production step, nitrogen production step and synthesis of ammonia in the reactor. After producing green ammonia, there is a need for the green ammonia to be stored at the storage system.

The process schematic of the green ammonia production plant is shown in Figure 2. Seawater enters MVC as **M0**, where it goes through a vapor compression system being desalinated and compressed. Desalinated water flow **M1** enters electrolyser, where hydrogen is being produced at the respective temperature of electrolysis of 80 ^oC. Then, produced hydrogen **M2** goes to the electrical preheater to obtain the required temperature for entering the ammonia reactor while oxygen **M9** as a subproduct of the reaction could be sold to the local buyers if they are present in the vicinity. **M4** air from the atmosphere enters ASU, where the temperature of Nitrogen separation is -172 ^oC. Separated Nitrogen **M5** goes to the Preheater for the same purpose as Hydrogen — to reach the required temperature for ammonia synthesis. Therefore, **M3** and **M6** are preheated hydrogen and nitrogen respectively, ready to enter the ammonia reactor. After the ammonia reactor, **M7** being ammonia obtained in the loop at the temperature of 450 °C and pressure of 150 bar, needs to be pressurized and stored. **M8** represents the final product as compressed ammonia at a temperature of -33 °C.

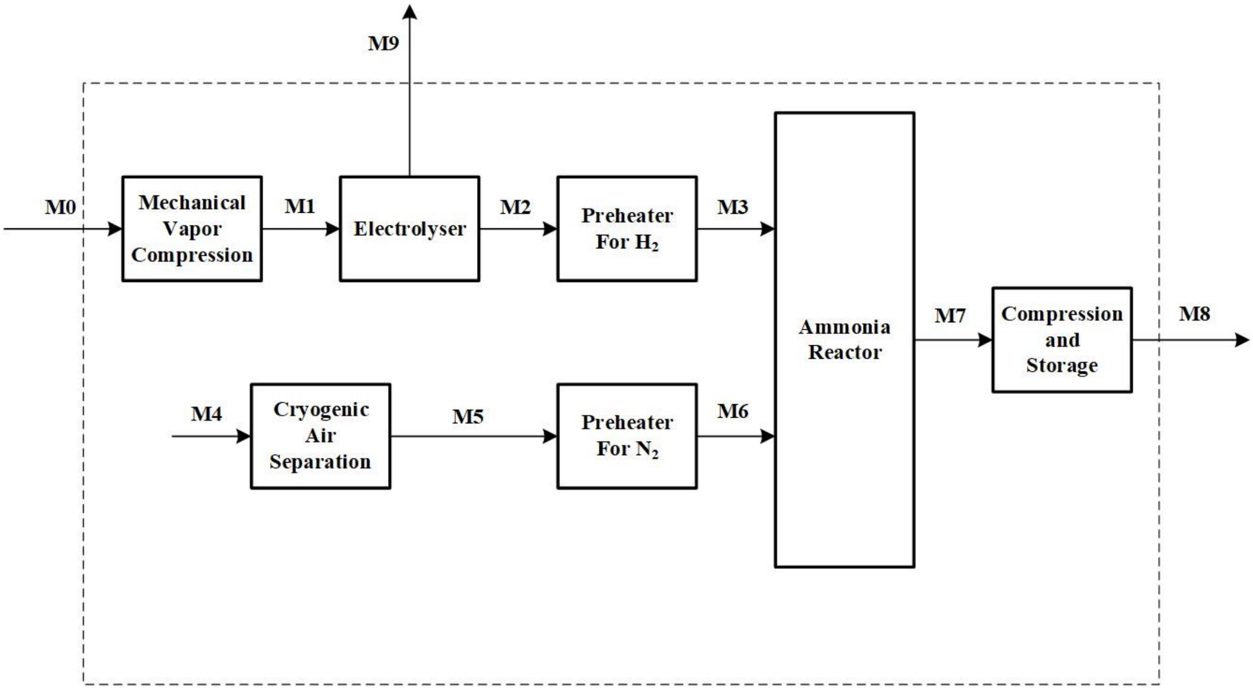

Figure 2. **Representative scheme of the green ammonia production plant.**

Also, in the frames of power requirements of the green ammonia plant, specific energy consumption was calculated, where electrolyser efficiency at stack was chosen to be 4.4 kWh/Nm^3 (since the range presented in the literature was found to be from 3.8 kWh/Nm^3 to 4.4 kWh/Nm^3, it was decided to consider for this part the upper limit of the range) [19]. Table 1 below represents a list of related assumptions made at this stage.

Table 1. List of assumptions for Energy & Material Balance calculations.

Name	Value	Units	Reference
Electrolyser efficiency at stack	3.8 (for the main model); 4.4 (for the SEC calculation part)	kWh/Nm^3	[19]
Rectifier efficiency	0.9	-	[20]
Cryogenic Air Separation (CAS) efficiency	0.552	kWh/Nm^3	[21]
Nitrogen efficiency of CAS	0.79	-	[22]

Power required per 1 ton of Nitrogen	0.00225	kWh/ton	[22]
Mechanical Vapor Compression (MVC) efficiency	9	kWh/m^3	[23]
Adiabatic efficiency of MVC	0.75	-	[23]
Driver efficiency of MVC	0.95	-	[23]
Power required per 1 ton of water	0.71	kW	[22]
The temperature of Hydrogen after electrolyzer	80	Celsius	[24]
The temperature of Nitrogen after CAS	-172	Celsius	[25]
The temperature of ammonia synthesis	450	Celsius	[26]
The pressure inside the ammonia loop	150	bar	[26]
Power required per 1 ton of ammonia inside a reactor	0.2633	kWh/ton	[27]
Mass of water for cooling purposes inside ammonia loop	9.5	Ton/minute	[22]
Power required for water pumping	0.0073	kW	[22]

Ammonia production requires hydrogen and nitrogen as feedstock for the ammonia reactor. Using Equation 1 above and stoichiometry of ammonia, the molecular mass of ammonia is about 17.03 g/mol meaning the molecular mass of hydrogen and nitrogen for ammonia reaction to being 3.03 g/mol and 14 g/mol respectively. Thereby, for 1 ton of ammonia, the demand for hydrogen is 0.17 ton and for nitrogen is 0.83 ton.

3.2.1. Hydrogen production step

Generally, the ammonia plant has certain requirements for reagents to enter the ammonia loop. Hydrogen feed is expected to be with the purity of a minimum 99.999 mol % while oxygen can be presented at the level of maximum 0.001 mol % [26], [28], [29]. Assuming water feedstock is brackish seawater with a salt concentration in the range from 1500 ppm to 10000 ppm, this introduces the need for desalination subsystem for water before being used in the electrolyzer. Consequently, mechanical vapor compression has been selected for this study due to its ability to desalinate seawater of any initial level of salt concentration alongside the flexibility and cost-effectiveness [30].

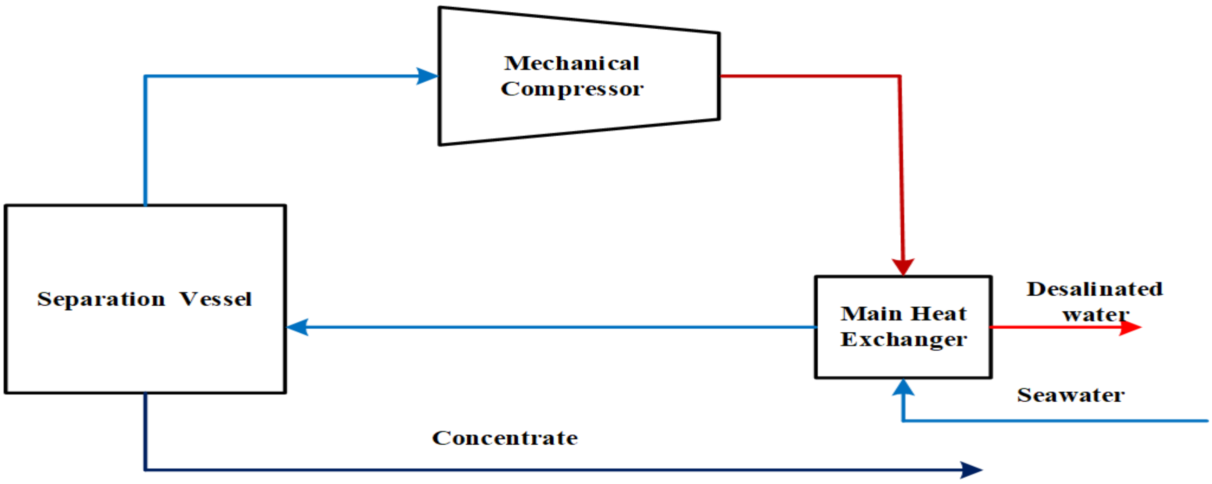

Figure 3. Mechanical vapor compression desalination process.

Figure 3 represents a general scheme of the mechanical vapor-compression distillation unit. Mechanical compressors are used as an energy input for evaporation of seawater with the following usage of compressed vapor as a heat source and also for increasing the pressure of seawater coming into the system [31]. Seawater enters the unit where it has to be separated into two substations: concentrate filled with waste (salts) and desalinated water [32].

After the MVC unit, distilled water enters the electrolyzer with specific power requirements. Nel Model A3880 Alkaline electrolyzer was chosen for this part of the analysis with power consumption at stack 3.8 kWh/Nm^3 [19].

3.2.2. Nitrogen production step

Considering hydrogen ratio 3:1, nitrogen feed is expected to be at a minimum 99.99 mol % with existing oxygen at a maximum of 0.01 mol % [26], [28], [29]. For this study, the volume of nitrogen production was preferred to be the main parameter for choosing appropriate equipment. Based on this, cryogenic air separation was selected for covering the demand for nitrogen requirements of the green ammonia plant.

Figure 4 shows the general scheme of the cryogenic air separation unit. Air enters the air compression subsystem where it is compressed by multistage compressors following interstage cooling. Then compressed and precooled air goes to the purification block to remove unwanted chemical components (to prevent freezing of H_2O and CO_2). The purified air enters the main heat exchanger where the air is cooled and partially liquified. After that, moves to the distillation column block consisting of two columns: high-pressure (5-6 bars) and low-pressure (around 1bar). Finally, gaseous nitrogen is liquified in the condenser [25].

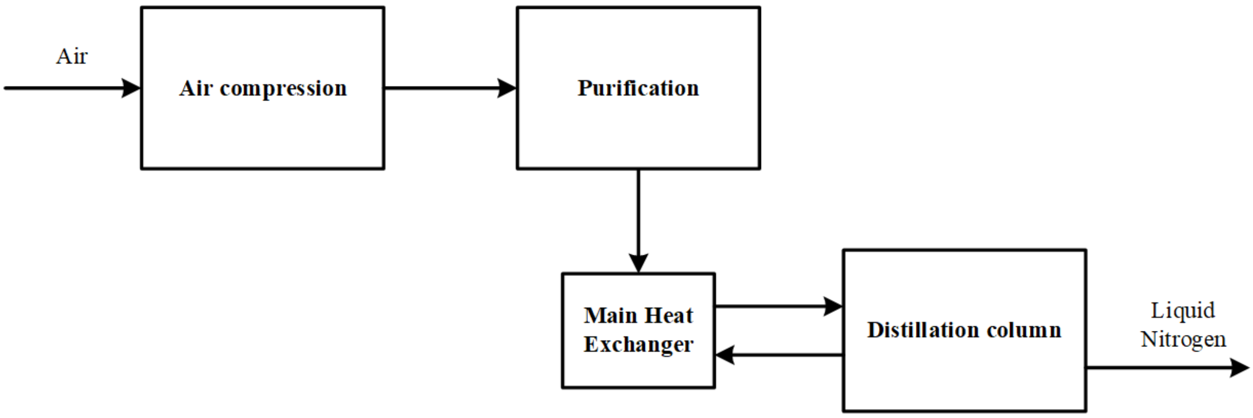

Figure 4. General scheme of an air separation unit

Hydrogen leaves the electrolyzer at a temperature in the range of 50-80 °C. Nitrogen, in turn, has an output temperature of -172 °C. On the other hand, the temperature inside the ammonia reactor is 450 °C, which determines preheating needs for the hydrogen and nitrogen before entering the reactor [33], [34].

3.2.3. Preheaters

Using the Shomate equation (equation 2), specific heat values were calculated, where the coefficients of the Shomate equation were taken from the NIST webbook [35]. Then, assuming electrical heaters and substituting mass of hydrogen and nitrogen as well as respective temperature difference in equation 3, the total amount of heat needed can be estimated.

$$Cp = A + B * t + C * t^2 + D * t^3 + E/t^2 \quad (2)$$

where A, B, C, D and E are the coefficients of the Shomate equation, t – the temperature of hydrogen/nitrogen after electrolyser/ASU.

$$Q = Cp * m * \Delta T \quad (3)$$

where Cp is a specific heat value of hydrogen/nitrogen, m is a mass of hydrogen/nitrogen required for ammonia formation and ΔT is a difference between the temperature inside ammonia reactor and temperature of hydrogen/nitrogen after electrolyser/ASU.

3.2.4. Ammonia reactor

Entering an adiabatic reactor with an operating temperature of 450 °C and operating pressure of 150 bar, pure hydrogen and nitrogen react to each other to form ammonia. An ammonia conversion range of 20-30 % is stated in the literature as an approximation for the end-product quality of 99.5 % [33], [34], [36].
In the case of the green ammonia plant, purities of ammonia reactants are slightly different from the conventional ammonia production. For instance, a mixture of

hydrogen and nitrogen has a smaller number of inert contaminants, which in turn may lead to the longer use of the catalyst, compared to the conventional ammonia production process.

3.2.5. Ammonia storage

At atmospheric pressure, a temperature of -33 °C is required to store ammonia as a cryogenic liquid. The literature review introduces the estimated period of storage to be 30 days [20].

3.3. Economic evaluation

An economic evaluation of the project requires a certain number of assumptions. It is vital to secure that all the costs can be correlated to the precise estimation of Net Present Value — major parameter which may give a better overall understanding of whether the project is economically feasible or not.

3.3.1. Capital cost estimation of the Base case

Equation 4 presents the scaling technique [37] which has been used coupled with [22] for capital cost estimation of the project. MVC, CAS, ammonia reactor, compression and storage were calculated assuming an annual inflation rate of 3%.

$$\frac{C_1}{C_2} = \left(\frac{S_1}{S_2}\right)^n (4),$$

where C_1 and C_2 are the respective ratios of the equipment costs, S_1 and S_2 are the sizes of a particular unit, and n-is scaling exponent assumed in this study to be 0.7.

Also, it was significant to reach uniform cost for the 2020 year since information for the cost of equipment was given in 2010 $. Equation 5 was used for this purpose,

$$A = (1 + \text{inflation rate})^k \ (5)$$

where A is an annuity coefficient and k is a target period of 10 years.

A lifetime of a typical alkaline electrolyzer is expected to be 90000 hours or 10.27 years [38]. Assuming the 20-year lifetime of the project, the replacement cost was considered in the economic assessment. The total capital cost of the green ammonia plant was a sum of all subcomponents' costs, mentioned above.

3.3.2. Annual expenses

Using conventional method of the economic evaluation of the project [39], operation and maintenance cost was assumed to be 5 % of total capital cost, rent of the land and property taxes 2 % of the total capital cost, while labor cost was chosen to be 0.005 $ million per year. For CO_2 emissions cost, it is assumed that the emissions from the Norwegian electrical grid are about 25 kg per MWh electricity since the major electricity source in Norway is hydropower with a typical electricity price for industries being 0.025 $ per kWh. Considering the price of water per ton (for purification), 10 $, and CO_2 emission price (in Norway) 56.18 $ per ton, it all provides the total annual expenses line.

3.3.3. Annual revenue

The main product of the project is ammonia with a market price of 233 $ per ton [40]. Also, O_2 from water electrolysis can be sold with the market price of around 40 $ per ton, which is used for medical needs. Thus, ammonia and oxygen are forming the revenue stream of the project.

3.3.4. Net Present Value calculations

The NPV was calculated using a discounted cashflow analysis, which is represented by equation 6.

$$NPV = \sum_{i=1}^{i} \frac{\mathrm{CF}}{(1+\mathrm{r})^{i}} \quad (6)$$

where CF, r and i, are respectively cumulative cash flow, discount rate and the lifetime of the project.
Table 2 provides all assumed values which have been used for the economic assessment.

Table 2. List of assumptions for economic calculations.

Name	Value	Units	Reference
Inflation rate	0.03	annually	[39]
Interest rate	0.1	annually	[39]
Exponent coefficient for equipment scaling	0.7	-	[39]
Electrolyser capital cost	628000	$ per MW H_2	[41]
Operating hours	8322	hours	[41]
Stack lifetime	90000	hours	[41]
Number of electrolyzer replacements during the lifetime of the project	1	-	[41]
Average currency for 2020	0.7877	$/€	Calculated
Operation & Maintenance cost	0.05 from the Capital Cost	$/year	[39]
Rent of the land	0.02 from the Capital Cost	$/year	[39]
Property taxes	0.02 from the Capital Cost	$/year	[39]
Labor cost	50000	$/year	[39]
Ammonia market price	233	$/ton	[40]

Electricity industrial price	0.025	$/kWh	[42]
Water usage cost	10	$/ton	Assumed
CO_2 emission cost	50	€	[43]
O_2 market price	40	$/ton	[44]
The emission factor of the grid	25	kg/MWh	[45]
Tax rate	0.4	-	Assumed

3.4. Sensitivity and uncertainty analysis

Mainly, there are two sources of uncertainties in the model inputs. The first group of uncertainties arises from the internationally traded items such as ammonia and oxygen market price. The second group represents the price of input parameters for the model i.e. electricity price and CO_2 emission cost. In this work, local sensitivity analysis was conducted with NPV being a target value for the base case, scenario 1 and scenario 2 [46]. The uncertain input parameters were varied by ±20% from the base values, and representative figures were obtained, which are shown in the following result section.

3.4.1. Scenario 1

In addition to ammonia and oxygen, there is one more revenue-forming element. Compared to conventional methods of ammonia production (SMR), green ammonia plants will save 7 kg of CO_2 per 1 kg produced hydrogen [10]. In this case, the Norwegian government may propose a plan with a fixed CO_2 price for an emission saving project, and the optimal price for this agreement will be discussed in the result section.

3.4.2. Scenario 2

Since ammonia produced using green electricity is not quite cost-competitive compared to the market price of conventional ammonia, it was suggested to create a separate market with a certain demand and higher selling price for the product [47]. This study estimates the minimum price at which the current project may become economically feasible.

4. Results

For a 400-ton per day green ammonia plant, roughly 71 tons of hydrogen and 329 tons of nitrogen are needed on daily basis. Assuming 8.981 kg of water per 1 kg of hydrogen and the efficiency of cryogenic separation unit being 0.79, this, in turn, leads to demand for water and air to be equal 635 ton/day and 417 ton/day respectively.

4.1. *Material and energy balance*

4.1.1. Mechanical Vapor Compression (MVC)

For 400-ton green ammonia plant, assuming adiabatic efficiency 0.75 and driver efficiency 0.95, the MVC power requirement was calculated to be 0.63 MW per day.

4.1.2. Electrolyzer

Assuming rectifier efficiency 0.9 [20], the daily power need of the electrolyzer unit equals 142.9 MW daily.

4.1.3. Cryogenic air separation

The main assumption in this section is power requirements per 1 ton of nitrogen. Coupled with adiabatic and rectifier efficiencies mentioned above, the power input for air separation is determined at 1.32 MW/day.

4.1.4. Preheaters

Specific heat values were calculated to be 29.08 J/mole K and 29.1 J/mole K for hydrogen and nitrogen respectively. Then, assuming electrical heaters and substituting mass of hydrogen and nitrogen as well as the respective temperature differences, the total amount of heat needed was estimated to be 6.89 MW per day.

4.1.5. Ammonia Reactor

Ammonia loop requires 0.026 MW energy per 1 ton of ammonia. Considering water for cooling purposes, the daily energy input will be 10.63 MW.

4.1.6. Compression and storage

Compressor power requirements were assumed to be 0.009 MW per day.

4.1.7. Total power requirements of the green ammonia plant

Summing up all energy requirements listed above, it is seen that the overall energy consumption of a 400-ton ammonia plant is 162.41 MW per day. Assuming linear relation between the size of ammonia plant versus total required power per day in Equation 7, Figure 5 shows how the energy input varies in range from 0 to 1000 ton per day capacity of a green ammonia plant, which is comparable to the results presented in the literature [48].

$$P_{NH3} = 0.406 * Capacity_{ammonia\ plant} \quad (7)$$

where P_{NH3} is a daily power requirement of a green ammonia plant.

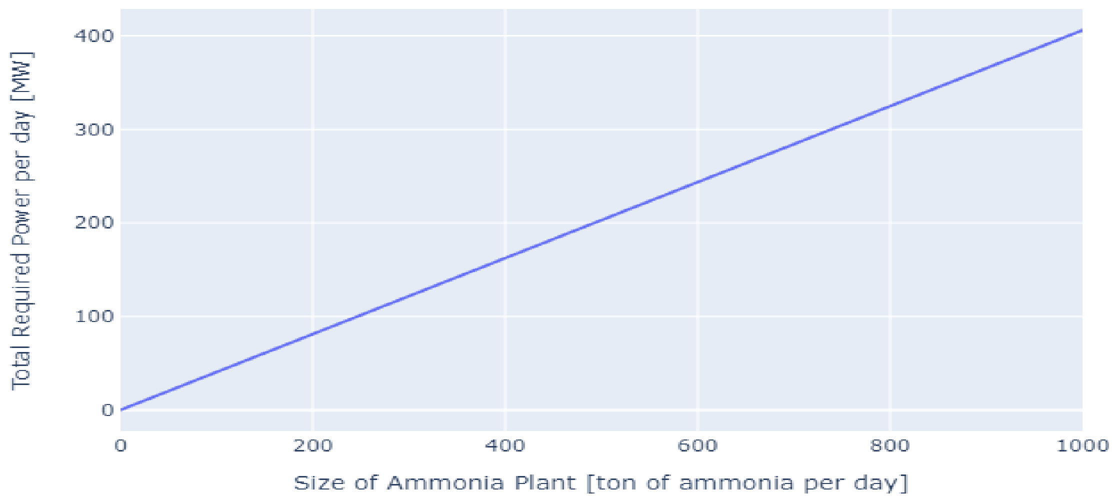

Figure 5. Power as a function of size for a green ammonia production plant.

4.2. Specific energy consumption

As Figure 6 shows, Electrolyser was determined as the most energy-consuming element in the scheme of a green ammonia production plant with an efficiency at stack 4.4 kWh/Nm^3. Roughly 94.2 % of overall specific energy consumption contributes to the electrolyzer needs for electricity, while the other subsystems as cryogenic air separation, preheaters for hydrogen and nitrogen, ammonia loop, and ammonia compression and storage require a relatively small amount of energy at the level of 5.8 % from the overall specific energy required per 1 ton of green ammonia to be produced.

According to table 3, the total specific energy needed for producing 1 ton of ammonia using electricity is 10.02 MWh/ton NH_3. Obtained value can be compared to the one from the literature [49], which represents an interval for the specific energy consumption of green ammonia plants from 10 to 12 MWh/ton NH_3.

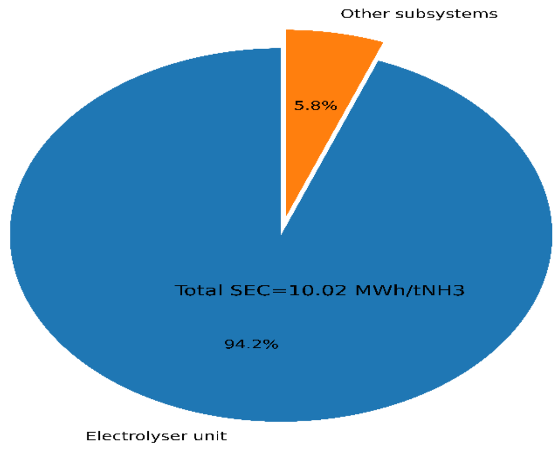

Figure 6. Specific energy consumption in percentage

Table 3. Energy & Material balance estimation. Specific Energy Consumption of the green ammonia plant.

Name	Value	Units
SEC for Hydrogen production by the electrolyzer.	53.14	MWh/ton H_2
SEC for water purification	$1*10^{-5}$	MWh/ton water
SEC for Nitrogen production by CAS	0.69	MWh/ton N_2
SEC of ammonia reaction coupled with preheaters for Hydrogen and Nitrogen and compression and storage system	$8.32*10^{-4}$	MWh/ton
Total SEC per 1 ton of ammonia	10.02	MWh/ton NH_3

4.3. Emissions

For this section, it was considered that the specific energy consumption of ammonia production by SMR is 8 MWh/ton NH_3 from which roughly 7.5 MWh is related to the SMR process and the remaining 0.5 MWh refers to other subsystems (ASU, preheaters, storage) [50]. Consequently, emissions associated with the ammonia loop in the case of SMR will be relatively higher than in the case of electrolyzer-based ammonia production. Also, the amount of CO_2 emissions using the SMR process will be comparable among the 5 chosen countries of study at the level of around 8.5-9 tCO_2/ton NH_3. Figure 7 demonstrates the comparison of associated CO_2 emissions for European countries. The respective CI-factors were used here for the calculations [45]. It is seen that in the case of Norway and Sweden there was a drastic difference in terms of associated CO_2 emissions if SMR compared to electrolyzer-based ammonia production (for instance, 0.025 tCO_2/ ton NH_3 if electrolyzer-based ammonia production technology was used and around 9 tCO_2/ton NH_3 using SMR for Norway). However, for Denmark and EU-28, due to the higher CI-factor of the grid, the ammonia production using an electrolyzer-based system was significantly higher at the level of 3.9 and 4.1 tCO_2/ton NH_3 respectively.

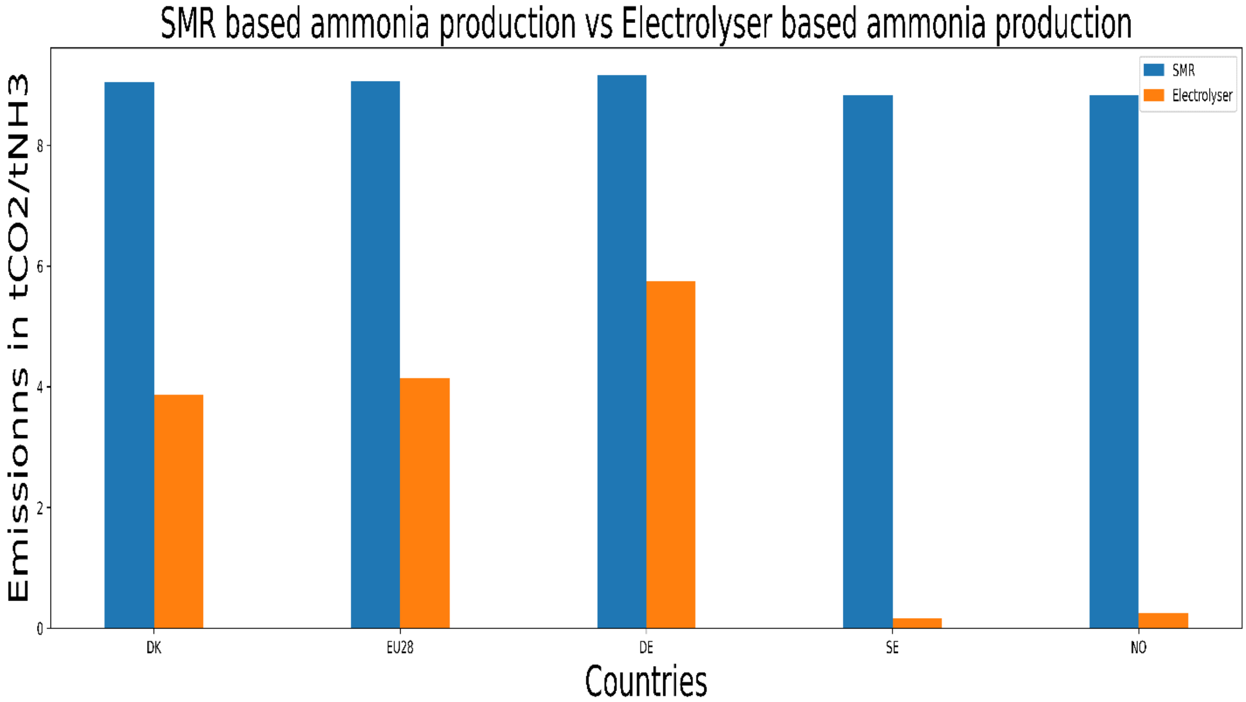

Figure 7. Emissions from the SMR (blue) and electrolyzer-based ammonia production technologies (orange) for 5 chosen European countries.

4.4. Economic evaluation

4.4.1. Capital cost estimation

The capital cost of each unit of the 400-tonn per day green ammonia production plant was estimated. Water purification unit expected to consist of compressor, driver, evaporator, 2 heat exchangers. In addition to these components, the flows of distillate and brine for the normal operational process are essential. Using standard sizing techniques [20], [51], [52] the capital cost of the MVC unit was estimated to be 13.86 million \$.The total cost of the electrolyzer subsystem, considering replacement cost at year 11 was estimated to be 78.74 million \$. The capital cost of air separation unit was estimated to be 11.23 million \$, which was calculated based on conventional design processes for unit producing in a range of 50-400 tons per day of nitrogen. For the ammonia reactor CAPEX calculations, the cost for 2 heat exchangers, 2 compressors, 2 drivers, 2 reactors, and 1 pump in total was estimated to be 14.72 million \$. The conventional cost scaling technique introduces the cost of

10.63 million $ for 30 days of storage with an overall capacity of 12000 tons of ammonia.
Table 4 represents the results from the CAPEX estimation for the 400-ton per day green ammonia plant. From table 4 it is seen that the most expensive unit is electrolyser. MVC, CAS, ammonia reactor and ammonia storage unit have comparable costs. The sum of capital costs for each unit presented above provides a total estimated cost of ammonia plant of 129.2 million $.

Table 4. CAPEX estimation of the 400-ton per day green ammonia production plant.

Name	Value	Units
MVC	13.86	million $
Electrolyser	78.74	million $
CAS	11.23	million $
Ammonia reactor	14.72	million $
Ammonia storage	10.63	million $
The total CAPEX	129.2	million $

4.4.2. Net Present Value calculations

For the base case, NPV of the green ammonia production plant was calculated to be -116.7 million $. This is in-line with the expectation because of the high capital cost since the electrolyser technologies are still too expensive. For the operational cost, electricity price is the main issue. According to basic economic decision rules, a project with NPV<0 is not economically feasible. Thus, further analysis of both scenario 1 and scenario 2 was conducted.

In the scenario 1, where the price of CO_2 savings was considered to be 100 $ per ton of CO_2, NPV increases in comparison to the base case scenario to -35.6 million $. However, even with this price, the project is still not economically effective.
In the scenario 2, the green ammonia selling price was increased from 233 $ per ton of NH_3 to 300 $ per ton of NH_3. Obtained value for NPV was higher than in the base case (-72.98 million $ compared to -116.7 million $). The NPV is still lower than 0. Thus, at the scenario 2 the 400-ton per day green ammonia production plant is not economically feasible.

4.5. Sensitivity analysis

The sensitivity analysis was conducted for the base case, scenario 1 and scenario 2. For the base case, tax rate, interest rate, ammonia price, the capacity of an ammonia plant, electricity cost, and electrolyzer efficiency were chosen as parameters of the analysis. Figure 8 illustrates the variations of NPV of the project in percentage when each of the parameters listed above varied by ±20% of their base value. The most important parameters affecting the NPV in the base case are electricity price, electrolyzer efficiency, ammonia price and capacity of the ammonia plant. Tax rate and discount rate have lower impact on the NPV. It is possible to divide the study parameters into two groups. The first one represents the effect of increasing ammonia price and capacity of an ammonia plant, which results in a considerable increase of NPV. The second group represents the impact of increasing electricity cost and electrolyzer specific energy consumption. In this case, an opposite effect on NPV can be observed.

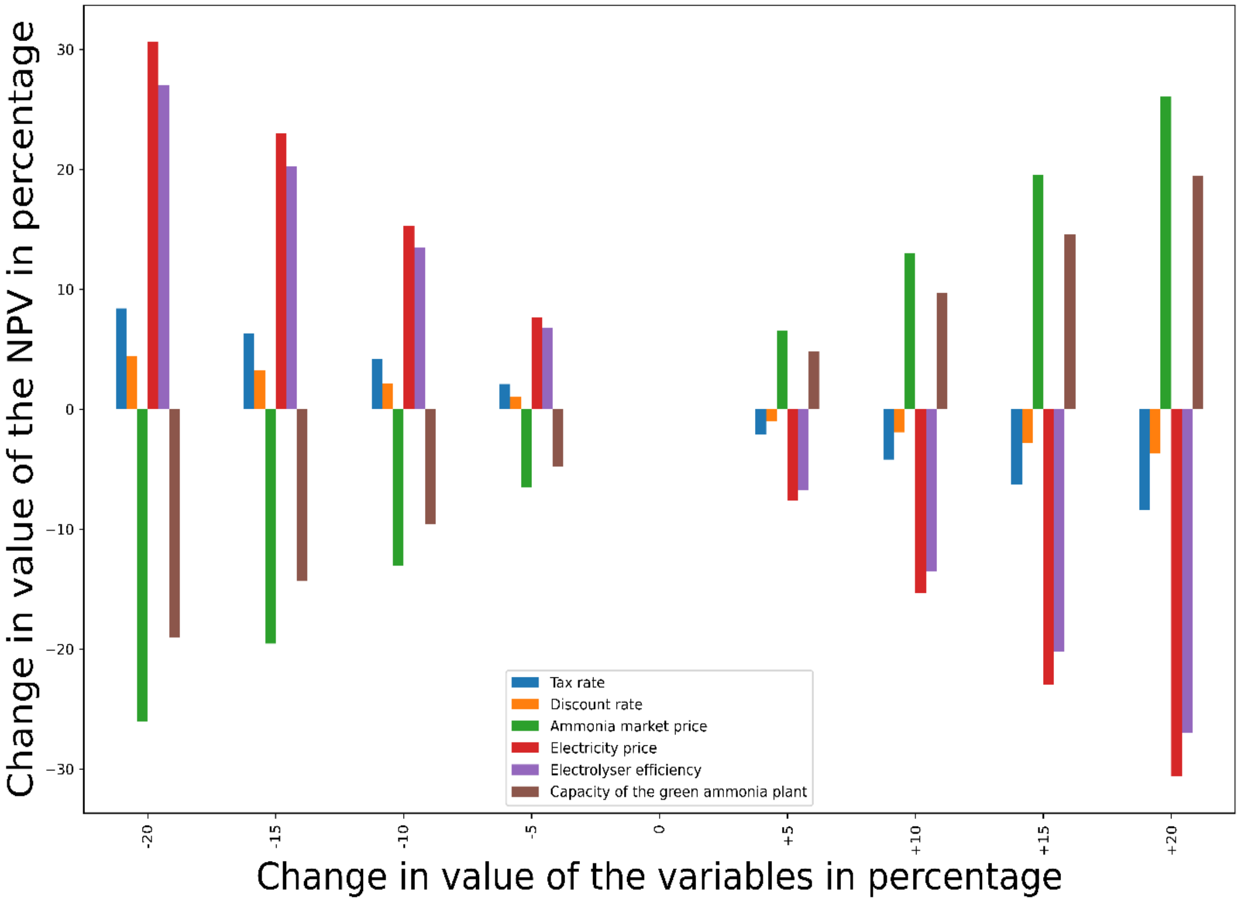

Figure 8. The results of the local sensitivity analysis for the Base case.

Figure 9 represents the results from the sensitivity analysis of the scenario 1. According to the figure 9, electricity price still plays the key role in terms of NPV fluctuations. However, in this case the ammonia selling price has more significant effect on the NPV compared to the base case. Price for emissions has higher impact on the NPV compared to tax rate and discount rate, but less than electricity price, electrolyser efficiency, ammonia selling price and capacity of the green ammonia plant. Also, tax rate in this scenario is much lower than in the base case and discount rate has considerably more effect on NPV compared to the base case sensitivity results.

Figure 10 shows the results from the sensitivity analysis of the scenario 2. Here, electricity price and ammonia selling price have comparable effect on NPV. In turn, tax rate and discount rate have much lower impact on the NPV compared to the base case.

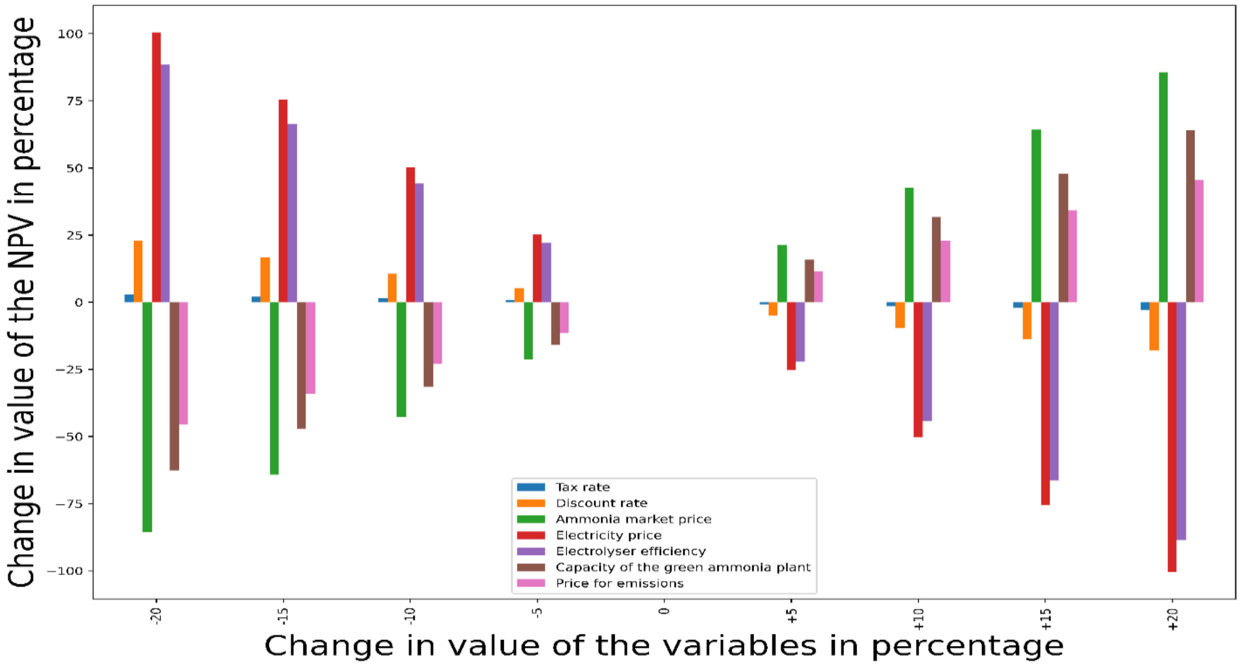

Figure 9. The results for the local sensitivity analysis of the scenario 1.

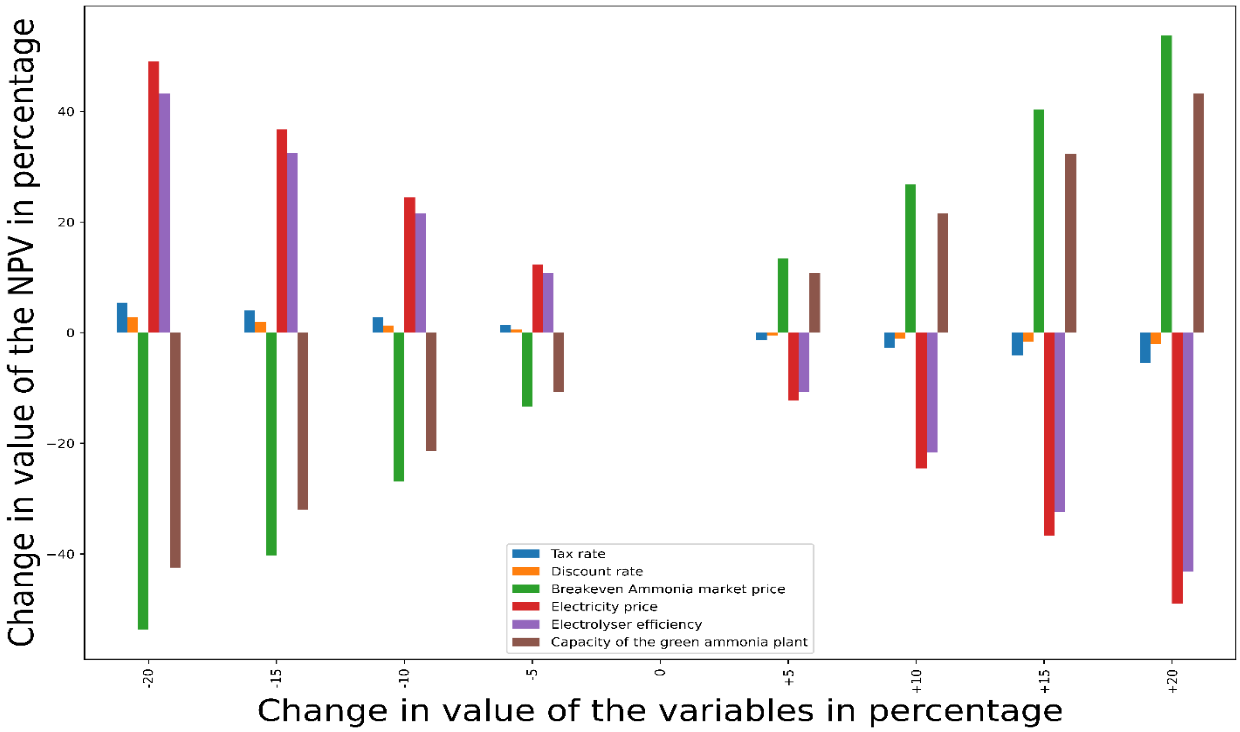

Figure 10. The results for the local sensitivity analysis of the scenario 2.

4.6. Breakeven studies of Scenario 1 and Scenario 2

In the frames of scenario 1, the breakeven point for emission price was calculated using equation 8 with a CO_2 price as an unknown parameter. According to obtained results, the CO_2 emission price 143.82 $ / ton CO_2 would result in NPV values equal to zero. Thus, the emission price higher than calculated value will result in positive NPV.

$$NPV(Scenario\ 1) = \sum_{i=1}^{i} \frac{\mathrm{CF(Scenario1\ )}}{(1+\mathrm{r})^{i}} = 0 \quad (8)$$

where NPV (scenario 1) is the NPV value at the respective cash flow of the scenario 1.

In scenario 2, the minimum selling price of the green ammonia was calculated using equation 9 with a green ammonia selling price being the target value. Thus, the selling price of green ammonia calculated to be 411.75 $ / ton NH_3 would lead to the positive economic results in the scenario 2.

$$NPV(Scenario\ 2) = \sum_{i=1}^{i} \frac{\mathrm{CF(Scenario2\ )}}{(1+\mathrm{r})^{i}} = 0 \quad (9)$$

where NPV (scenario 2) is the NPV value at the respective cash flow of the scenario 2.

5. Discussion

Specific energy consumption of the 400-ton per green ammonia plant was found to be 10.02 MWh/ton NH_3. A range of 10-12 MWh/ ton NH_3 is generally presented in the literature [49], [53], [54]. The most energy consuming element in the green ammonia production process is electrolyser unit. Further development of the electrolyser technologies (i.e. efficiency at stack) may result in the significant reduction in the overall energy consumption of the green ammonia production.

From the total capital cost estimation, it is seen that the electrolyser unit is the most expensive part of the green ammonia production plant scheme with a respective cost of 78.74 million $. Future reduction of the alkaline electrolyser cost would reduce the CAPEX of the green ammonia production projects. This would result in an essential decrease in the LCOA [55]. For the operational costs, the cost of electricity is the major issue. The analysis of the day-ahead electricity market may reduce the effect from the electricity cost on the LCOA. Also, the operational costs of the green ammonia production plant could be reduced by selecting countries with low electricity prices for further analysis.

The NPV value of the green ammonia production plant is very sensible to the changes in the electricity price, electrolyser efficiency, ammonia selling price and the capacity of the plant. Tax rate and discount rate have the minor effect on the NPV.

Norway has been selected for this study as a country with low prices on the electricity for the industrial users and low CI-factor of the Norwegian electricity grid compared to other European countries [42], [45], [56]. The alliance of Yara, Statkraft and Aker Horizons established the first large scale ammonia production plant in Porsgrunn with a capacity of 500,000-ton NH_3/year [57]. The main target of the companies is to reduce CO_2 from the shipping industry, which nowadays accounts for 2% of global GHG emissions [58]. There is another partnership of Finnish technology group Wartsila and Norwegian Grieg Edge which received funding support by the Norwegian government to build the world's first ammonia-fueled tanker [59]. Thus, the green ammonia production will play a key role in the decarbonization of the shipping, chemical and petrochemical industries in Norway.

In turn, among the countries in Central Asian region, Uzbekistan seems to be a foreseeable location to put green ammonia production plant. Masdar being one of the world's leading renewable energy company signed an agreement with the Government of the Republic of Uzbekistan to extend the capacity of the Zarafshan wind farm project from previously stated 500 MW to 1.5 GW [60]. Zarafshon district with the total mean capacity factor for wind resources of 34.8 % and capacity of the wind farm of 500 MW would provide electricity to 500,000 homes and would displace 1.1 million tons of CO_2 [60], [61]. Also, Uzbekistan has approved a presidential decree to establish a strategy to boost development of renewable and hydrogen energy [62]. Considering recent development of the ammonia and urea chemical complex at the Navoiazot plant with a capacity of ammonia production of 660,000 tons per year, Uzbekistan could be chosen for the techno-economic assessment of the viability of green ammonia production in the near future [63].

6. Conclusion

A techno-economic evaluation was conducted to access the feasibility of the electrolyser-based ammonia production. The study was conducted under three economic scenarios, and results were compared. In the base case scenario, the NPV of the project was found to be -116.7 million $. In scenario 1, where an emission saving price of 100 $/ton CO_2 was introduced in addition to the annual revenue of the base case, NPV was calculated to be -35.6 million $. Scenario 2, where the green ammonia selling price was increased from 233 $/ ton NH_3 (base case value) to 300 $/ ton NH_3, the NPV of -72.98 million $ was presented.

The minimum CO_2 saving price was established in the current master thesis work to be 143 $/ ton CO_2. This is in-line recent news from the Norwegian government to increase CO_2 emission price up to 200 €/ton CO_2 by 2030 [64]. The minimum green ammonia selling price of 411.75 $/ ton NH_3 presented in this paper is comparable to the LCOA was found in the literature with a respective range of 400-1000 $/ ton NH_3 [15], [65], [66], [67]. All the obtained results in this work have been proposed a list of main conclusions:

1. Electrolyser-based ammonia production has a much lower carbon footprint compared to SMR;
2. SEC of the green ammonia production process is relatively higher than the one of SMR (due to continuous development of the electrolyser technologies which are currently the most energy-consuming part of the equipment);
3. The economic evaluation represented in this paper as the NPV analysis has shown that the green ammonia production can hardly compete with the conventional production technologies;
4. It is essential to put green ammonia power plants to the regions with low electricity prices since the electricity price has the most significant impact on the NPV of the project;
5. Scenario 1 of the paper seems to be the most desirable option to be considered to obtain a positive NPV value (when the prices on CO_2 emission will be increased in the near future);
6. In Norway, governmental support for emission-saving projects can make a 400-ton green ammonia production plant viable for potential investors;
7. The green ammonia market with much higher prices on green NH_3 would have a positive effect on the project economy.

Appendix.

The appendix contains the Jupyter Notebook required for obtaining the values and graphs related to the current master thesis work. The files listed below are ordered as they had been created for the research purposes.

To run the program, all files must be implemented into Jupyter Notebook followed the respective line order below.

Line 1:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import plotly.express as px
import numpy_financial as npf
from sklearn.linear_model import LinearRegression
import scipy.optimize as opt
from scipy import optimize
from scipy.optimize import differential_evolution
from scipy import interpolate
import plotly.graph_objects as go
from highlight_text.htext import htext, fig_htext
def money_format(money):
   return '$ {:,.2f}'.format(money)
def number_format(number):
   return '{:,.4f}'.format(number)
def power_format(power):
   return '{:,.2f} MW'.format(power)
from IPython.display import IFrame
import SALib
# Part 1: Input energy requirements for 400-ton per day green ammonia production plant.
#Setting input parameters.
#Molecular weights of components and conversion constants.
```

Line 2:

```
mol_weight_h2 = 1.00794 #g/mol
mol_weight_n2 = 14.0067 #g/mol
mol_weight_o2 = 15.999 #g/mol
mol_weight_h20 = 18.01488 #g/mol
mol_wt_nh3 = 17.03052 #g/mol
conversion1 = 0.058718113 #Mol in 1 g NH3
conversion2 = 0.08281 #from kg to Nm3
conversion3 = 0.055509668 #mol per 1 kg H2O
electrolyser_efficiency = 3.8
#Parameters for Cp calculations.
```

```
# Hydrogen
T2 = 450 # temperature of Ammonia Reactor (fixed for standard process)
A_h = 33.066178
B_h = -11.363417
C_h = 11.432816
D_h = -2.772874
E_h = -0.158558
F_h = -9.980797
G_h = 172.707974
H_h = 0
T1_h = 80 # Hydrogen temperature after electrolyzer (range from 60 to 80 Degree Celcius)
t_h = (T1_h + 273.15)/1000;
### Nitrogen
A_n = 28.98641
B_n = 1.853978
C_n = -9.647459
D_n = 16.63537
E_n = 0.000117
F_n = -8.671914
G_n = 226.4168
H_n = 0
T1_n = -172# Nitrogen temperature after Cryogenic Air Separation (fixed for standard process)
t_n = (T1_n + 273.15)/1000;
```

Line 3:

```
# "Green ammonia" function.
def greenammonia(capacity):
## Determination of Hydrogen and Nitrogen demand.
    h2_demand = capacity*(3*mol_weight_h2*conversion1) # ton/day
    n2_demand = capacity*(mol_weight_n2*conversion1) # ton/day
    h2_demand_kgh = (h2_demand/24)*1000 # kg/h
    n2_demand_kgh = (n2_demand/24)*1000 # kg/h
    h2_demand_Nm3_per_h = h2_demand_kgh/conversion2 #Nm3/h
    print (h2_demand)
    print (n2_demand)
## Electrolyser energy requirements
     #KWh/Nm3
    electrolyser_capacity_MW = (h2_demand_Nm3_per_h*electrolyser_efficiency)/1000
    rectifier_efficiency = 0.95
    total_electrolyser_power = electrolyser_capacity_MW/rectifier_efficiency
## 1.2.3: Amount of water for purification (Mechanical Vapor Compression). Energy
requirements of water purification unit.
    mass_water_el = h2_demand/(mol_weight_h2*conversion3*2) # ton/day
    mass_o2 = mol_weight_o2*conversion3*mass_water_el #ton/day
    adiabatic_efficiency = 0.75
```

```
    driver_efficiency = 0.95
    power_per_1_ton = 0.71 #kW
    kg_hydrogen = 8.981 # 1kg hydrogen = 8.981 #kg water
    p_water =
power_per_1_ton*mass_water_el/(adiabatic_efficiency*driver_efficiency*1000) #MW
## Amount of air for separation (Cryogenic). Energy requirements of air separation unit.
    n2_eff = 0.79
    m_air = n2_demand/n2_eff #ton
    power_ton_n2 = 2.25 #kW/ton
    power_air_separation =
(m_air*power_ton_n2/(adiabatic_efficiency*driver_efficiency))/1000 #MW
## Heaters for Hydrogen and Nitrogen before entering the ammonia reactor.
    ## Heater for Hydrogen.
#Cp calculation for Hydrogen
    Cp_h = A_h + B_h*t_h + C_h*t_h**2 + D_h*t_h**3 + E_h/t_h**2  #J/mol*K
    Q_h = ((h2_demand*1000)/(24*3600)*Cp_h*500*(T2 - T1_h))/10**6 #MW
    ## Heater for Nitrogen.
    Cp_n = A_n + B_n*t_n + C_n*t_n**2 + D_n*t_n**3 + E_n/t_n**2 #J/mol*K
    Q_n = ((n2_demand*1000)/(24*3600)*Cp_n*1000/28*(T2 - T1_n))/10**6 #MW
    total_heaters_power = Q_h + Q_n
## : Ammonia Reactor energy requirements.
    P_per_ammonia = 0.02633 # per ton
    P_per_day=P_per_ammonia*capacity
    M_water_per_minute = 9.5 #ton/minute
    P_water_pumping = 7.3*10**-6 #MW
    P_water_total = M_water_per_minute*60*24*P_water_pumping
    P_reactor = P_per_day + P_water_total
## Ammonia compression and storage.
    P_compressor = 0.0091
   ## "Green Hydrogen" SEC
    h2_cost_green = electrolyser_efficiency/conversion2
    ## Hydrogen cost using Steam Methane Reforming and associated amount of CO2.
    h2_cost_smr = 4.5 * conversion2 #kg
    CO2_per_1_kg_h2 = 7 #kg
    CO2_smr = CO2_per_1_kg_h2*h2_demand*1000 #kg/day
## Total power of Green Ammonia Plant per day.
    total_power = (total_electrolyser_power + p_water + power_air_separation +
total_heaters_power + P_reactor + P_compressor)
    return(total_power)
p = greenammonia(400)
print('Total power of Ammonia Plant:', power_format(p))
```

Line 4:

```
#Power as a function of Size.
i_array = np.arange(start=0, stop=1050, step=50);
```

```
array_length = i_array.shape[0];
a_array = np.zeros(i_array.shape);
for idx in range(array_length):
   i = i_array[idx];
   a_array[idx] = greenammonia(i);
fig1 = px.line(x=i_array, y=a_array, labels={'x':'Size of Ammonia Plant [ton of ammonia per day]','y':'Total Required Power per day [MW]'})
fig1.show()
```

Line 5:

```
# Part 2: Economic evaluation of the green ammonia plant.
#Setting input parameters.
def cost_estimation(C1, S2, S1, exponent):
      return C1*(S2/S1)**exponent
capacity = 300
h2_demand_1 = capacity*(3*mol_weight_h2*conversion1) # ton/day
n2_demand_1 = capacity*(mol_weight_n2*conversion1) # ton/day
mass_water_el_1 = h2_demand_1/(mol_weight_h2*conversion3*2) # ton/day
mass_o2_1 = mol_weight_o2*conversion3*mass_water_el_1 #ton/day
inflation_rate = 0.03
interest_rate = 0.1
Capacity_new = input("Enter new capacity of ammonia plant: ");
capacity_new = float(Capacity_new)
exponent = 0.7 # commonly used for Chemical Industry
## Please type '400' here.
```

Line 6:

```
# MVC capital cost.
# Compressors
compressor_mvc_capacity = 0.676 * mass_water_el_1
compressor_mvc_inst_cost_2020 = 984.3 * (1+inflation_rate)**10*mass_water_el_1
compressor_mvc_uninst_cost_2020 = 328.1 * (1+inflation_rate)**10*mass_water_el_1
print('Total compressors (MVC) Estimated Capital Cost is:', money_format(cost_estimation(compressor_mvc_inst_cost_2020, capacity_new, capacity, exponent)))
# Drivers
driver_mvc_capacity = 0.95 * mass_water_el_1
driver_mvc_inst_cost_2020 = 307.9 * (1+inflation_rate)**10*mass_water_el_1
driver_mvc_uninst_cost_2020 = 205.26 * (1+inflation_rate)**10*mass_water_el_1
print('Total drivers (MVC) Estimated Capital Cost is:', money_format(cost_estimation(driver_mvc_inst_cost_2020, capacity_new, capacity, exponent)))
# Evaporators.
evaporator_mvc_capacity = 5.7 * mass_water_el_1
evaporator_mvc_inst_cost_2020 = 7411.34 * (1+inflation_rate)**10*mass_water_el_1
```

```
evaporator_mvc_uninst_cost_2020 = 2470.44 * (1+inflation_rate)**10*mass_water_el_1
print('Total evaporator (MVC) Estimated Capital Cost is:',
money_format(cost_estimation(evaporator_mvc_inst_cost_2020, capacity_new, capacity,
exponent)))
# Distillate.
distillate_mvc_capacity = 0.496 * mass_water_el_1
distillate_mvc_inst_cost_2020 = 2729.62 * (1+inflation_rate)**10*mass_water_el_1
distillate_mvc_uninst_cost_2020 = 909.88 * (1+inflation_rate)**10*mass_water_el_1
print('Total distillate (MVC) Estimated Capital Cost is:',
money_format(cost_estimation(distillate_mvc_inst_cost_2020, capacity_new, capacity,
exponent)))
# Brine.
brine_mvc_capacity = 1.724 * mass_water_el_1
brine_mvc_inst_cost_2020 = 6286.42 * (1+inflation_rate)**10*mass_water_el_1
brine_mvc_uninst_cost_2020 = 2095.48 * (1+inflation_rate)**10*mass_water_el_1
print('Total brine (MVC) Estimated Capital Cost is:',
money_format(cost_estimation(brine_mvc_inst_cost_2020, capacity_new, capacity,
exponent)))
# Capital cost of MVC unit.
total_mvc_inst_cost = compressor_mvc_inst_cost_2020 + driver_mvc_inst_cost_2020 +
evaporator_mvc_inst_cost_2020 + distillate_mvc_inst_cost_2020 +
brine_mvc_inst_cost_2020
total_mvc_uninst_cost = compressor_mvc_uninst_cost_2020 +
driver_mvc_uninst_cost_2020 + evaporator_mvc_uninst_cost_2020 +
distillate_mvc_uninst_cost_2020 + brine_mvc_uninst_cost_2020
print('Total MVC Estimated Capital Cost is:',
money_format(cost_estimation(total_mvc_inst_cost, capacity_new, capacity, exponent)))
```

Line 7:

```
# CAS capital cost.
# Heat exchangers.
heat_exchanger_asep_1_capacity = 0.48528 * n2_demand_1
heat_exchanger_asep_1_inst_cost_2020 = 332.8 * (1+inflation_rate)**10 * n2_demand_1
heat_exchanger_asep_1_uninst_cost_2020 = 153.36 * (1+inflation_rate)**10 *
n2_demand_1

heat_exchanger_asep_2_capacity = 0.51024 * n2_demand_1
heat_exchanger_asep_2_inst_cost_2020 = 340.96 * (1+inflation_rate)**10 * n2_demand_1
heat_exchanger_asep_2_uninst_cost_2020 = 153.12 * (1+inflation_rate)**10 *
n2_demand_1

heat_exchanger_asep_3_capacity = 0.75276 * n2_demand_1
heat_exchanger_asep_3_inst_cost_2020 = 421.4 * (1+inflation_rate)**10 * n2_demand_1
heat_exchanger_asep_3_uninst_cost_2020 = 194.2 * (1+inflation_rate)**10 * n2_demand_1

heat_exchanger_asep_4_capacity = 11.42276 * n2_demand_1
```

```
heat_exchanger_asep_4_inst_cost_2020 = 5715.52 * (1+inflation_rate)**10 * n2_demand_1
heat_exchanger_asep_4_uninst_cost_2020 = 2059.64 * (1+inflation_rate)**10 *
n2_demand_1

total_heat_exchangers_asep_inst_cost = heat_exchanger_asep_1_inst_cost_2020 +
heat_exchanger_asep_2_inst_cost_2020 + heat_exchanger_asep_3_inst_cost_2020 +
heat_exchanger_asep_4_inst_cost_2020
total_heat_exchangers_asep_uninst_cost = heat_exchanger_asep_1_uninst_cost_2020 +
heat_exchanger_asep_2_uninst_cost_2020 + heat_exchanger_asep_3_uninst_cost_2020 +
heat_exchanger_asep_4_uninst_cost_2020
print('Total heat exchangers (Air Separation) Estimated Capital Cost is:',
money_format(cost_estimation(total_heat_exchangers_asep_inst_cost, capacity_new,
capacity, exponent)))
# Compressors.
compressor_asep_1_capacity = 1.516 * n2_demand_1
compressor_asep_1_inst_cost_2020 = 1987.92 * (1+inflation_rate)**10 * n2_demand_1
compressor_asep_1_uninst_cost_2020 = 722.88 * (1+inflation_rate)**10 * n2_demand_1

compressor_asep_2_capacity = 1.4492 * n2_demand_1
compressor_asep_2_inst_cost_2020 = 1914.76 * (1+inflation_rate)**10 * n2_demand_1
compressor_asep_2_uninst_cost_2020 = 698.28 * (1+inflation_rate)**10 * n2_demand_1

compressor_asep_3_capacity = 1.654 * n2_demand_1
compressor_asep_3_inst_cost_2020 = 2136.32 * (1+inflation_rate)**10 * n2_demand_1
compressor_asep_3_uninst_cost_2020 = 776.84 * (1+inflation_rate)**10 * n2_demand_1

total_compressors_asep_inst_cost = compressor_asep_1_inst_cost_2020 +
compressor_asep_2_inst_cost_2020 + compressor_asep_3_inst_cost_2020
total_compressors_asep_uninst_cost = compressor_asep_1_uninst_cost_2020 +
compressor_asep_2_uninst_cost_2020 + compressor_asep_3_uninst_cost_2020
print('Total compressors (Air Separation) Estimated Capital Cost is:',
money_format(cost_estimation(total_compressors_asep_inst_cost, capacity_new, capacity,
exponent)))
# Drivers.
driver_asep_1_capacity = 2.0212 * n2_demand_1
driver_asep_1_inst_cost_2020 = 630.32 * (1+inflation_rate)**10 * n2_demand_1
driver_asep_1_uninst_cost_2020 = 420.2 * (1+inflation_rate)**10 * n2_demand_1

driver_asep_2_capacity = 1.9324 * n2_demand_1
driver_asep_2_inst_cost_2020 = 620.16 * (1+inflation_rate)**10 * n2_demand_1
driver_asep_2_uninst_cost_2020 = 413.44 * (1+inflation_rate)**10 * n2_demand_1

driver_asep_3_capacity = 2.2052 * n2_demand_1
driver_asep_3_inst_cost_2020 = 650 * (1+inflation_rate)**10 * n2_demand_1
driver_asep_3_uninst_cost_2020 = 433.32 * (1+inflation_rate)**10 * n2_demand_1
```

```
total_drivers_asep_inst_cost = driver_asep_1_inst_cost_2020 +
driver_asep_2_inst_cost_2020 + driver_asep_3_inst_cost_2020
total_drivers_asep_uninst_cost = driver_asep_1_uninst_cost_2020 +
driver_asep_2_uninst_cost_2020 + driver_asep_3_uninst_cost_2020
print('Total drivers (Air Separation) Estimated Capital Cost is:',
money_format(cost_estimation(total_drivers_asep_inst_cost, capacity_new, capacity,
exponent)))
# Towers.
tower_asep_1_capacity = 0.5396 * n2_demand_1
tower_asep_1_inst_cost_2020 = 6634.68 * (1+inflation_rate)**10 * n2_demand_1
tower_asep_1_uninst_cost_2020 = 483.2 * (1+inflation_rate)**10 * n2_demand_1

tower_asep_2_capacity = 0.7012 * n2_demand_1
tower_asep_2_inst_cost_2020 = 5040.2 * (1+inflation_rate)**10 * n2_demand_1
tower_asep_2_uninst_cost_2020 = 614.8 * (1+inflation_rate)**10 * n2_demand_1

tower_asep_3_capacity = 0.09 * n2_demand_1
tower_asep_3_inst_cost_2020 = 908.68 * (1+inflation_rate)**10 * n2_demand_1
tower_asep_3_uninst_cost_2020 = 110.84 * (1+inflation_rate)**10 * n2_demand_1

total_towers_asep_inst_cost = tower_asep_1_inst_cost_2020 +
tower_asep_2_inst_cost_2020 + tower_asep_3_inst_cost_2020
total_towers_asep_uninst_cost = tower_asep_1_uninst_cost_2020 +
tower_asep_2_uninst_cost_2020 + tower_asep_3_uninst_cost_2020
print('Total towers (Air Separation) Estimated Capital Cost is:',
money_format(cost_estimation(total_towers_asep_inst_cost, capacity_new, capacity,
exponent)))
# Turbines.
turbine_asep_capacity = 0.1 * n2_demand_1
turbine_asep_inst_cost_2020 = 356 * (1+inflation_rate)**10 * n2_demand_1
turbine_asep_uninst_cost_2020 = 58.36 * (1+inflation_rate)**10 * n2_demand_1

total_turbine_asep_inst_cost = turbine_asep_inst_cost_2020
total_turbine_asep_uninst_cost = turbine_asep_uninst_cost_2020
print('Total turbine (Air Separation) Estimated Capital Cost is:',
money_format(cost_estimation(total_turbine_asep_inst_cost, capacity_new, capacity,
exponent)))
# CAS capital cost.
total_asep_cost = total_heat_exchangers_asep_inst_cost +
total_compressors_asep_inst_cost + total_drivers_asep_inst_cost +
total_towers_asep_inst_cost + total_turbine_asep_inst_cost
print('Total Air Separation unit Estimated Capital Cost is:',
money_format(cost_estimation(total_asep_cost, capacity_new, capacity, exponent)))
```

Line 8:

```
# Electrolyser capital cost.
h2_demand = capacity_new*(3*mol_weight_h2*conversion1) # ton/day
lhv_h2 = 120.1 #MJ/kg
h2_per_second_kg = h2_demand*1000/86400
h2_capacity_MW = h2_per_second_kg * lhv_h2
h2_investment_MW_h2 = 628000 # $ per MW H2
electrolyser_cost = h2_investment_MW_h2 * h2_capacity_MW
operating_hours = 365*24*0.95
stack_lifetime = 90000 # hours
stack_replacement_year = stack_lifetime/operating_hours
stack_replacement_number = 1
replacement_cost = electrolyser_cost*0.6*0.45*np.round(stack_replacement_number)
total_electrolyser_cost = electrolyser_cost + replacement_cost
print('Total Electrolyser unit Estimated Capital Cost is:',
money_format(total_electrolyser_cost))
```

Line 9:

```
# Ammonia loop capital cost.
# Heat exchangers.
heat_exchanger_ar_1_capacity = 5.6333 * capacity
heat_exchanger_ar_1_inst_cost_2020 = 10.055333 * (1+inflation_rate)**10 * capacity

heat_exchanger_ar_2_capacity = 0.095 * capacity
heat_exchanger_ar_2_inst_cost_2020 = 732.0333 * (1+inflation_rate)**10 * capacity

heat_exchanger_ar_3_capacity = 1.10733 * capacity
heat_exchanger_ar_3_inst_cost_2020 = 2.035066 * (1+inflation_rate)**10 * capacity

heat_exchanger_ar_4_capacity = 0.65833 * capacity
heat_exchanger_ar_4_inst_cost_2020 = 1425.9 * (1+inflation_rate)**10 * capacity

heat_exchanger_ar_5_capacity = 0.58933 * capacity
heat_exchanger_ar_5_inst_cost_2020 = 1334.833 * (1+inflation_rate)**10 * capacity

total_heat_exchanger_ar_cost = heat_exchanger_ar_1_inst_cost_2020 +
heat_exchanger_ar_2_inst_cost_2020 + heat_exchanger_ar_4_inst_cost_2020 +
heat_exchanger_ar_4_inst_cost_2020
print('Total heat exchangers (Ammonia Loop) Estimated Capital Cost is:',
money_format(cost_estimation(total_heat_exchanger_ar_cost, capacity_new, capacity,
exponent)))
# Compressors.
compressor_ar_1_capacity = 0.0034 * capacity
compressor_ar_1_inst_cost_2020 = 7387.33 * (1+inflation_rate)**10 * capacity

compressor_ar_2_capacity = 5*0.001733 * capacity
```

```
compressor_ar_2_inst_cost_2020 = 4734 * (1+inflation_rate)**10 * capacity

total_compressor_ar_cost = compressor_ar_1_inst_cost_2020 +
compressor_ar_2_inst_cost_2020
print('Total compressors (Ammonia Loop) Estimated Capital Cost is:',
money_format(cost_estimation(total_compressor_ar_cost, capacity_new, capacity,
exponent)))
# Drivers.
driver_ar_1_capacity = 0.005133 * capacity
driver_ar_1_inst_cost_2020 = 943.367 * (1+inflation_rate)**10 * capacity

driver_ar_2_capacity = 5*0.0026 * capacity
driver_ar_2_inst_cost_2020 = 702.4 * (1+inflation_rate)**10 * capacity

total_driver_ar_cost = driver_ar_1_inst_cost_2020 + driver_ar_2_inst_cost_2020
print('Total drivers (Ammonia Loop) Estimated Capital Cost is:',
money_format(cost_estimation(total_driver_ar_cost, capacity_new, capacity, exponent)))
# Reactors.
reactor_ar_1_capacity = 0.02513 * capacity
reactor_ar_1_inst_cost_2020 = 5623 * (1+inflation_rate)**10 * capacity

reactor_ar_2_capacity = 0.02033 * capacity
reactor_ar_2_inst_cost_2020 = 4926.33 * (1+inflation_rate)**10 * capacity

total_reactor_ar_cost = reactor_ar_1_inst_cost_2020 + reactor_ar_2_inst_cost_2020
print('Total reactors (Ammonia Loop) Estimated Capital Cost is:',
money_format(cost_estimation(total_reactor_ar_cost, capacity_new, capacity, exponent)))
# Pumps.
pump_ar_capacity = 0.3733 * capacity
pump_ar_inst_cost_2020 = 1939.46 * (1+inflation_rate)**10 * capacity

total_pump_ar_cost = pump_ar_inst_cost_2020
print('Total pump (Ammonia Loop) Estimated Capital Cost is:',
money_format(cost_estimation(total_pump_ar_cost, capacity_new, capacity, exponent)))
# Capital cost of ammonia loop.
total_ar_cost = total_heat_exchanger_ar_cost + total_compressor_ar_cost +
total_driver_ar_cost + total_reactor_ar_cost + total_pump_ar_cost
print('Total  Ammonia Loop Estimated Capital Cost is:',
money_format(cost_estimation(total_ar_cost, capacity_new, capacity, exponent)))
```

Line 10:

```
# Ammonia storage capital cost.
storage_as_capacity = capacity*30 ## for 30 days
storage_as_inst_cost_2020 = 718.89*storage_as_capacity*(1+inflation_rate)**10
```

```
total_as_cost = storage_as_inst_cost_2020
print('Total  Ammonia Storage Estimated Capital Cost is:',
money_format(cost_estimation(total_as_cost, capacity_new, capacity, exponent)))
```

Line 11:

```
# Total capital cost of the green ammonia plant.
total_capital_cost = (total_mvc_inst_cost + total_asep_cost + total_ar_cost + total_as_cost)
total_estimated_capital_cost_2020 = cost_estimation(total_capital_cost, capacity_new,
capacity, exponent) + total_electrolyser_cost
print('Total Capital Cost of the Ammonia Plant:',
money_format(total_estimated_capital_cost_2020))
```

Line 12:

```
# Net present value calculations.
# Input parameters.
h2_demand_2 = capacity_new*(3*mol_weight_h2*conversion1) # ton/day
n2_demand_2 = capacity_new*(mol_weight_n2*conversion1) # ton/day
mass_water_el_2 = h2_demand_2/(mol_weight_h2*conversion3*2) # ton/day
mass_o2_2 = mol_weight_o2*conversion3*mass_water_el_2 #ton/day
average_currency_2020 = (0.900264 + 0.915870 + 0.903361 +0.919940 + 0.915750 +
0.888883 + 0.874205 + 0.845664 + 0.845587)/9
labor_cost = 50000  ## $ per year
ammonia_market_price_2019 = 233 ## $ per ton (Yara)
electricity_price = 0.025/average_currency_2020 ## $ per kWh
water_cost = 10 ## $ per ton
O2_market_price = 40 # $ per ton
O2_cost = mass_o2_2 * O2_market_price*365
electrolyser_efficiency = 3.8
price_for_emmisions = 50
benefit_from_CO2_savings = 7*h2_demand*365*price_for_emmisions
CO2_cost = 56.183114002779696 ## $ per ton CO2
CO2_emmisions_from_electricity_grid_norway = 0.020 * 1000
exponent = 0.7
tax_rate = 0.4
plant_life = 20
DiscountRate = 0.1
inflation_rate = 0.03
print(average_currency_2020)
```

Line 13:

```
# Base case scenario.
def npv_base(tax_rate, DiscountRate, ammonia_market_price_2019, electricity_price,
electrolyser_efficiency, capacity_new):
    def greenammonia(capacity):
        h2_demand = capacity*(3*mol_weight_h2*conversion1)
```

```
        n2_demand = capacity*(mol_weight_n2*conversion1)
        h2_demand_kgh = (h2_demand/24)*1000
        n2_demand_kgh = (n2_demand/24)*1000
        h2_demand_Nm3_per_h = h2_demand_kgh/conversion2
        electrolyser_capacity_MW = (h2_demand_Nm3_per_h*electrolyser_efficiency)/1000
        rectifier_efficiency = 0.95
        total_electrolyser_power = electrolyser_capacity_MW/rectifier_efficiency
        mass_water_el = h2_demand/(mol_weight_h2*conversion3*2)
        mass_o2 = mol_weight_o2*conversion3*mass_water_el
        adiabatic_efficiency = 0.75
        driver_efficiency = 0.95
        power_per_1_ton = 0.71
        kg_hydrogen = 8.981
        p_water =
power_per_1_ton*mass_water_el/(adiabatic_efficiency*driver_efficiency*1000)
        n2_eff = 0.79
        m_air = n2_demand/n2_eff
        power_ton_n2 = 2.25
        power_air_separation =
(m_air*power_ton_n2/(adiabatic_efficiency*driver_efficiency))/1000

        Cp_h = A_h + B_h*t_h + C_h*t_h**2 + D_h*t_h**3 + E_h/t_h**2
        Q_h = ((h2_demand*1000)/(24*3600)*Cp_h*500*(T2 - T1_h))/10**6

        Cp_n = A_n + B_n*t_n + C_n*t_n**2 + D_n*t_n**3 + E_n/t_n**2
        Q_n = ((n2_demand*1000)/(24*3600)*Cp_n*1000/28*(T2 - T1_n))/10**6
        total_heaters_power = Q_h + Q_n

        P_per_ammonia = 0.02633
        P_per_day=P_per_ammonia*capacity
        M_water_per_minute = 9.5
        P_water_pumping = 7.3*10**-6
        P_water_total = M_water_per_minute*60*24*P_water_pumping
        P_reactor = P_per_day + P_water_total
        P_compressor = 0.006827
        h2_cost_green = electrolyser_efficiency/conversion2
        h2_cost_smr = 4.5 * conversion2
        CO2_per_1_kg_h2 = 7
        CO2_smr = CO2_per_1_kg_h2*h2_demand*1000
        total_power = (total_electrolyser_power + p_water + power_air_separation +
total_heaters_power + P_reactor + P_compressor)
        return(total_power)
    p = greenammonia(400)

    ## 1. Total Capital Cost
```

```
capacity = 300
capaciity_new  = 400

compressor_mvc_inst_cost_2020 = 984.3 * (1+inflation_rate)**10*mass_water_el_1
driver_mvc_inst_cost_2020 = 307.9 * (1+inflation_rate)**10*mass_water_el_1
evaporator_mvc_inst_cost_2020 = 7411.34 * (1+inflation_rate)**10*mass_water_el_1
distillate_mvc_inst_cost_2020 = 2729.62 * (1+inflation_rate)**10*mass_water_el_1
brine_mvc_inst_cost_2020 = 6286.42 * (1+inflation_rate)**10*mass_water_el_1
total_mvc_inst_cost = compressor_mvc_inst_cost_2020 + driver_mvc_inst_cost_2020 + evaporator_mvc_inst_cost_2020 + distillate_mvc_inst_cost_2020 + brine_mvc_inst_cost_2020

heat_exchanger_asep_1_inst_cost_2020 = 332.8 * (1+inflation_rate)**10 * n2_demand_1
heat_exchanger_asep_2_inst_cost_2020 = 340.96 * (1+inflation_rate)**10 * n2_demand_1
heat_exchanger_asep_3_inst_cost_2020 = 421.4 * (1+inflation_rate)**10 * n2_demand_1
heat_exchanger_asep_4_inst_cost_2020 = 5715.52 * (1+inflation_rate)**10 * n2_demand_1
total_heat_exchangers_asep_inst_cost = heat_exchanger_asep_1_inst_cost_2020 + heat_exchanger_asep_2_inst_cost_2020 + heat_exchanger_asep_3_inst_cost_2020 + heat_exchanger_asep_4_inst_cost_2020

compressor_asep_1_inst_cost_2020 = 1987.92 * (1+inflation_rate)**10 * n2_demand_1
compressor_asep_2_inst_cost_2020 = 1914.76 * (1+inflation_rate)**10 * n2_demand_1
compressor_asep_3_inst_cost_2020 = 2136.32 * (1+inflation_rate)**10 * n2_demand_1
total_compressors_asep_inst_cost = compressor_asep_1_inst_cost_2020 + compressor_asep_2_inst_cost_2020 + compressor_asep_3_inst_cost_2020

driver_asep_1_inst_cost_2020 = 630.32 * (1+inflation_rate)**10 * n2_demand_1
driver_asep_2_inst_cost_2020 = 620.16 * (1+inflation_rate)**10 * n2_demand_1
driver_asep_3_inst_cost_2020 = 650 * (1+inflation_rate)**10 * n2_demand_1
total_drivers_asep_inst_cost = driver_asep_1_inst_cost_2020 + driver_asep_2_inst_cost_2020 + driver_asep_3_inst_cost_2020

tower_asep_1_inst_cost_2020 = 6634.68 * (1+inflation_rate)**10 * n2_demand_1
tower_asep_2_inst_cost_2020 = 5040.2 * (1+inflation_rate)**10 * n2_demand_1
tower_asep_3_inst_cost_2020 = 908.68 * (1+inflation_rate)**10 * n2_demand_1
total_towers_asep_inst_cost = tower_asep_1_inst_cost_2020 + tower_asep_2_inst_cost_2020 + tower_asep_3_inst_cost_2020

turbine_asep_inst_cost_2020 = 356 * (1+inflation_rate)**10 * n2_demand_1
total_turbine_asep_inst_cost = turbine_asep_inst_cost_2020
total_asep_cost = total_heat_exchangers_asep_inst_cost + total_compressors_asep_inst_cost + total_drivers_asep_inst_cost + total_towers_asep_inst_cost + total_turbine_asep_inst_cost
```

```
lhv_h2 = 120.1 #MJ/kg
h2_per_second_kg = h2_demand_2*1000/86400
h2_capacity_MW = h2_per_second_kg * lhv_h2

h2_investment_MW_h2 = 628000 # $ per MW H2
electrolyser_cost = h2_investment_MW_h2 * h2_capacity_MW

operating_hours = 365*24*0.95
stack_lifetime = 90000 # hours
stack_replacement_year = stack_lifetime/operating_hours
stack_replacement_number = 1
replacement_cost = electrolyser_cost*0.6*0.45*np.round(stack_replacement_number)

total_electrolyser_cost = electrolyser_cost + replacement_cost
heat_exchanger_ar_1_inst_cost_2020 = 10.055333 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_2_inst_cost_2020 = 732.0333 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_3_inst_cost_2020 = 2.035066 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_4_inst_cost_2020 = 1425.9 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_5_inst_cost_2020 = 1334.833 * (1+inflation_rate)**10 * capacity
total_heat_exchanger_ar_cost = heat_exchanger_ar_1_inst_cost_2020 +
heat_exchanger_ar_2_inst_cost_2020 + heat_exchanger_ar_4_inst_cost_2020 +
heat_exchanger_ar_4_inst_cost_2020

compressor_ar_1_inst_cost_2020 = 7387.33 * (1+inflation_rate)**10 * capacity
compressor_ar_2_inst_cost_2020 = 4734 * (1+inflation_rate)**10 * capacity
total_compressor_ar_cost = compressor_ar_1_inst_cost_2020 +
compressor_ar_2_inst_cost_2020

driver_ar_1_inst_cost_2020 = 943.367 * (1+inflation_rate)**10 * capacity
driver_ar_2_inst_cost_2020 = 702.4 * (1+inflation_rate)**10 * capacity
total_driver_ar_cost = driver_ar_1_inst_cost_2020 + driver_ar_2_inst_cost_2020

reactor_ar_1_inst_cost_2020 = 5623 * (1+inflation_rate)**10 * capacity
reactor_ar_2_inst_cost_2020 = 4926.33 * (1+inflation_rate)**10 * capacity
total_reactor_ar_cost = reactor_ar_1_inst_cost_2020 + reactor_ar_2_inst_cost_2020

pump_ar_inst_cost_2020 = 1939.46 * (1+inflation_rate)**10 * capacity
total_pump_ar_cost = pump_ar_inst_cost_2020
total_ar_cost = total_heat_exchanger_ar_cost + total_compressor_ar_cost +
total_driver_ar_cost + total_reactor_ar_cost + total_pump_ar_cost

storage_as_inst_cost_2020 = 718.89*storage_as_capacity*(1+inflation_rate)**10
total_as_cost = storage_as_inst_cost_2020
```

```
    total_capital_cost = total_mvc_inst_cost + total_asep_cost + total_ar_cost + total_as_cost
    total_estimated_capital_cost_2020 = cost_estimation(total_capital_cost, capacity_new,
capacity, exponent) + total_electrolyser_cost
    ## 2. Electricity cost
    electricity_cost_y = electricity_price * 1000 * p * 365 * 24/10**6
    ## 3. Water cost
    total_water_cost = water_cost * mass_water_el_2 * 365/10**6
    ## 4. CO2 cost
     ## kg/MWh
    emission_cost_annual = (p * CO2_emmisions_from_electricity_grid_norway * 365 *
CO2_cost)/(1000*10**6)
    ## 5. Operation and Maintenance
    operation_maintance_cost_annual = 0.05 * total_estimated_capital_cost_2020/10**6
    ## 6. Property tax
    property_taxes_annual = 0.02 * total_estimated_capital_cost_2020/10**6
    ## 7. Rent of the land
    rent_of_land_annual = 0.02 * total_estimated_capital_cost_2020/10**6
    ## 8. Labor cost
    labor_cost_annual = 50000/10**6 ## $ per year
    ## 9. Revenue
    total_revenue = (ammonia_market_price_2019 * capacity_new * 365 + O2_cost)/10**6
    ## 10. Depreciation cost
    depreciation_cost_y = total_estimated_capital_cost_2020/(plant_life*10**6)

    ## 11. Total Revenue
    years = np.arange(0, plant_life, 1)
    Years = np.round(years, 0)
    capital_cost_yr=np.repeat(0,plant_life).tolist()
    capital_cost_yr[0]=(total_estimated_capital_cost_2020*0.5 - replacement_cost)/10**6
    capital_cost_yr[1]=(total_estimated_capital_cost_2020*0.5 - replacement_cost)/10**6
    capital_cost_yr[11]=replacement_cost/10**6


operation_maintance_cost=np.repeat(operation_maintance_cost_annual,plant_life).tolist()
    labor_cost=np.repeat(labor_cost_annual,plant_life).tolist()
    property_tax_cost=np.repeat(property_taxes_annual,plant_life).tolist()
    emission_cost=np.repeat(emission_cost_annual,plant_life).tolist()
    depreciation_cost=np.repeat(depreciation_cost_y,plant_life).tolist()
    electricity_cost = np.repeat(electricity_cost_y,plant_life).tolist()
    total_water_cost = np.repeat(total_water_cost, plant_life).tolist()
    total_revenue=np.repeat(total_revenue,plant_life).tolist()

    for i in range(2):
        #ammonia_production[i] = 0
        operation_maintance_cost[i] = 0
```

```
        labor_cost[i] = 0
        emission_cost[i] = 0
        depreciation_cost[i] = 0
        property_tax_cost[i] = 0
        electricity_cost[i] = 0
        total_water_cost[i] = 0
        total_revenue[i] = 0

    tax_cost=[]
    for i in range(len(operation_maintance_cost)):
        tax_cost.append((total_revenue[i]-(capital_cost_yr[i] +
operation_maintance_cost[i]+emission_cost[i]+
                                labor_cost[i]  + total_water_cost[i] +  depreciation_cost[i]  +
property_tax_cost[i]   +  electricity_cost[i]))*tax_rate)
    ## $ million
    tax_cost[0]=0
    tax_cost[1]=0

    cash_flow=[]
    for i in range(len(total_revenue)):
        cash_flow.append(total_revenue[i]-(capital_cost_yr[i] + emission_cost[i]
                                + operation_maintance_cost[i] + labor_cost[i] + total_water_cost[i]
+ property_tax_cost[i] +
                                electricity_cost[i] +tax_cost[i]))

    npv = npf.npv(DiscountRate, cash_flow)
    irr=npf.irr(cash_flow)
    return [npv, irr, cash_flow]
```

Line 14:

```
# Sensitivity analysis of the Base case.
baseline=[tax_rate, DiscountRate, ammonia_market_price_2019, electricity_price,
electrolyser_efficiency, capacity_new]
baseline_npv=npv_base(*baseline)[0]
baseline_npv
```

Line 15:

```
variation=np.arange(-20,21,5)
t=[1+(i/100) for i in variation] # P varies from -1.20 to 1.20
change=[x*100 for x in t]
tax_rate_n=[tax_rate*x for x in t]
DiscountRate_n=[DiscountRate*x for x in t]
ammonia_market_price_2019_n=[ammonia_market_price_2019*x for x in t]
electricity_price_n=[electricity_price*x for x in t]
electrolyser_efficiency_n=[electrolyser_efficiency*x for x in t]
```

```
capacity_new_n=[capacity_new*x for x in t]
```

Line 16:

```
#Tax_rate is varied
tax_rate_s=[npv_base(tax_rate_n[i],DiscountRate,ammonia_market_price_2019,
            electricity_price,electrolyser_efficiency,capacity_new)[0]
        for i in range(len(tax_rate_n))]
#Interest rate is varied
DiscountRate_s=[npv_base(tax_rate,DiscountRate_n[i],ammonia_market_price_2019,
            electricity_price,electrolyser_efficiency,capacity_new)[0]
        for i in range(len(DiscountRate_n))]

#Ammonia market price is varied
ammonia_market_price_2019_s=[npv_base(tax_rate,DiscountRate,ammonia_market_pric
e_2019_n[i],
            electricity_price,electrolyser_efficiency,capacity_new)[0]
        for i in range(len(ammonia_market_price_2019_n))]

#Electricity price is varied
electricity_price_s=[npv_base(tax_rate,DiscountRate,ammonia_market_price_2019,
            electricity_price_n[i],electrolyser_efficiency,capacity_new)[0]
        for i in range(len(electricity_price_n))]

#Electrolyser efficiency is varied
electrolyser_efficiency_s=[npv_base(tax_rate,DiscountRate,ammonia_market_price_2019,
            electricity_price,electrolyser_efficiency_n[i],capacity_new)[0]
        for i in range(len(electrolyser_efficiency_n))]

#Capacity new is varied
capacity_new_s=[npv_base(tax_rate,DiscountRate,ammonia_market_price_2019,
            electricity_price,electrolyser_efficiency,capacity_new_n[i])[0]
        for i in range(len(capacity_new_n))]
```

Line 17:

```
if baseline_npv<0:
   tax_rate_npv=[((x -baseline_npv)/(baseline_npv))*100 for x in tax_rate_s]

   DiscountRate_npv=[((x-baseline_npv)/(baseline_npv))*100 for x in DiscountRate_s]

   ammonia_market_price_2019_npv=[((baseline_npv-x)/(baseline_npv))*100 for x in
ammonia_market_price_2019_s]

   electricity_price_npv=[((baseline_npv-x)/(baseline_npv))*100 for x in electricity_price_s]
```

```
   electrolyser_efficiency_npv=[((baseline_npv-x)/(baseline_npv))*100 for x in
electrolyser_efficiency_s]

   capacity_new_npv=[((baseline_npv-x)/(baseline_npv))*100 for x in capacity_new_s]
```

Line 18:

```
change_variables_per=['-20', '-15', '-10', '-5', '0', '+5', '+10', '+15', '+20']
```

Line 19:

```
npv_dict={'Tax rate': tax_rate_npv,'Discount rate':DiscountRate_npv,'Ammonia market
price':ammonia_market_price_2019_npv,
        'Electricity price':electricity_price_npv,'Electrolyser
efficiency':electrolyser_efficiency_npv,
        'Capacity of the green ammonia
plant':capacity_new_npv,'%Change':change_variables_per}
```

Line 20:

```
df_npv=pd.DataFrame.from_dict(npv_dict,orient='index')
df_npv=df_npv.transpose()
df_npv
```

Line 21:

```
change_variables_per=['-20', '-15', '-10', '-5', '0', '+5', '+10', '+15', '+20']
```

Line 22:

```
df_npv.plot.bar(x='%Change',y=['Tax rate', 'Discount rate', 'Ammonia market price',
                    'Electricity price', 'Electrolyser efficiency', 'Capacity of the green
ammonia plant']
            ,stacked=False,figsize=(15,10))
plt.xlabel('Change in value of the variables in percentage',Fontsize=25)
plt.ylabel('Change in value of the NPV in percentage',Fontsize=25)
plt.legend(loc='best')
plt.savefig('NPV_percentage_change_base_case.pdf',dpi=1200, bbox_inches='tight')
plt.savefig('NPV_percentage_change_base_case.png',dpi=1200, bbox_inches='tight')
```

Line 23:

```
# Scenario 1.
# Sensitivity analysis.
price_for_emissions_1 = 100
```

Line 24:

```
def npv_sc1(tax_rate, DiscountRate, ammonia_market_price_2019, electricity_price,
electrolyser_efficiency, capacity_new, price_for_emissions_1):
   def greenammonia(capacity):
      h2_demand = capacity*(3*mol_weight_h2*conversion1)
```

```
n2_demand = capacity*(mol_weight_n2*conversion1)
h2_demand_kgh = (h2_demand/24)*1000
n2_demand_kgh = (n2_demand/24)*1000
h2_demand_Nm3_per_h = h2_demand_kgh/conversion2
electrolyser_capacity_MW = (h2_demand_Nm3_per_h*electrolyser_efficiency)/1000
rectifier_efficiency = 0.95
total_electrolyser_power = electrolyser_capacity_MW/rectifier_efficiency

mass_water_el = h2_demand/(mol_weight_h2*conversion3*2)
mass_o2 = mol_weight_o2*conversion3*mass_water_el
adiabatic_efficiency = 0.75
driver_efficiency = 0.95
power_per_1_ton = 0.71
kg_hydrogen = 8.981
p_water =
power_per_1_ton*mass_water_el/(adiabatic_efficiency*driver_efficiency*1000)

n2_eff = 0.79
m_air = n2_demand/n2_eff
power_ton_n2 = 2.25
power_air_separation =
(m_air*power_ton_n2/(adiabatic_efficiency*driver_efficiency))/1000
#print('power_air_separation:', power_air_separation)

Cp_h = A_h + B_h*t_h + C_h*t_h**2 + D_h*t_h**3 + E_h/t_h**2
Q_h = ((h2_demand*1000)/(24*3600)*Cp_h*500*(T2 - T1_h))/10**6

Cp_n = A_n + B_n*t_n + C_n*t_n**2 + D_n*t_n**3 + E_n/t_n**2
Q_n = ((n2_demand*1000)/(24*3600)*Cp_n*1000/28*(T2 - T1_n))/10**6
total_heaters_power = Q_h + Q_n

P_per_ammonia = 0.02633
P_per_day=P_per_ammonia*capacity
M_water_per_minute = 9.5
P_water_pumping = 7.3*10**-6
P_water_total = M_water_per_minute*60*24*P_water_pumping
P_reactor = P_per_day + P_water_total

P_compressor = 0.006827
h2_cost_green = electrolyser_efficiency/conversion2
h2_cost_smr = 4.5 * conversion2
CO2_per_1_kg_h2 = 7
CO2_smr = CO2_per_1_kg_h2*h2_demand*1000
total_power = (total_electrolyser_power + p_water + power_air_separation +
total_heaters_power + P_reactor + P_compressor)
```

```
    return(total_power)
  p = greenammonia(400)

  capacity = 300
  capaciity_new  = 400

  compressor_mvc_inst_cost_2020 = 984.3 * (1+inflation_rate)**10*mass_water_el_1
  driver_mvc_inst_cost_2020 = 307.9 * (1+inflation_rate)**10*mass_water_el_1
  evaporator_mvc_inst_cost_2020 = 7411.34 * (1+inflation_rate)**10*mass_water_el_1
  distillate_mvc_inst_cost_2020 = 2729.62 * (1+inflation_rate)**10*mass_water_el_1
  brine_mvc_inst_cost_2020 = 6286.42 * (1+inflation_rate)**10*mass_water_el_1
  total_mvc_inst_cost = compressor_mvc_inst_cost_2020 + driver_mvc_inst_cost_2020 +
evaporator_mvc_inst_cost_2020 + distillate_mvc_inst_cost_2020 +
brine_mvc_inst_cost_2020

  heat_exchanger_asep_1_inst_cost_2020 = 332.8 * (1+inflation_rate)**10 * n2_demand_1
  heat_exchanger_asep_2_inst_cost_2020 = 340.96 * (1+inflation_rate)**10 *
n2_demand_1
  heat_exchanger_asep_3_inst_cost_2020 = 421.4 * (1+inflation_rate)**10 * n2_demand_1
  heat_exchanger_asep_4_inst_cost_2020 = 5715.52 * (1+inflation_rate)**10 *
n2_demand_1
  total_heat_exchangers_asep_inst_cost = heat_exchanger_asep_1_inst_cost_2020 +
heat_exchanger_asep_2_inst_cost_2020 + heat_exchanger_asep_3_inst_cost_2020 +
heat_exchanger_asep_4_inst_cost_2020

  compressor_asep_1_inst_cost_2020 = 1987.92 * (1+inflation_rate)**10 * n2_demand_1
  compressor_asep_2_inst_cost_2020 = 1914.76 * (1+inflation_rate)**10 * n2_demand_1
  compressor_asep_3_inst_cost_2020 = 2136.32 * (1+inflation_rate)**10 * n2_demand_1
  total_compressors_asep_inst_cost = compressor_asep_1_inst_cost_2020 +
compressor_asep_2_inst_cost_2020 + compressor_asep_3_inst_cost_2020

  driver_asep_1_inst_cost_2020 = 630.32 * (1+inflation_rate)**10 * n2_demand_1
  driver_asep_2_inst_cost_2020 = 620.16 * (1+inflation_rate)**10 * n2_demand_1
  driver_asep_3_inst_cost_2020 = 650 * (1+inflation_rate)**10 * n2_demand_1
  total_drivers_asep_inst_cost = driver_asep_1_inst_cost_2020 +
driver_asep_2_inst_cost_2020 + driver_asep_3_inst_cost_2020

  tower_asep_1_inst_cost_2020 = 6634.68 * (1+inflation_rate)**10 * n2_demand_1
  tower_asep_2_inst_cost_2020 = 5040.2 * (1+inflation_rate)**10 * n2_demand_1
  tower_asep_3_inst_cost_2020 = 908.68 * (1+inflation_rate)**10 * n2_demand_1
  total_towers_asep_inst_cost = tower_asep_1_inst_cost_2020 +
tower_asep_2_inst_cost_2020 + tower_asep_3_inst_cost_2020

  turbine_asep_inst_cost_2020 = 356 * (1+inflation_rate)**10 * n2_demand_1
  total_turbine_asep_inst_cost = turbine_asep_inst_cost_2020
```

```
total_asep_cost = total_heat_exchangers_asep_inst_cost +
total_compressors_asep_inst_cost + total_drivers_asep_inst_cost +
total_towers_asep_inst_cost + total_turbine_asep_inst_cost

lhv_h2 = 120.1
h2_per_second_kg = h2_demand_2*1000/86400
h2_capacity_MW = h2_per_second_kg * lhv_h2

h2_investment_MW_h2 = 628000
electrolyser_cost = h2_investment_MW_h2 * h2_capacity_MW

operating_hours = 365*24*0.95
stack_lifetime = 90000
stack_replacement_year = stack_lifetime/operating_hours
stack_replacement_number = 1
replacement_cost = electrolyser_cost*0.6*0.45*np.round(stack_replacement_number)

total_electrolyser_cost = electrolyser_cost + replacement_cost

heat_exchanger_ar_1_inst_cost_2020 = 10.055333 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_2_inst_cost_2020 = 732.0333 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_3_inst_cost_2020 = 2.035066 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_4_inst_cost_2020 = 1425.9 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_5_inst_cost_2020 = 1334.833 * (1+inflation_rate)**10 * capacity
total_heat_exchanger_ar_cost = heat_exchanger_ar_1_inst_cost_2020 +
heat_exchanger_ar_2_inst_cost_2020 + heat_exchanger_ar_4_inst_cost_2020 +
heat_exchanger_ar_4_inst_cost_2020

compressor_ar_1_inst_cost_2020 = 7387.33 * (1+inflation_rate)**10 * capacity
compressor_ar_2_inst_cost_2020 = 4734 * (1+inflation_rate)**10 * capacity
total_compressor_ar_cost = compressor_ar_1_inst_cost_2020 +
compressor_ar_2_inst_cost_2020

driver_ar_1_inst_cost_2020 = 943.367 * (1+inflation_rate)**10 * capacity
driver_ar_2_inst_cost_2020 = 702.4 * (1+inflation_rate)**10 * capacity
total_driver_ar_cost = driver_ar_1_inst_cost_2020 + driver_ar_2_inst_cost_2020

reactor_ar_1_inst_cost_2020 = 5623 * (1+inflation_rate)**10 * capacity
reactor_ar_2_inst_cost_2020 = 4926.33 * (1+inflation_rate)**10 * capacity
total_reactor_ar_cost = reactor_ar_1_inst_cost_2020 + reactor_ar_2_inst_cost_2020

pump_ar_inst_cost_2020 = 1939.46 * (1+inflation_rate)**10 * capacity
total_pump_ar_cost = pump_ar_inst_cost_2020
total_ar_cost = total_heat_exchanger_ar_cost + total_compressor_ar_cost +
total_driver_ar_cost + total_reactor_ar_cost + total_pump_ar_cost
```

```
storage_as_inst_cost_2020 = 718.89*storage_as_capacity*(1+inflation_rate)**10
total_as_cost = storage_as_inst_cost_2020

total_capital_cost = total_mvc_inst_cost + total_asep_cost + total_ar_cost + total_as_cost
total_estimated_capital_cost_2020 = cost_estimation(total_capital_cost, capacity_new,
capacity, exponent) + total_electrolyser_cost
## 1. Electricity cost
electricity_cost_y = electricity_price * 1000 * p * 365 * 24/10**6
## 2. Water cost
total_water_cost = water_cost * mass_water_el_2 * 365/10**6
## 3. CO2 cost
benefit_from_CO2_savings = 7*h2_demand*365*price_for_emissions_1
emission_cost_annual = (p * CO2_emmisions_from_electricity_grid_norway * 365 *
CO2_cost)/(1000*10**6)
## 4. Operation and Maintance
operation_maintance_cost_annual = 0.05 * total_estimated_capital_cost_2020/10**6
## 5. Property tax
property_taxes_annual = 0.02 * total_estimated_capital_cost_2020/10**6
## 6. Rent of the land
rent_of_land_annual = 0.02 * total_estimated_capital_cost_2020/10**6
## 7. Labor cost
labor_cost_annual = 50000/10**6
## 8. Revenue
total_revenue = (ammonia_market_price_2019 * capacity_new * 365 + O2_cost +
benefit_from_CO2_savings)/10**6
## 9. Depreciation cost
depreciation_cost_y = total_estimated_capital_cost_2020/(plant_life*10**6)
## 10. Total Revenue
years = np.arange(0, plant_life, 1)
Years = np.round(years, 0)
#ammonia_production=np.repeat((capacity_new*365)/10**6,plant_life).tolist()
capital_cost_yr=np.repeat(0,plant_life).tolist()
capital_cost_yr[0]=(total_estimated_capital_cost_2020*0.5 - replacement_cost)/10**6
capital_cost_yr[1]=(total_estimated_capital_cost_2020*0.5 - replacement_cost)/10**6
capital_cost_yr[11]=replacement_cost/10**6

operation_maintance_cost=np.repeat(operation_maintance_cost_annual,plant_life).tolist()
labor_cost=np.repeat(labor_cost_annual,plant_life).tolist()
property_tax_cost=np.repeat(property_taxes_annual,plant_life).tolist()
emission_cost=np.repeat(emission_cost_annual,plant_life).tolist()
depreciation_cost=np.repeat(depreciation_cost_y,plant_life).tolist()
electricity_cost = np.repeat(electricity_cost_y,plant_life).tolist()
total_water_cost = np.repeat(total_water_cost, plant_life).tolist()
total_revenue=np.repeat(total_revenue,plant_life).tolist()
```

```
    for i in range(2):
        #ammonia_production[i] = 0
        operation_maintance_cost[i] = 0
        labor_cost[i] = 0
        emission_cost[i] = 0
        depreciation_cost[i] = 0
        property_tax_cost[i] = 0
        electricity_cost[i] = 0
        total_water_cost[i] = 0
        total_revenue[i] = 0

    tax_cost=[]
    for i in range(len(operation_maintance_cost)):
        tax_cost.append((total_revenue[i]-(capital_cost_yr[i] +
operation_maintance_cost[i]+emission_cost[i]+
                                labor_cost[i]  + total_water_cost[i] +  depreciation_cost[i]  +
property_tax_cost[i]   +  electricity_cost[i]))*tax_rate)
    tax_cost[0]=0
    tax_cost[1]=0

    cash_flow=[]
    for i in range(len(total_revenue)):
        cash_flow.append(total_revenue[i]-(capital_cost_yr[i] + emission_cost[i]
                               + operation_maintance_cost[i] + labor_cost[i] + total_water_cost[i]
+ property_tax_cost[i] +
                               electricity_cost[i] +tax_cost[i]))

    npv = npf.npv(DiscountRate, cash_flow)
    irr=npf.irr(cash_flow)
    return [npv, irr, cash_flow]
```

Line 25:

```
baseline1=[tax_rate, DiscountRate, ammonia_market_price_2019, electricity_price,
electrolyser_efficiency, capacity_new, price_for_emissions_1]
baseline_npv1=npv_sc1(*baseline1)[0]
```

Line 26:

```
variation=np.arange(-20,21,5)
t=[1+(i/100) for i in variation] # P varies from -1.20 to 1.20
change=[x*100 for x in t]
tax_rate_n=[tax_rate*x for x in t]
DiscountRate_n=[DiscountRate*x for x in t]
ammonia_market_price_2019_n=[ammonia_market_price_2019*x for x in t]
electricity_price_n=[electricity_price*x for x in t]
```

```
electrolyser_efficiency_n=[electrolyser_efficiency*x for x in t]
capacity_new_n=[capacity_new*x for x in t]
price_for_emissions_1_n=[price_for_emissions_1*x for x in t]
```

Line 27:

```
#Tax_rate is varied
tax_rate_s=[npv_sc1(tax_rate_n[i],DiscountRate,ammonia_market_price_2019,
        electricity_price,electrolyser_efficiency,capacity_new, price_for_emissions_1)[0]
    for i in range(len(tax_rate_n))]
#Interest rate is varied
DiscountRate_s=[npv_sc1(tax_rate,DiscountRate_n[i],ammonia_market_price_2019,
        electricity_price,electrolyser_efficiency,capacity_new, price_for_emissions_1)[0]
    for i in range(len(DiscountRate_n))]

#Ammonia market price is varied
ammonia_market_price_2019_s=[npv_sc1(tax_rate,DiscountRate,ammonia_market_price_
2019_n[i],
        electricity_price,electrolyser_efficiency,capacity_new, price_for_emissions_1)[0]
    for i in range(len(ammonia_market_price_2019_n))]

#Electricity price is varied
electricity_price_s=[npv_sc1(tax_rate,DiscountRate,ammonia_market_price_2019,
        electricity_price_n[i],electrolyser_efficiency,capacity_new,
price_for_emissions_1)[0]
    for i in range(len(electricity_price_n))]

#Electrolyser efficiency is varied
electrolyser_efficiency_s=[npv_sc1(tax_rate,DiscountRate,ammonia_market_price_2019,
        electricity_price,electrolyser_efficiency_n[i],capacity_new,
price_for_emissions_1)[0]
    for i in range(len(electrolyser_efficiency_n))]

#Capacity new is varied
capacity_new_s=[npv_sc1(tax_rate,DiscountRate,ammonia_market_price_2019,
        electricity_price,electrolyser_efficiency,capacity_new_n[i],
price_for_emissions_1)[0]
    for i in range(len(capacity_new_n))]

#Price for emissions is varied
price_for_emissions_1_s=[npv_sc1(tax_rate,DiscountRate,ammonia_market_price_2019,
        electricity_price,electrolyser_efficiency,capacity_new,
price_for_emissions_1_n[i])[0]
    for i in range(len(price_for_emissions_1_n))]
```

Line 28:

```
if baseline_npv1>0:
   tax_rate_npv=[-((baseline_npv1-x)/(baseline_npv1))*100 for x in tax_rate_s]

   DiscountRate_npv=[-((baseline_npv1-x)/(baseline_npv1))*100 for x in DiscountRate_s]

   ammonia_market_price_2019_npv=[-((baseline_npv1-x)/(baseline_npv1))*100 for x in
ammonia_market_price_2019_s]

   electricity_price_npv=[-((baseline_npv1-x)/(baseline_npv1))*100 for x in
electricity_price_s]

   electrolyser_efficiency_npv=[-((baseline_npv1-x)/(baseline_npv1))*100 for x in
electrolyser_efficiency_s]

   capacity_new_npv=[-((baseline_npv1-x)/(baseline_npv1))*100 for x in capacity_new_s]

   price_for_emissions_1_npv=[-((baseline_npv1-x)/(baseline_npv1))*100 for x in
price_for_emissions_1_s]
elif baseline_npv<0:
   tax_rate_npv=[((baseline_npv1-x)/(baseline_npv1))*100 for x in tax_rate_s]

   DiscountRate_npv=[((baseline_npv1-x)/(baseline_npv1))*100 for x in DiscountRate_s]

   ammonia_market_price_2019_npv=[((baseline_npv1-x)/(baseline_npv1))*100 for x in
ammonia_market_price_2019_s]

   electricity_price_npv=[((baseline_npv1-x)/(baseline_npv1))*100 for x in
electricity_price_s]

   electrolyser_efficiency_npv=[((baseline_npv1-x)/(baseline_npv1))*100 for x in
electrolyser_efficiency_s]

   capacity_new_npv=[((baseline_npv1-x)/(baseline_npv1))*100 for x in capacity_new_s]

   price_for_emissions_1_npv=[((baseline_npv1-x)/(baseline_npv1))*100 for x in
price_for_emissions_1_s]
```

Line 29:

```
change_variables_per=['-20', '-15', '-10', '-5', '0', '+5', '+10', '+15', '+20']
```

Line 30:

```
npv_dict={'Tax rate': tax_rate_npv,'Discount rate':DiscountRate_npv,'Ammonia market
price':ammonia_market_price_2019_npv,
        'Electricity price':electricity_price_npv,'Electrolyser
efficiency':electrolyser_efficiency_npv,
```

```
'Capacity of the green ammonia plant':capacity_new_npv, 'Price for emissions':price_for_emissions_1_npv,'%Change':change_variables_per}
```

Line 31:

```
df_npv=pd.DataFrame.from_dict(npv_dict,orient='index')
df_npv=df_npv.transpose()
df_npv
```

Line 32:

```
change_variables_per=['-20', '-15', '-10', '-5', '0', '+5', '+10', '+15', '+20']
```

Line 33:

```
df_npv.plot.bar(x='%Change',y=['Tax rate', 'Discount rate', 'Ammonia market price',
                     'Electricity price', 'Electrolyser efficiency', 'Capacity of the green ammonia plant', 'Price for emissions']
            ,stacked=False,figsize=(15,10))
plt.xlabel('Change in value of the variables in percentage',Fontsize=25)
plt.ylabel('Change in value of the NPV in percentage',Fontsize=25)
plt.legend(loc='best')
plt.savefig('NPV_percentage_change_sc1.pdf',dpi=1200, bbox_inches='tight')
plt.savefig('NPV_percentage_change_sc1.png',dpi=1200, bbox_inches='tight')
```

Line 34:

```
# Breakeven study.
price_for_emissions = 100
```

Line 35:

```
def npv_sc1_f(tax_rate, DiscountRate, ammonia_market_price_2019, electricity_price, electrolyser_efficiency, capacity_new, price_for_emissions):
   def greenammonia(capacity):
      h2_demand = capacity*(3*mol_weight_h2*conversion1)
      n2_demand = capacity*(mol_weight_n2*conversion1)
      h2_demand_kgh = (h2_demand/24)*1000
      n2_demand_kgh = (n2_demand/24)*1000
      h2_demand_Nm3_per_h = h2_demand_kgh/conversion2
      electrolyser_capacity_MW = (h2_demand_Nm3_per_h*electrolyser_efficiency)/1000
      rectifier_efficiency = 0.95
      total_electrolyser_power = electrolyser_capacity_MW/rectifier_efficiency
      mass_water_el = h2_demand/(mol_weight_h2*conversion3*2)
      mass_o2 = mol_weight_o2*conversion3*mass_water_el
      adiabatic_efficiency = 0.75
      driver_efficiency = 0.95
      power_per_1_ton = 0.71
      kg_hydrogen = 8.981
```

```
    p_water =
power_per_1_ton*mass_water_el/(adiabatic_efficiency*driver_efficiency*1000)
    n2_eff = 0.79
    m_air = n2_demand/n2_eff
    power_ton_n2 = 2.25
    power_air_separation =
(m_air*power_ton_n2/(adiabatic_efficiency*driver_efficiency))/1000
    Cp_h = A_h + B_h*t_h + C_h*t_h**2 + D_h*t_h**3 + E_h/t_h**2
    Q_h = ((h2_demand*1000)/(24*3600)*Cp_h*500*(T2 - T1_h))/10**6

    Cp_n = A_n + B_n*t_n + C_n*t_n**2 + D_n*t_n**3 + E_n/t_n**2
    Q_n = ((n2_demand*1000)/(24*3600)*Cp_n*1000/28*(T2 - T1_n))/10**6
    total_heaters_power = Q_h + Q_n
    P_per_ammonia = 0.02633
    P_per_day=P_per_ammonia*capacity
    M_water_per_minute = 9.5
    P_water_pumping = 7.3*10**-6
    P_water_total = M_water_per_minute*60*24*P_water_pumping
    P_reactor = P_per_day + P_water_total
    P_compressor = 0.006827
    h2_cost_green = electrolyser_efficiency/conversion2
    h2_cost_smr = 4.5 * conversion2
    CO2_per_1_kg_h2 = 7
    CO2_smr = CO2_per_1_kg_h2*h2_demand*1000
    total_power = (total_electrolyser_power + p_water + power_air_separation +
total_heaters_power + P_reactor + P_compressor)
    return(total_power)
  p = greenammonia(400)

  capacity = 300
  capaciity_new  = 400

  compressor_mvc_inst_cost_2020 = 984.3 * (1+inflation_rate)**10*mass_water_el_1
  driver_mvc_inst_cost_2020 = 307.9 * (1+inflation_rate)**10*mass_water_el_1
  evaporator_mvc_inst_cost_2020 = 7411.34 * (1+inflation_rate)**10*mass_water_el_1
  distillate_mvc_inst_cost_2020 = 2729.62 * (1+inflation_rate)**10*mass_water_el_1
  brine_mvc_inst_cost_2020 = 6286.42 * (1+inflation_rate)**10*mass_water_el_1
  total_mvc_inst_cost = compressor_mvc_inst_cost_2020 + driver_mvc_inst_cost_2020 +
evaporator_mvc_inst_cost_2020 + distillate_mvc_inst_cost_2020 +
brine_mvc_inst_cost_2020

  heat_exchanger_asep_1_inst_cost_2020 = 332.8 * (1+inflation_rate)**10 * n2_demand_1
  heat_exchanger_asep_2_inst_cost_2020 = 340.96 * (1+inflation_rate)**10 *
n2_demand_1
  heat_exchanger_asep_3_inst_cost_2020 = 421.4 * (1+inflation_rate)**10 * n2_demand_1
```

```
    heat_exchanger_asep_4_inst_cost_2020 = 5715.52 * (1+inflation_rate)**10 *
n2_demand_1
    total_heat_exchangers_asep_inst_cost = heat_exchanger_asep_1_inst_cost_2020 +
heat_exchanger_asep_2_inst_cost_2020 + heat_exchanger_asep_3_inst_cost_2020 +
heat_exchanger_asep_4_inst_cost_2020

    compressor_asep_1_inst_cost_2020 = 1987.92 * (1+inflation_rate)**10 * n2_demand_1
    compressor_asep_2_inst_cost_2020 = 1914.76 * (1+inflation_rate)**10 * n2_demand_1
    compressor_asep_3_inst_cost_2020 = 2136.32 * (1+inflation_rate)**10 * n2_demand_1
    total_compressors_asep_inst_cost = compressor_asep_1_inst_cost_2020 +
compressor_asep_2_inst_cost_2020 + compressor_asep_3_inst_cost_2020

    driver_asep_1_inst_cost_2020 = 630.32 * (1+inflation_rate)**10 * n2_demand_1
    driver_asep_2_inst_cost_2020 = 620.16 * (1+inflation_rate)**10 * n2_demand_1
    driver_asep_3_inst_cost_2020 = 650 * (1+inflation_rate)**10 * n2_demand_1
    total_drivers_asep_inst_cost = driver_asep_1_inst_cost_2020 +
driver_asep_2_inst_cost_2020 + driver_asep_3_inst_cost_2020

    tower_asep_1_inst_cost_2020 = 6634.68 * (1+inflation_rate)**10 * n2_demand_1
    tower_asep_2_inst_cost_2020 = 5040.2 * (1+inflation_rate)**10 * n2_demand_1
    tower_asep_3_inst_cost_2020 = 908.68 * (1+inflation_rate)**10 * n2_demand_1
    total_towers_asep_inst_cost = tower_asep_1_inst_cost_2020 +
tower_asep_2_inst_cost_2020 + tower_asep_3_inst_cost_2020

    turbine_asep_inst_cost_2020 = 356 * (1+inflation_rate)**10 * n2_demand_1
    total_turbine_asep_inst_cost = turbine_asep_inst_cost_2020
    total_asep_cost = total_heat_exchangers_asep_inst_cost +
total_compressors_asep_inst_cost + total_drivers_asep_inst_cost +
total_towers_asep_inst_cost + total_turbine_asep_inst_cost


    lhv_h2 = 120.1
    h2_per_second_kg = h2_demand_2*1000/86400
    h2_capacity_MW = h2_per_second_kg * lhv_h2

    h2_investment_MW_h2 = 628000
    electrolyser_cost = h2_investment_MW_h2 * h2_capacity_MW

    operating_hours = 365*24*0.95
    stack_lifetime = 90000
    stack_replacement_year = stack_lifetime/operating_hours
    stack_replacement_number = 1
    replacement_cost = electrolyser_cost*0.6*0.45*np.round(stack_replacement_number)

    total_electrolyser_cost = electrolyser_cost + replacement_cost
```

```
    heat_exchanger_ar_1_inst_cost_2020 = 10.055333 * (1+inflation_rate)**10 * capacity
    heat_exchanger_ar_2_inst_cost_2020 = 732.0333 * (1+inflation_rate)**10 * capacity
    heat_exchanger_ar_3_inst_cost_2020 = 2.035066 * (1+inflation_rate)**10 * capacity
    heat_exchanger_ar_4_inst_cost_2020 = 1425.9 * (1+inflation_rate)**10 * capacity
    heat_exchanger_ar_5_inst_cost_2020 = 1334.833 * (1+inflation_rate)**10 * capacity
    total_heat_exchanger_ar_cost = heat_exchanger_ar_1_inst_cost_2020 +
heat_exchanger_ar_2_inst_cost_2020 + heat_exchanger_ar_4_inst_cost_2020 +
heat_exchanger_ar_4_inst_cost_2020

    compressor_ar_1_inst_cost_2020 = 7387.33 * (1+inflation_rate)**10 * capacity
    compressor_ar_2_inst_cost_2020 = 4734 * (1+inflation_rate)**10 * capacity
    total_compressor_ar_cost = compressor_ar_1_inst_cost_2020 +
compressor_ar_2_inst_cost_2020

    driver_ar_1_inst_cost_2020 = 943.367 * (1+inflation_rate)**10 * capacity
    driver_ar_2_inst_cost_2020 = 702.4 * (1+inflation_rate)**10 * capacity
    total_driver_ar_cost = driver_ar_1_inst_cost_2020 + driver_ar_2_inst_cost_2020


    reactor_ar_1_inst_cost_2020 = 5623 * (1+inflation_rate)**10 * capacity
    reactor_ar_2_inst_cost_2020 = 4926.33 * (1+inflation_rate)**10 * capacity
    total_reactor_ar_cost = reactor_ar_1_inst_cost_2020 + reactor_ar_2_inst_cost_2020

    pump_ar_inst_cost_2020 = 1939.46 * (1+inflation_rate)**10 * capacity
    total_pump_ar_cost = pump_ar_inst_cost_2020
    total_ar_cost = total_heat_exchanger_ar_cost + total_compressor_ar_cost +
total_driver_ar_cost + total_reactor_ar_cost + total_pump_ar_cost

    storage_as_inst_cost_2020 = 718.89*storage_as_capacity*(1+inflation_rate)**10
    total_as_cost = storage_as_inst_cost_2020

    total_capital_cost = total_mvc_inst_cost + total_asep_cost + total_ar_cost + total_as_cost
    total_estimated_capital_cost_2020 = cost_estimation(total_capital_cost, capacity_new,
capacity, exponent) + total_electrolyser_cost
    electricity_cost_y = electricity_price * 1000 * p * 365 * 24/10**6
    total_water_cost = water_cost * mass_water_el_2 * 365/10**6
    emission_cost_annual = (p * CO2_emmisions_from_electricity_grid_norway * 365 *
CO2_cost)/(1000*10**6)
    operation_maintance_cost_annual = 0.05 * total_estimated_capital_cost_2020/10**6
    property_taxes_annual = 0.02 * total_estimated_capital_cost_2020/10**6
    rent_of_land_annual = 0.02 * total_estimated_capital_cost_2020/10**6
    labor_cost_annual = 50000/10**6 ## $ per year
    benefit_from_CO2_savings = 7*h2_demand*365*price_for_emissions
```

```
    total_revenue = (ammonia_market_price_2019 * capacity_new * 365 + O2_cost +
benefit_from_CO2_savings)/10**6
    depreciation_cost_y = total_estimated_capital_cost_2020/(plant_life*10**6)

    years = np.arange(0, plant_life, 1)
    Years = np.round(years, 0)
    capital_cost_yr=np.repeat(0,plant_life).tolist()
    capital_cost_yr[0]=(total_estimated_capital_cost_2020*0.5 - replacement_cost)/10**6
    capital_cost_yr[1]=(total_estimated_capital_cost_2020*0.5 - replacement_cost)/10**6
    capital_cost_yr[11]=replacement_cost/10**6


operation_maintance_cost=np.repeat(operation_maintance_cost_annual,plant_life).tolist()
    labor_cost=np.repeat(labor_cost_annual,plant_life).tolist()
    property_tax_cost=np.repeat(property_taxes_annual,plant_life).tolist()
    emission_cost=np.repeat(emission_cost_annual,plant_life).tolist()
    depreciation_cost=np.repeat(depreciation_cost_y,plant_life).tolist()
    electricity_cost = np.repeat(electricity_cost_y,plant_life).tolist()
    total_water_cost = np.repeat(total_water_cost, plant_life).tolist()
    total_revenue=np.repeat(total_revenue,plant_life).tolist()

    for i in range(2):
        operation_maintance_cost[i] = 0
        labor_cost[i] = 0
        emission_cost[i] = 0
        depreciation_cost[i] = 0
        property_tax_cost[i] = 0
        electricity_cost[i] = 0
        total_water_cost[i] = 0
        total_revenue[i] = 0

    tax_cost=[]
    for i in range(len(operation_maintance_cost)):
        tax_cost.append((total_revenue[i]-(capital_cost_yr[i] +
operation_maintance_cost[i]+emission_cost[i]+
                                labor_cost[i]  + total_water_cost[i] +  depreciation_cost[i]  +
property_tax_cost[i]   +  electricity_cost[i]))*tax_rate)
    tax_cost[0]=0
    tax_cost[1]=0

    cash_flow = [total_revenue[i]-(capital_cost_yr[i] + emission_cost[i]
                               + operation_maintance_cost[i] + labor_cost[i] + total_water_cost[i]
+ property_tax_cost[i] +
                                electricity_cost[i] +tax_cost[i]) for i in range(len(total_revenue))]
```

```
    npv1 = npf.npv(DiscountRate, cash_flow)
    return (npv1)
```

Line 36:

```
npv_sc1_f(tax_rate, DiscountRate, ammonia_market_price_2019, electricity_price,
electrolyser_efficiency, capacity_new, price_for_emissions)
```

Lime 37:

```
price_for_emmisions_array = np.arange(start=50,stop=200,step=0.1)
NPV_end_array = np.zeros(price_for_emmisions_array.shape)
for i in range(price_for_emmisions_array.shape[0]):
    NPV_end_array[i] =npv_sc1_f(tax_rate, DiscountRate, ammonia_market_price_2019,
electricity_price, electrolyser_efficiency, capacity_new, price_for_emmisions_array[i])
fig_SA = px.line(x=price_for_emmisions_array, y=NPV_end_array, labels={'x':'Price for
emission savings','y':'Net Present Value of green ammonia plant'})
fig_SA.show()
```

Line 38:

```
NPV_target = 0
squared_error = lambda x_price_for_emmisions : (npv_sc1_f(tax_rate, DiscountRate,
ammonia_market_price_2019, electricity_price, electrolyser_efficiency, capacity_new,
x_price_for_emmisions)-NPV_target)**2
#result = optimize.minimize(fun=squared_error, x0=51, method='SLSQP', bounds=((50, 200),))
result = differential_evolution(func=squared_error, bounds=[(141, 148),])
benefit = result.x[0]
print('Breakeven price for CO2 savings is:', money_format(benefit))
```

Line 39:

```
# Scenario 2.
# Sensitivity analysis.
ammonia_market_price_2019_breakeven=300
def npv_sc2(tax_rate, DiscountRate, ammonia_market_price_2019_breakeven,
electricity_price, electrolyser_efficiency, capacity_new):
    def greenammonia(capacity):
        h2_demand = capacity*(3*mol_weight_h2*conversion1)
        n2_demand = capacity*(mol_weight_n2*conversion1)
        h2_demand_kgh = (h2_demand/24)*1000
        n2_demand_kgh = (n2_demand/24)*1000
        h2_demand_Nm3_per_h = h2_demand_kgh/conversion2
        electrolyser_capacity_MW = (h2_demand_Nm3_per_h*electrolyser_efficiency)/1000
        rectifier_efficiency = 0.95
        total_electrolyser_power = electrolyser_capacity_MW/rectifier_efficiency
        mass_water_el = h2_demand/(mol_weight_h2*conversion3*2)
        mass_o2 = mol_weight_o2*conversion3*mass_water_el
```

```
        adiabatic_efficiency = 0.75
        driver_efficiency = 0.95
        power_per_1_ton = 0.71
        kg_hydrogen = 8.981
        p_water = 
power_per_1_ton*mass_water_el/(adiabatic_efficiency*driver_efficiency*1000)
        n2_eff = 0.79
        m_air = n2_demand/n2_eff
        power_ton_n2 = 2.25
        power_air_separation = 
(m_air*power_ton_n2/(adiabatic_efficiency*driver_efficiency))/1000
        Cp_h = A_h + B_h*t_h + C_h*t_h**2 + D_h*t_h**3 + E_h/t_h**2
        Q_h = ((h2_demand*1000)/(24*3600)*Cp_h*500*(T2 - T1_h))/10**6

        Cp_n = A_n + B_n*t_n + C_n*t_n**2 + D_n*t_n**3 + E_n/t_n**2
        Q_n = ((n2_demand*1000)/(24*3600)*Cp_n*1000/28*(T2 - T1_n))/10**6
        total_heaters_power = Q_h + Q_n
        P_per_ammonia = 0.02633
        P_per_day=P_per_ammonia*capacity
        M_water_per_minute = 9.5
        P_water_pumping = 7.3*10**-6
        P_water_total = M_water_per_minute*60*24*P_water_pumping
        P_reactor = P_per_day + P_water_total
        P_compressor = 0.006827
        h2_cost_green = electrolyser_efficiency/conversion2
        h2_cost_smr = 4.5 * conversion2
        CO2_per_1_kg_h2 = 7
        CO2_smr = CO2_per_1_kg_h2*h2_demand*1000
        total_power = (total_electrolyser_power + p_water + power_air_separation + 
total_heaters_power + P_reactor + P_compressor)
        return(total_power)
    p = greenammonia(400)

    capacity = 300
    capaciity_new  = 400

    compressor_mvc_inst_cost_2020 = 984.3 * (1+inflation_rate)**10*mass_water_el_1
    driver_mvc_inst_cost_2020 = 307.9 * (1+inflation_rate)**10*mass_water_el_1
    evaporator_mvc_inst_cost_2020 = 7411.34 * (1+inflation_rate)**10*mass_water_el_1
    distillate_mvc_inst_cost_2020 = 2729.62 * (1+inflation_rate)**10*mass_water_el_1
    brine_mvc_inst_cost_2020 = 6286.42 * (1+inflation_rate)**10*mass_water_el_1
    total_mvc_inst_cost = compressor_mvc_inst_cost_2020 + driver_mvc_inst_cost_2020 + 
evaporator_mvc_inst_cost_2020 + distillate_mvc_inst_cost_2020 + 
brine_mvc_inst_cost_2020
```

heat_exchanger_asep_1_inst_cost_2020 = 332.8 * (1+inflation_rate)10 * n2_demand_1**
heat_exchanger_asep_2_inst_cost_2020 = 340.96 * (1+inflation_rate)10 * n2_demand_1**
heat_exchanger_asep_3_inst_cost_2020 = 421.4 * (1+inflation_rate)10 * n2_demand_1**
heat_exchanger_asep_4_inst_cost_2020 = 5715.52 * (1+inflation_rate)10 * n2_demand_1**
total_heat_exchangers_asep_inst_cost = heat_exchanger_asep_1_inst_cost_2020 + heat_exchanger_asep_2_inst_cost_2020 + heat_exchanger_asep_3_inst_cost_2020 + heat_exchanger_asep_4_inst_cost_2020

compressor_asep_1_inst_cost_2020 = 1987.92 * (1+inflation_rate)10 * n2_demand_1**
compressor_asep_2_inst_cost_2020 = 1914.76 * (1+inflation_rate)10 * n2_demand_1**
compressor_asep_3_inst_cost_2020 = 2136.32 * (1+inflation_rate)10 * n2_demand_1**
total_compressors_asep_inst_cost = compressor_asep_1_inst_cost_2020 + compressor_asep_2_inst_cost_2020 + compressor_asep_3_inst_cost_2020

driver_asep_1_inst_cost_2020 = 630.32 * (1+inflation_rate)10 * n2_demand_1**
driver_asep_2_inst_cost_2020 = 620.16 * (1+inflation_rate)10 * n2_demand_1**
driver_asep_3_inst_cost_2020 = 650 * (1+inflation_rate)10 * n2_demand_1**
total_drivers_asep_inst_cost = driver_asep_1_inst_cost_2020 + driver_asep_2_inst_cost_2020 + driver_asep_3_inst_cost_2020

tower_asep_1_inst_cost_2020 = 6634.68 * (1+inflation_rate)10 * n2_demand_1**
tower_asep_2_inst_cost_2020 = 5040.2 * (1+inflation_rate)10 * n2_demand_1**
tower_asep_3_inst_cost_2020 = 908.68 * (1+inflation_rate)10 * n2_demand_1**
total_towers_asep_inst_cost = tower_asep_1_inst_cost_2020 + tower_asep_2_inst_cost_2020 + tower_asep_3_inst_cost_2020

turbine_asep_inst_cost_2020 = 356 * (1+inflation_rate)10 * n2_demand_1**
total_turbine_asep_inst_cost = turbine_asep_inst_cost_2020
total_asep_cost = total_heat_exchangers_asep_inst_cost + total_compressors_asep_inst_cost + total_drivers_asep_inst_cost + total_towers_asep_inst_cost + total_turbine_asep_inst_cost

lhv_h2 = 120.1
h2_per_second_kg = h2_demand_2*1000/86400
h2_capacity_MW = h2_per_second_kg * lhv_h2

h2_investment_MW_h2 = 628000
electrolyser_cost = h2_investment_MW_h2 * h2_capacity_MW

operating_hours = 365*24*0.95
stack_lifetime = 90000
stack_replacement_year = stack_lifetime/operating_hours

```
stack_replacement_number = 1
replacement_cost = electrolyser_cost*0.6*0.45*np.round(stack_replacement_number)

total_electrolyser_cost = electrolyser_cost + replacement_cost

heat_exchanger_ar_1_inst_cost_2020 = 10.055333 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_2_inst_cost_2020 = 732.0333 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_3_inst_cost_2020 = 2.035066 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_4_inst_cost_2020 = 1425.9 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_5_inst_cost_2020 = 1334.833 * (1+inflation_rate)**10 * capacity
total_heat_exchanger_ar_cost = heat_exchanger_ar_1_inst_cost_2020 +
heat_exchanger_ar_2_inst_cost_2020 + heat_exchanger_ar_4_inst_cost_2020 +
heat_exchanger_ar_4_inst_cost_2020

compressor_ar_1_inst_cost_2020 = 7387.33 * (1+inflation_rate)**10 * capacity
compressor_ar_2_inst_cost_2020 = 4734 * (1+inflation_rate)**10 * capacity
total_compressor_ar_cost = compressor_ar_1_inst_cost_2020 +
compressor_ar_2_inst_cost_2020

driver_ar_1_inst_cost_2020 = 943.367 * (1+inflation_rate)**10 * capacity
driver_ar_2_inst_cost_2020 = 702.4 * (1+inflation_rate)**10 * capacity
total_driver_ar_cost = driver_ar_1_inst_cost_2020 + driver_ar_2_inst_cost_2020


reactor_ar_1_inst_cost_2020 = 5623 * (1+inflation_rate)**10 * capacity
reactor_ar_2_inst_cost_2020 = 4926.33 * (1+inflation_rate)**10 * capacity
total_reactor_ar_cost = reactor_ar_1_inst_cost_2020 + reactor_ar_2_inst_cost_2020

pump_ar_inst_cost_2020 = 1939.46 * (1+inflation_rate)**10 * capacity
total_pump_ar_cost = pump_ar_inst_cost_2020
total_ar_cost = total_heat_exchanger_ar_cost + total_compressor_ar_cost +
total_driver_ar_cost + total_reactor_ar_cost + total_pump_ar_cost

storage_as_inst_cost_2020 = 718.89*storage_as_capacity*(1+inflation_rate)**10
total_as_cost = storage_as_inst_cost_2020

total_capital_cost = total_mvc_inst_cost + total_asep_cost + total_ar_cost + total_as_cost
total_estimated_capital_cost_2020 = cost_estimation(total_capital_cost, capacity_new,
capacity, exponent) + total_electrolyser_cost
electricity_cost_y = electricity_price * 1000 * p * 365 * 24/10**6
total_water_cost = water_cost * mass_water_el_2 * 365/10**6
emission_cost_annual = (p * CO2_emmisions_from_electricity_grid_norway * 365 *
CO2_cost)/(1000*10**6)
operation_maintance_cost_annual = 0.05 * total_estimated_capital_cost_2020/10**6
property_taxes_annual = 0.02 * total_estimated_capital_cost_2020/10**6
```

```
    rent_of_land_annual = 0.02 * total_estimated_capital_cost_2020/10**6
    labor_cost_annual = 50000/10**6 ## $ per year
    total_revenue = (ammonia_market_price_2019_breakeven * capacity_new * 365 +
O2_cost)/10**6
    depreciation_cost_y = total_estimated_capital_cost_2020/(plant_life*10**6)

    years = np.arange(0, plant_life, 1)
    Years = np.round(years, 0)
    capital_cost_yr=np.repeat(0,plant_life).tolist()
    capital_cost_yr[0]=(total_estimated_capital_cost_2020*0.5 - replacement_cost)/10**6
    capital_cost_yr[1]=(total_estimated_capital_cost_2020*0.5 - replacement_cost)/10**6
    capital_cost_yr[11]=replacement_cost/10**6


operation_maintance_cost=np.repeat(operation_maintance_cost_annual,plant_life).tolist()
    labor_cost=np.repeat(labor_cost_annual,plant_life).tolist()
    property_tax_cost=np.repeat(property_taxes_annual,plant_life).tolist()
    emission_cost=np.repeat(emission_cost_annual,plant_life).tolist()
    depreciation_cost=np.repeat(depreciation_cost_y,plant_life).tolist()
    electricity_cost = np.repeat(electricity_cost_y,plant_life).tolist()
    total_water_cost = np.repeat(total_water_cost, plant_life).tolist()
    total_revenue=np.repeat(total_revenue,plant_life).tolist()

    for i in range(2):
        #ammonia_production[i] = 0
        operation_maintance_cost[i] = 0
        labor_cost[i] = 0
        emission_cost[i] = 0
        depreciation_cost[i] = 0
        property_tax_cost[i] = 0
        electricity_cost[i] = 0
        total_water_cost[i] = 0
        total_revenue[i] = 0

    tax_cost=[]
    for i in range(len(operation_maintance_cost)):
        tax_cost.append((total_revenue[i]-(capital_cost_yr[i] +
operation_maintance_cost[i]+emission_cost[i]+
                                labor_cost[i]  + total_water_cost[i] +  depreciation_cost[i]  +
property_tax_cost[i]   +  electricity_cost[i]))*tax_rate)
    tax_cost[0]=0
    tax_cost[1]=0

    cash_flow=[]
    for i in range(len(total_revenue)):
```

```
        cash_flow.append(total_revenue[i]-(capital_cost_yr[i] + emission_cost[i]
                                  + operation_maintance_cost[i] + labor_cost[i] + total_water_cost[i]
+ property_tax_cost[i] +
                                   electricity_cost[i] +tax_cost[i]))


    npv = npf.npv(DiscountRate, cash_flow)
    irr = npf.irr(cash_flow)
    return [npv, irr, cash_flow]
```

Line 40:

```
baseline2=[tax_rate, DiscountRate, ammonia_market_price_2019_breakeven,
electricity_price, electrolyser_efficiency, capacity_new]
baseline_npv2=npv_sc2(*baseline2)[0]
baseline_npv2
```

Line 41:

```
variation=np.arange(-20,21,5)
t=[1+(i/100) for i in variation] # P varies from -1.20 to 1.20
change=[x*100 for x in t]
tax_rate_n=[tax_rate*x for x in t]
DiscountRate_n=[DiscountRate*x for x in t]
ammonia_market_price_2019_breakeven_n=[ammonia_market_price_2019_breakeven*x
for x in t]
electricity_price_n=[electricity_price*x for x in t]
electrolyser_efficiency_n=[electrolyser_efficiency*x for x in t]
capacity_new_n=[capacity_new*x for x in t]
```

Line 42:

```
#Tax_rate is varied
tax_rate_s=[npv_sc2(tax_rate_n[i],DiscountRate,ammonia_market_price_2019_breakeven,
                electricity_price,electrolyser_efficiency,capacity_new)[0]
            for i in range(len(tax_rate_n))]
#Interest rate is varied
DiscountRate_s=[npv_sc2(tax_rate,DiscountRate_n[i],ammonia_market_price_2019_break
even,
                electricity_price,electrolyser_efficiency,capacity_new)[0]
            for i in range(len(DiscountRate_n))]

#Ammonia market price is varied
ammonia_market_price_2019_breakeven_s=[npv_sc2(tax_rate,DiscountRate,ammonia_ma
rket_price_2019_breakeven_n[i],
                electricity_price,electrolyser_efficiency,capacity_new)[0]
            for i in range(len(ammonia_market_price_2019_breakeven_n))]
```

```
#Electricity price is varied
electricity_price_s=[npv_sc2(tax_rate,DiscountRate,ammonia_market_price_2019_breake
ven,
              electricity_price_n[i],electrolyser_efficiency,capacity_new)[0]
          for i in range(len(electricity_price_n))]

#Electrolyser efficiency is varied
electrolyser_efficiency_s=[npv_sc2(tax_rate,DiscountRate,ammonia_market_price_2019_b
reakeven,
              electricity_price,electrolyser_efficiency_n[i],capacity_new)[0]
          for i in range(len(electrolyser_efficiency_n))]


#Capacity new is varied
capacity_new_s=[npv_sc2(tax_rate,DiscountRate,ammonia_market_price_2019_breakeven
,
              electricity_price,electrolyser_efficiency,capacity_new_n[i])[0]
          for i in range(len(capacity_new_n))]
```

Line 43:

```
if baseline_npv2<0:
   tax_rate_npv=[((x -baseline_npv2)/(baseline_npv2))*100 for x in tax_rate_s]

   DiscountRate_npv=[((baseline_npv2-x)/(baseline_npv2))*100 for x in DiscountRate_s]

   ammonia_market_price_2019_breakeven_npv=[((baseline_npv2-x)/(baseline_npv2))*100
for x in ammonia_market_price_2019_breakeven_s]

   electricity_price_npv=[((baseline_npv2-x)/(baseline_npv2))*100 for x in
electricity_price_s]

   electrolyser_efficiency_npv=[((baseline_npv2-x)/(baseline_npv2))*100 for x in
electrolyser_efficiency_s]

   capacity_new_npv=[((baseline_npv2-x)/(baseline_npv2))*100 for x in capacity_new_s]
```

Line 44:

```
change_variables_per=['-20', '-15','-10', '-5', '0', '+5', '+10', '+15', '+20']
```

Line 45:

```
npv_dict={'Tax rate': tax_rate_npv,'Discount rate':DiscountRate_npv,'Breakeven
Ammonia market price':ammonia_market_price_2019_breakeven_npv,
        'Electricity price':electricity_price_npv,'Electrolyser
efficiency':electrolyser_efficiency_npv,
        'Capacity of the green ammonia
plant':capacity_new_npv,'%Change':change_variables_per}
```

Line 46:

```
df_npv=pd.DataFrame.from_dict(npv_dict,orient='index')
df_npv=df_npv.transpose()
df_npv
```

Line 47:

```
change_variables_per=['-20', '-15','-10', '-5', '0', '+5', '+10', '+15', '+20']
```

Line 48:

```
df_npv.plot.bar(x='%Change',y=['Tax rate', 'Discount rate', 'Breakeven Ammonia market price',
                    'Electricity price', 'Electrolyser efficiency', 'Capacity of the green ammonia plant']
            ,stacked=False,figsize=(15,10))
plt.xlabel('Change in value of the variables in percentage',Fontsize=25)
plt.ylabel('Change in value of the NPV in percentage',Fontsize=25)
plt.legend(loc='best')
plt.savefig('NPV_percentage_change_sc2.pdf',dpi=1200, bbox_inches='tight')
plt.savefig('NPV_percentage_change_sc2.png',dpi=1200, bbox_inches='tight')
```

Line 49:

```
#Breaeven study Scenario 2
def npv_sc2_f(tax_rate, DiscountRate, ammonia_market_price_2019, electricity_price, electrolyser_efficiency, capacity_new):
    def greenammonia(capacity):
        h2_demand = capacity*(3*mol_weight_h2*conversion1)
        n2_demand = capacity*(mol_weight_n2*conversion1)
        h2_demand_kgh = (h2_demand/24)*1000
        n2_demand_kgh = (n2_demand/24)*1000
        h2_demand_Nm3_per_h = h2_demand_kgh/conversion2
        electrolyser_capacity_MW = (h2_demand_Nm3_per_h*electrolyser_efficiency)/1000
        rectifier_efficiency = 0.95
        total_electrolyser_power = electrolyser_capacity_MW/rectifier_efficiency
        mass_water_el = h2_demand/(mol_weight_h2*conversion3*2
        mass_o2 = mol_weight_o2*conversion3*mass_water_el
        adiabatic_efficiency = 0.75
        driver_efficiency = 0.95
        power_per_1_ton = 0.71
        kg_hydrogen = 8.981
        p_water = power_per_1_ton*mass_water_el/(adiabatic_efficiency*driver_efficiency*1000)
        n2_eff = 0.79
        m_air = n2_demand/n2_eff
        power_ton_n2 = 2.25
```

```
        power_air_separation =
(m_air*power_ton_n2/(adiabatic_efficiency*driver_efficiency))/1000
        Cp_h = A_h + B_h*t_h + C_h*t_h**2 + D_h*t_h**3 + E_h/t_h**2
        Q_h = ((h2_demand*1000)/(24*3600)*Cp_h*500*(T2 - T1_h))/10**6
        Cp_n = A_n + B_n*t_n + C_n*t_n**2 + D_n*t_n**3 + E_n/t_n**2
        Q_n = ((n2_demand*1000)/(24*3600)*Cp_n*1000/28*(T2 - T1_n))/10**6
        total_heaters_power = Q_h + Q_n
        P_per_ammonia = 0.02633
        P_per_day=P_per_ammonia*capacity
        M_water_per_minute = 9.5
        P_water_pumping = 7.3*10**-6
        P_water_total = M_water_per_minute*60*24*P_water_pumping
        P_reactor = P_per_day + P_water_total
        P_compressor = 0.006827
        h2_cost_green = electrolyser_efficiency/conversion2
        h2_cost_smr = 4.5 * conversion2
        CO2_per_1_kg_h2 = 7
        CO2_smr = CO2_per_1_kg_h2*h2_demand*1000
        total_power = (total_electrolyser_power + p_water + power_air_separation +
total_heaters_power + P_reactor + P_compressor)
        return(total_power)
    p = greenammonia(400)

    capacity = 300
    capaciity_new  = 400

    compressor_mvc_inst_cost_2020 = 984.3 * (1+inflation_rate)**10*mass_water_el_1
    driver_mvc_inst_cost_2020 = 307.9 * (1+inflation_rate)**10*mass_water_el_1
    evaporator_mvc_inst_cost_2020 = 7411.34 * (1+inflation_rate)**10*mass_water_el_1
    distillate_mvc_inst_cost_2020 = 2729.62 * (1+inflation_rate)**10*mass_water_el_1
    brine_mvc_inst_cost_2020 = 6286.42 * (1+inflation_rate)**10*mass_water_el_1
    total_mvc_inst_cost = compressor_mvc_inst_cost_2020 + driver_mvc_inst_cost_2020 +
evaporator_mvc_inst_cost_2020 + distillate_mvc_inst_cost_2020 +
brine_mvc_inst_cost_2020

    heat_exchanger_asep_1_inst_cost_2020 = 332.8 * (1+inflation_rate)**10 * n2_demand_1
    heat_exchanger_asep_2_inst_cost_2020 = 340.96 * (1+inflation_rate)**10 *
n2_demand_1
    heat_exchanger_asep_3_inst_cost_2020 = 421.4 * (1+inflation_rate)**10 * n2_demand_1
    heat_exchanger_asep_4_inst_cost_2020 = 5715.52 * (1+inflation_rate)**10 *
n2_demand_1
    total_heat_exchangers_asep_inst_cost = heat_exchanger_asep_1_inst_cost_2020 +
heat_exchanger_asep_2_inst_cost_2020 + heat_exchanger_asep_3_inst_cost_2020 +
heat_exchanger_asep_4_inst_cost_2020
```

```
compressor_asep_1_inst_cost_2020 = 1987.92 * (1+inflation_rate)**10 * n2_demand_1
compressor_asep_2_inst_cost_2020 = 1914.76 * (1+inflation_rate)**10 * n2_demand_1
compressor_asep_3_inst_cost_2020 = 2136.32 * (1+inflation_rate)**10 * n2_demand_1
total_compressors_asep_inst_cost = compressor_asep_1_inst_cost_2020 +
compressor_asep_2_inst_cost_2020 + compressor_asep_3_inst_cost_2020

driver_asep_1_inst_cost_2020 = 630.32 * (1+inflation_rate)**10 * n2_demand_1
driver_asep_2_inst_cost_2020 = 620.16 * (1+inflation_rate)**10 * n2_demand_1
driver_asep_3_inst_cost_2020 = 650 * (1+inflation_rate)**10 * n2_demand_1
total_drivers_asep_inst_cost = driver_asep_1_inst_cost_2020 +
driver_asep_2_inst_cost_2020 + driver_asep_3_inst_cost_2020

tower_asep_1_inst_cost_2020 = 6634.68 * (1+inflation_rate)**10 * n2_demand_1
tower_asep_2_inst_cost_2020 = 5040.2 * (1+inflation_rate)**10 * n2_demand_1
tower_asep_3_inst_cost_2020 = 908.68 * (1+inflation_rate)**10 * n2_demand_1
total_towers_asep_inst_cost = tower_asep_1_inst_cost_2020 +
tower_asep_2_inst_cost_2020 + tower_asep_3_inst_cost_2020

turbine_asep_inst_cost_2020 = 356 * (1+inflation_rate)**10 * n2_demand_1
total_turbine_asep_inst_cost = turbine_asep_inst_cost_2020
total_asep_cost = total_heat_exchangers_asep_inst_cost +
total_compressors_asep_inst_cost + total_drivers_asep_inst_cost +
total_towers_asep_inst_cost + total_turbine_asep_inst_cost


lhv_h2 = 120.1 #MJ/kg
h2_per_second_kg = h2_demand_2*1000/86400
h2_capacity_MW = h2_per_second_kg * lhv_h2

h2_investment_MW_h2 = 628000
electrolyser_cost = h2_investment_MW_h2 * h2_capacity_MW

operating_hours = 365*24*0.95
stack_lifetime = 90000
stack_replacement_year = stack_lifetime/operating_hours
stack_replacement_number = 1
replacement_cost = electrolyser_cost*0.6*0.45*np.round(stack_replacement_number)

total_electrolyser_cost = electrolyser_cost + replacement_cost

heat_exchanger_ar_1_inst_cost_2020 = 10.055333 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_2_inst_cost_2020 = 732.0333 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_3_inst_cost_2020 = 2.035066 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_4_inst_cost_2020 = 1425.9 * (1+inflation_rate)**10 * capacity
heat_exchanger_ar_5_inst_cost_2020 = 1334.833 * (1+inflation_rate)**10 * capacity
```

```
    total_heat_exchanger_ar_cost = heat_exchanger_ar_1_inst_cost_2020 +
heat_exchanger_ar_2_inst_cost_2020 + heat_exchanger_ar_4_inst_cost_2020 +
heat_exchanger_ar_4_inst_cost_2020

    compressor_ar_1_inst_cost_2020 = 7387.33 * (1+inflation_rate)**10 * capacity
    compressor_ar_2_inst_cost_2020 = 4734 * (1+inflation_rate)**10 * capacity
    total_compressor_ar_cost = compressor_ar_1_inst_cost_2020 +
compressor_ar_2_inst_cost_2020

    driver_ar_1_inst_cost_2020 = 943.367 * (1+inflation_rate)**10 * capacity
    driver_ar_2_inst_cost_2020 = 702.4 * (1+inflation_rate)**10 * capacity
    total_driver_ar_cost = driver_ar_1_inst_cost_2020 + driver_ar_2_inst_cost_2020


    reactor_ar_1_inst_cost_2020 = 5623 * (1+inflation_rate)**10 * capacity
    reactor_ar_2_inst_cost_2020 = 4926.33 * (1+inflation_rate)**10 * capacity
    total_reactor_ar_cost = reactor_ar_1_inst_cost_2020 + reactor_ar_2_inst_cost_2020

    pump_ar_inst_cost_2020 = 1939.46 * (1+inflation_rate)**10 * capacity
    total_pump_ar_cost = pump_ar_inst_cost_2020
    total_ar_cost = total_heat_exchanger_ar_cost + total_compressor_ar_cost +
total_driver_ar_cost + total_reactor_ar_cost + total_pump_ar_cost

    storage_as_inst_cost_2020 = 718.89*storage_as_capacity*(1+inflation_rate)**10
    total_as_cost = storage_as_inst_cost_2020

    total_capital_cost = total_mvc_inst_cost + total_asep_cost + total_ar_cost + total_as_cost
    total_estimated_capital_cost_2020 = cost_estimation(total_capital_cost, capacity_new,
capacity, exponent) + total_electrolyser_cost
    electricity_cost_y = electricity_price * 1000 * p * 365 * 24/10**6
    total_water_cost = water_cost * mass_water_el_2 * 365/10**6

    emission_cost_annual = (p * CO2_emmisions_from_electricity_grid_norway * 365 *
CO2_cost)/(1000*10**6)
    operation_maintance_cost_annual = 0.05 * total_estimated_capital_cost_2020/10**6
    property_taxes_annual = 0.02 * total_estimated_capital_cost_2020/10**6
    rent_of_land_annual = 0.02 * total_estimated_capital_cost_2020/10**6
    labor_cost_annual = 50000/10**6
    total_revenue = (ammonia_market_price_2019 * capacity_new * 365 + O2_cost)/10**6
    depreciation_cost_y = total_estimated_capital_cost_2020/(plant_life*10**6)

    years = np.arange(0, plant_life, 1)
    Years = np.round(years, 0)
    #ammonia_production=np.repeat((capacity_new*365)/10**6,plant_life).tolist()
    capital_cost_yr=np.repeat(0,plant_life).tolist()
```

```
    capital_cost_yr[0]=(total_estimated_capital_cost_2020*0.5 - replacement_cost)/10**6
    capital_cost_yr[1]=(total_estimated_capital_cost_2020*0.5 - replacement_cost)/10**6
    capital_cost_yr[11]=replacement_cost/10**6


operation_maintance_cost=np.repeat(operation_maintance_cost_annual,plant_life).tolist()
    labor_cost=np.repeat(labor_cost_annual,plant_life).tolist()
    property_tax_cost=np.repeat(property_taxes_annual,plant_life).tolist()
    emission_cost=np.repeat(emission_cost_annual,plant_life).tolist()
    depreciation_cost=np.repeat(depreciation_cost_y,plant_life).tolist()
    electricity_cost = np.repeat(electricity_cost_y,plant_life).tolist()
    total_water_cost = np.repeat(total_water_cost, plant_life).tolist()
    total_revenue=np.repeat(total_revenue,plant_life).tolist()

    for i in range(2):
        #ammonia_production[i] = 0
        operation_maintance_cost[i] = 0
        labor_cost[i] = 0
        emission_cost[i] = 0
        depreciation_cost[i] = 0
        property_tax_cost[i] = 0
        electricity_cost[i] = 0
        total_water_cost[i] = 0
        total_revenue[i] = 0

    tax_cost=[]
    for i in range(len(operation_maintance_cost)):
        tax_cost.append((total_revenue[i]-(capital_cost_yr[i] +
operation_maintance_cost[i]+emission_cost[i]+
                                labor_cost[i]  + total_water_cost[i] +  depreciation_cost[i]  +
property_tax_cost[i]   +  electricity_cost[i]))*tax_rate)

    tax_cost[0]=0
    tax_cost[1]=0

    cash_flow=[]
    for i in range(len(total_revenue)):
        cash_flow.append(total_revenue[i]-(capital_cost_yr[i] + emission_cost[i]
                              + operation_maintance_cost[i] + labor_cost[i] + total_water_cost[i]
+ property_tax_cost[i] +
                               electricity_cost[i] +tax_cost[i]))


    npv_sc2_f = npf.npv(DiscountRate, cash_flow)
    return (npv_sc2_f)
```

Line 50:

```
ammonia_market_price_2019_array = np.arange(start=200,stop=500,step=0.1)
npv_end_array = np.zeros(ammonia_market_price_2019_array.shape)
for i in range(ammonia_market_price_2019_array.shape[0]):
    npv_end_array[i] =npv_sc2_f(tax_rate, DiscountRate,
ammonia_market_price_2019_array[i], electricity_price, electrolyser_efficiency,
capacity_new)
fig_SA = px.line(x=ammonia_market_price_2019_array, y=npv_end_array,
labels={'x':'Ammonia market price','y':'Net Present Value of green ammonia plant'})
fig_SA.show()
```

Line 51:

```
NPV_target = 0
squared_error = lambda x_ammonia_market_price_2019 : (npv_sc2_f(tax_rate,
DiscountRate, x_ammonia_market_price_2019, electricity_price, electrolyser_efficiency,
capacity_new)-NPV_target)**2
result = differential_evolution(func=squared_error, bounds=[(200, 500),])
ammonia_price = result.x[0]
print('Breakeven ammonia price is:', money_format(ammonia_price))
```

Line: 52

```
#Part 3. Emission calculation and SEC of the green ammonia production plant.
# Emission calculation.
CI_countries = np.array([0.386,0.413,0.574,0.016,0.025]) ## t/MWh
countries=['DK','EU28','DE','SE','NO']
NG_based_ammonia_plant = np.array([1.19*7.4155, 1.19*7.4155, 1.19*7.4155, 1.19*7.4155,
1.19*7.4155]) ## tonCO2/tonNH3
other_sys = np.array([0.5845, 0.5845, 0.5845, 0.5845, 0.5845]) #MWh
ty  = CI_countries*other_sys
total_ng_based_co2 = ty + NG_based_ammonia_plant
print(total_ng_based_co2)
print (ty)
elecrtolyser_based = np.array([10.02, 10.02, 10.02, 10.02, 10.02])
total_elect_co2 = CI_countries*elecrtolyser_based
print(total_elect_co2)
```

Line 53:

```
emission_countries=np.array([total_ng_based_co2, total_elect_co2])
emission_countries
```

Line 54:

```
df_emission=pd.DataFrame(emission_countries).T
df_emission.columns=['total_ng_based_co2', 'total_elect_co2']
```

```
df_emission['Countries']=countries
```

Line 55:

```
labels = ['DK','EU28','DE','SE','NO']
m_means = [9.050062 , 9.0658435, 9.159948 , 8.833797 , 8.8390575]
w_means = [3.86772  , 4.13826  , 5.75148  , 0.16032  , 0.2505  ]

x = np.arange(len(labels))  # the label locations
width = 0.25 # the width of the bars

fig, ax = plt.subplots()
rects1 = ax.bar(x - width/2, m_means, width, label='SMR')
rects2 = ax.bar(x + width/2, w_means, width, label='Electrolyser')

ax.set_ylabel('Emissionns in tCO2/tNH3', fontsize=15)
ax.set_xlabel('Countries', fontsize=15)
#ax.set_title()
ax.set_xticks(x)
ax.set_xticklabels(labels)
ax.legend()

fig.set_figwidth(10)
fig.set_figheight(8)

fig.tight_layout()
plt.savefig('SMR vs Electrolyser emissions.pdf',dpi=1200,bbox_inches='tight')
```

Line 56:

```
# Specific energy consumption.
capacity = 1 # kg
mol_weight_h2 = 1.00794 #g/mol
mol_weight_n2 = 14.0067 #g/mol
mol_weight_o2 = 15.999 #g/mol
mol_weight_h20 = 18.01488 #g/mol
mol_wt_nh3 = 17.03052 #g/mol
conversion1 = 0.058718113 #Mol in 1 g NH3
conversion2 = 0.08281 #from kg to Nm3 Hydrogen
conversion3 = 0.055509668 #mol per 1 kg H2O
conversion4 = 0.79961 # kg to Nm3 Nitrogen
electrolyser_efficiency = 4.4 # kWh/Nm3
cryogenic_efficiency = 0.552 # kWh/Nm3
MVC_efficiency = 10 # kWh/m3
T2 = 450
t2 = (T2 + 273.15)/1000
A_h = 33.066178
```

```
B_h = -11.363417
C_h = 11.432816
D_h = -2.772874
E_h = -0.158558
F_h = -9.980797
G_h = 172.707974
H_h = 0
T1_h = 80
t_h = (T1_h + 273.15)/1000;
A_n = 28.98641
B_n = 1.853978
C_n = -9.647459
D_n = 16.63537
E_n = 0.000117
F_n = -8.671914
G_n = 226.4168
H_n = 0
T1_n = -172
t_n = (T1_n + 273.15)/1000;
```

Line 57:

```
# Demand for nitrogen and hydrogen.
h2_demand = capacity*(3*mol_weight_h2*conversion1)
n2_demand = capacity*(mol_weight_n2*conversion1)
h2_demand_kgh = h2_demand/24
n2_demand_kgh = n2_demand/24
h2_demand_Nm3_per_h = h2_demand_kgh/conversion2
print('Hydrogen demand for 1 kg of ammonia', h2_demand, 'kg')
print('Nitrogen demand for 1 kg of ammonia', n2_demand, 'kg')
```

Line 58:

```
# Electrolyser power requirements.
electrolyser_capacity_MW = (h2_demand_Nm3_per_h*electrolyser_efficiency)/1000
rectifier_efficiency = 0.9
total_electrolyser_power = electrolyser_capacity_MW/rectifier_efficiency
```

Line 59:

```
# SEC of hydrogen production.
SEC_hydrogen = electrolyser_efficiency/(1000*conversion2) # per 1 kg of NH3
SEC_hydrogen_ton = SEC_hydrogen * 1000 # per 1 ton of NH3
print('Specific Energy Consumption for Hydrogen production:', SEC_hydrogen, 'MWh/kg H2')
print('Specific Energy Consumption for Hydrogen production:', SEC_hydrogen_ton, 'MWh/ton H2')
```

Line 60:

```
# Amount of water for purification.
mass_water_el = h2_demand/(mol_weight_h2*conversion3*2)
mass_o2 = mol_weight_o2*conversion3*mass_water_el
adiabatic_efficiency = 0.75
driver_efficiency = 0.95
power_per_1_ton = 0.71
kg_hydrogen = 8.981
p_water = power_per_1_ton*mass_water_el/(adiabatic_efficiency*driver_efficiency*1000)
```

Line 61:

```
# SEC of water purification unit.
SEC_water = MVC_efficiency/1000000
print('Specific Energy Consumption for Water purification:', SEC_water, 'kWh/kgwater')
```

Line 62:

```
# Amount of air for CAS.
n2_eff = 0.79
m_air = n2_demand/n2_eff
power_ton_n2 = 2.25/1000
power_air_separation =
(m_air*power_ton_n2/(adiabatic_efficiency*driver_efficiency))/1000#MW
```

Line 63:

```
# SEC of nitrogen production.
SEC_nitrogen = cryogenic_efficiency/(1000*conversion4)
SEC_nitrogen_ton = SEC_nitrogen*1000
print('Specific Energy Consumption for Nitrogen production:', SEC_nitrogen,
'MWh/kgN2')
print('Specific Energy Consumption for Nitrogen production:', SEC_nitrogen_ton,
'MWh/tonN2')
```

Line 64:

```
# Heater for hydrogen.
Cp_h = A_h + B_h*t_h + C_h*t_h**2 + D_h*t_h**3 + E_h/t_h**2
Q_h = ((h2_demand*1000)/(24*3600)*Cp_h*500*(T2 - T1_h))/10**6

# Heater for nitrogen
Cp_n = A_n + B_n*t_n + C_n*t_n**2 + D_n*t_n**3 + E_n/t_n**2
Q_n = ((n2_demand*1000)/(24*3600)*Cp_n*1000/28*(T2 - T1_n))/10**6
total_heaters_power = Q_h + Q_n
# Delta enthalpy Hydrogen
h_1_hydrogen = A_h*t_h + B_h*t_h**2/2 + C_h*t_h**3/3 + D_h*t_h**4/4 - E_h/t_h + F_h
- H_h
h_2_hydrogen = A_h*t2 + B_h*t2**2/2 + C_h*t2**3/3 + D_h*t2**4/4 - E_h/t2 + F_h - H_h
```

```
delta_h_hydrogen = h_2_hydrogen - h_1_hydrogen # kJ/mol

# Heater for nitrogen.
Cp_n = A_n + B_n*t_n + C_n*t_n**2 + D_n*t_n**3 + E_n/t_n**2
Q_n = ((n2_demand*1000)/(24*3600)*Cp_n*1000/28*(T2 - T1_n))/10**6
total_heaters_power = Q_h + Q_n
# Delta enthalpy Nitrogen
h_1_nitrogen = A_n*t_n + B_n*t_n**2/2 + C_n*t_n**3/3 + D_n*t_n**4/4 - E_n/t_n + F_n -
H_n
h_2_nitrogen = A_n*t2 + B_n*t2**2/2 + C_n*t2**3/3 + D_n*t2**4/4 - E_n/t2 + F_n - H_n
delta_h_nitrogen = h_2_nitrogen - h_1_nitrogen # kJ/mol
```

Line 65:

```
# Ammonia reactor.
P_per_ammonia = 0.02633/1000
P_per_day=P_per_ammonia*capacity
M_water_per_minute = 9.5*1000
P_water_pumping = 7.3*10**-6
P_water_total = M_water_per_minute*60*24*P_water_pumping
P_reactor = P_per_day + P_water_total
```

Line 66:

```
# SEC of ammonia conversion.
delta_h_ammonia = 46.2 # kJ/mol
kWh_conversion = 3.6 * 10**6 # 3.6 * 10**6 Joules per 1 kWh
SEC_ammonia = ((delta_h_ammonia*1000/mol_wt_nh3)/(1000*kWh_conversion))
print('Specific Energy Consumption of Ammonia Reaction:', SEC_ammonia,
'MWh/kgNH3')

SEC_heater_h2 = ((delta_h_hydrogen*1000/mol_weight_h2)/kWh_conversion)
#print(SEC_heater_h2)

SEC_heater_n2 = ((delta_h_nitrogen*1000/mol_weight_n2)/kWh_conversion)
#print(SEC_heater_n2)

h = (h2_demand * SEC_heater_h2 + n2_demand * SEC_heater_n2 + SEC_ammonia)/1000
print('Specific Energy Consumption of ammonia reaction coupled with preheaters for
Hydrogen and Nitrogen:', h, 'MWh/kg NH3')
```

Line 67:

```
# Overall SEC
SEC_overall = SEC_hydrogen*h2_demand + SEC_water*mass_water_el +
SEC_nitrogen*n2_demand + h
print('Total Specific Energy Consumption for 1 kg of NH3:', SEC_overall, 'MWh/kg NH3')
SEC_overall_ton = SEC_overall * 1000
```

```
print('Total Specific Energy Consumption for 1 ton of NH3:', SEC_overall_ton, 'MWh/ton NH3')
```

Line 68:

```
labels = 'Electrolyser unit','Other subsystems'
sizes =
[SEC_hydrogen*h2_demand*1000,SEC_water*mass_water_el*1000+SEC_nitrogen*n2_demand*1000+h*1000]
explode = (0, 0.1) # only "explode" the 2nd slice (i.e. 'Hogs')

fig1, ax1 = plt.subplots(figsize=(15,10))
ax1.pie(sizes,labels=labels, autopct='%1.1f%%',explode=explode,
    shadow=False, startangle=90,textprops={'fontsize': 15})
ax1.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.
plt.annotate('Total SEC=10.02 MWh/tNH3', xy=(0.39, 0.34), xycoords='axes fraction',fontsize=18)
plt.savefig('SEC_NH3.pdf',dpi=1200)
plt.savefig('SEC_NH3.png',dpi=1200)
plt.savefig('SEC_NH3_ppt.png',dpi=300)
```

References.

[1] IPCC 2018, “Summary for Policymakers. In: Global Warming of 1.5°C. An IPCC Special Report on the impacts of global warming of 1.5°C above pre-industrial levels and related global greenhouse gas emission pathways, in the context of strengthening the global, Tech. rep.,” 2018.

[2] IPCC 2014, “Climate Change 2014: Synthesis Report. Contribution of Working Groups I, II and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change,” 2014.

[3] H. Ritchie, “Sector by sector: where do global greenhouse gas emissions come from?,” *Our World Data*, 2020, [Online]. Available: https://ourworldindata.org/ghg-emissions-by-sector.

[4] T. Brown, “Ammonia production causes 1% of total global GHG emissions,” 2016. .

[5] I. Wilkinson, “Green Ammonia can replace fossil fuel storage at scale,” 2020. .

[6] H. Ritchie, “How many people does synthetic fertilizer feed?,” *Our World Data*, 2017, [Online]. Available: https://ourworldindata.org/how-many-people-does-synthetic-fertilizer-feed.

[7] G. Thomas and G. Parks, “Potential Roles of Ammonia in a Hydrogen Economy,” 2006.

[8] S. H. Crolius, “Ammonia’s Role in the Hydrogen Society,” *Ammon. Energy Assoc.*, 2019, [Online]. Available: https://www.ammoniaenergy.org/articles/ammonias-role-in-the-hydrogen-society/.

[9] Julia Hansson et al., “The Potential Role of Ammonia as Marine Fuel—Based on Energy Systems Modeling and Multi-Criteria Decision Analysis,” *Sustainability*, 2020.

[10] R. Soltani et al, “Assessment of CO2 capture options from various points in steam methane reforming for hydrogen production,” *Int. J. Hydrogen Energy*, pp. 1–10, 2014.

[11] A. Scott, “Ammonia on route to fuel ships and planes.” .

[12] European Commission, “2030 climate & energy framework.” .

[13] Hanfei Zhang et al, “Techno-economic comparison of green ammonia production processes,” *Appl. Energy*, vol. 259, 2020.

[14] C. Fúnez Guerra et al, “Technical-economic analysis for a green ammonia production plant in Chile and its subsequent transport to Japan,” *Renew.*

Energy, vol. 157, pp. 404–414, 2020.

[15] Richard Michael Nayak-Luke et al, "Techno-economic viability of islanded green ammonia as a carbo-free energy vector and as a substitute for conventional production," *Energy Environ. Sci.*, 2020.

[16] G. Thomassen et al, "How to assess the potential of emerging green technologies? Towards a prospective environmental and techno-economic assessment framework," *Green Chem.*, vol. 21, no. 18, pp. 4868–4886, 2019.

[17] Statistics Market Research Consulting Pvt Ltd, "Green Ammonia - Global Market Outlook (2019-2027)," 2020.

[18] Trevor Brown, "Industry report sees multi-billion ton market for green ammonia," *Ammonia Energy Association*, 2020. .

[19] NEL official website, "Athmospheric Alkaline Electrolyser." .

[20] G. D. Ulrich, *A Guide to Chemical Engineering Process Design and Economics*, 1st ed. New York: John Wiley and Sons, 1984.

[21] Y. Alsultannty et al, "Oxygen specific power consumption comparison for air separation units," *Eng. J.*, vol. 18, no. 2, pp. 67–80, 2014.

[22] E. R. Morgan, "Techno-Economic Feasibility Study of Ammonia Plants Powered by Offshore Wind," University of Massachusetts Amherst, 2013.

[23] E. Tzen, "Wind-Powered Desalination-Principles, Configurations, Design, and Implementation," in *Renewable Energy Powered Desalination Handbook: Application and Thermodynamics*, Elsevier Inc., 2018, pp. 91–139.

[24] Marangio F. et al, "Concept of a high pressure PEM electrolyser prototype," *Int. J. Hydrogen Energy*, vol. 36, pp. 7807–7015, 2011.

[25] Stefanie Tesch et al, "Comparative Evaluation of Cryogenic Air Separation Units from the Exergetic and Economic Points of View." .

[26] Anders Nielsen, *Ammonia. Catalysis and Manufacture*. New York: Springer-Verlag, 1995.

[27] J. Ruddock et al, "Energy integration in ammonia production." .

[28] Max Appl, *Ammonia: Principles and Industrial Practice*, 1st ed. Weinheim,: Wiley-VCH Verlag GmbH & Co.:, 1999.

[29] Dubey M. et al, "Conversion and Storage of Wind Energy as Nitrogenous Fertilizer, Technical Report LR 28338," 1978.

[30] G. Fiorenza et al, "Techno-economic evaluation of a solar powered water desalination plant," *Energy Convers. Manag.*, vol. 44, pp. 2217–2240, 2003.

[31] Mirna Rahman Lubis, "Desalination using vapor-compression distillation,"

Texas A&M University, 2009.

[32] Victor McLafferty, “Comparison of Mechanical Vapour Compression Evaporator Technologies,” Dalhousie University, Halifax, Nova Scotia, 2018.

[33] J. R. Jennings, *Catalytic Ammonia Synthesis: Fundamentals and Practice*. New York: Plenum Press, 1991.

[34] European Commission, “Reference Document on Best Available Techniques for the Manufacture of Large Volume Inorganic Chemicals - Ammonia, Acids and Fertilizers,” 2007.

[35] M. W. Chase, *NIST-JANAF thermochemical Tables*, 4th ed., vol. 9. American institute of Physics for the National institute of standards and technology, 1998.

[36] D. Borgars, *Advances in Ammonia Plant Technology*, Borgars DJ. London: The Macmillam Press, Ltd, 1971.

[37] D. W. Green and R. H. Perry, *Perry’s chemical engineers’ handbook*, 8th ed. New York: NY: McGraw-Hill, 2008.

[38] Schmidt O. et al, “Future cost and performance of water electrolysis: An expert elicitation study,” *Int. J. Hydrogen Energy*, vol. 42, pp. 30470–30492, 2017.

[39] G. Towler and R. Sinnott, *Chemical Engineering Design*, 2nd ed. Elsevier, 2013.

[40] Yara Annual Report 2019, “Crop Nutrition Company for the Future,” 2019.

[41] European Commission, “Hydrogen generation in Europe: Overview of costs and key benefits,” Luxembourg, 2020.

[42] Fraunhofer-ISI and Ecofys, “Electricity costs of energy-intensive industries in Norway – a comparison with energy-intensive industries in selected countries,” 2016.

[43] N. Buli and N. Adamaitis, “Norway’s plans to raise carbon tax draw oil industry ire,” 2021. .

[44] C. C. Dorris et al, “High-Purity Oxygen Production Using Mixed Ionic-Electronic Conducting Sorbents,” 2016. [Online]. Available: https://repository.upenn.edu/cgi/viewcontent.cgi?article=1080&context=cbe_sdr.

[45] Alberto Moro et al, “Electricity carbon intensity in European Member States: Impacts on GHG emissions of electric vehicles. Transportation Research Part D: Transport and Environment,” vol. 64, pp. 5–14, 2018.

[46] D. M. Hamby, “A review of techniques for parameter sensitivity analysis of

environmental models," *Environ. Monit. Assess.*, vol. 32, no. 2, pp. 135–154, 1994.

[47] K. Lindblad, "An economic feasibility study of hydrogen production by electrolysis in relation to offshore wind energy at Oxelösund," KTH ROYAL INSTITUTE OF TECHNOLOGY SCHOOL OF ENGINEERING SCIENCES, 2019.

[48] E. R. Morgan et al, "Sustainable Ammonia Production from U.S. Offshore Wind Farms: A Techno-Economic Review," *ACS Sustain. Chem. Eng.*, 2017.

[49] T. Brown, "Round-trip Efficiency of Ammonia as a Renewable Energy Transportation Media," *Ammon. Energy Assoc.*, 2017, [Online]. Available: https://www.ammoniaenergy.org/articles/round-trip-efficiency-of-ammonia-as-a-renewable-energy-transportation-media/#:~:text=Energy input required to produce,-12 MWh%2Fton ammonia.

[50] F. Dolci, "Green hydrogen opportunities in selected industrial processes," Brussels, Belgium, 2018.

[51] S. M. Walas, *Chemical Process Equipment. Selection and Design*. 1988.

[52] R. Turton et al, *Analysis, Synthesis and Design of Chemical Processes*, 3rd ed. Boston, MA: Prentice Hall, 2009.

[53] ISPT, "Power to Ammonia. Feasibility study for the value chains and business cases to produce CO2 -free ammonia suitable for various market applications," 2017.

[54] T. Brown, "Australian solar-ammonia exports to Germany," 2016. .

[55] IRENA, "GREEN HYDROGEN COST REDUCTION. SCALING UP ELECTROLYSERS TO MEET THE 1.5°C H CLIMATE GOAL," 2020. [Online]. Available: https://irena.org/-/media/Files/IRENA/Agency/Publication/2020/Dec/IRENA_Green_hydrogen_cost_2020.pdf.

[56] Fraunhofer-ISI, Ecofys, and GWS, "Electricity Costs of Energy Intensive Industries An International Comparison," 2015.

[57] Argus Blog, "Yara plans green ammonia production in Norway," 2020. .

[58] Yara official website, "Yara partners with Statkraft and Aker Horizons to establish Europe's first large-scale green ammonia project in Norway," 2021. .

[59] SAFETY4SEA, "Norway-funded partnership to build first green ammonia tanker," 2020. .

[60] REVE, "Masdar Agrees to Expand Uzbekistan Wind Farm Capacity to 1.5 GW," 2021. .

[61] Renewables.ninja, “Wind mounthly capacity factor for Zarafshan district (Uzbekistan),” 2021. .

[62] RENEWS.BIZ, “Uzbekistan reveals hydrogen and renewables strategy,” 2021. .

[63] The Tashkent Times, “Navoiazot commissions US$ 1 billion ammonia and urea chemical complex,” 2020. .

[64] Bellona Europa, “Norway proposes €200 per ton CO2 tax by 2030,” 2021. .

[65] R. Nayak-Luke, “Green ammonia: AI optimised decarbonised ammonia production for conventional use and as a viable renewable energy storage solution,” University of Oxford;, 2020.

[66] J. Ikäheimo, J. Kiviluoma, R. Weiss, and H. Holttinen, “Power-toammonia in future North-European 100% renewable power and heat system,” *Int. J. Hydrogen Energy*, vol. 43, pp. 17295–17308, 2018.

[67] S. Beerbühl, M. Fröhling, and F. Schultmann, “Combined scheduling and capacity planning of electricity-based ammonia production to integrate renewable energies,” *Eur. J. Oper. Res.*, vol. 241, pp. 851–862, 2014.