Escuela Politécnica Superior

ARIMA and Artificial Neural Networks to forecast CO_2 emissions allowances price: application to the design of petrochemical supply chain under uncertainty.

Máster en Ingeniería Química

Trabajo Fin de Máster

Autor:
Adrián Amat Bernabéu

Tutor/es:
Rubén Ruiz Femenia

Raquel Salcedo Díaz

Universitat d'Alacant
Universidad de Alicante

Enero 2019

Index

Abstract

To face up the threat of global climate change, governments and regulatory agencies are implementing policies to reduce the greenhouse gas emissions. A common climate change policy is to cap CO_2 emissions and establishing a price through trading. The idea behind this cap-and-trading scheme is to set a price on carbon emissions and in consequence, a financial incentive to decrease them. After a cap is fixed on emissions, companies are allowed to buy or sell from each other the allowances to emit CO_2. Firms exceeding their emissions cap must buy extra credits to cover the excess. Meanwhile, those that do not use all their allowances can sell them, providing the least-polluting firms with an extra revenue. It is expected that the CO_2 emissions allowances price increases with time forcing company towards more sustainable technologies. Hence, the key parameter for a successful result is the evolution of that price with time, which is difficult to predict due to the uncertainty associated to it. One approach to predict that trend is to assume that the emissions allowance price exhibits a similar behavior to a share in the stock market. In this work, we analyze the forecasting performance of the autoregressive integrated moving average (ARIMA) and artificial neural networks by comparing with CO_2 European Emission Allowances prices reported for previous year. To illustrate the usefulness of this methodology, we design a petrochemical supply chain under uncertainty in the emission allowances price.

Keywords

ARIMA; Neural network; forecasting; CO_2 allowances; uncertainty optimization; supply chain.

1. INTRODUCTION

Several research studies on stock predictions have been conducted with various solution techniques proposed over the years [1]. The prominent techniques fall into two broad categories, namely, statistical and soft computing techniques. Statistical techniques include, among others, exponential smoothing and autoregressive integrated moving average (ARIMA) [1]. The ARIMA model, also known as the Box-Jenkins model or methodology, is commonly used in analysis and forecasting. It is widely regarded as the most efficient forecasting technique in social science and is used extensively for time series. The use of ARIMA for forecasting time series is essential with uncertainty as it does not assume knowledge of any underlying model or relationships as in some other methods. ARIMA essentially relies on past values of the series as well as previous error terms for forecasting. However, ARIMA models are relatively more robust and efficient than more complex structural models in relation to short-run forecasting [1].

Predicting the return of financial times series using traditional technical analysis and widely used economic models, has proven to be difficult. Machine learning is a subfield of computer science, a concept that is frequently being used within different domains and recurrently delivers successful results. Artificial neural network is a product from the field of machine learning, a black box model that if properly designed processes data, learns its dynamic and subsequently provides an informative output. Artificial neural networks (ANNs) as a soft computing technique are the most accurate and widely used as forecasting models in many areas including social, engineering, economic, business, finance, foreign exchange, and stock problems [2].

The EU Emissions Trading System (EU ETS) has been the cornerstone of the EU's strategy for reducing greenhouse gas (GHG) emissions from industry and the power sector since 2005. It contributes significantly to the achievement of the EU's target of cutting GHG emissions by 20% from 1990 levels by 2020. The EU ETS operates in the 31 countries of the European Economic Area (EEA). It limits emissions from nearly 11,000 power plants and manufacturing installations. It covers around 45% of the EU's GHG emissions. The emission reduction target has into account the number of allowances put in circulation over a trading period. The 2013 cap for emissions from stationary installations was set at 2 084 301 856 allowances. This cap decreases each year by a linear reduction factor of 1.74% of the average total quantity of allowances issued annually in 2008-2012, thus ensuring that the number of allowances that can be used by stationary installations will be 21% lower in 2020 than in 2005 [3].

In this scenario, the opportunity to manage the CO_2 emissions effectively depends on the capacity of companies to have a global view of its responsibilities and associated cost. Limiting greenhouse gas emissions and establishing a price to trade with them is the essential foundation for Carbon Trading climate change policy. The idea underlying this scheme becomes the reduction of CO_2 emissions in an economic incentive, since it is a limit on emissions, companies can buy or sell among them rights to emit greenhouse gases [4].

The objective of this work is to develop ARIMA and Artificial Neural Networks models that predict CO_2 European Emission Allowances [5] taking into account that exhibit a similar behavior to a share in the stock market. This prediction will be used later in order to design a petrochemical supply chain using a stochastic optimization model with emission allowances price as an uncertain parameter.

2. PROBLEM STATEMENT

In this work, the problem of forecasting the CO_2 European Emission Allowances price and design a petrochemical supply chain with the predicted values were addressed in two different sections.

2.1 Forecasting

The study used the published CO_2 European Emission Allowances price [5] on ARIMA and ANN models developed. MATLAB Econometrics Toolbox and Neural Network Toolbox were used for developing ARIMA and ANNs models, respectively. The data considered here are exclusively the closing prices available from November 26th 2009 until September 28th 2018. These historical prices are shown in *Figure 1*.

Figure 1. Historical prices CO_2 emission allowances in Europe (€/ton CO_2).

2.1 Application of the predicted values in the design of a petrochemical supply chain

Any design depends on the parameters used to achieve the final solution. In the design process of a supply chain or a chemical plant, there is always uncertainty in the data given. For this reason, an accurate forecast model to predict the values of the uncertainty parameters is required in order to design a reliable system; in this work, a petrochemical supply chain. In particular, this supply chain is a generic three-echelon SC (production-storage-market) as the one depicted in *Figure 2*. This network includes: a set of plants with a set of available technologies, where products are manufactured; a set of warehouses where products are stored before being shipped to final markets; and a set of markets where products become available to customers. The case study is better detailed below.

.

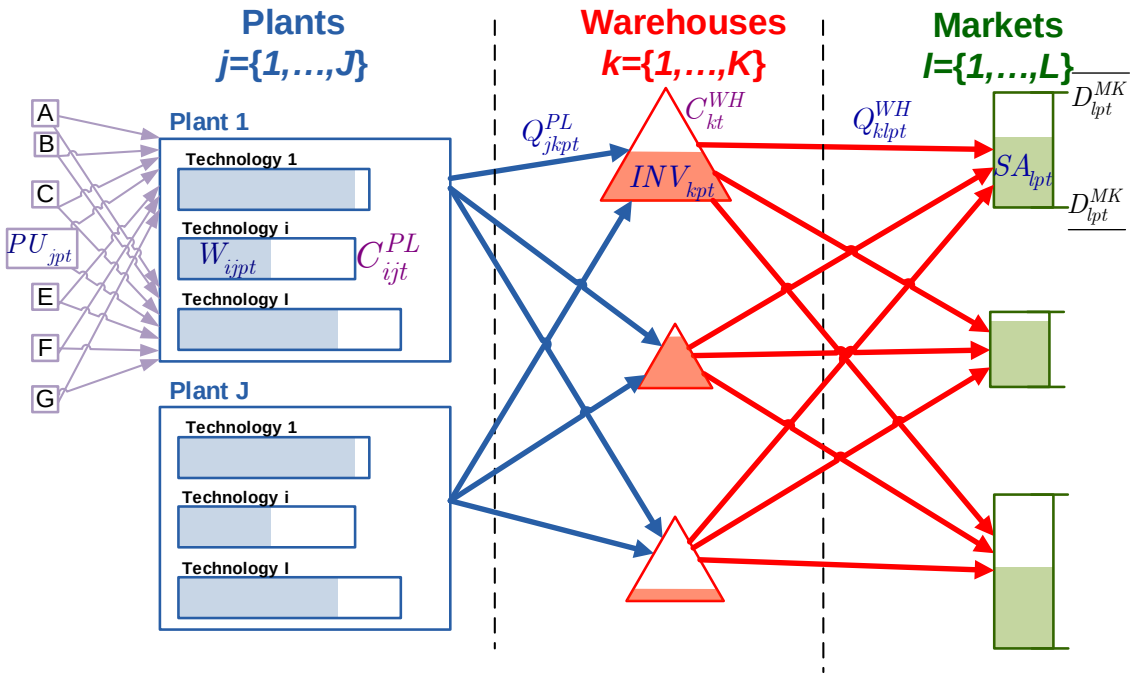


Figure 2. Superstructure of the three-echelon SC taken as reference.

3. THEORICAL FRAMEWORK

This section presents relevant theory regarding the objective of this work. It is divided into three parts: Economic theory, Time Series and Artificial Neural Networks.

3.1 Economic theory

To understand the domain of the forecast models, this section aims to introduce relevant economic theory that will aid the ARIMA model and the Artificial Neural Network models.

3.1.1 Efficient Market Hypothesis (EMH)

The EMH is a theory that stems from Fama (1970), which implies that prices on liquid capital markets fully reflect all available information, and that new information instantly will be incorporated in the price. An implication of EMH is that it is impossible to beat the market, since all available information already is incorporated in the price, and the results by any attempts to do so is subject to the domain of chance.

There exists three common forms of the EMH, namely the weak form, semi-strong form and strong form of EMH.

- **Weak form:** The weak form of EMH suggests that the prices on capital markets fully reflect all past prices and volumes, and as such implies that the market cannot be beaten using technical analysis.

- **Semi-strong form:** The semi-strong form of EMH suggests that the prices on capital markets fully reflect all public information. A consequence of this hypothesis is that neither technical nor fundamental analysis can be used to beat the market.
- **Strong form:** The strong form of EMH suggests that prices on capital markets fully reflect all public and private information. This means that consistent excess returns of the market is impossible to achieve consistently, regardless of whether the investor has insider information or not.

3.2 Time series

In order to approach time series analysis and forecasting, it must first be answered the question regarding what constitutes time series data. A time series is a sequence of data points, typically consisting of successive measurements made over a time interval. Examples of time series are solar activity, ocean tides, stock market behavior, and the spread of disease. Time series data are found in any domain of applied science and engineering which involves time-based measurements. In time series, data are a collection of ordered observations recorded at a specific time, for instance, hours, months, or years (like *Figure 1* shown previously).

Most often, the observations are made at regular time intervals. Time series analysis accounts for the fact that data points taken over time may have an internal structure, such as autocorrelation, trend or seasonal variation.

3.2.1 Time series analysis

Time series analysis comprises methods for analyzing time series data to extract meaningful statistics and other characteristics of time series data. It focuses on comparing values of a single time series or multiple dependent time series at different points in time.

3.2.2 Time series analysis forecasting

Time series forecasting is the use of a time series model to predict future values based on previously observed values in the series. It is used for many applications where pertinent time series data can be collected, such as: Budget Analysis, Financial Market Analysis, Economic Forecasting, Marketing and Sales Forecasting, Workload Projections.

There are two main goals of time series analysis. First, identifying the nature of the phenomenon represented by the sequence of observations in the data. Second, using the data to forecast or predict future values of the time series variable. Both of these goals require that we identify the pattern of observed time series data and more or less formally describe it. Once the pattern is established, it can be interpreted and integrated with other data (i.e., use it in our theory of the investigated phenomenon, seasonal commodity prices). Regardless of the depth of our understanding and the validity of our interpretation of the phenomenon, we can extrapolate the identified pattern to predict future events with this caveat: the further out in time we try to predict, the less accurate is the forecast.

3.2.3 Techniques used in time series analysis

The fitting of time series models can be an ambitious matter. It requires much more data preparation than the usual statistical models applied to other ordinary models (e.g., standard linear regression). Two important characteristics in time series data are that data are not necessarily independent and not necessarily identically distributed.

There are a number of different methods for modeling time series data including the following:

- Box-Jenkins ARIMA models
- Box-Jenkins Multivariate Models
- Holt-Winters Exponential Smoothing (single, double, triple)
- Unobserved Components Model

In our particular case, Box-Jenkins ARIMA is the used model due to the fact that we only used univariate data and it is a very robust model [6], which includes other models mentioned as it will be seen below.

3.2.4 Box-Jenkins ARIMA model

ARIMA models (Autoregressive Integrated Moving Average) are a kind of time series models that relate the present value of a series to past values and past prediction errors. It can be decomposed in three different components:

- **The AR part of ARIMA**. This first component AR indicates that ARIMA models are Autorregressive. That is to say, the evolving variable of interest is regressed on its own lagged (i.e., prior) values. For example:

$$\widehat{Y}_t = \alpha Y_{t-1} \tag{1}$$

where,

- $\widehat{Y}_t$ is the forecasted value of the model at the time t
- α is a parameter of the autorregressive part of the model
- Y_{t-1} is a past data of Y, at time $t-1$

- **The MA part of ARIMA**. It means moving average (MA). This second component indicates that the regression error is actually a linear combination of error terms whose values occurred contemporaneously and at various times in the past. For example:

$$\widehat{Y}_t = \beta e_{t-1} = \beta(Y_{t-1} - \hat{Y}_{t-1}) \tag{2}$$

where,

- $\widehat{Y}_t$ is the forecasted value of the model at the time t
- β is a parameter of the MA part of the model
- e_{t-1} error of the past forecasted values, at $t-1$
- Y_{t-1} is a past data of Y, at time $t-1$

- $\hat{Y}_{t-1}$ is the forecasted value of the model at the time $t-1$

- **The I part of ARIMA**. This component is called integrated (I). It indicates that the data values have been replaced with the difference between their values and the previous values (and this differencing process may have been performed more than once). For example:

$$Y_t - Y_{t-1} = \mu \rightarrow Y_t = \mu + Y_{t-1} \tag{3}$$

where,

- Y_t is the value of the model at the time t
- Y_{t-1} is a past data of Y, at time $t-1$
- μ is the average value of the data set

It is extremely important and worth emphasizing that the first step in applying ARIMA methodology is to check for stationarity. "Stationarity" implies that the series remains at a fairly constant level over time. If a trend exists, as in most economic or business applications, then data is not stationary. The data should also show a constant variance in its fluctuations over time. This is easily seen with a series that is heavily seasonal and growing at a faster rate. In such a case, the ups and downs in the seasonality will become more dramatic over time. Without these stationarity conditions being met, many of the calculations associated with the process cannot be computed. A data set can be converted into stationary taking into account this integrated part of ARIMA model, that it to say, differencing the series, subtracting the observation in the current period from the previous one.

Equations (1), (2) and (3) are just simple examples that illustrate the three components of ARIMA models. However, ARIMA models are linear combinations of these three parts, where there are different degrees. Non-seasonal ARIMA models are generally denoted ARIMA(p,d,q) where parameters p, d, and q are non-negative integers:

- p is the order (number of time lags) of the autoregressive model (AR).
- d is the degree of differencing, the number of times the data have had past values subtracted (I)
- q is the order of the moving-average model (MA).

A general equation of an ARIMA model is shown below:

$$\hat{y}_t = \mu + \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \cdots + \alpha_p y_{t-p} - \beta_1 e_{t-1} - \beta_2 e_{t-2} - \cdots - \beta_q e_{t-q} \tag{4}$$

where,

- The autoregressive term (AR):
  - $\alpha_1, \alpha_2, \dots, \alpha_p$ are the parameters of autorregresive terms (there are as many parameters as the order of p)
  - $y_{t-1}, y_{t-2}, \dots, y_{t-p}$ are past auxiliary variables of Y, at time $t-1, t-2, \dots, t-p$ (there are as many terms as the order of p). These auxiliary variables depend on the order of differencing (d) as shown in *Table 1* below.
- The moving average term (MA):
  - $\beta_1, \beta_2, \dots, \beta_q$ are the parameters of moving average terms (there are as many parameters as the order of q).
  - $e_{t-1}, e_{t-2}, \dots, e_{t-q}$ are the errors of the past forecasted values (there are as many terms as the order of q). These errors are equivalent to the terms in *Table 2.*

- The integrated term (I):
  - $\hat{y}_t$ is an auxiliary forecasted variable that can be changed depending on the degree of differencing as shown in *Table 3.*
  - μ is the average value of the data set

Table 1. Auxiliary variable y as a function of d^{th}difference (AR).

d^{th} order of Y	Error of the past forecasted
$d = 0$	$y_t = Y_t$
$d = 1$	$y_t = Y_t - Y_{t-1}$
$d = 2$	$y_t = (Y_t - Y_{t-1}) - (Y_{t-1} - Y_{t-2})$ $y_t = Y_t - 2Y_{t-1} + Y_{t-2}$

Table 2. Errors of the past forecasted values (MA).

q^{th} order of MA	Error of the past forecasted
$q = 0$	$e_t = 0$
$q = 1$	$e_{t-1} = Y_{t-1} - \hat{Y}_{t-1}$
$q = 2$	$e_{t-2} = Y_{t-2} - \hat{Y}_{t-2}$

Table 3. Auxiliary forecasted variable as function of d^{th}diference (I).

d^{th} difference of Y	Auxiliary forecasted value $\hat{y}_t$
$d = 0$	$\hat{y}_t = \hat{Y}_t$
$d = 1$	$\hat{y}_t = \hat{Y}_t - Y_{t-1}$
$d = 2$	$\hat{y}_t = (\hat{Y}_t - Y_{t-1}) - (Y_{t-1} - Y_{t-2})$ $\hat{y}_t = \hat{Y}_t - 2Y_{t-1} - Y_{t-2}$

3.2.5 Examples of ARIMA models

Some types of ARIMA models that are commonly encountered are shown below.

- **ARIMA(1,0,0)** = **First-order autoregressive model**

$$\hat{Y}_t = \mu + \alpha_1 Y_{t-1} \tag{5}$$

The series is predicted as a multiple of its own previous value, plus a constant.

- **ARIMA(0,1,0)** = **random walk**

$$\hat{Y}_t - Y_{t-1} = \mu \tag{6}$$

Or equivalently:

$$\hat{Y}_t = \mu + Y_{t-1} \tag{7}$$

If the series Y is not stationary, the simplest possible model for it is a random walk model.

- **ARIMA(1,1,0)** = **differenced first-order autoregressive model**

$$\hat{Y}_t - Y_{t-1} = \mu + \alpha_1 (Y_{t-1} - Y_{t-2}) \tag{8}$$

Equivalent to:

$$\hat{Y}_t = \mu + Y_{t-1} + \alpha_1 (Y_{t-1} - Y_{t-2}) \tag{9}$$

If the errors of a random walk model are autocorrelated, perhaps the problem can be fixed by adding one lag of the dependent variable to the prediction equation.

- **ARIMA(0,1,1) without constant** = **simple exponential smoothing**

$$\hat{Y}_t - Y_{t-1} = -\beta_1 e_{t-1}$$

(10)

That expanding and leaving $\hat{Y}_t$ alone:

$$\hat{Y}_t = Y_{t-1} - \beta_1(Y_{t-1} - \hat{Y}_{t-1})$$

(11)

It is a strategy for correcting autocorrelated errors in a random walk model. Ones that exhibit noisy fluctuations around a slowly-varying mean, the random walk model does not perform as well as a moving average of past values. It is better to use an average of the last few observations in order to filter out the noise and more accurately estimate the local mean.

- **ARIMA(0,1,1) with constant = simple exponential smoothing with growth**

$$\hat{Y}_t = \mu + Y_{t-1} - \beta_1(Y_{t-1} - \hat{Y}_{t-1})$$

(12)

There is the option of including a constant term in the ARIMA model if the trajectory of the long-term forecasts is typically a sloping line (whose slope is equal to μ) rather than a horizontal line.

- **ARIMA(0,2,2) without constant = linear exponential smoothing**

$$\hat{Y}_t - 2Y_{t-1} + Y_{t-2} = -\beta_1 e_{t-1} - \beta_2 e_{t-2}$$

(13)

It uses two nonseasonal differences in conjunction with MA terms. This is a general linear exponential smoothing model, essentially the same as Holt's model, and Brown's model mentioned above.

- **ARIMA(1,1,2) without constant = damped-trend linear exponential smoothing**

$$\hat{Y}_t - Y_{t-1} = \alpha_1(Y_{t-1} - Y_{t-2}) - \beta_1 e_{t-1} - \beta_2 e_{t-2}$$

(14)

It extrapolates the local trend at the end of the series but flattens it out at longer forecast horizons.

A few references to further information on such techniques can be found in [6] and [7].

What has been explained so far, there are three parts to the model, the AR, I and MA. In order to predict future values from the time series there are a number of requirements on the time series data. It must be stationary, that is to say, have the following properties:

1. Constant variance.
2. Void of seasonality.
3. No trend.

These properties can be achieved by differencing to remove the trend and seasonality and logs can be taken to flatten the variance.

Once that our data has been transform into a "white noise" signal (an stationarize the series) it can be performed a prediction. Before doing so, it must be determined the order of the AR (autoregressive) and MA (moving average) components in order to make an accurate prediction. These orders can be determined by taking the ACF (autocorrelation function) and PACF (partial autocorrelation function) of the "white noise" data, which are described in the next section.

3.2.6 Autocorrelation (ACF) and Partial Autocorrelation (PACF)

To determine the order of the AR (autoregressive) and MA (moving average) components, it is usual to build a time series basic diagnostics chart with the autocorrelation (ACF) and partial autocorrelation functions (PACF). Autocorrelations are numerical values that indicate how a data series is related to itself over time. More precisely, it measures how strongly data values at a specified number of periods apart are correlated to each other over time. The number of periods apart is usually called the lag**.** For example, an autocorrelation at lag 1 measures how values 1 period apart are correlated to one another throughout the series. An autocorrelation at lag 2 measures how the data two periods apart are correlated throughout the series.

The autocorrelation for the k^{th} lag, is calculated as follows:

$$r_k = \frac{c_k}{c_0} \tag{15}$$

with,

$$c_k = \frac{1}{N}\sum_{t=k+1}^{N}(Y_t - \bar{Y})(Y_{t-k} - \bar{Y}) \tag{16}$$

$$c_0 = \sum_{t=1}^{N}(Y_t - \bar{Y})^2 \tag{17}$$

where,

- k is the number of lags
- r_k is the autocorrelation value for the k^{th} lag
- $\bar{Y}$ is the mean of the N points in the time series
- Y_t is the value for each time t

On the other hand, the partial autocorrelation measures the linear dependence of one variable after removing the effect of other variable(s) that affect both variables. For example, the partial autocorrelation of order 2 measures the effect (linear dependence) of y_{t-2} on y_t after removing the effect of y_{t-1}on both y_t and y_{t-2} (the amount of correlation between a variable and a lag of itself that is not explained by correlations at all lower-order-lags). Another simpler example to clarify this concept would be regressing a variable Y on other variables X_1, X_2, and X_3. The partial correlation between Y and X_3 is the amount of correlation between Y and X_3 that is not explained by their common correlations with X_1 and X_2. The sample partial autocorrelations are computed by solving the following system for each order k:

$$\begin{pmatrix} r_0 & r_1 & \dots & r_{k-1} \\ r_1 & r_0 & \dots & r_{k-2} \\ \vdots & \vdots & \vdots & \vdots \\ r_{k-1} & r_{k-2} & \dots & r_0 \end{pmatrix} \cdot \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_k \end{pmatrix} = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_k \end{pmatrix} \tag{18}$$

where

- $p_1, p_2, \dots, p_k$ are the sample autocorrelations

These autocorrelations and partial autocorrelations values are normally graphically depicted by a bars chart as shown in *Figure 3*. The blue lines indicate when an autocorrelation or partial autocorrelation sample is significant, calculating the p-value (statistical parameter that takes into account the statistical significance of evidence).

Once the ACF and PACF have been built, it can be determined AR and MA terms needed. In Appendix (section 7.1.1), it is explained how to carry out this step. Moreover,

Figure 4 shows the steps to be followed in order to build an ARIMA model capable of modelling the time series data.

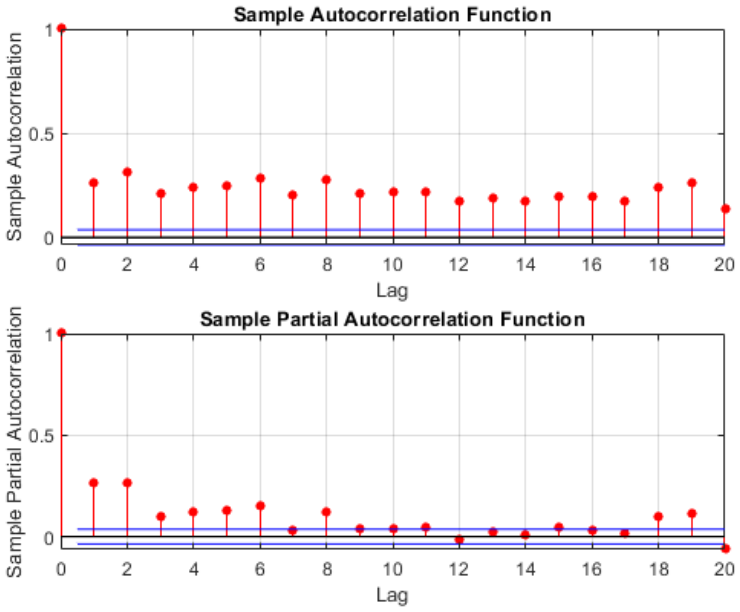


Figure 3. Example of Autocorrelation and Partial Autocorrelation Function.

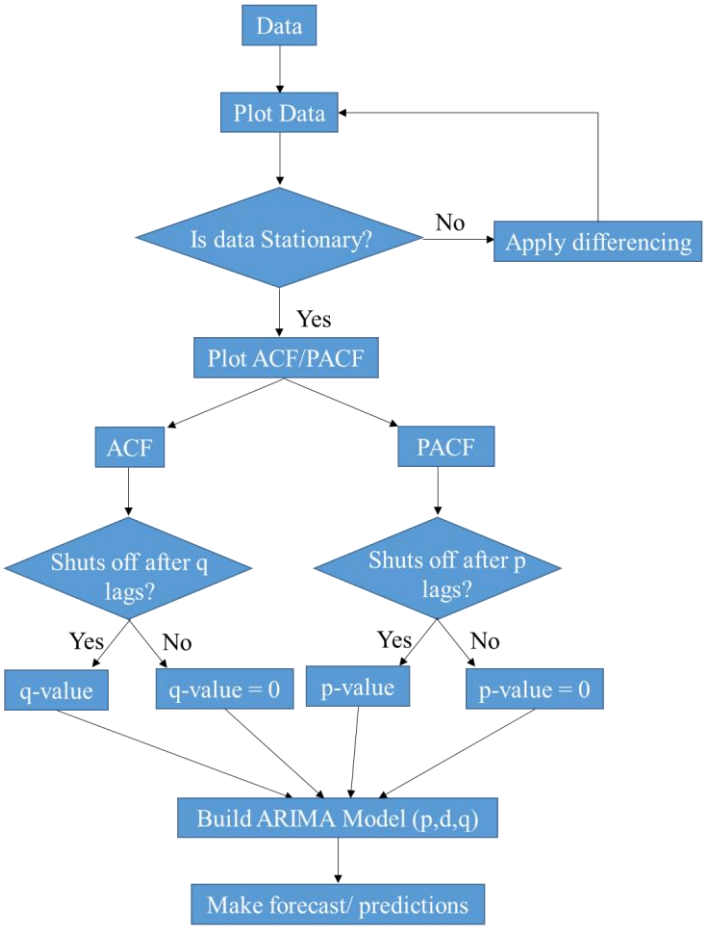


Figure 4. Steps to build an ARIMA model.

3.3 Artificial Neural Networks

Among others highlights, neural networks have the property of managing non stationarity between data. The neural networks have been shown to be universal approximators of mathematical functions. They are model-free, which means that the functional relationship will take the form that fits best to the data without explicitly defining the function beforehand.

The highest and simplest abstract representation of a neural network, can be presented as a black box with 2 methods, learn and predict as *Figure 5* shows.

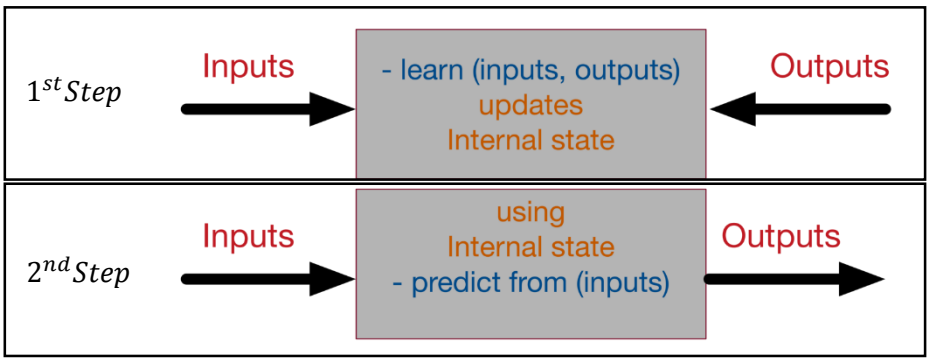


Figure 5. Basic concept of neural network.

The learning process takes the inputs and the desired outputs and updates its internal state accordingly, so the calculated output gets as close as possible from the desired output. The predict process takes input and generate, using the internal state, the most likely output according to its past training experience. It just like a kind of sophisticated fitting model.

3.3.1 Artificial Neural Networks components

Basically, any neural network is built on three different layers:

1. **Input Layer**. All the inputs are fed in the model through this layer.
2. **Hidden Layers**. There can be more than one hidden layer. They are used for processing the inputs received from the input layer.
3. **Output Layer**. The data after processing is made available at the output layer.

All these layers are subdivided in nodes which are interconnected with nodes of other layers. In *Figure* 6 an example of interconnection between nodes is shown.

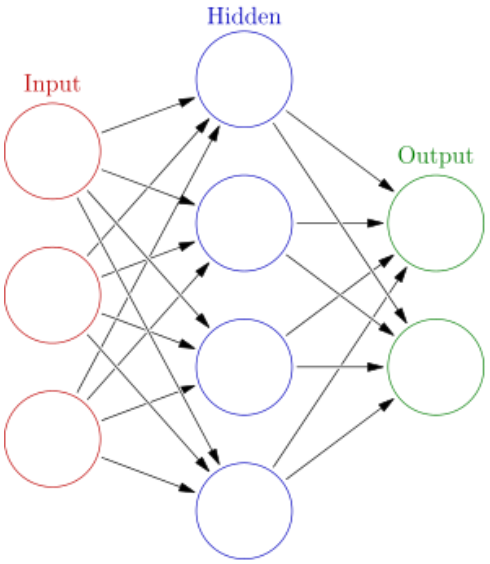


Figure 6. Interconnection between nodes of different layers in an ANN.

Once a general understanding of artificial neural networks is provided, it is appropriate to explore them in greater detail. Instead of describing the internal element of each node, which can be abstract, an illustrative example is shown to easily understand the concepts. *Figure 7 shows* an example of a NN made in an Excel file once the neural network has been trained (after learning, at second step).

As can be seen, in Figure 7 there are:

- 3 inputs(X_1, X_2, X_3).
- 4 nodes in the hidden layer (A_1, A_2, A_3, A_4).
- 1 output (Y).
- At each line that connects nodes, there are also parameters (a total of 16). 12 input parameters to the hidden layer (3 input parameters for each hidden node) and 4 parameters to the output. These parameters are called weights in terms of machine learning.
- 4 parameters for each node of the hidden layer and 1 parameter for the output. These parameters are called bias in terms of machine learning.

With this NN model, it is possible to predict the output value (Y) as a function of the input values (X_1, X_2, X_3), These inputs and output are related by very complex mathematical relationships that take into account the different connections between nodes. The final model implemented is shown below in its mathematical formulation.

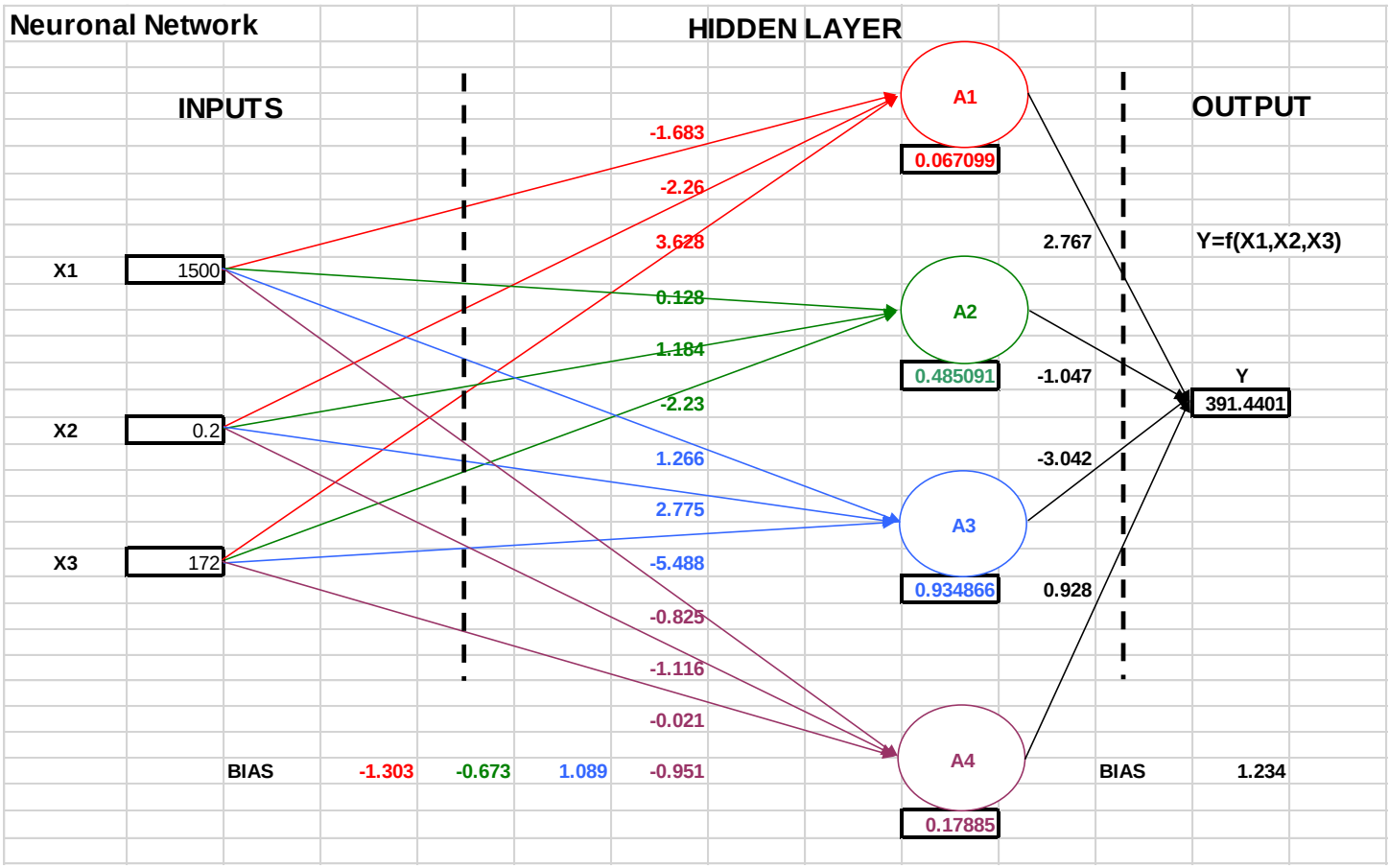


Figure 7. Example of a Neural Network after training.

The output value (Y) is computed as:

$$Y = \gamma_1 A_1 + \gamma_2 A_2 + \gamma_3 A_3 + \gamma_4 A_4 + \theta_y \tag{19}$$

where,

- $\gamma_1, \gamma_2, \gamma_3, \gamma_4$ are weights (the output parameters from hidden layer to the output).
- θ_y is the output bias

In particular,

$$\gamma_1 = 2.767,\ \gamma_2 = -1.047,\ \gamma_3 = -3.042,\ \gamma_4 = 0.928$$

$$\theta_y = 1.234$$

The nodes of the hidden layers are calculated as following:

$$A1 = \tanh(\delta_1^1\widehat{X_1} + \delta_2^1\widehat{X_2} + \delta_3^1\widehat{X_3} + \theta_1)$$

$$A2 = \tanh(\delta_1^2\widehat{X_1} + \delta_2^2\widehat{X_2} + \delta_3^2\widehat{X_3} + \theta_2)$$

$$A3 = \tanh(\delta_1^3\widehat{X_1} + \delta_2^3\widehat{X_2} + \delta_3^3\widehat{X_3} + \theta_3)$$

$$A4 = \tanh(\delta_1^4\widehat{X_1} + \delta_2^4\widehat{X_2} + \delta_3^4\widehat{X_3} + \theta_4)$$

(20)

where,

- δ_i^j are the input parameters from each input variable j $(\mathrm{X}_1, \mathrm{X}_2, \mathrm{X}_3)$ to each node i in the hidden layer.

In particular,

$$\delta_1^1 = -1.683,\ \delta_2^1 = -2.26, \delta_3^1 = 3.626$$

$$\delta_1^2 = 0.128,\ \delta_2^2 = 1.184, \delta_3^2 = -2.23$$

$$\delta_1^3 = 1.266,\ \delta_2^3 = 2.775, \delta_3^3 = -5.488$$

$$\delta_1^4 = -0.825,\ \delta_2^4 = -1.116, \delta_3^4 = -0.021$$

- $\widehat{X_1}, \widehat{X_2}, \widehat{X_3}$ are the normalized input variables

This $tanh$ is what is called in machine learning as activation functions. It is equivalent to calculate:

$$\tanh(x) = \frac{1}{1 + e^{-x}}$$

(21)

It is the most typical function used due to the fact that it has a behavior similar to a computer chip circuit, which can be "ON" (1) or "OFF" (0), depending on the input. This is not the only function used, hyperbolic tangent and rectified linear functions are used as well. See *Figure 8.*

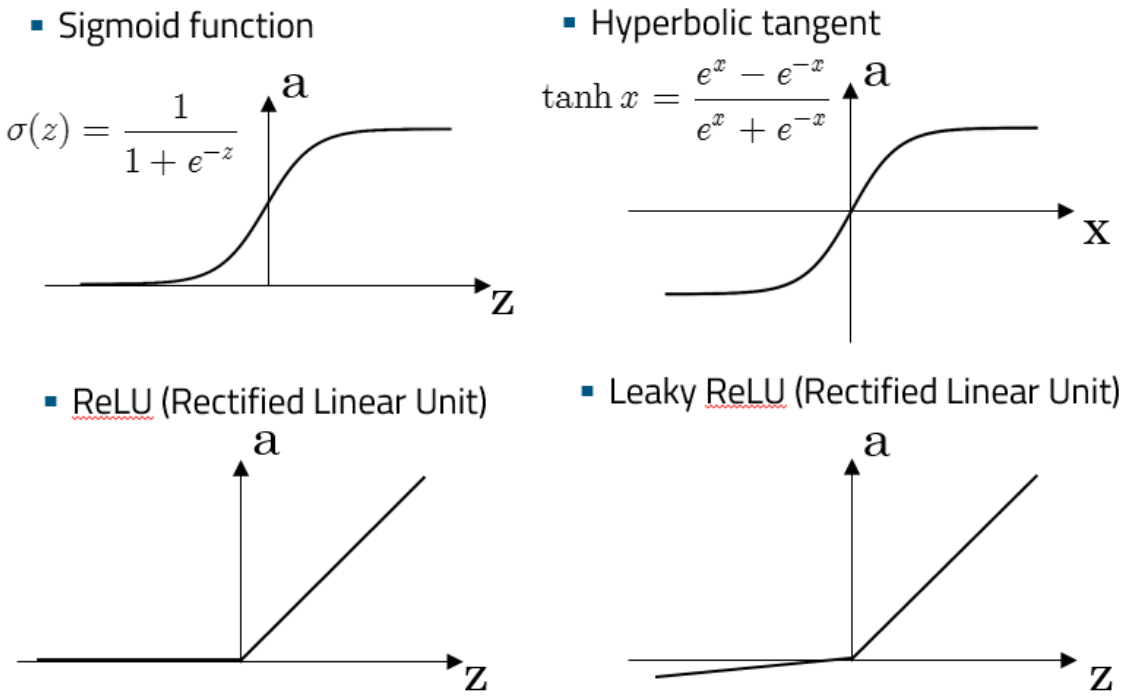


Figure 8. Examples of activation functions.

Up to now, it is known how a neural network works. It can predict a final output value (Y) using a very complex NN model, whose structure has been seen in the example above (equations (19) and (20)). However, the model presented uses parameters (weights and bias) that have been calculated as first step, the training step.

The process of training a neural network (the first step according to *Figure 5*) involves tuning the values of the weights and biases of the network to optimize network performance. The default performance function for feedforward networks is mean square error (MSE), the average squared error between the network outputs and the target outputs.

$$Min\ MSE = \frac{1}{N}\sum_{i=1}^{m}\left(Y_i - Y_i^{target}\right)^2 \tag{22}$$

Subject to equations (19) and (20) presented above, with bias and weights as free variables. The gradient descent is usually used as learning algorithm.

3.3.2 Artificial Neural Networks used in Time Series.

Dynamic recurrent neural network like the nonlinear autoregressive (NAR) are neural network structures that can be useful in Time Series forecasting. It is a non-linear autoregressive neural networks (NAR), which is able to forecast samples framed in a one-dimensional time series. Due to the data set available (1868 samples of CO_2 closing prices), NAR is the indicated neural network to use. The main characteristic of these Neural Networks is that they can accept dynamic inputs represented by time series sets. They used as time series forecasters feed-forward networks which employ a sliding window over the input sequence. In this technique, the N-tuple input slides over the full training set. *Figure 9* gives the basic idea of NAR.

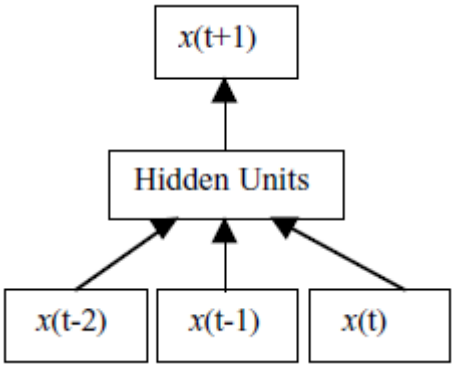


Figure 9. The standard method of performing time series prediction using a sliding window of, in this case, three time steps.

This technique can be seen as an extension of auto-regressive time series modelling, in which the function is assumed to be a linear combination of a fixed number of previous series values. Such a restriction does not apply with the non-linear neural network approach as such networks are general function approximators.

3.3.3 Nonlinear autoregressive neural network (NAR)

In the majority of cases, time series applications are characterized by high variations and fleeting transient periods. This fact makes it difficult to model time series using a liner model, therefore a nonlinear approach should be suggested. NAR models uses the past values of the time series to predict future values. Equation (23) describes the mathematic relations that occur between the data of subsequent levels.

$$y(t) = f\big(y(t-1), y(t-2), y(t-3), \dots, y(t-p)\big) + \in (t) \tag{23}$$

Where $y(t)$ is the model output that depends on the p temporally previous values of the output signal. Besides, the function f is unknown in advance, and the training of the neural network aims to approximate the function by means of the optimization of the network weights and neuron bias,

as it was explained above. Finally, the term $\in (t)$ stands for the error of the approximation of the series y at time t.

The topology of a NAR network is shown in *Figure 10*. The p features

$$y(t-1), y(t-2), \ldots, y(t-p)$$

are called feedback delays. The number of hidden layers and neurons per layer are completely flexible, and can be optimized through a trial-and-error procedure to obtain the network topology that can provide the best performance. Nevertheless, it is important to bear in mind that increasing the number of neurons makes the system more complex, while a low number of neurons may restrict the generalization capabilities and computing power of the network.

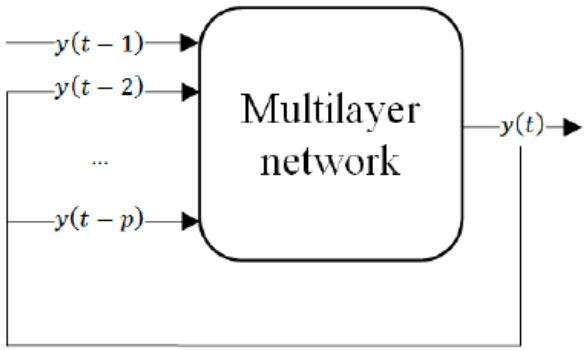


Figure 10. Basic non-linear autoregressive (NAR) network.

3.3.4 Mathematical NAR formulation

In the world of neural network, the function f presented in equation (23) can be achieved by means of the weights and bias optimizations. Given the importance and the widespread use of the backpropagation algorithm, let see what it is based upon and its development in order to better understand how the network is learning in each iteration (see section 7.1.2.1 of Appendix).

The internal structure of the NAR network is divided into layers as showing in
Figure 11. The first layer defines the delay line that is, the number of previous samples used in the model, which are linked to the input layer. Each neuron has p inputs associated with a weight parameter, w, plus the bias, b. Also, each neuron has an activation function, $g(u_k)$, which acts on the sums of the inputs and weight parameters,

$$u_k = b_k \sum_{i}^{p} w_{i,k} \cdot v_i \tag{24}$$

where, k is the number of the neuron in the hidden layer; g is a non-linear function in the case of neurons in the hidden layer, $g(u_k) = tanh\,(u_k)$, and it is a linear function in the case of the output neuron, $g\,(u_k) = u_k$.

Training is a cyclic process, where internal weights w and b are modified following a rule that makes the network behave in a specific manner. The rule considers an error signal that is generated by the difference between the actual output of the network and the expected output (i.e., the desired signal). In the case of the NAR network, the set-up consists in training with the feedback loop open (no feedback) and comparing the output $y(t + 1)$ with the same training signal with a shift (see section 7.1.2.1 of Appendix as was said before). Once the error reaches a certain minimum value the training process ends. Once the error reaches a certain minimum value the training process ends. With the trained network, it is possible to forecast multiple future values by closing the loop, allowing the network to feed back its own predictions [10].

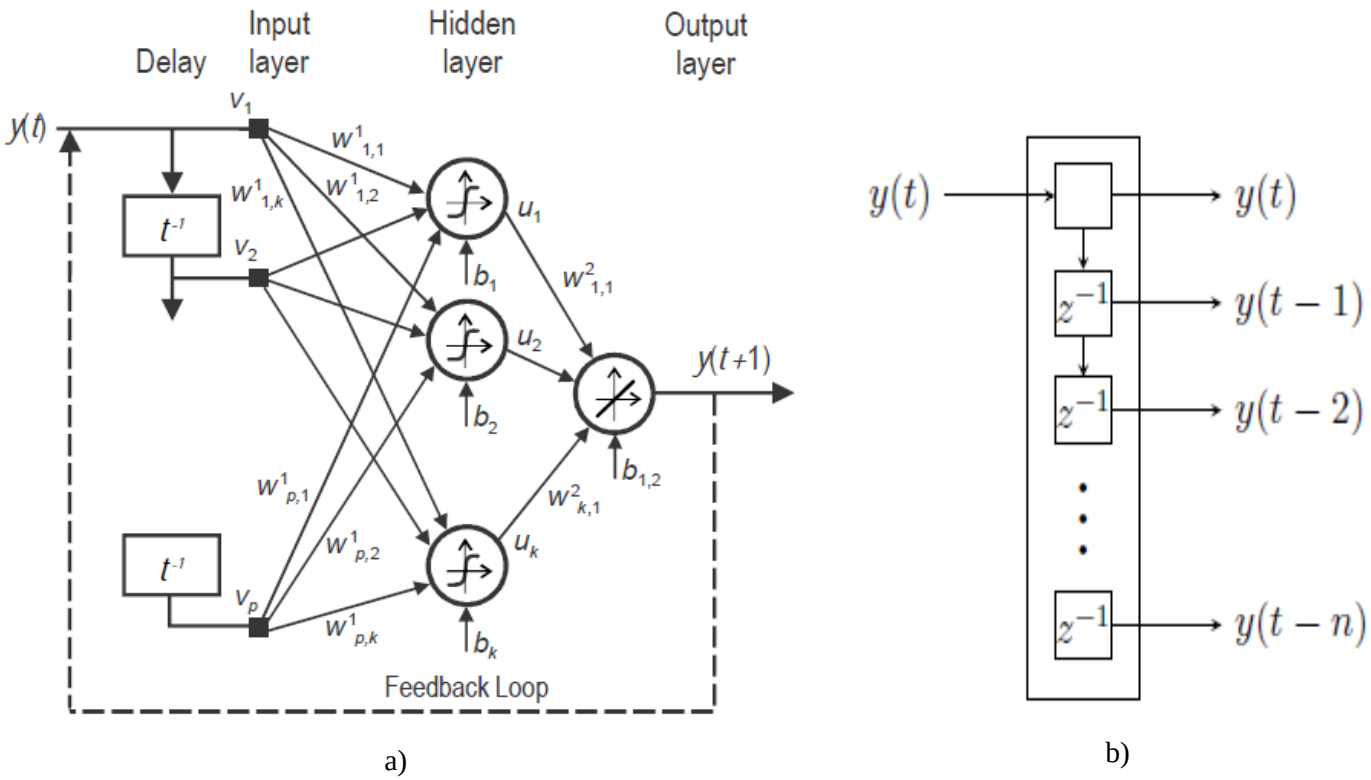


Figure 11. a) Nonlinear autoregressive network architecture. Feedback loop is open for training and closed for predictions; b) *Tapped delay line.*

4. OPTIMIZATION MODEL REFORMULATION

Given a set of technologies to manufacture different products, their demand and prices, the investment and operating costs of the network and the emissions, wastes and feedstock requirements, the objective of the model is obtain an optimal supply chain taking into account the economic and environmental impact (which is considered as a monetized constraint) under uncertainty . Therefore, the optimization problem is formulated as:

$$Max\,\{E[NPV]\}$$
$$s.t\ constraints$$

That is to say, maximize the expected value (E[NPV]) of the net present value (NPV) for a petrochemical supply chain subject to capacity, mass balance, raw material availability and product demand constraints, taking into account the environmental impact assessed by the Global Warming Potential (GWP) through is monetization by a carbon trading system. Here is where uncertainty comes into play, due to the need to forecast the carbon emissions rights price for the network horizon time.

Taking into account all the characteristics of the supply chain presented in section 2.1, the most important equations of the model are presented below. The complete model can be found in section 7.2 of Appendixes. The sets make reference to: the process technologies (i), the production plants (j), the warehouses (k), the markets (l), the chemicals products (p), the scenarios (s) and the time periods (t).

s.t

$$GWP_{total} \leq \varepsilon_p \quad \text{(M2)}$$

$$\underline{\varepsilon} \leq \varepsilon_p \leq \overline{\varepsilon} \quad \text{(M3)}$$

$$PU_{j,p,t,s} + \sum_{i\in OUT(p)} W_{i,j,p,t,s} = \sum_k Q^{PL}_{j,k,p,t,s} + \sum_{i\in OIN(p)} W_{i,j,p,t,s}\ \forall j,p,t,s \quad \text{(M4)}$$

$$\underline{CE^{PL}_{i,j,t}} X^{PL}_{i,j,t} \geq CE^{PL}_{i,j,t} \leq \overline{CE^{PL}_{i,j,t}} X^{PL}_{i,j,t}\ \forall \mathrm{i,j,t} \quad \text{(M5)}$$

$$\underline{CE^{WH}_{k,t}} X^{WH}_{k,t} \leq CE^{WH}_{k,t} \leq \overline{CE^{WH}_{k,t}} X^{WH}_{k,t} \quad \forall k,t \quad \text{(M6)}$$

$$\underline{Q^{PL}_{j,k,t,s}} Y^{PL}_{j,k,t} \leq \sum_p Q^{PL}_{j,k,p,t,s} \leq \overline{Q^{PL}_{j,k,p,t,s}} Y^{PL}_{j,k,t} \quad \forall j,k,t,s \quad \text{(M7)}$$

$$GWP_s^{total} = GWP_s^{RM} + GWP_s^{EN} + GWP_s^{TR}. \quad \text{(M8)}$$

$$GWP_s^{total} = \sum_t GWP_t^s$$

$$Sales^{CO2}_{t,s} + GWP_{t,s} = Max^{CO2}_t + Buy^{CO2}_{t,s} \quad \forall t,s \quad \text{(M9)}$$

$$Net^{CO2}_{t,s} = Price^{CO2}_{t,s}\, Sales^{CO2}_{t,s} - Cost^{CO2}_{t,s} Buy^{CO2}_{t,s} \quad \forall t,s \quad \text{(M10)}$$

At the end of the time horizon, different NPV values are obtained for scenario considered and the model maximizes the expected value ($E\ [NPV]$) of the NPV_s distribution. The probability of occurrence of scenario s (equation (M1)) is denoted by $prob_s$. On the other hand, to measure the environmental performance of the supply chain, the GWP 100a indicator is used (equation (M2)). It is a parameter based on the principles of Life Cycle Analysis. The Life Cycle Analysis is a tool used to quantify the potential environmental impact derived from the use of a certain industrial technology throughout the study system.

The mass balance (equation (M4)) must be satisfied for each node embedded in the network. Thus, the purchases ($PU_{j,p,t,s}$) made and the amount produced ($\sum_{i\in OUT(p)} W_{i,j,p,t,s}$) must equal the amount transported from the plant to the warehouses ($QPL^{PL}_{j,k,p,t,s}$) plus the amount consumed $\sum_{i\in OIN(p)} W_{i,j,p,t,s}$). The model also includes capability constraints of the different plants and warehouses, where there is possibility of increasing the actual capacity (equations (M5) and (M6)). In equation (M5), $\overline{CE^{PL}_{i,j,t}}$ and $\underline{CE^{PL}_{i,j,t}}$ are the upper and lower bounds of the plant capacity expansions. The binary variable $X^{PL}_{i,j,t}$ indicates the occurrence of the capacity expansion. In equation (M6), $\overline{CE^{WH}_{k,t}}$ and $\underline{CE^{WH}_{k,t}}$ are the upper and lower bounds of warehouse capacity expansion. This constraint makes use of the binary variable $X^{WH}_{k,t}$. Equation (M7) defines a transportation link between plant j and warehouse k in period t and scenario s, which existence is represented by the binary variable $Y^{PL}_{j,k,t}$ (). $\overline{Q^{PL}_{j,k,p,t,s}}$ and $\underline{Q^{PL}_{j,k,t,s}}$ are the upper and lower bounds for the amount of product p transported from warehouse k to market l in period t and scenario s.

Besides, the GWP 100a indicator (equation (M8)) takes into account the main emission sources, which are the consumption of raw materials, GWP^{RM}_s , the energy consumed in the auxiliary facilities GWP^{EN}, and the transport of products between the nodes of the supply chain GWP^{TR}. Finally, in order not to exceed the imposed emission limit, the $GWP_{t,s}$ index is included in the balance of CO_2 emissions (equation (M9)), which states that, the emission rights that are sold plus the CO_2 that is emitted, equals the emission limit ($Max^{CO_2}_t$) plus the rights that are bought.

To optimize the objective function ($E\ [NPV]$) taking into account GWP_{total}, the CO_2 emissions must be monetized, which is done through the emissions trading equation (equation (M10)). Thus, the income related to trade with CO_2 emissions ($Net^{CO2}_{t,s}$) must be equal to the sale of emissions minus the purchase of emissions. It is important to mention that the uncertainty in the model comes from the price of CO_2 allowances ($Price^{CO2}_{t,s}$ and $Cost^{CO2}_{t,s}$), which is going to be forecasted using the ARIMA and ANN explained above.

5. RESULTS

5.1 CO_2 allowances price data statistical analysis

First of all, it is presented the CO_2 allowance price in *Figure 12* [5]. With the data in the graph, the initial statistical analysis presented below is carried out.

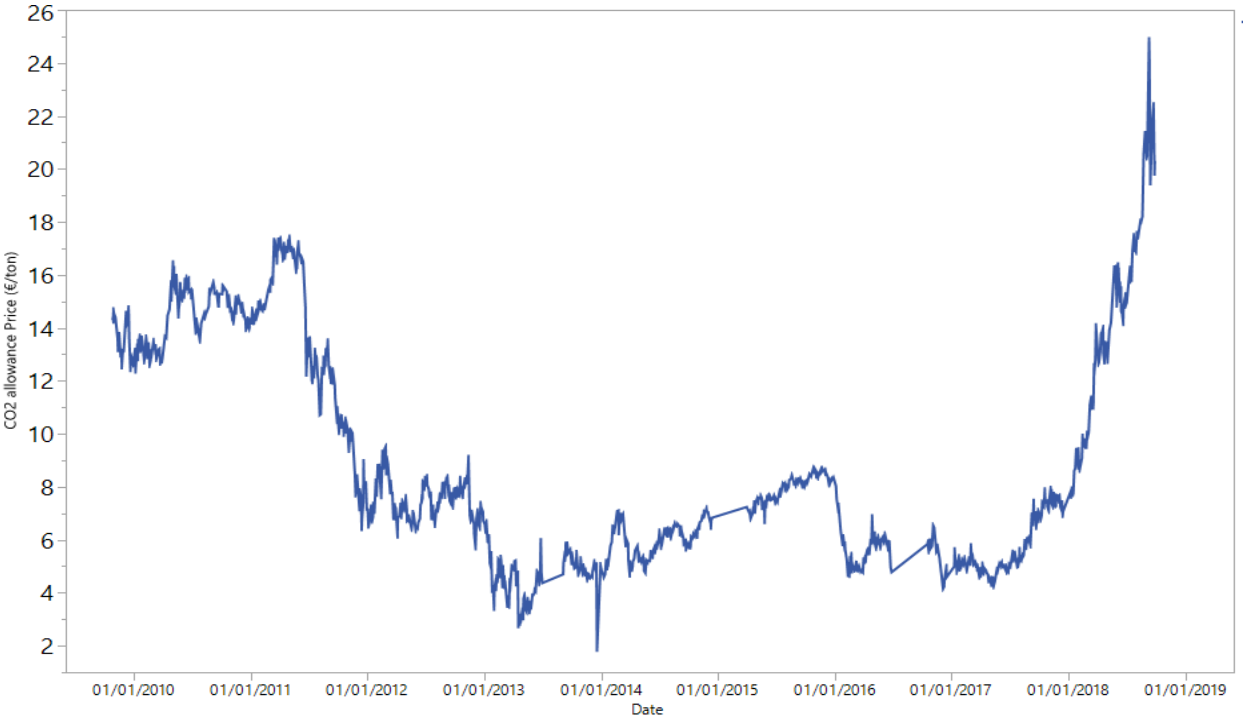
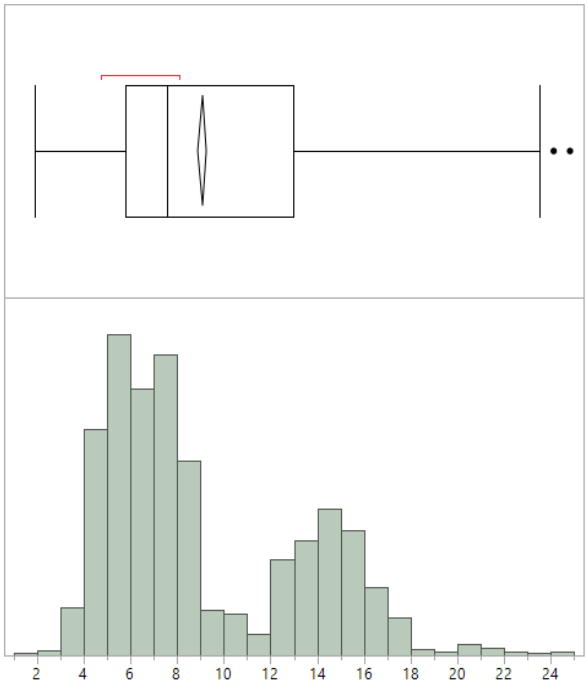


Figure 12. Initial data of CO_2 allowance Price (€/ton).

As can be seen in *Figure 12*, CO_2 allowance price has changed along years. At the beginning of 2009, prices were around 14 €/ton. In 2010, prices rose to a maximum of 16.42 €/ton and at the end of 2011 they decreased until a minimum amount of 6.46 €/ton. The price along 2012, 2013, 2014, 2015, 2016 and 2017 is ranging between 9.52 €/ton to 1.9 €/ton. Finally, on 2018 CO_2 allowances prices rose to a record levels with a maximum of 24.85 €/ton. *Table 4* shows some statistical data about the total data set, and in *Figure 13* the distribution of prices is shown.

Table 4. Statistical data of CO_2 allowance Price data set.

Average	9,065
Standard desviation	4,189
Standard error	0,098
Upper end of the 95% CI for the mean	9,257
Lower end of the 95% CI for the mean	8,872
Number of data points	1822



100.0%	maximum	24,850
99.5%		21,278
97.5%		17,204
90.0%		15,344
75.0%	quartile	12,993
50.0%	median	7,555
25.0%	quartile	5,760
10.0%		4,860
2.5%		4,000
0.5%		3,100
0.0%	minimum	1,900

Figure 13. Distribution of CO_2 allowance Price (€/ton).

As can be seen, most of the prices are between 5 €/ton to 9 €/ton. These prices correspond to years 2012-2017, where the prices were relatively cheap. the price distribution of each year can be seen in *Table 5.* in year 2013, the prices were the cheapest, while in 2018 prices are been the highest ever.

Table 5. Distribution of CO_2 allowances prices by years.

	maximum				**quartile**	**median**	**quartile**				**minimum**
Year	**100.00%**	**99.50%**	**97.50%**	**90.00%**	**75.00%**	**50.00%**	**25.00%**	**10.00%**	**2.50%**	**0.50%**	**0.00%**
2009	14.720	14.720	14.708	14.506	14.340	13.685	13.088	12.671	12.478	12.460	12.460
2010	16.420	16.406	15.867	15.594	15.198	14.515	13.333	12.966	12.716	12.414	12.400
2011	17.400	17.387	17.252	16.970	16.260	13.180	10.700	8.376	7.228	6.546	6.460
2012	9.520	9.499	9.074	8.360	8.010	7.460	6.910	6.608	6.349	5.787	5.730
2013	6.650	6.650	6.371	5.614	5.105	4.650	4.010	3.476	3.006	1.900	1.900
2014	7.190	7.190	7.090	6.862	6.500	6.070	5.505	5.100	4.729	4.590	4.590
2015	8.720	8.720	8.660	8.574	8.360	8.070	7.530	7.216	6.844	6.720	6.720
2016	8.070	8.070	7.588	6.340	5.983	5.650	4.998	4.800	4.309	4.150	4.150
2017	7.910	7.910	7.700	7.430	6.920	5.260	4.960	4.702	4.412	4.261	4.260
2018	24.850	24.850	22.536	20.505	16.903	14.570	10.913	8.925	7.790	7.680	7.680

5.2 Forecasting CO_2 allowance prices

A forecasting of CO_2 allowance prices is carried out with MATLAB Econometric Toolbox used to build the ARIMA model and Neuronal Toolboox used to build the neural network as follows. These models will be used to generate the CO_2 parameters in the optimization model.

5.2.1 ARIMA model for forecasting CO_2 allowance prices

Following the structure given by *Figure 14* the first step is differencing in order to get a stationary data set. As can be seen in *Figure 12*, the time series is clearly non-stationary. Applying a difference of order d equal to 1, the data set becomes stationary as shown in *Figure 14* .

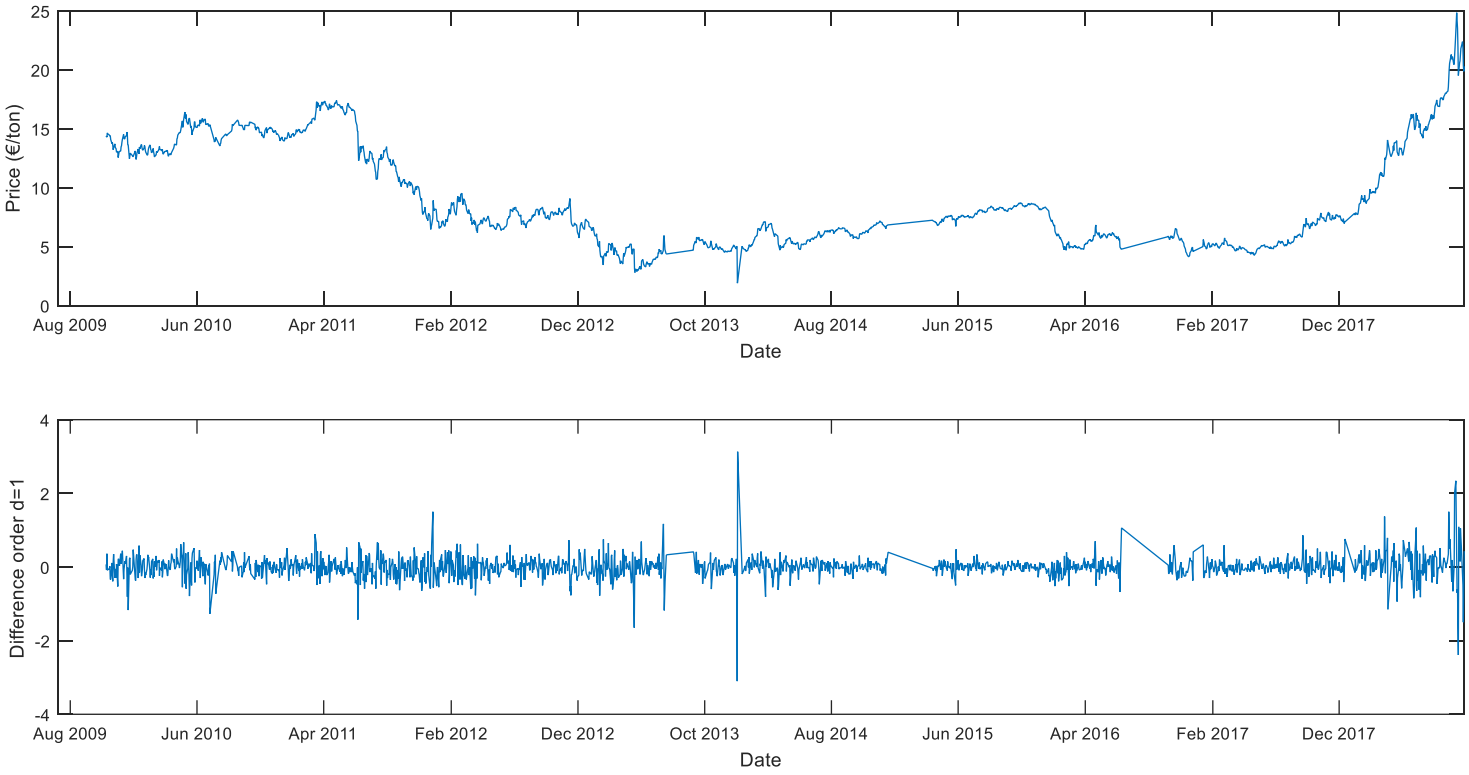


Figure 14. Result of differencing CO_2 allowance price data set.

In *Figure 15*, the ACF and PACF of the difference function are represented. This graph gives some idea about the p and q coefficients of the ARIMA model. According to the ACF and PACF theory (section 7.1.1 of Appendix), an AR(4) should be applied. Significance (blue line) is 0.0469 and -0.0469 for ACF and PACF. The correlation at lag 1 is not significant, but it is positive, so AR terms should be added. From lag 2, the ACF and PACF are negative (see *Table 6*) and significant until lag 4. Therefore, an ARIMA(4,1,0) would be a good combination of parameters as first approach.

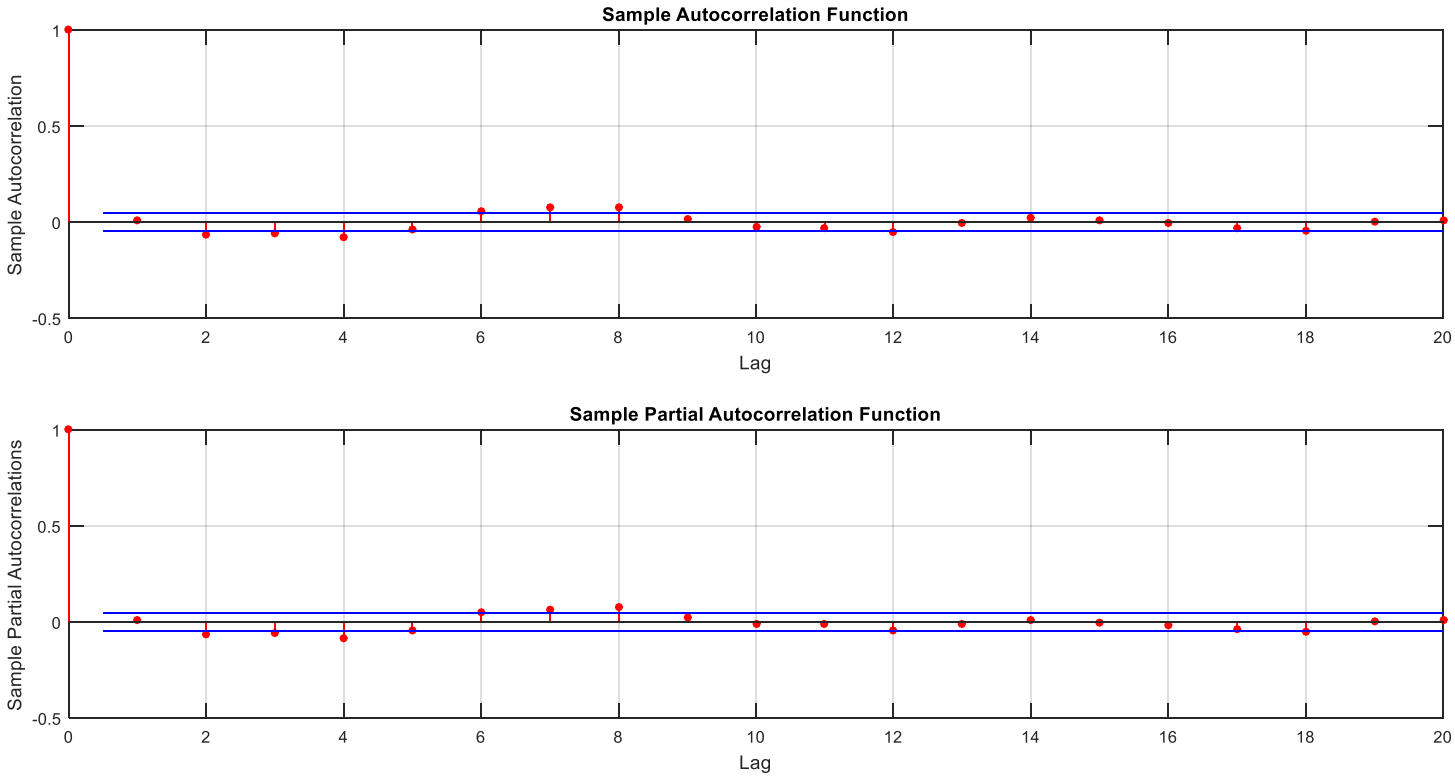


Figure 15. ACF and PACF for ARIMA(0,1,0).

Table 6. ACF, PACF values for ARIMA(0,1,0).

Lag	ACF	Parcial	Significance
0	1.0000	1.0000	Significance
1	0.0078	0.0078	
2	-0.0666	-0.0667	Significance
3	-0.0598	-0.0589	Significance
4	-0.0751	-0.0795	Significance
5	-0.0346	-0.0427	
6	0.0599	0.0463	Significance
7	0.0769	0.0636	Significance
8	0.0751	0.0742	Significance
9	0.0145	0.0263	
10	-0.0270	-0.0019	
11	-0.0286	-0.0040	
12	-0.0481	-0.0371	
13	-0.0051	-0.0102	
14	0.0235	0.0015	
15	0.0068	-0.0147	
16	-0.0046	-0.0173	
17	-0.0275	-0.0305	
18	-0.0447	-0.0384	
19	0.0022	0.0065	
20	0.0093	0.0059	

Then, an ARIMA(4,1,0) is built, as well as different variations of ARIMA(p,d,q) models which ensure that the best ARIMA model is used. To explore this issue, this work evaluated the performance of the two commonly used model selection criteria, Akaike information criteria (AIC) and Bayesian information criteria (BIC). In *Table 7* the AlC, SBC results, as well as, Rsqr, the log-2likelihood ($-2log$LH), weigh comparison between models, the mean absolute percentage error (MAPE) and the mean absolute error (MAE) are collected. It can be found further information about all the parameters in Appendix (section 7.1.3) and [17].

Table 7. ARIMA models comparison.

Model	Variance	AIC	SBC	Rsquare	-2LogLH	Weigh	MAPE	MAE
ARIMA(4, 1, 2)	0,087	721,100	759,650	0,995	707,100	0,341	2,457	0,191
ARIMA(2, 1, 4)	0,087	721,441	759,991	0,995	707,441	0,288	2,456	0,191
ARIMA(4, 1, 3)	0,087	723,073	767,130	0,995	707,073	0,127	2,457	0,191
ARIMA(3, 1, 4)	0,087	723,235	767,292	0,995	707,235	0,117	2,456	0,191
ARIMA(4, 1, 4)	0,087	725,052	774,616	0,995	707,052	0,047	2,457	0,191
ARIMA(3, 1, 2)	0,087	725,266	758,309	0,995	713,266	0,042	2,459	0,191
ARIMA(8, 1, 4)	0,087	725,561	797,154	0,995	699,561	0,037	2,459	0,191
ARI(4, 1)	0,088	747,825	775,361	0,995	737,825	0,000	2,443	0,190
ARIMA(4, 1, 1)	0,088	748,524	781,567	0,995	736,524	0,000	2,443	0,190
IMA(1, 4)	0,088	752,681	780,217	0,995	742,681	0,000	2,438	0,190
ARIMA(1, 1, 5)	0,088	754,309	792,859	0,995	740,309	0,000	2,438	0,190
ARIMA(1, 1, 4)	0,088	754,533	787,575	0,995	742,533	0,000	2,438	0,190
ARIMA(1, 1, 2)	0,088	755,034	777,063	0,995	747,034	0,000	2,432	0,190
IMA(1, 2)	0,089	760,893	777,414	0,995	754,893	0,000	2,429	0,190
I(1)	0,089	766,394	771,901	0,995	764,394	0,000	2,440	0,190

The models are sorted by the AIC and SBC statistic, in decreasing order. As mentioned in Appendix, a lower number of AIC and SBC indicates a better model. Therefore, an ARIMA model including moving average terms improve the final result. That is to say, an ARIMA(4,1,2) is a better model than an ARIMA(4,1,0). Once, ARIMA(4,1,2) model has been decided as appropriate model to forecast future CO_2 allowance prices, it is developed a simulation taking into account this model. Besides, it is also developed a simulation of the ARIMA(4,1,2) as a MMSE (minimum mean square error) forecast. A Monte Carlo simulation (that has into account this ARIMA(4,1,2)) is also developed in order to compare results (see sections 7.1.4 of Appendix or [18] and [19] for further information). Monte Carlo simulation generates different scenarios in which each one has the same probability of occurrence. On the other hand, MMSEE generates a mean scenario with de confident intervals. Monte Carlo simulation would be used to generate the scenarios, but first it is convenient compared both predictions in order to ensure the result is the same.

Figure 16 shows the forecast prediction of CO_2 allowance price with MMSE method. The mean price tends to increase slightly. The probability interval of future values of the process is also given. It has been forecasted 500 path forward, from September 30th 2018 to March 9th 2021. The first forecast value predicts 20.727 €/ton on 30th September 2018. This quantity increases until 22.766 €/ton on March 9th 2021. That is to say, ARIMA(4,1,2) predicts that in 2,34 years the price will increase a mean of 2.039 €/ton.

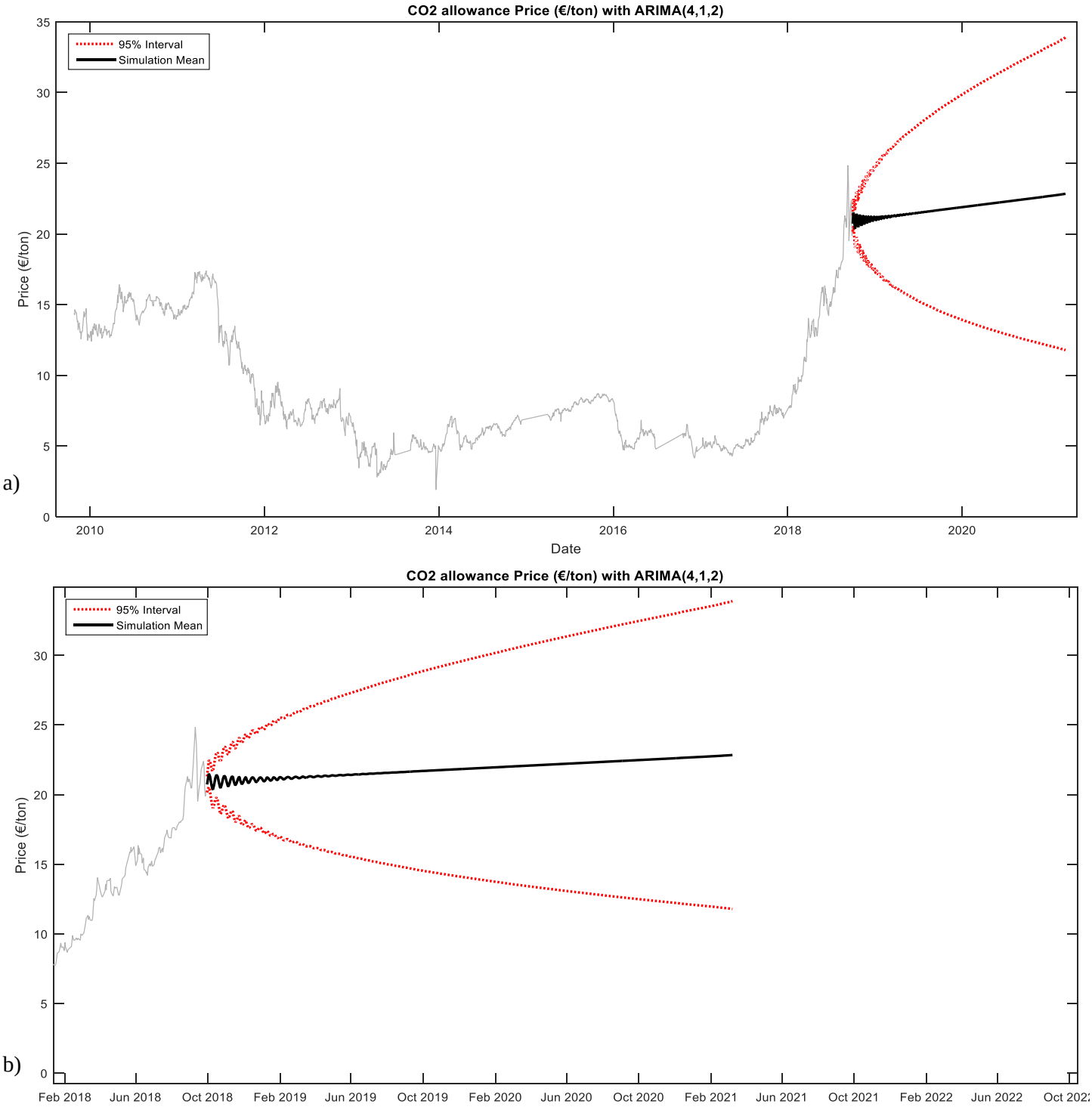
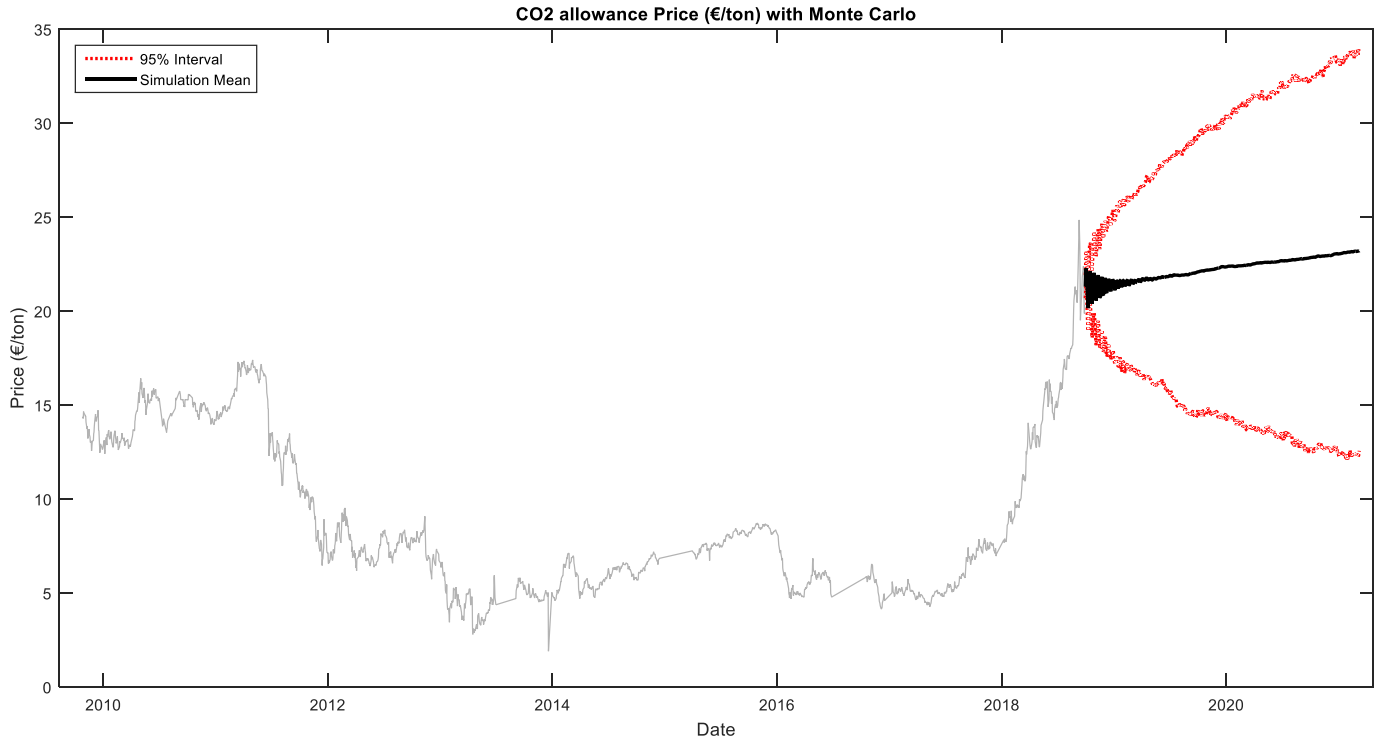


Figure 16. CO_2 allowance Price (€/ton) obtained with MMSE : a) Without zoom ; b) With zoom.

a)



b)

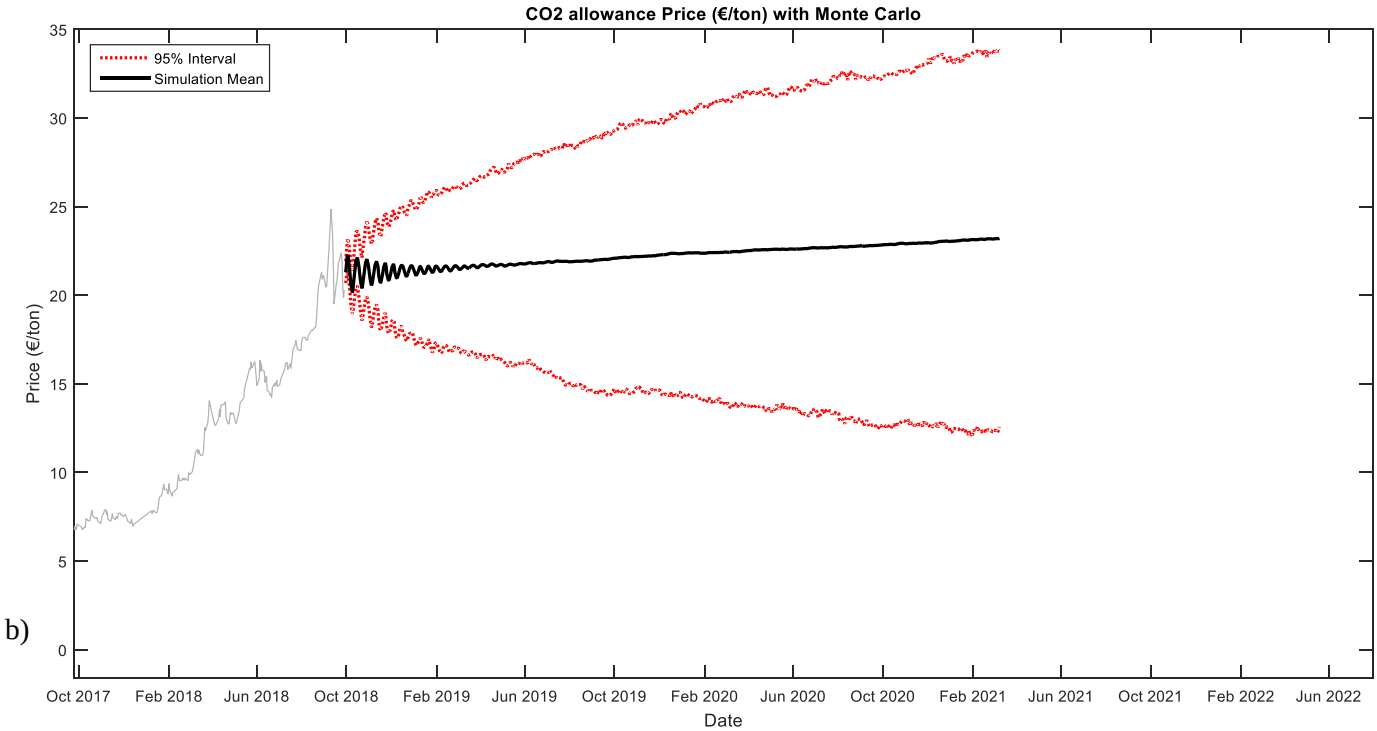


Figure 17. CO_2 allowance Price (€/ton) obtained with Monte Carlo simulation. a) Without zoom ; b) With zoom.

Figure 17 shows the prediction given by a Monte Carlo simulation. Repeating the forecast with another method ensures a safer prediction. As can be seen, the trend followed by the CO_2

allowance price seems to be very similar with respect to MMSE. In fact, if it is represented both predictions, they are practically equal. In *Figure 18* can be seen a comparison between MMSE and Monte Carlo simulation.

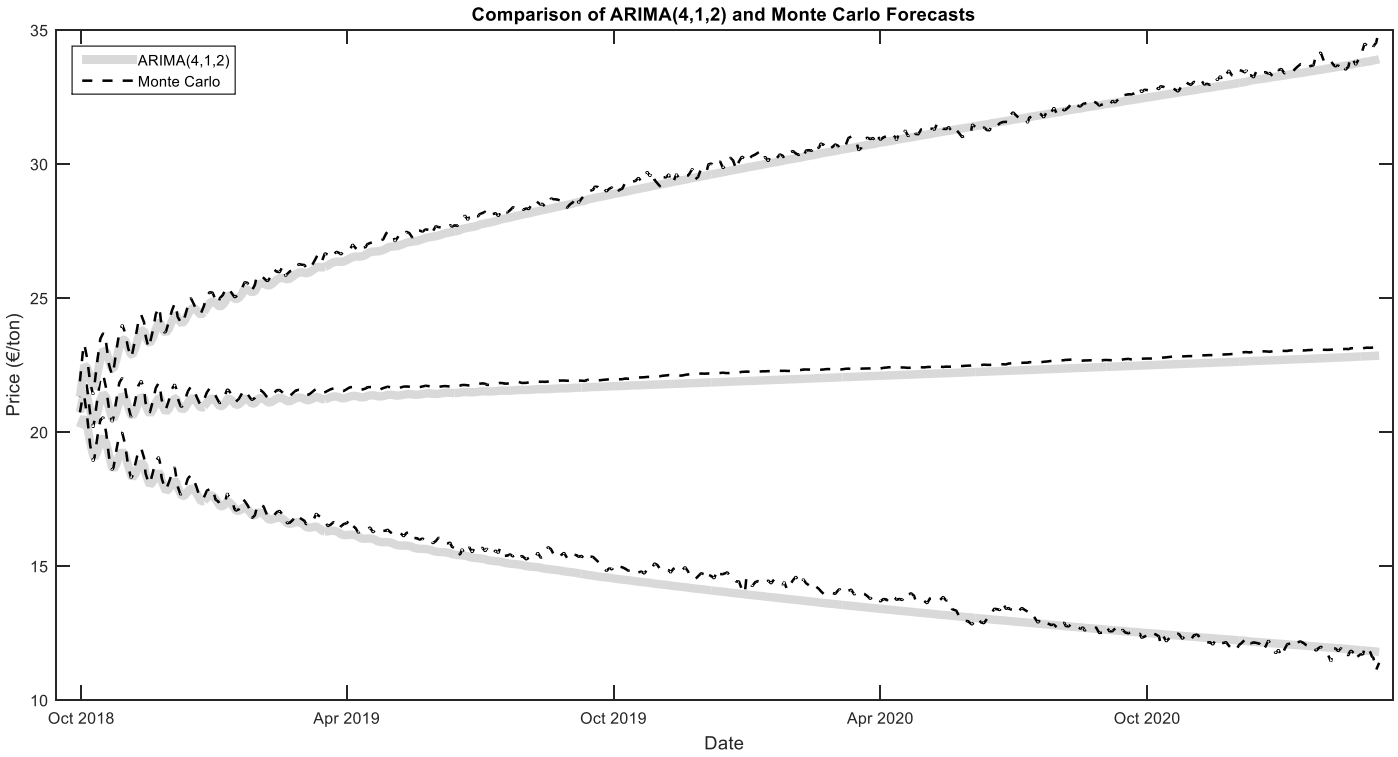


Figure 18. Comparison of ARIMA(4,1,2) and Monte Carlo Forecasts.

The final forecasted solution presents the distribution showed in *Figure 19*. Along these 2.34 years predicted, if we have into account the confidence interval, the most repeated price will be 21.5 €/ton, that could vary between a maximum of 33.9 €/ton and a minimum of 11.78 €/ton within a confidence interval of 95%.

Figure 19. Distribution of CO_2 allowance prices forecasted.

5.2.2 Neural Network for forecasting CO_2 allowance prices

Our neural network was developed using MATLAB software's neural network toolbox package dynamic feedback nonlinear autoregressive neural network (NARNET). Testing different combinations of feedback delays and number of neurons in hidden layer resulted in some findings:

- From the model predictions, it is obtained a good match between the real and predicted values. However, looking in more detail, the metrics and model evaluation can be very misleading.
- The CO_2 historical allowance prices data is a completely stochastic process. Due to this fact, the idea of using historical data as a training set in order to learn the behavior and predict future outcomes is simply not possible.
- Time series data tend to be correlated in time and exhibit a significant autocorrelation. In this case, that means that the index at time "t+1" is quite likely close to the index at time "t". What the model is actually doing is that when predicting the value at time "t+1", it simply uses the value at time "t" as its prediction.
- Defining the model to predict the difference in values between time steps rather than the value itself, is a much stronger test of the model predictive powers. In that case, it cannot simply use the data that has a strong autocorrelation and use the value at time "t" as the prediction for "t+1". This check provides a better test of the model and if it has learnt anything useful from the training phase, and whether analyzing historical data can actually help the model predict future changes. The results indicate that the model is not able to predict future changes based on historical events, which is the expected result in this case, since the data is generated using a completely stochastic random walk process.

The equations presented in sections 7.1.3.1 and 7.1.3.2 of Appendix for AIC and BIC have the disadvantage of calculating the maximum log-likelihood. When it is analyzed the performance of different NN architectures, AIC and BIC can be also computed as follows [20]:

$$AIC = Nln\left(\frac{SSE}{N}\right) + 2k$$

$$BIC = Nln\left(\frac{SSE}{N}\right) + kln(N)$$

where

- k is the number of parameters
- SSE is the sum of square errors
- N is the number of data points (observations).

Table 8 shows the performance of the different architectures checked. There are combination of 3, 5 and 10 hidden neurons with a number of delays from 2 to 8. The red numbers are the best values of the architectures according with AIC, BIC, MAE and MAPE criterion. The rest of values in each architecture are also marked.

Table 8. Comparison of different architectures.

		Hidden Neurons														
		3					5					10				
		AIC	BIC	Rsq	MAE	MAPE	AIC	BIC	Rsq	MAE	MAPE	AIC	BIC	Rsq	MAE	MAPE
Number of delays	2	-4464	-4392	0.998	0.189	2.435	-4417	-4301	0.998	0.193	2.482	**-4533**	**-4307**	**0.998**	**0.186**	**2.412**
	3	-4179	-4091	0.997	0.213	2.910	-4483	-4339	0.998	0.192	2.486	-4478	-4197	0.998	0.191	2.459
	4	-4442	-4337	0.998	0.189	2.430	-4544	-4373	0.998	0.189	2.447	-4498	-4162	0.998	0.186	2.425
	5	**-4542**	**-4421**	**0.998**	**0.189**	**2.446**	-4437	-4239	0.998	0.193	2.493	-4381	-3990	0.998	0.193	2.504
	6	-4431	-4294	0.998	0.191	2.448	-4386	-4160	0.998	0.196	2.563	-4425	-3979	0.998	0.189	2.475
	7	-3468	-3313	0.996	0.192	2.464	**-4567**	**-4314**	**0.998**	**0.192**	**2.472**	-4276	-3775	0.998	0.193	2.465
	8	-4546	-4375	0.998	0.188	2.431	-4460	-4179	0.998	0.189	2.441	-4430	-3873	0.998	0.188	2.439

According to the criteria chosen (AIC, BIC or MAE/MAPE criteria) there are different models that could be selected. NN(1:5,3) has the better BIC, while NN(1:7,5) has the better AIC and NN(1:2,10) has a smallest error (Section 7.1.3 explains why this models are better according with the parameter criteria) . When, the predicted values are represented, they tend to a constant value that is different in each architecture, as shown in *Figure 20.*

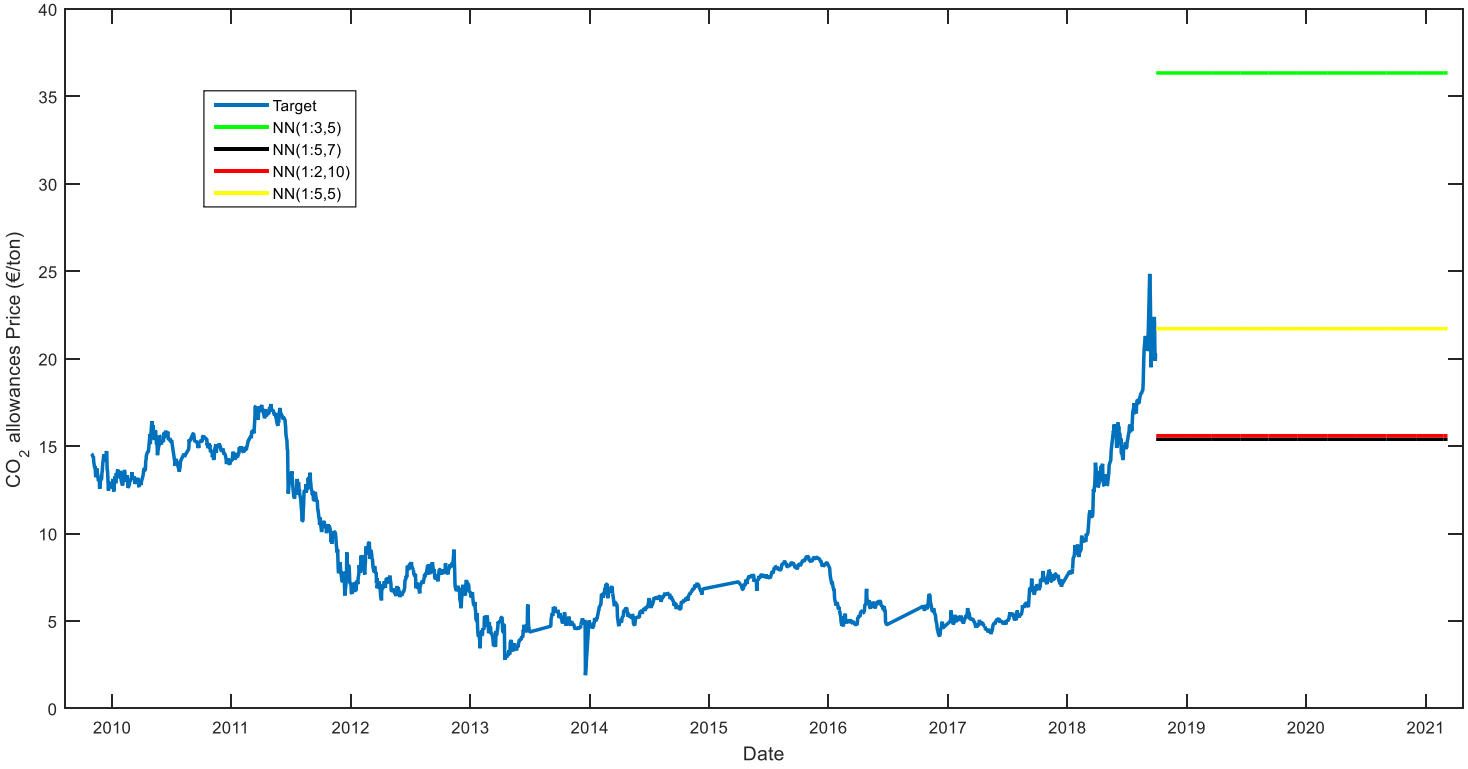


Figure 20. Neural Network Times Series ahead prediction of different architectures.

The different constant value predictions make one suspect that, from this data set, patterns for predicting future values cannot be extracted. Despite these results, futher work should be done to ensure that this model can be discarded.

Following the procedure stated above, if we consider AIC as criteria, the best architecture is NN(1:7,5). In *Figure 21* the response of output and error of NN(1:7,5) is shown.

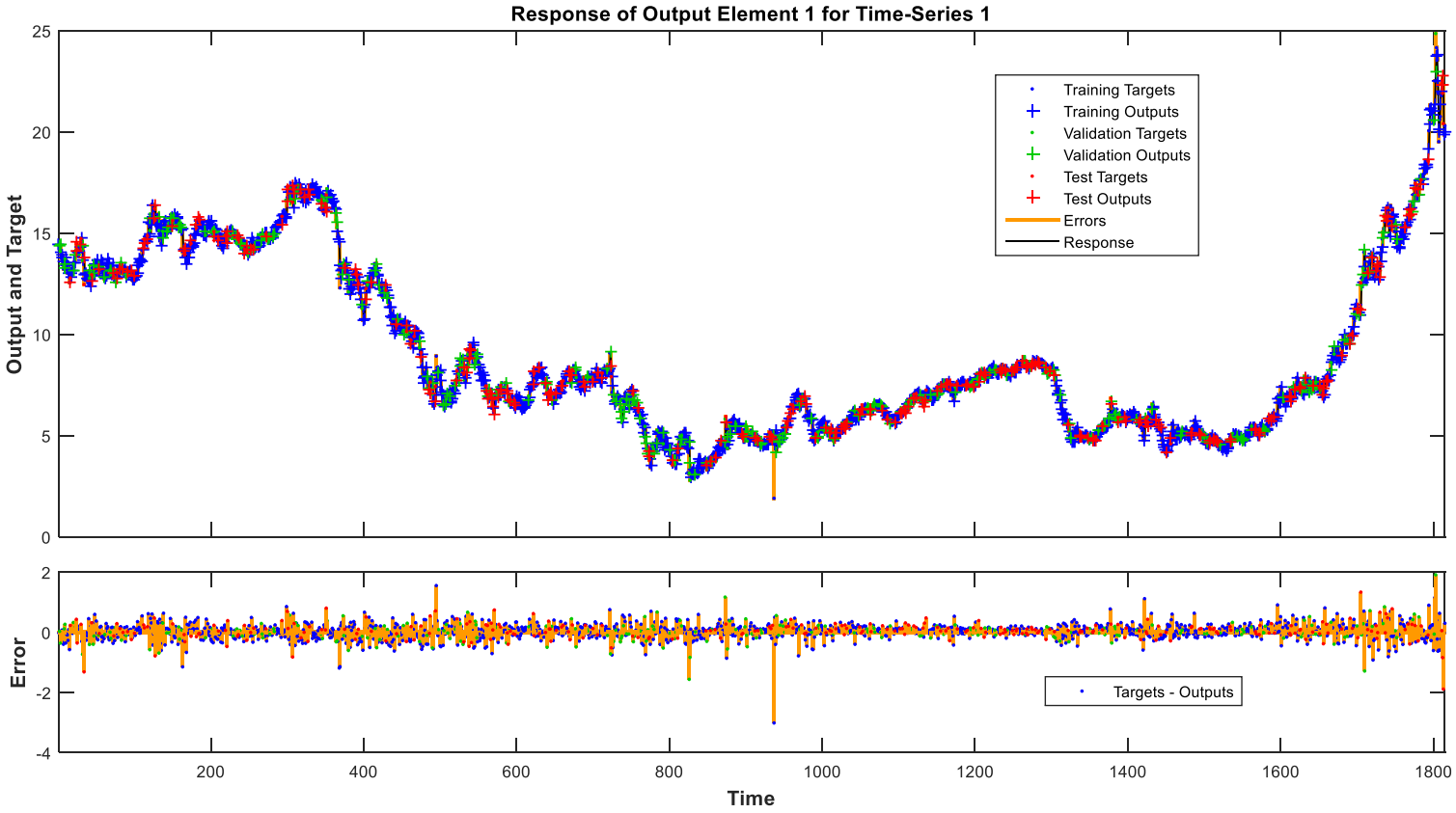


Figure 21. NN(1:7,5) response and error.

Table 9 presents a comparison between the performance of the ARIMA and NN best models according to AIC criteria. It is important to mention that SSE and log-likelihood are different statistic methods. Therefore, in order to compare the AIC and BIC values of ARIMA model with respect to NN ones, they must be computed using the same criteria. It was computed the value of AIC and BIC taking into account the sum of square errors.

Table 9. Comparison of ARIMA and NN performance.

	AIC	BIC	Rsq	MAE	MAPE	SSE
ARIMA(4,1,2)	-4450.51	-4411.96	0.995	0.191	2.457	157.174
NN(1:2,10)	-4566.8	-4313.5	0.998	0.192	2.472	141.274

As can be seen in *Table 9*, ARIMA(4,1,2) has a better value of BIC and MAE or MAPE, while NN(1:7,5) gives a better performance in AIC. Based on the results, ARIMA(4,1,2) predicts safer forecasting values. It offers a more realistic trend, that shows a possible increase in CO_2 allowance prices with its confidence intervals. Meanwhile, NN(1:7,5) does not offer a significant trend. Like

all other NN models, it arrives at a steady state constant value that varies in a significant manner in function of the architecture chosen.

Despite the parameters obtained, it is very important to be very careful when evaluating the model performance in terms of prediction accuracy. As can be seen in

Figure 22, time series data tends to be correlated in time and exhibit a significant autocorrelation ("*t*+1" is quite likely close to the index at time "*t*"). This indicates that the value at time "*t*+1" simply uses the value at time "*t*" as its prediction

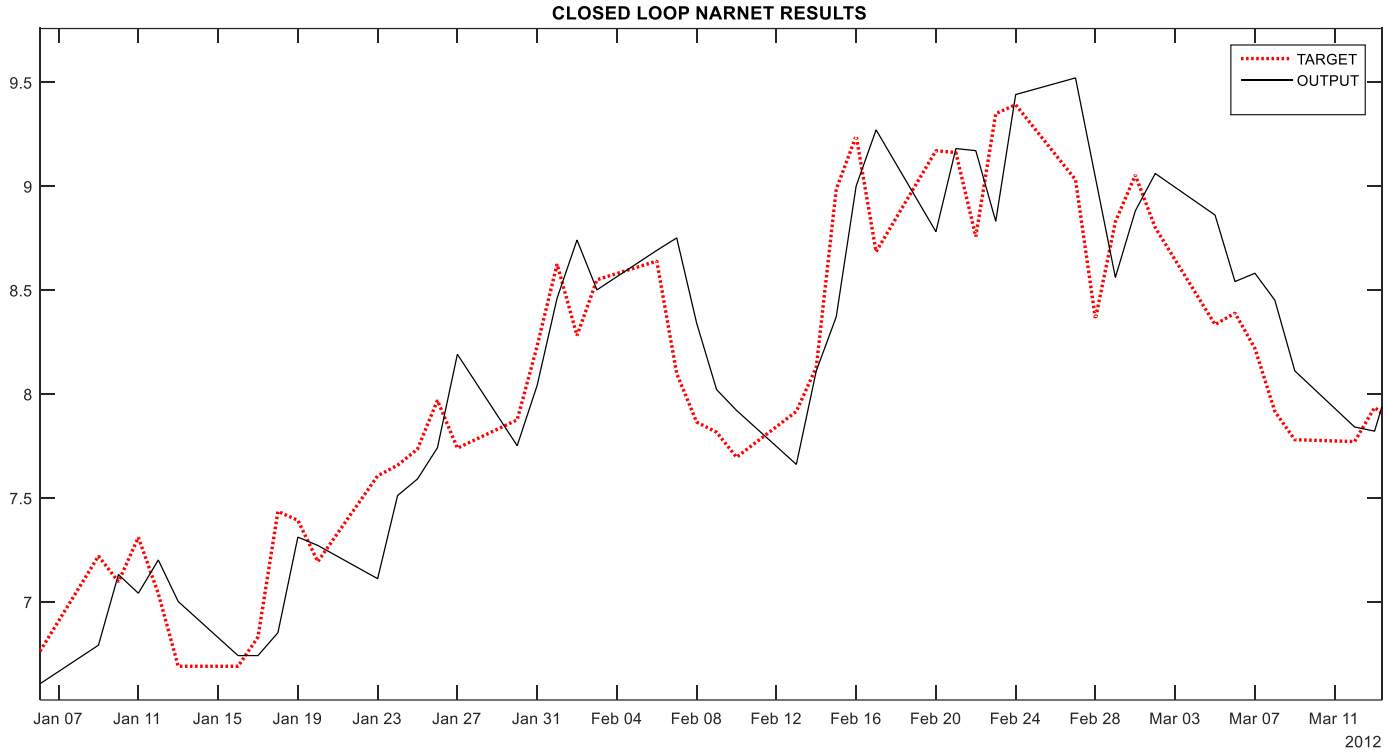


Figure 22. Time delayed predictions autocorrelated.

On the other hand, being able to predict the time-differenced data, rather than the data directly, is a much stronger indication of the predictive power of the model.

Figure 23 indicates that the model is not able to predict future changes based on historical events, which is the expected result in this case, since the data is generated using a completely stochastic random walk process. In other words, the time series is non-stationary and making it stationary shows no obviously learnable structure in the data. This can be concluded once the output and target of the differenced data has such poor correlation.

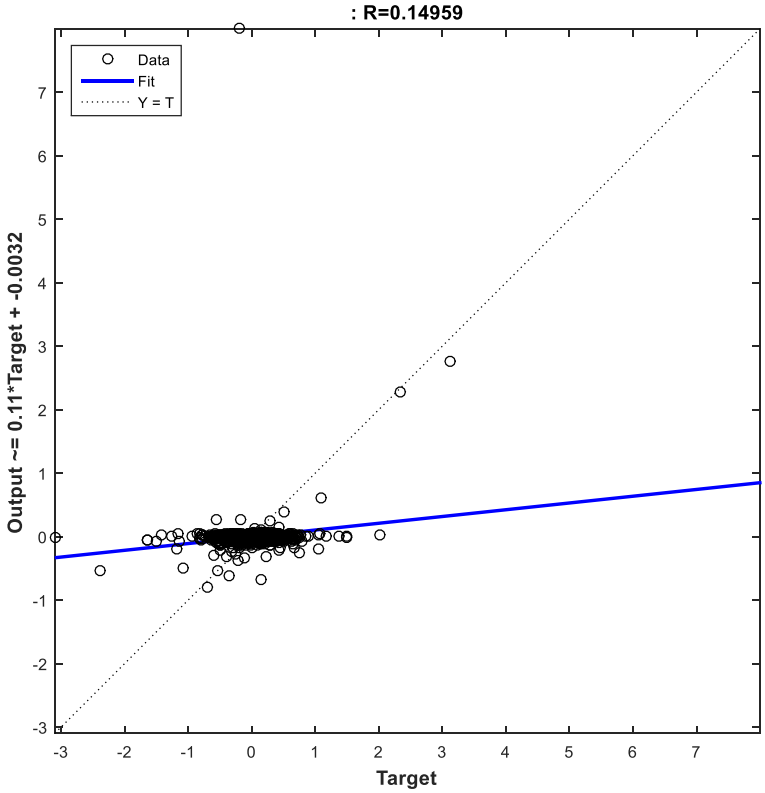


Figure 23. Time differenced output vs target.

As can be seen, the fit of the data is very poor and with this test is finally confirmed that artificial neural network is not capable of learning a patron in the CO_2 data as it is totally stochastic, and therefore, these predictions are not going to be used as parameters under uncertainty in the design of the petrochemical supply chain.

5.3 Supply chain results

The models described in the previous sections have been applied to the design of a SC with two plants, located in Neratovice and Tarragona, which produce 6 main products: acetaldehyde, acetone, acrylonitrile, cumene, isopropanol and phenol which sell in the following European cities: Leuna (Germany), Neratovice (Czech Republic), Sines (Portugal) and Tarragona as shown in *Figure 24.*

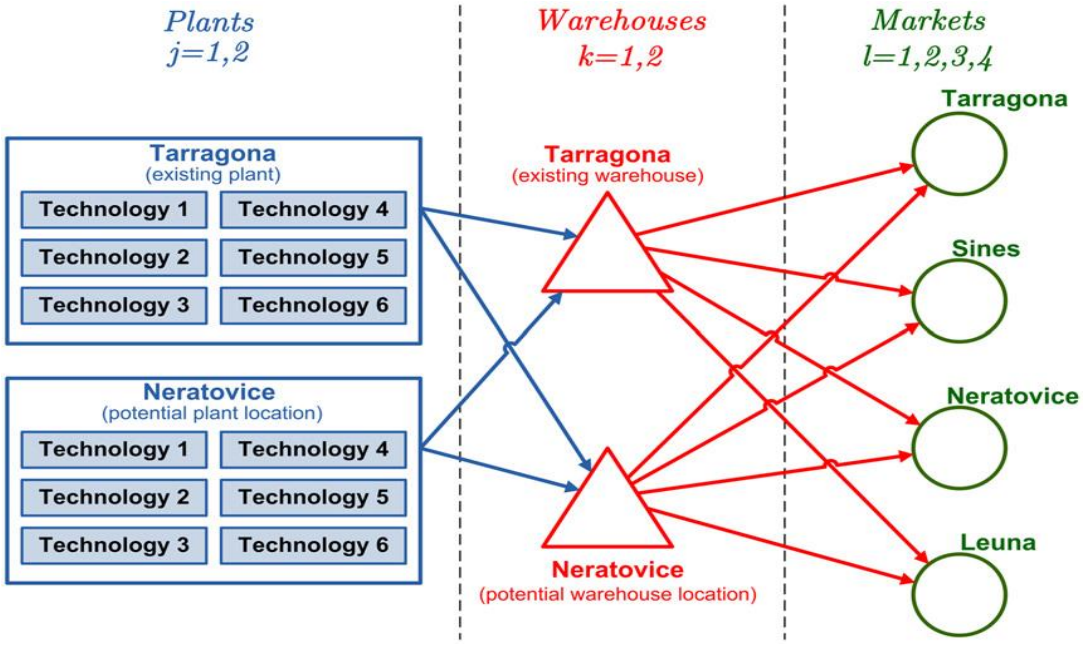


Figure 24. Supply chain studied.

5.3.1 Deterministic supply chain results

Uncertainty appears in all systems, in this case, it is due to parameters that represent information about the future. All this kind of problems could be solved by ignoring the inherent uncertainty of each of them, but it could constitute a violation in reality and it may design a supply chain far away from the one needed. In order to discuss the use of stochastic optimization, the risk curve of the deterministic solution is presented together with other stochastic solutions (see

5). In deterministic optimization the problem parameters are supposed to be known with certainty. To solve the presented deterministic model, it is used the mean forecasted CO_2 allowance prices as parameters (given in *Figure 18).* Note that the deterministic model can be easily obtained from the stochastic one (Eqs. (61)-(89) of section 7.2.1.2) by defining only one single scenario which corresponds to this mean value mentioned.

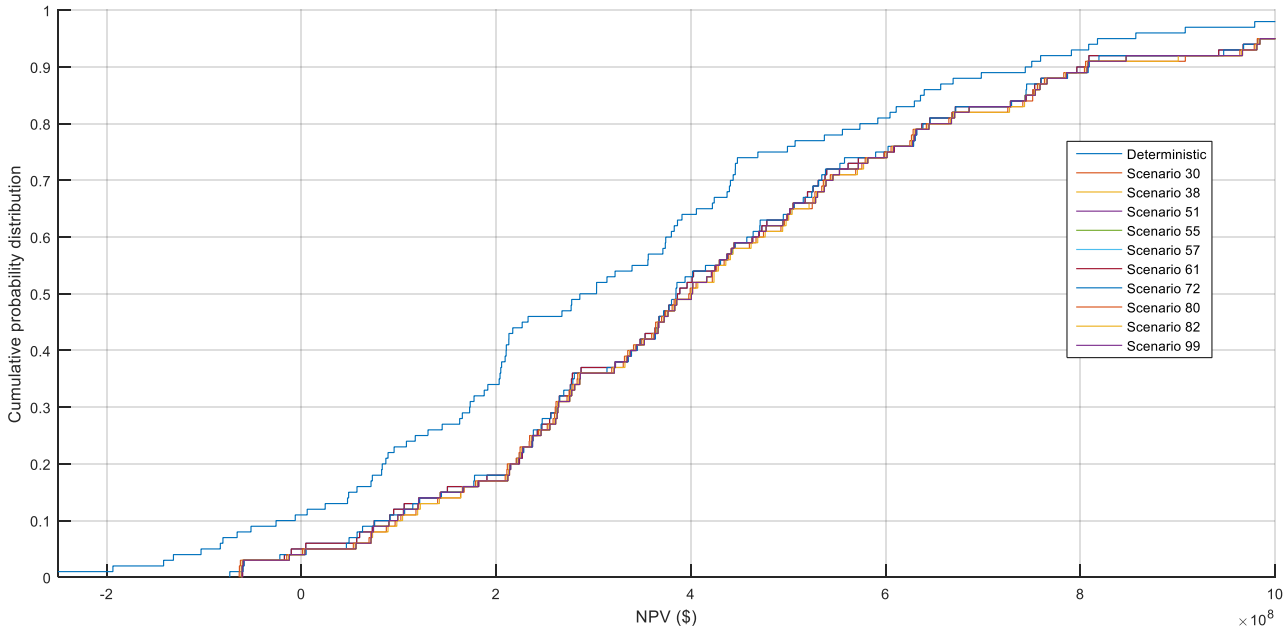


Figure 24. Deterministic solution vs stochastic solution.

As can be seen, the deterministic solution (scenario 1) is always worse than the stochastic solutions. It is clear that if someone had to choose between the represented lines, the blue line, that is to say, the deterministic solution will be always excluded since, for the same NPV value, it has a greater risk, and in the same way, for a same risk value the NPV is always lower. Apart from this, it is also important to mention that solving the stochastic problem ensures a more robust solution that takes into account the possible fluctuation of parameters. For these reason, stochastic optimization is a better guarantee to find better and more robust solutions.

5.3.2 Stochastic supply chain results

As it has been said above, the problem under study is stochastic (depends on chance), since it has an uncertain parameter (the CO_2 emission rights price). For this reason, it cannot be solved for a single price scenario, but it has to be solved for a set of possible price scenarios that will have a certain probability of occurrence associated (it is considered that each scenario has exactly the same probability of occurrence). In this way, a robust model will be obtained and that includes the source of uncertainty.

This work follows the two-stage stochastic programming approach with discretization of the uncertainty CO_2 emissions rights price by ARIMA model estimations (see section 7.1.5 and [22]). Once it has been demonstrated that ANN are not capable of forecasting the future CO_2 allowance prices, ARIMA(4,1,2) will compute them in order to formulate the scenarios. shows the evolution of prices generated when 100 scenarios are considered. *Figure 26* shows the evolution of prices generated when 100 scenarios are considered.

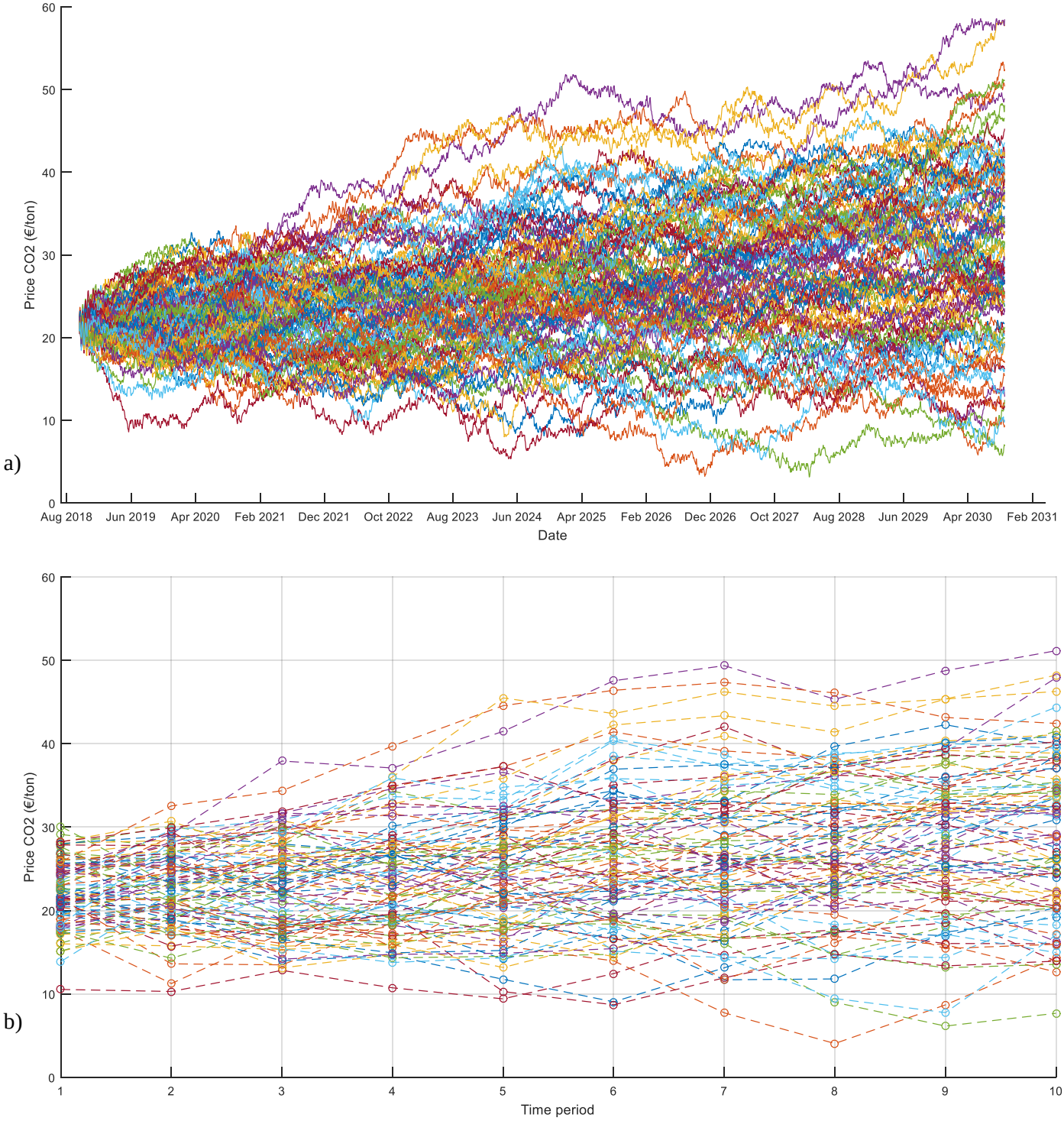


Figure 26. Generation of scenarios with ARIMA(4,1,2): a) Price evolution ; b) Price related with time period.

The difference between *Figure 26 a)* and *b)* is the number of data points considered. *Figure 26 a)* shows all the forecasted data points obtained with the ARIMA(4,1,2) model (almost a CO_2 allowance price each day) and *Figure 26 b)* only shows the CO_2 allowance price at the end of each year, when it has to be decided whether rights must be bought or sold. The optimization model only takes into account the data in *Figure 26 b).*

5.3.2.1 Objective function results

Taking into account the price of each scenario for the year end period (September 29th), it is solved the stochastic problem iteratively for an increasing number of scenarios. It has been established 100 scenarios as a good quantity, due to the fact that results have changed comparing with less scenarios.

The set of risk curves generated at the end of the solution process includes a number of curves equal to the number of scenarios explored, although some of them are dominated by others. For this reason, it is necessary to filter the risk curves (see section 7.2.2.4). Besides, in order to find the best risk curves, some financial metrics are used (see section 7.1.6 and [21] and [23]).

In *Table 10* the most interesting scenarios according to the different metrics explained below are presented. It has been recollected the greatest values of opportunity value (OV) in absolute value, as well as in relative value, opportunity value differenced (OVD), the maximum expected NPV (E[NPV]), the scenario with greater value at risk (VaR) and worst case (WC). Moreover, it has been collected the scenario with lower downside risk (DSR) and value at risk difference (VaRD).

Table 10. Interesting scenarios after filtering solutions.

OV ($)	WC ($)	VaR ($)	DSR ($)	OVD ($)	VaRD ($)	E[NPV] ($)
s30, s38, s55, s99	s51, s57, s61	s72	s82	s30	s30	s53, s80

Each metric gives a different perspective to evaluate risk. In this work, it is presented all these metrics that allow management of financial risk according to the decision maker's preference. In this way, the propose to use the concept of minimum downside risk (DSR) and value at risk difference (VaRD) to measure the recourse cost variability and obtain solutions appealing to a risk-averse investor. However, a risk-taking attitude would be to focus in the maximum values of opportunity values (OV) and opportunity value differenced (OVD), which would be interesting for a slightly more conservative risk-taker who wants to make sure that the average behavior is acceptable (good E[NPV]).

As can be seen in *Table 10*, the most repeated scenario is s30. It appears in three of the measures. *Table 11* shows the different results obtained.

Table 11. Risk metrics results from interesting scenarios.

Scenario	OV ($)	WC ($)	VaR ($)	DSR ($)	OVD ($)	VaRD ($)	E[NPV] ($)
Determ.	$8.084 \cdot 10^8$	$-2.867 \cdot 10^8$	$-1.031 \cdot 10^8$	$3.750 \cdot 10^8$	$2.724 \cdot 10^8$	$5.360 \cdot 10^8$	$5.360 \cdot 10^8$
s30	$9.811 \cdot 10^8$	$-6.122 \cdot 10^7$	$3.666 \cdot 10^6$	$8.796 \cdot 10^7$	$5.596 \cdot 10^8$	$4.178 \cdot 10^8$	$4.215 \cdot 10^8$
s38	$9.811 \cdot 10^8$	$-6.122 \cdot 10^7$	$3.666 \cdot 10^6$	$8.796 \cdot 10^7$	$4.336 \cdot 10^8$	$5.439 \cdot 10^8$	$5.476 \cdot 10^8$
s51	$9.671 \cdot 10^8$	$-6.048 \cdot 10^7$	$-1.043 \cdot 10^7$	$1.008 \cdot 10^8$	$4.871 \cdot 10^8$	$4.904 \cdot 10^8$	$4.800 \cdot 10^8$
s53	$9.811 \cdot 10^8$	$-6.122 \cdot 10^7$	$3.666 \cdot 10^6$	$8.796 \cdot 10^7$	$4.054 \cdot 10^8$	$5.721 \cdot 10^8$	$5.758 \cdot 10^8$
S57	$9.671 \cdot 10^8$	$-6.048 \cdot 10^7$	$-1.043 \cdot 10^7$	$1.008 \cdot 10^8$	$4.820 \cdot 10^8$	$4.955 \cdot 10^8$	$4.851 \cdot 10^8$
s61	$9.671 \cdot 10^8$	$-6.048 \cdot 10^7$	$-1.043 \cdot 10^7$	$1.008 \cdot 10^8$	$4.169 \cdot 10^8$	$5.606 \cdot 10^8$	$5.502 \cdot 10^8$
s72	$9.675 \cdot 10^8$	$-7.358 \cdot 10^7$	$4.839 \cdot 10^6$	$9.527 \cdot 10^7$	$4.173 \cdot 10^8$	$5.454 \cdot 10^8$	$5.503 \cdot 10^8$
s80	$9.785 \cdot 10^8$	$-6.391 \cdot 10^7$	$9.814 \cdot 10^5$	$7.969 \cdot 10^7$	$3.882 \cdot 10^8$	$5.893 \cdot 10^8$	$5.902 \cdot 10^8$
s82	$9.802 \cdot 10^8$	$-6.216 \cdot 10^7$	$2.731 \cdot 10^6$	$7.785 \cdot 10^7$	$4.044 \cdot 10^8$	$5.730 \cdot 10^8$	$5.758 \cdot 10^8$
s99	$9.811 \cdot 10^8$	$-6.122 \cdot 10^7$	$3.666 \cdot 10^6$	$8.796 \cdot 10^7$	$4.182 \cdot 10^8$	$5.593 \cdot 10^8$	$5.629 \cdot 10^8$

As can be seen above, the variation in the metrics seems to be relatively low for OV, however, the difference between scenarios $(\max(OV_s) - OV_s)$ ranges from $1.360 \cdot 10^7$ to $1.404 \cdot 10^7$\$, which are large amounts of money. The maximum variation occurs with VaR, especially in s51, s57 and s61, which have different orders of magnitude, even there are negative values. When it is compared the relative metric OVD, the variation between scenarios is more significative than OV. Comparing *Table 10* and *Table 11* it can be found the maximum and minimum values of each scenario according to the measure, however, the results could be presented much more visually as a whole. *Table 12* shows the normalized values with colors, which gives a better perspective about what is happening.

Strong color cells are related with the greater values and light color with the lower values. The numbers have been normalized for each financial metric and they vary from 0 to 1 (being 0 the minimum value among scenarios and 1 the maximum value among scenarios). Selecting a scenario, it can be seen the most important information of each risk curve and it can also be compared with different scenarios.

Table 12. Normalized data from interesting scenarios.

Scenario	OV ($)	WC ($)	VaR ($)	DSR ($)	OVD ($)	VaRD ($)	E[NPV] ($)
s30	1,000	0,944	0,923	0,441	1,000	0,000	0,000
s38	1,000	0,944	0,923	0,441	0,265	0,735	0,747
s51	0,000	1,000	0,000	1,000	0,577	0,423	0,347
s55	1,000	0,944	0,923	0,441	0,100	0,900	0,914
s57	0,000	1,000	0,000	1,000	0,547	0,453	0,377
s61	0,000	1,000	0,000	1,000	0,167	0,833	0,763
s72	0,031	0,000	1,000	0,760	0,169	0,744	0,763
s80	0,809	0,739	0,747	0,080	0,000	1,000	1,000
s82	0,933	0,872	0,862	0,000	0,095	0,905	0,914
s99	1,000	0,944	0,923	0,441	0,175	0,825	0,838

For example, scenario s30 is clearly for risk-takers. It has great values of OV, which means it could be obtained a greater quantity of NPV than in scenarios s51, s57, s61, s72, s80 and s82 (difference of $1.36 - 1.40 \cdot 10^7$\$ more that could be earned). Besides, in the worst-case scenario the NPV would not lose as much as in others. The great value of VaR indicates that at 5% of cumulative probability, the NPV quantity would be good. This happens especially since other scenarios still would not have positive NPV (s51, s57, s61). Apart from this, downside risk shows a value below 0.5, which means there is a relatively low risk in order to get the target value imposed of $1.592 \cdot 10^8$\$. Moreover, OVD is the greatest of all scenarios, what means that there is a great distance between the E[NPV] (which in fact is the lowest) and the opportunity value that could be obtained (the greatest in relative and absolute terms). Finally, the VaRD indicates that the distance between the E[NPV] and NPV at 5% is the lowest of all scenarios. In conclusion, scenario s30 could obtain the greatest NPV of all scenarios. However, the risk taken is also high since the difference between the expected value and the opportunity value is large. It would be more likely to earn $3.666 \cdot 10^6$\$ than $9.811 \cdot 10^8$ \$ due to VaRD is nearer to the E[NPV] than OVD is.

Taking into account the criterion shown with scenario 30 and *Table 12*, it can be identified among the different scenarios the best ones. *Figure 27* shows a classification of scenarios according to risk-taker and risk-averse decision makers, which, in turn, divides the different scenarios considering the risk metrics. Looking at this hierarchical classification, it is clear that a risk-taker, which wants to earn the maximum amount of money, would choose scenario 99, since it may be earned the same quantity that scenarios 30, 38 and 55, however, scenario 99 has a lower value of OVD which ensures lower risk to obtain the opportunity value of $9.811 \cdot 10^8$ \$. On the other hand, a risk- averse decision maker that only wants to ensure a good NPV without taking too many risks, will look only at the best VaR and would choose scenario 72.

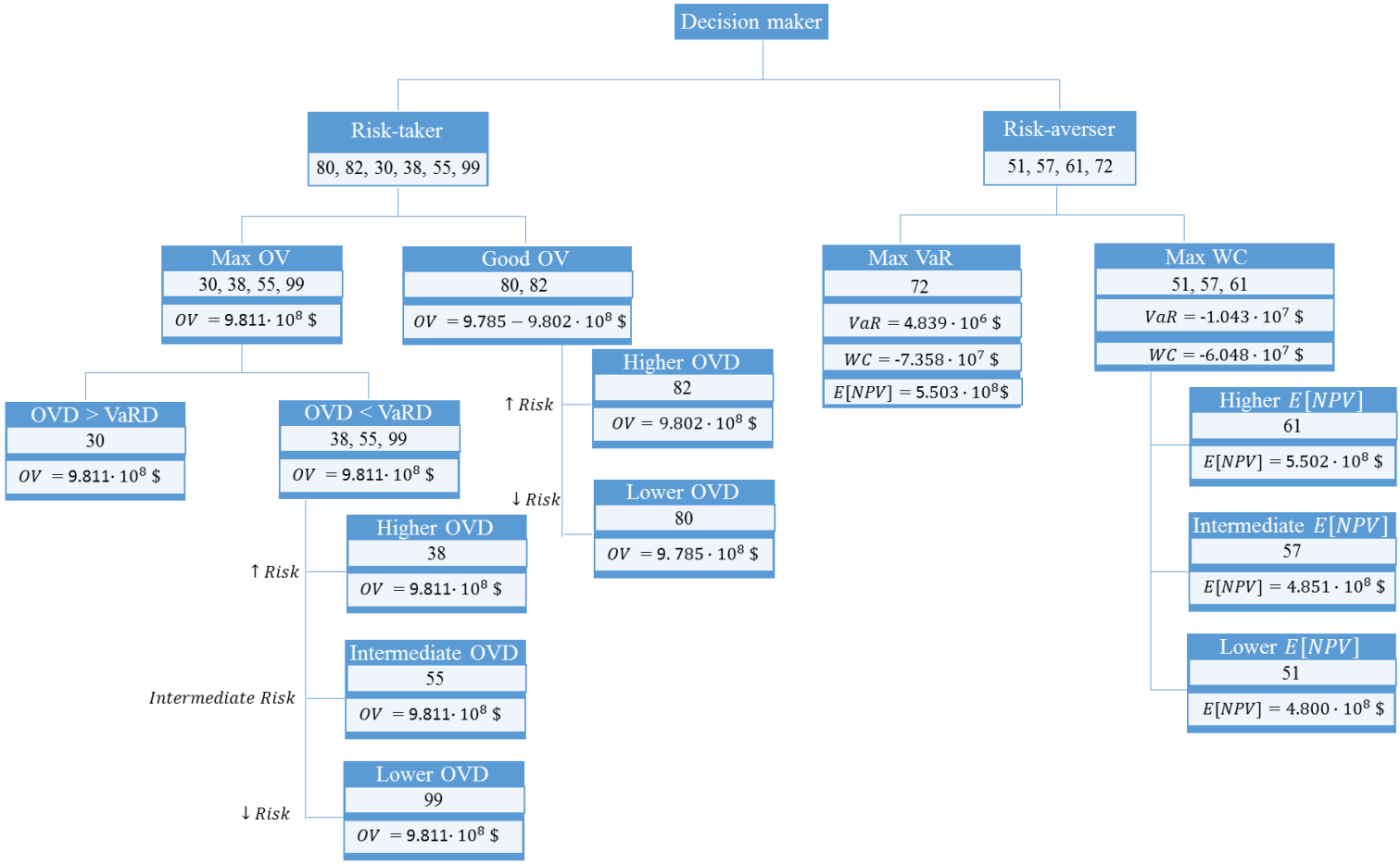


Figure 27. Classification of scenarios.

Looking at the entire curve is important because, even when one is a risk-averse decision maker, and, consequently, concerned with the profit distribution at low profit expectations, one can also assess the effect of risk-related decisions in the downside region of the profit distribution on the loss of profit potential at the other end of the spectrum. For this reason, the risk curves of all filtered scenarios are presented. *Figure 28*, *Figure 29* and *Figure 30* show these risk curves. It seems that they are very similar, however, the difference between metrics are remarkable.

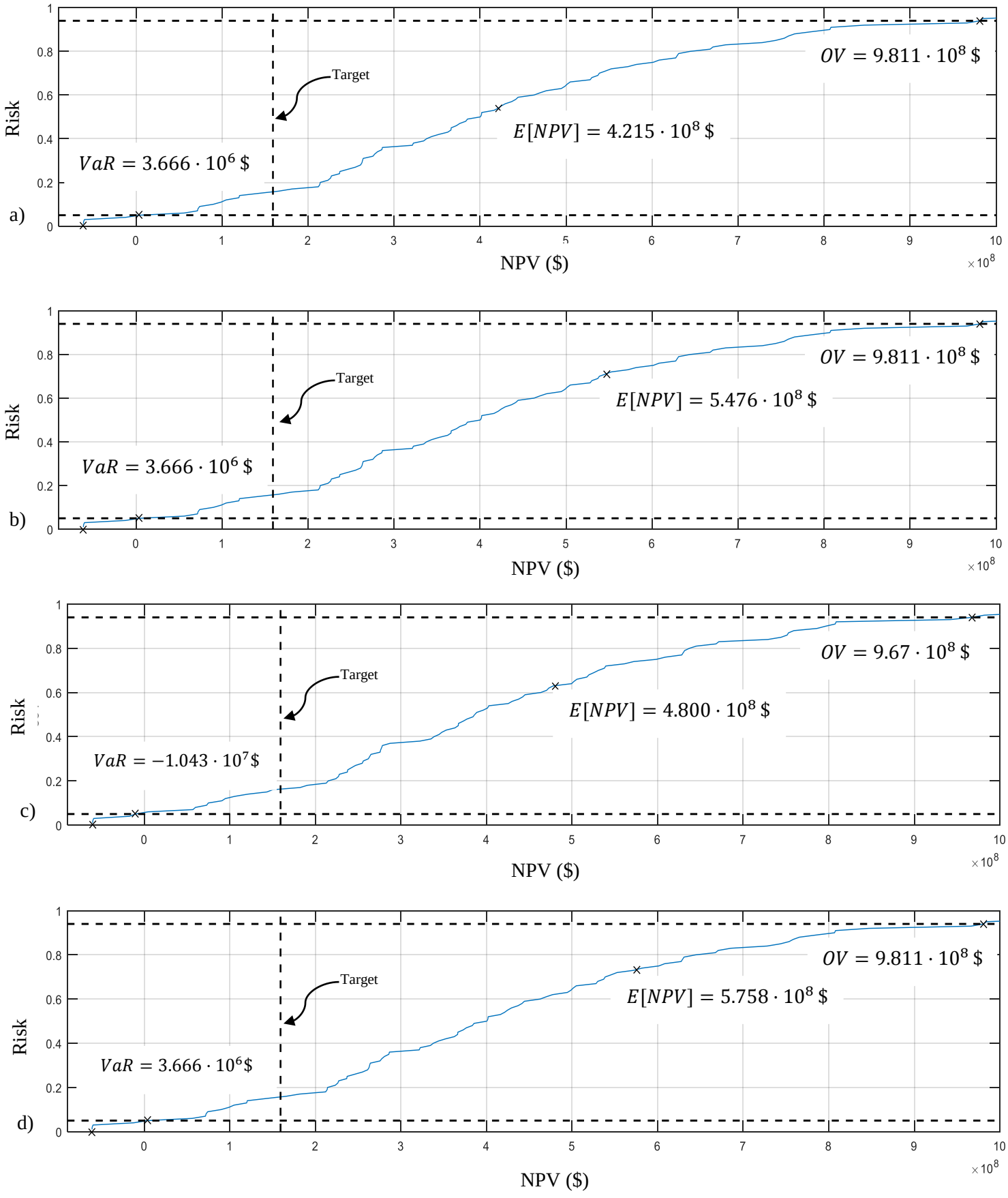


Figure 28. Risk curve of scenarios: a) 30; b) 38; c) 51; d) 55.

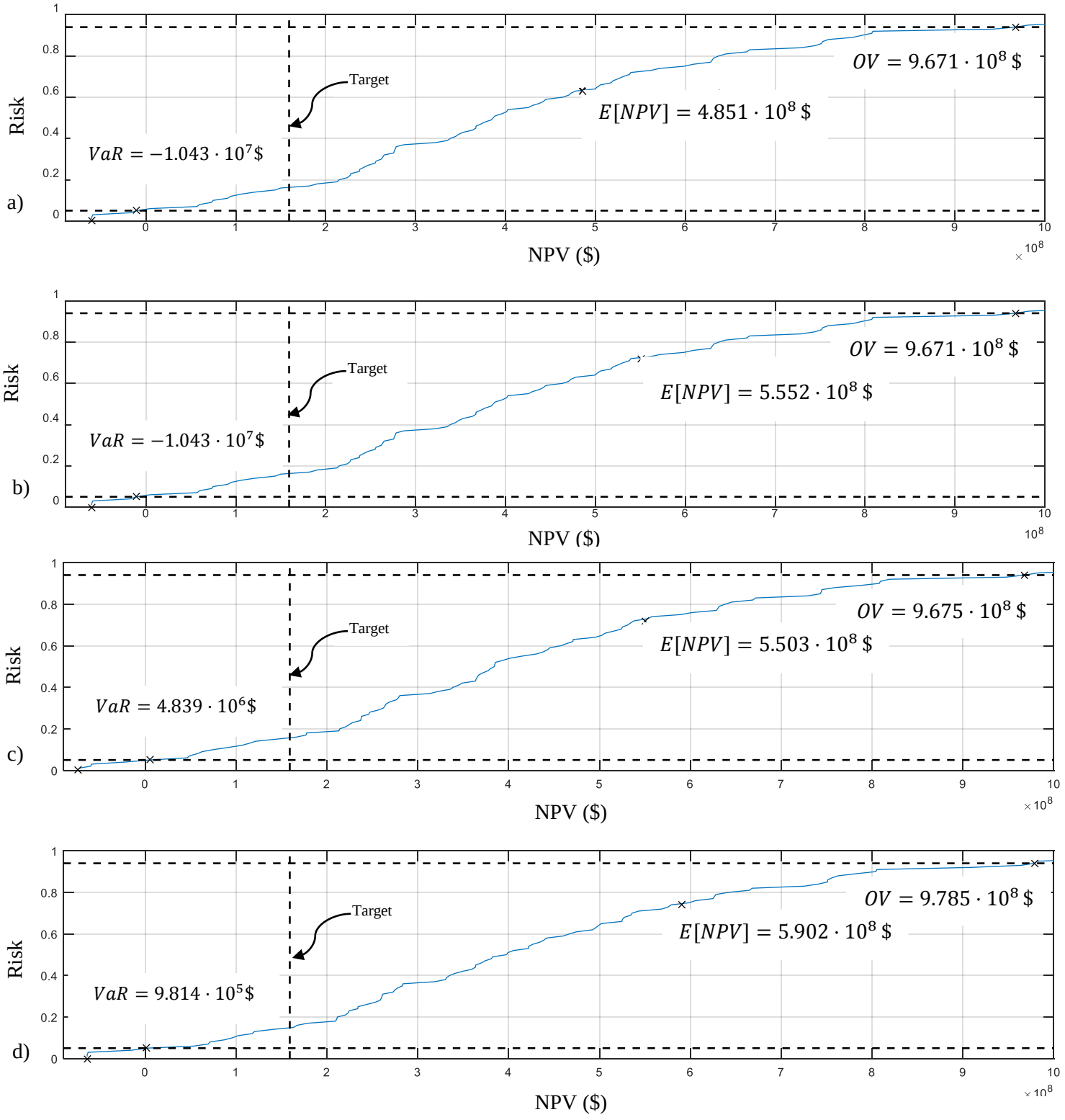


Figure 29. Risk curves of scenarios: a) 57; b) 61; c) 72; d) 80.

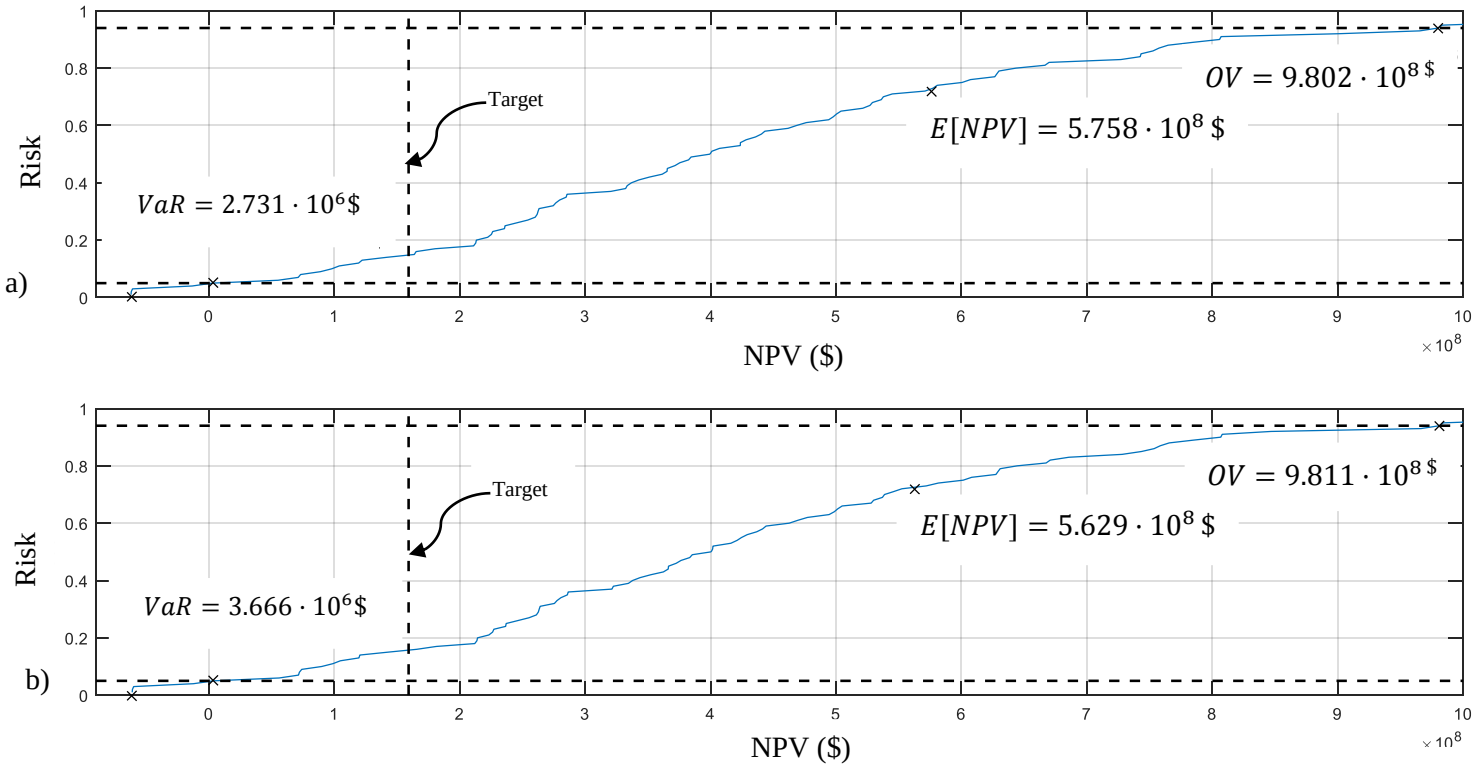


Figure 30. Risk curves of scenarios: a) 82; b) 99.

5.3.2.2 Structural variables results

Apart from presenting the objective function of our stochastic problem, it is even more important to present the structural variables determined, the variables that do not depend on the scenario. The variables that depend on the scenario are only used to solve the model and determine the structural variables. These structural variables would be used to design the supply chain. They are basically capacities of process and warehouses, as well as, its expansions in the different time periods. *Table 13* and *Table 14* show the results obtained.

Table 13. Capacity of process in all time periods at plants.

Scenario	Technology	Plant (tons)	
		Neratovice	Tarragona
Deterministic	Tech. 1	10000	100000
	Tech. 2		100000
	Tech. 3	10000	100000
	Tech. 4	10000	100000
	Tech. 5	10960	100000
	Tech. 6	10000	100000
30	Tech. 1	17983	100000
	Tech. 2		100000
	Tech. 3	23978	100000
	Tech. 4	11989	100000
	Tech. 5	49822	100000
	Tech. 6	23585	100000
38	Tech. 1	17983	100000
	Tech. 2		100000
	Tech. 3	23978	100000
	Tech. 4	11989	100000
	Tech. 5	49822	100000
	Tech. 6	23585	100000
51	Tech. 1	14027	100000
	Tech. 2		100000
	Tech. 3	18703	100000
	Tech. 4	10000	100000
	Tech. 5	38863	100000
	Tech. 6	18397	100000
55	Tech. 1	17983	100000
	Tech. 2		100000
	Tech. 3	23978	100000
	Tech. 4	11989	100000
	Tech. 5	49822	100000
	Tech. 6	23585	100000
57	Tech. 1	14027	100000
	Tech. 2		100000
	Tech. 3	18703	100000
	Tech. 4	10000	100000
	Tech. 5	38863	100000
	Tech. 6	18397	100000

Scenario	Technology	Plant (tons)	
		Neratovice	Tarragona
61	Tech. 1	14027	100000
	Tech. 2		100000
	Tech. 3	18703	100000
	Tech. 4	10000	100000
	Tech. 5	38863	100000
	Tech. 6	18397	100000
72	Tech. 1	11445	100000
	Tech. 2		100000
	Tech. 3	15260	100000
	Tech. 4	10000	100000
	Tech. 5	31709	100000
	Tech. 6	15010	100000
80	Tech. 1	32074	100000
	Tech. 2		100000
	Tech. 3	42765	100000
	Tech. 4	21382	100000
	Tech. 5	88860	100000
	Tech. 6	42064	100000
82	Tech. 1	22890	100000
	Tech. 2		100000
	Tech. 3	30520	100000
	Tech. 4	15260	100000
	Tech. 5	63417	100000
	Tech. 6	30020	100000
99	Tech. 1	17983	100000
	Tech. 2		100000
	Tech. 3	23978	100000
	Tech. 4	11989	100000
	Tech. 5	49822	100000
	Tech. 6	23585	100000

The capacity of process at plants remains constant along all time periods. The results show that only the Tarragona plant capacity remains constant in all scenarios and technologies at a value of 100000 ton of each product. On the other hand, the capacity in Neratovice depends on the scenario as well as the technology used. It is also important to mention that the risk-taker scenarios (scenarios 30, 38, 55, 80, 82 and 99) present greater capacity in their plants that the averse scenarios (scenarios 51, 57, 61 and 72), which is logical as risk-taker scenarios try to earn extra-money taking more risks.

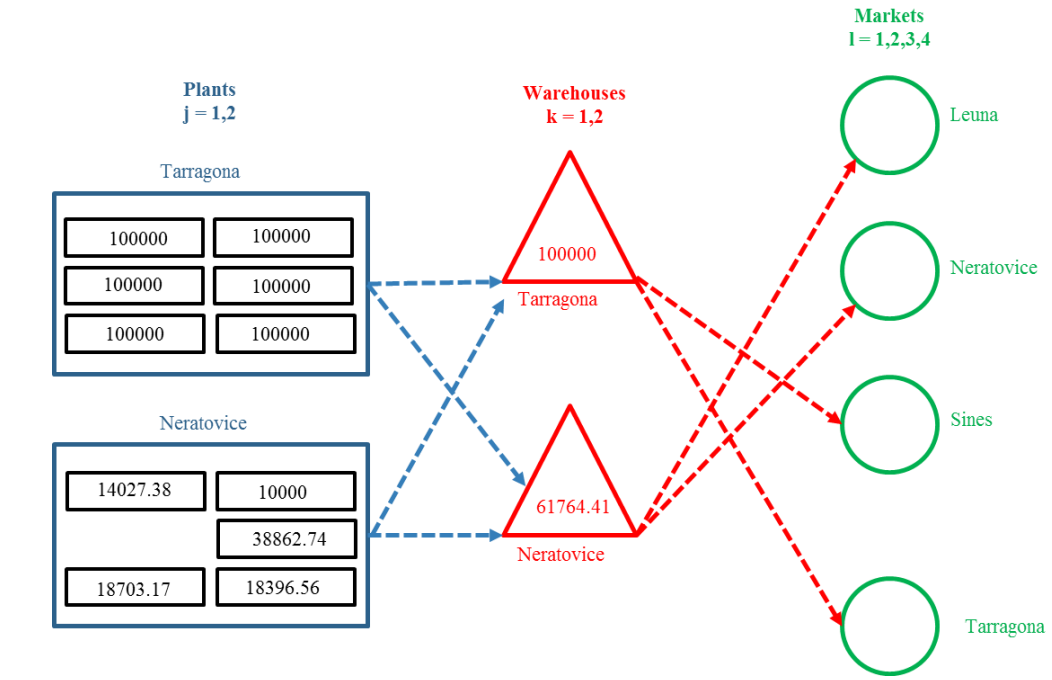
Table 14. Capacity of warehouse in time period t.

Scenario	Warehouse (tons)	
	Neratovice	**Tarragona**
Deterministic	5000	100000
30	84444.82	100000
38	84444.82	100000
51	61764.41	100000
55	84444.82	100000
57	61764.41	100000
58	22680.42	100000
61	61764.41	100000
72	80423.64	100000
80	169268.79	100000
82	115118.82	100000
99	84444.82	100000

The capacity of warehouses does not change along the Tarragona warehouse scenarios, as shown in *Table 14*. Equally as it happens with the capacity of process, the capacity of Neratovice warehouse is greater for the risk-taker scenarios and lower with the risk-averse scenarios. By increasing the production capacity, the warehouse capacity also needs to increase.

By way of example, the final result obtained with scenarios 52 (risk-averse) and 99 (risk-taker) shows in *Figure 31*. As was mentioned before, structural variables are related with the capacities of plant and warehouses, in which decision maker has the opportunity to choose the most appropriate configuration according to their risk criteria. However, it is important to mention that the links between plants, warehouses and markets are not structural variables, they are operational (they depends on the scenario) and all possible links have to be considered.

a)



b)

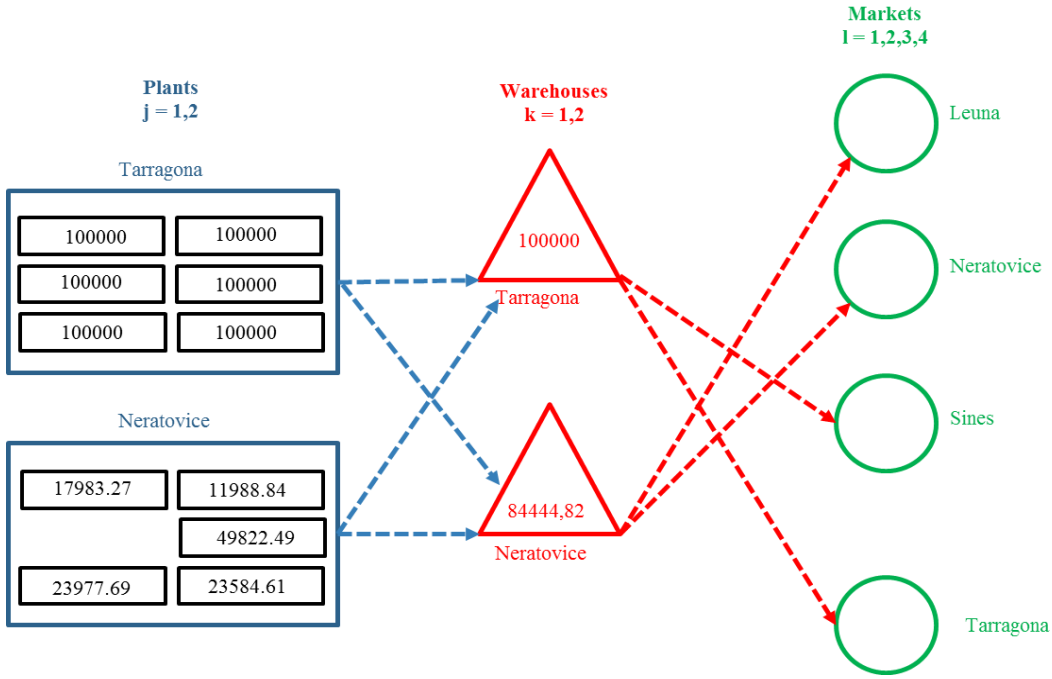


Figure 31. Structural variable results for scenarios: a) 52 (risk-averse); b) 99 (risk-taker).

5.3.2.3 Operational variables results

The operational variables are related to planning decisions and they are affected by the uncertainty in the price of CO_2 emission rights. In the following sections, it is shown what happens along the different scenarios, where some trends are observed. However, it is important to emphasise that there is no reason why the predicted trend happens, any scenario is possible individually.

5.3.2.3.1 Economic results

It is interesting to observe the probability distribution of the variables that depend on the scenario. As an example, *Figure 32* shows the probability distribution of cumene sales in the different markets during the first and last years.

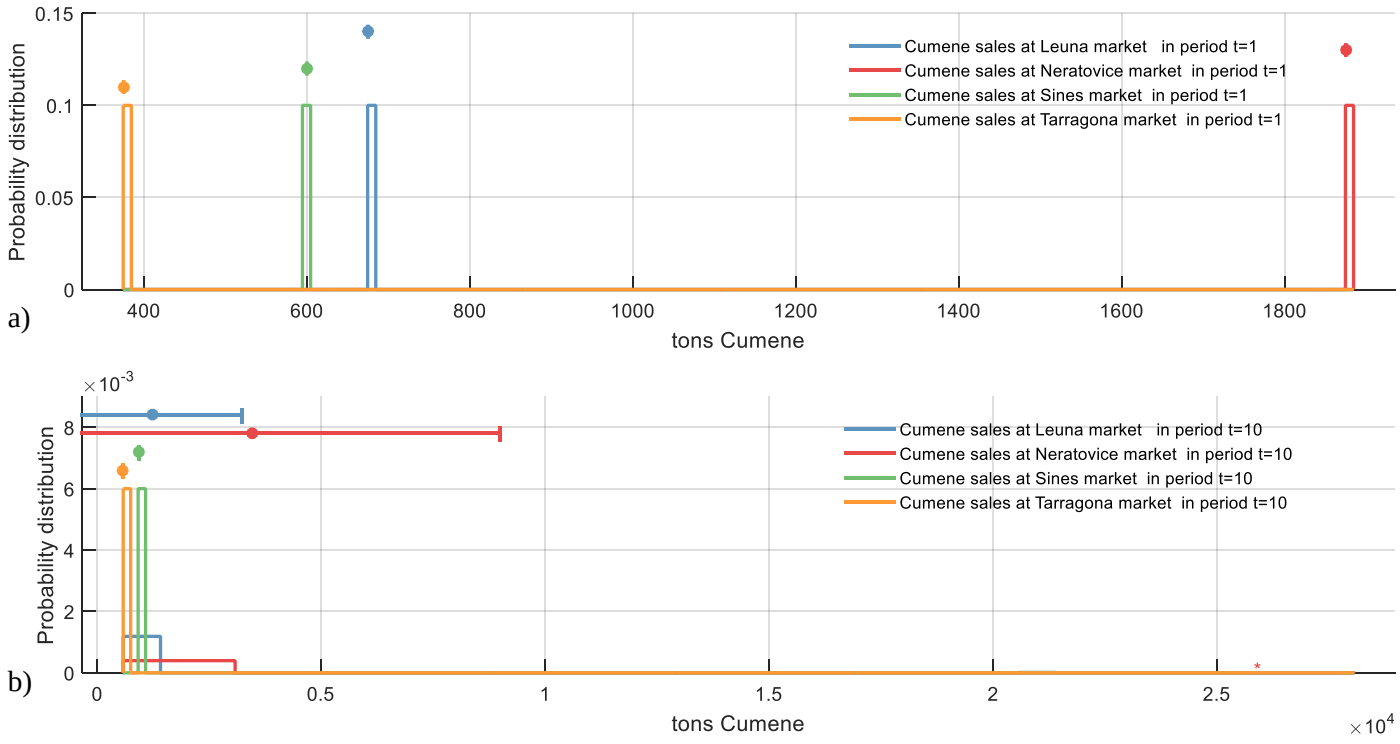


Figure 32. Probability distribution for all the scenarios for the cumene sales in the different markets during: a) the first year; b) the last year.

During the first year the sales are constant in every scenario, regardless there are a total of 100 scenarios, what shows that it is clear how to act at the beginning of the time horizon considered. Cumene sales at Neratovice market in period t=1 are the greatest, with a mean value of 1875 tons, followed by the Leuna market with a mean of 675 tons, Sines market with 600 tons and Tarragona market with 375 tons. The standard deviation in all cases is 0, however, at time period 10 dispersion appears in Neratovice and Leuna markets. In these cases, the cumene sales are 1246 and 3461 tons for Neratovice and Leuna, respectively. Cumene sales at Sines and Tarragona markets are 930.80 and 581.75 tons respectively, without dispersion. presents the sales of all chemicals at the different markets. In all cases the amount tends to be increased at the last time period.

Table 15. Sales at markets of chemicals in first and last period with its respective standard deviation.

		Acetaldehyde		Acetona		Acrylonitrile		Cumene		Isopropanol		Phenol	
	Markets (tons)	**Mean**	**St. desv**	**Mean**	**St. desv**	**Mean**	**St. desv**	**Mean**	**St. desv**	**Mean**	**St. desv**	**Mean**	**St. desv**
t_1	**Leuna**	675,00	0,00	540,00	0,00	900,00	0,00	675,00	0,00	450,00	0,00	630,00	0,00
	Neratovice	1875,00	0,00	1500,00	0,00	2500,00	0,00	1875,00	0,00	1250,00	0,00	3047,54	685,57
	Sines	600,00	0,00	480,00	0,00	800,00	0,00	600,00	0,00	400,00	0,00	568,72	87,22
	Tarragona	375,00	0,00	300,00	0,00	500,00	0,00	375,00	0,00	250,00	0,00	1007,82	962,07
t_{10}	**Leuna**	1246,10	1989,58	996,88	1591,66	1661,47	2652,77	1246,10	1989,58	830,74	1326,39	1163,03	1856,94
	Neratovice	3461,40	5526,61	2769,12	4421,29	4615,20	7368,81	3461,40	5526,61	2307,60	3684,40	4239,29	5100,75
	Sines	1107,65	1768,51	744,64	0,00	1476,86	2358,02	930,80	0,00	738,43	1179,01	988,15	1194,03
	Tarragona	692,28	1105,32	465,40	0,00	923,04	1473,76	581,75	0,00	461,52	736,88	886,14	281,61

5.3.2.3.1 Environmental results

Carbon trading is also analyzed taking into account the trends forecasted by scenarios. In order to optimize the model, CO_2 emissions were monetized, which is done through the emissions trading equation (M10) or (89). This equation is presented again:

$$Net_{t,s}^{CO_2} = Price_{t,s}^{CO_2} Sales_{t,s}^{CO_2} - Cost_{t,s}^{CO_2} Buy_{t,s}^{CO_2} \quad \forall t, s$$

Thus, net incomes related to trade with CO_2 emissions are equal to the sale of emissions minus the purchase of emissions. Each of these terms are analyzed below. *Figure 33* shows the CO_2 allowance prices of each scenario at each time period. It is also represented the price trend with dashed line. As can be seen, it is expected that price increases with time. The average price changes from 21.69 €/ton in the first year to 29.64 €/ton in the last year. The maximum price used is located in the last time period with a value of 58.55 €/ton and the minimum price used is 6.08 €/ton in scenario 34 in time period 9.

In *Figure 34* the GWP of each scenario and in each time period is represented. The average GWP changes from $7.27 \cdot 10^7$ kg $CO_2 -$ eq in first year to $6.29 \cdot 10^7$ kg $CO_2 -$ eq in last year. That is to say, it is expected an average reduction of $9.85 \cdot 10^6$kg $CO_2 -$ eq, a considerable reduction of GWP (a 36.65% increase of CO_2 prices caused a 13.48% decrease of GWP). Therefore, it is clear that, the increase of CO_2 allowance prices produces a reduction in GWP.

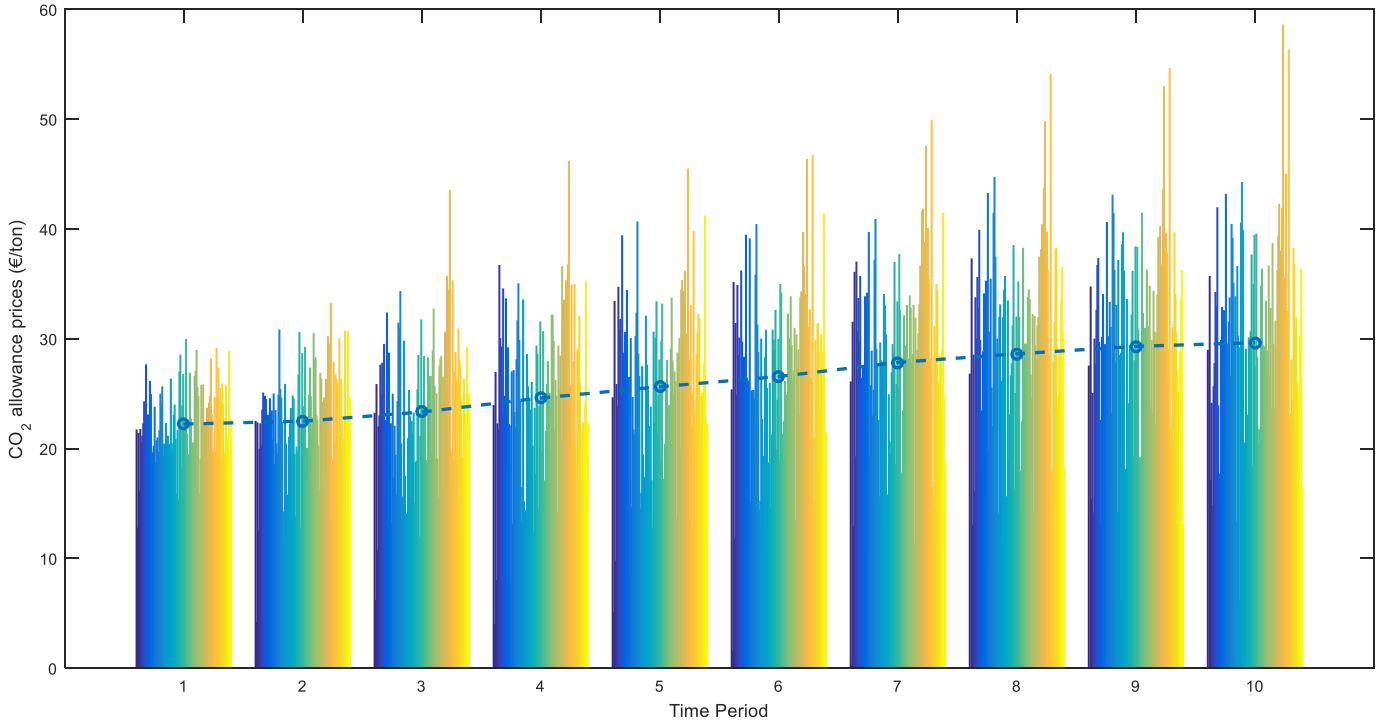


Figure 33. CO_2 allowance prices (€/ton) of each scenario during the different time periods.

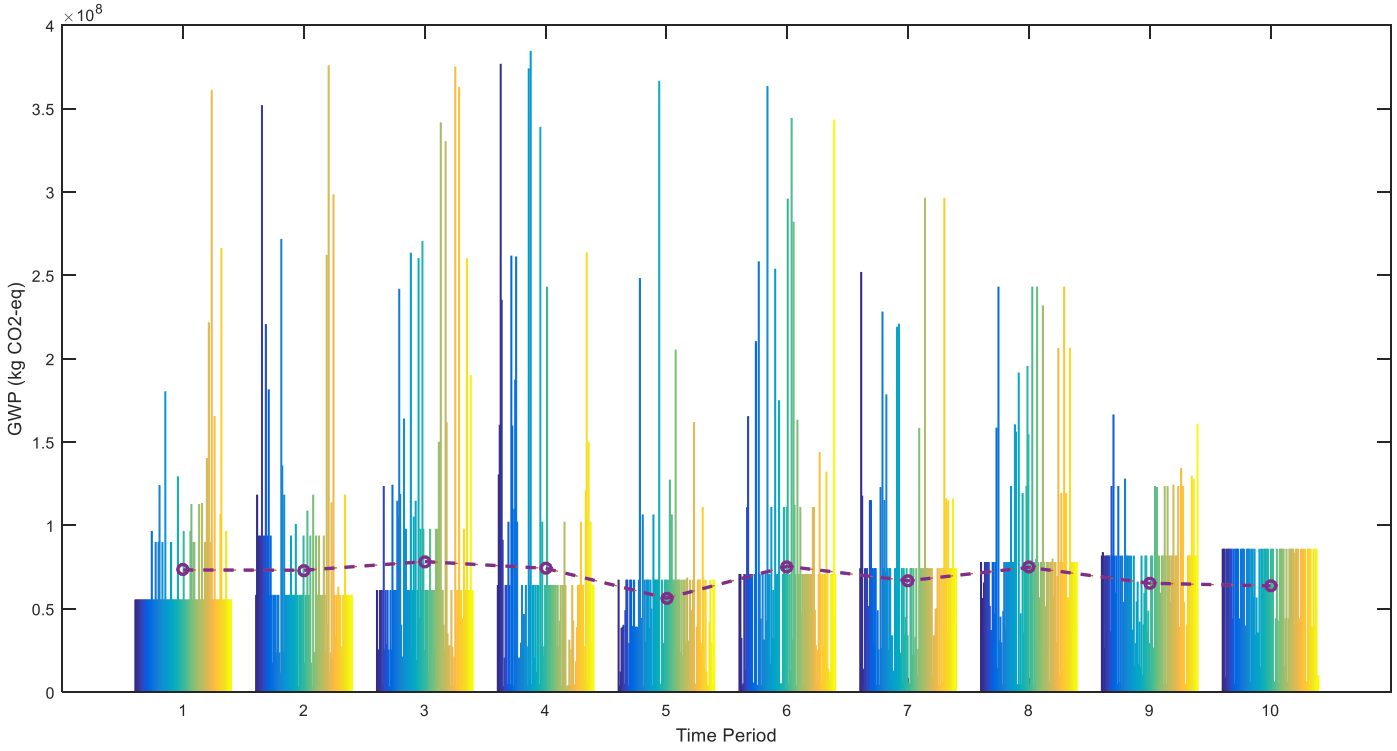


Figure 34. GWP of each scenario in each time period.

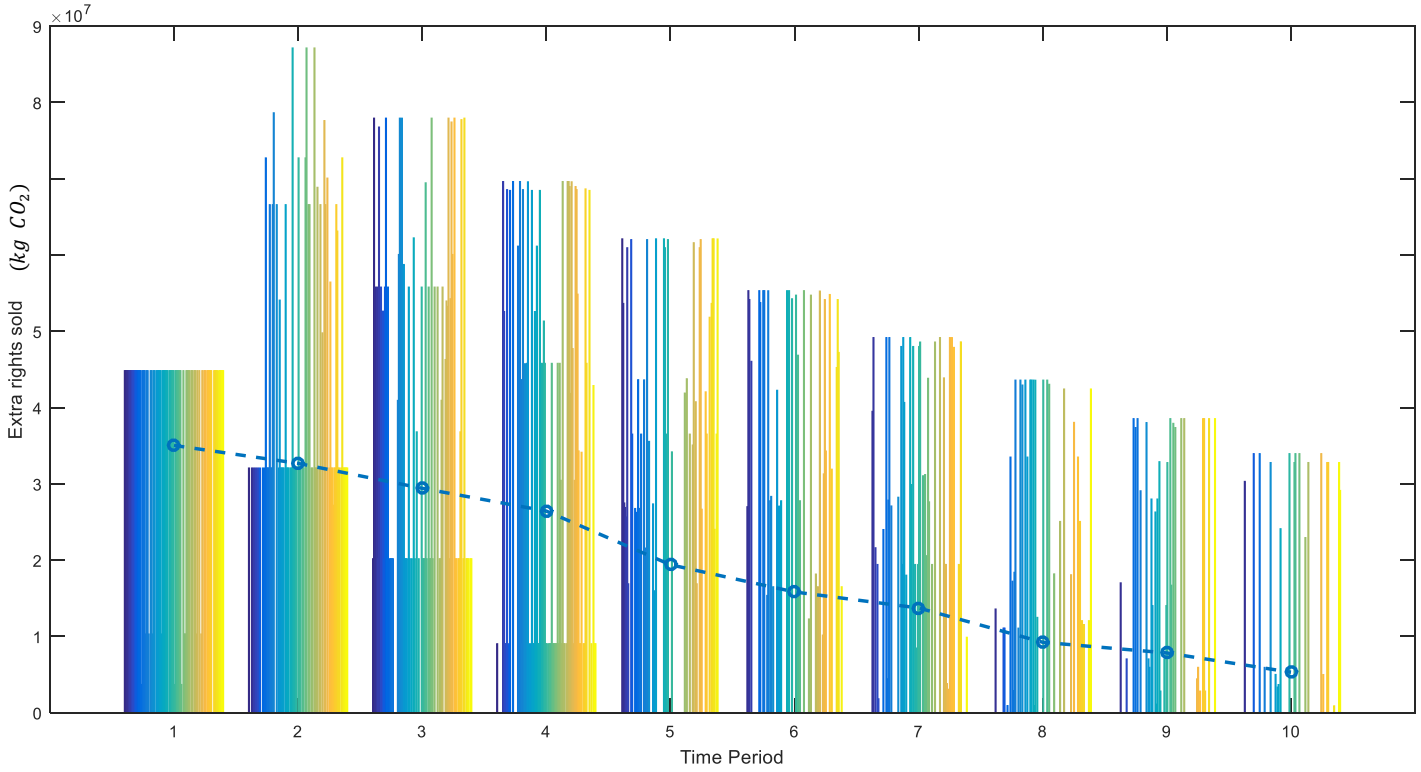


*Figure 35. A*mount of emissions extra rights sold ($kg\ CO_2$) in each period.

As can be seen in *Figure 35*, the amount of emissions extra rights sold is expected to be reduced with time. It is explained mainly because of the fact that it is important not to exceed the emission limit, however, the demand increase year after year, which causes a decrease in sold emissions and an average increase in bought.

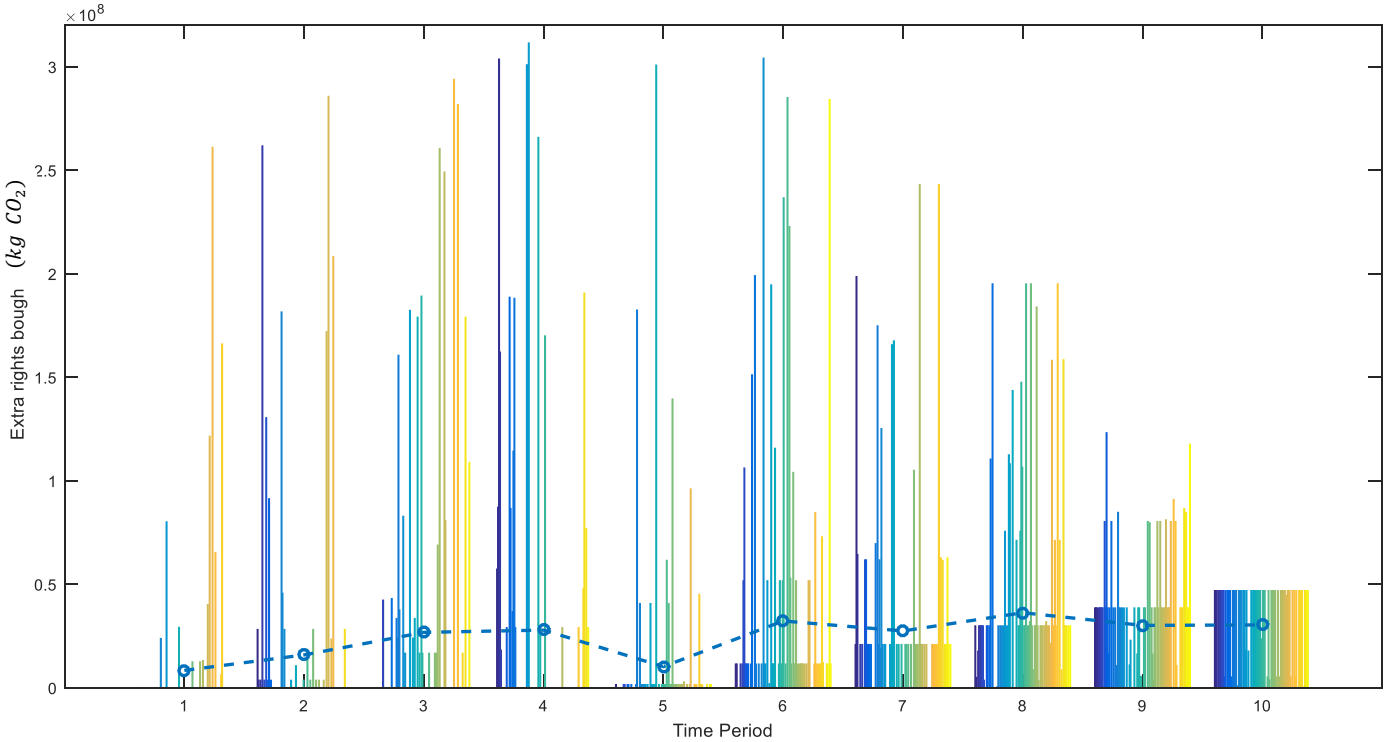


Figure 36. Amount of emissions extra rights bought in each time period.

Moreover, as can be seen in *Figure 36* from time period 1 to 10 the amount of emission right bought can be very small, as well as very great, especially in time periods 1 to 9, while in the last time period (time period 10) the bought amount seems to be very homogeneous in comparison with the other ones. Finally, the average quantity bought increases at the final time period (although it increases and falls along the different time periods). As a result, the extra rights that have to be bought increase. It goes from an average of $8.3 \cdot 10^6$ ton in first year to $5.55 \cdot 10^7$ ton in last year. Finally, the net income due to emissions trading is found to be really important. See *Figure 37.*

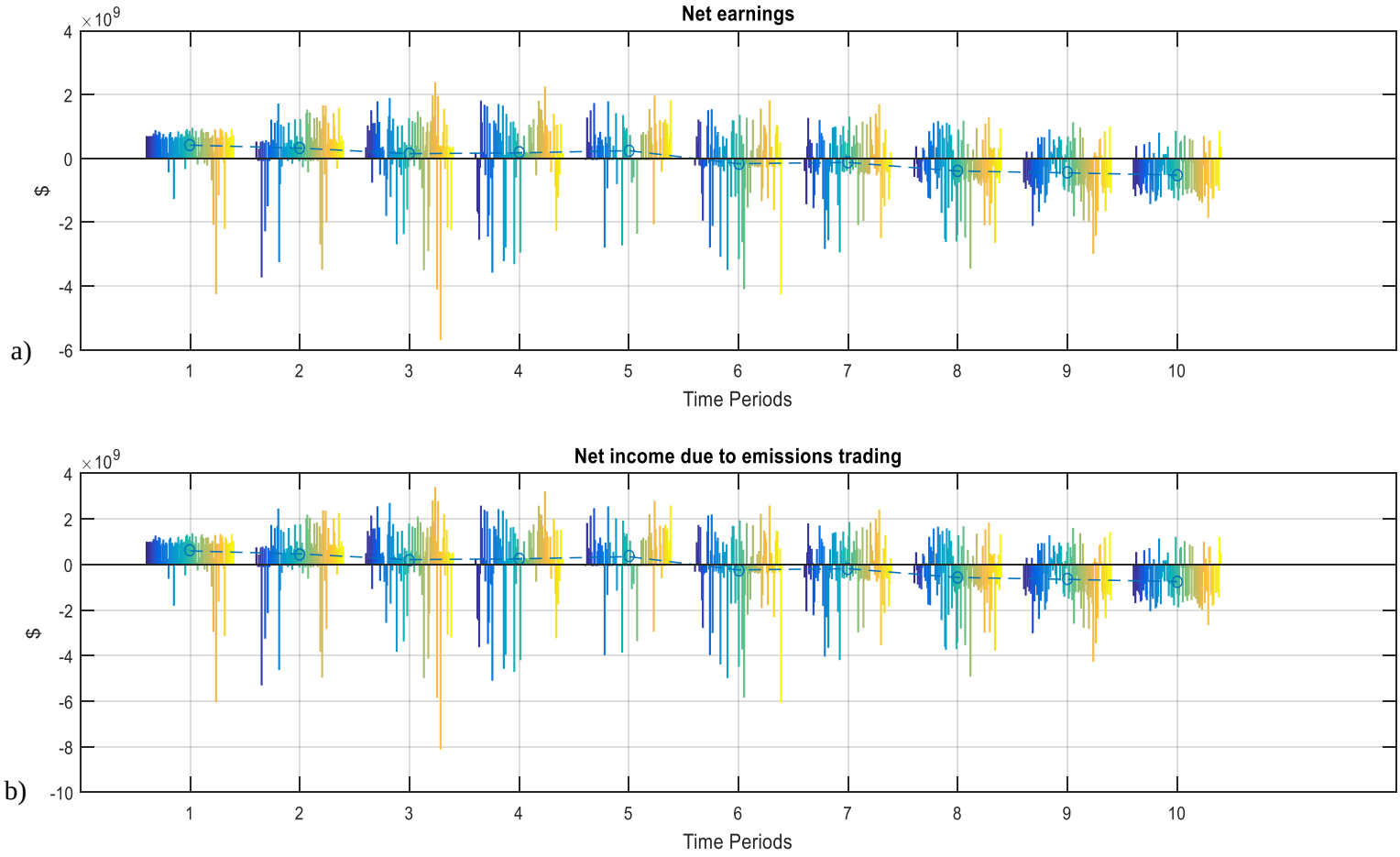


Figure 37. a) Net earnings; b) Net income due to emission trading.

As can be seen, the total net income due to emissions trading follows the same trend that the total net earnings. The values have the same order and direction, which is an indication of the great importance of emission trading in the total net income. Then, it can be concluded that imposing an increase in CO_2 emissions right price will probably cause losses in our supply chain (in most of the scenarios). Demands in markets increase on a par with the emission cost. The demand must be satisfied and prices in products do not increase enough to offset the increase in CO_2 emission prices, what finally cause losses in most of the scenarios. That means that companies would be forced to adapt their technologies to other more sustainable in order not to lost money year after year.

6. CONCLUSIONS

This work has addressed the forecasting of future CO_2 allowance prices and the design and planning of chemical SC under uncertainty, taking into account the CO_2 prices as uncertain parameter. The problem was mathematically formulated as a stochastic MILP that accounts for the maximization of the expected NPV. Results show that the stochastic design improves the deterministic one and therefore, it should be the preferred choice in practice (see *Figure 25*).

ARIMA and artificial neural networks were tried to make a forecast for the CO_2 allowance prices in the along the future 2 years. Based on the results of this work, estimating emission prices with artificial neural networks cannot be drawn, as the model shows no evidence of a good performance. As it has been proven, for a completely random process, where predicting future outcomes is by definition impossible, time series is not a good tool. By simply defining a model, making some predictions and calculating common accuracy metrics, one could seemingly have a good model and decide to put it into production. In reality, the model might have no predictive power whatsoever. On the other hand, the results obtained with ARIMA model demonstrated a satisfactory prediction. Through this work, it was explained the process of building and selecting the appropriate ARIMA model, which finally proved to be an ARIMA(4,1,2). This ARIMA(4,1,2) was the tool used to introduce uncertainty in the design of the supply chain since it has been proven to be appropriated since the price of CO_2 emission rights with ARIMA was suitable.

As it was presented in the results, solving the stochastic model gives as a result a great quantity of possible solutions. Among all these solutions, the decision maker has the opportunity to choose the most appropriate configuration according to their risk criteria. During this work, the best scenarios have been divided into risk-averse and risk-taking scenarios in order to provide a easier way to choose a final decision. Moreover, among the solutions for the operational variables, it is important to emphasize that it has been found that an increase of CO_2 allowance prices involves a reduction in GWP (a 36.65% increase of CO_2 prices caused a 13.48% decrease of GWP), which is in fact the final objective of the risen in emission prices. However, it also has been demonstrated that increase these CO_2 allowance prices will probably cause in long term huge losses. This is due to the greater demand generated along years that must be satisfied, at the same time that products prices remain constant and CO_2 allowance prices increase. The designed supply chain has to produce and transport the products to a greater extent of consumers, while the transporting and producing cost due to CO_2 emission prices increase damage competitiveness of the supply chain. As a result, most of the scenarios predict negative net earnings, which would force to companies to adapt technologies to other more sustainable in order not to lost money year after year.

References

[1] Adebiyi, A. A. (2014). Comparison of ARIMA and artificial neural. Journal of Applied Mathematics Article ID 614342.

[2] Guoqiang Zhang, B. E. (1998). Forecasting with artificial neural networks: The state of the art.

[3] Report on the functioning of the European carbon market. Brussels, (2017): EUROPEAN COMMISSION.

[4] STERN, N. (2006;). Stern review on the economics of climate change. London, UK: HM Treasury.

[5] CO2 EUROPEAN EMISSION ALLOWANCES. (n.d.). Retrieved from https://markets.businessinsider.com/commodities/co2-emissionsrechte

[6] Robert H. Shumway, D. S. (n.d.). Time Series Analysis and Its Applications. Springer Texts in Statistics.

[7] STERN, N. (2006;). Stern review on the economics of climate change. London, UK: HM Treasury.

[8] Luis Gonzaga Baca Ruiz, M. P.-F. (2016). An Application of Non-Linear Autoregressive Neural Networks to Predict Energy Consumption in Public Buildings. Granada: Department of Computer Science and Artificial Intelligence, University of Granada.

[9] Valeri Mladenov, C. J. (September 5-7, 2014). Engineering Applications of Neural Networks . Bulgaria: Lazaros Iliadis.

[10] Erik Molino-Minero-Re, J. G.-M.-F.-A.-C. (2014). Comparison of artificial neural networks and harmonic analysis for sea level forecasting.

[11] GUILLÉN-GOSÁLBEZ, G. and GROSSMANN, I.E. Optimal design and planning of sustainable chemical supply chains under uncertainty. University Rovira i Virgili and Carnegie Mellon University, 2008.

[12] Shapiro, J. F. *Modeling the Supply Chain*. Duxbury, 2001.

[13] AZAPAGIC, A. and CLIFT, R. *Life cycle assessment and multiobjective optimization.*Journal of Cleaner Production. 135-143. 1999.

[14] QI, Y. and CHEN, X. *Improved Life Cycle Assessment of Recycling Organic Wastes for Practice*. Journal of Cleaner Production. August 2017.

[15] Frischknecht, R., Jungbluth, N., Althaus, H.-J., Doka, G., Dones, R., Heck, T., Hellweg, S., Hischier, R., Nemecek, T., Rebitzer, G., Spielmann, M., 2005. The ecoinvent database: overview and methodological framework (7 pp). Int. J. Life Cycle Assess. 10, 3–9.

[16] R. Ruiz-Femenia, G.-G. L. (2013). Multi-objective optimization of environmentally conscious chemical supply chains under demand uncertainty. ELSEVIER.

[17] Acquah, H. d.-G. (2009). Comparison of Akaike information criterion (AIC) and Bayesian information criterion (BIC) in selection of an asymmetric price relationship.

[18] Sokolowski, J.A. (2010) Monte Carlo Simulation. In: Sokolowski, J.A. and Banks, C.M., Eds., *Modelling and Simulation Fundamentals*: *Theoretical Underpinnings and Practical Domains*, Wiley & Sons Inc., New Jersey, 131-145. http://dx.doi.org/10.1002/9780470590621.ch5

[19] Box, G. E. P., G. M. Jenkins, and G. C. Reinsel. Time Series Analysis: Forecasting and Control. 3rd ed. Englewood Cliffs, NJ: Prentice Hall, 1994

[20] Acquah, H. d.-G. (2009). Comparison of Akaike information criterion (AIC) and Bayesian information criterion (BIC) in selection of an asymmetric price relationship.

[21] Bagajewicz, A. B. (May 2004). Managing Financial Risk in Planning under Uncertainty. AIChE Journal, Wiley InterScience, School of Chemical Engineering and Materials Science, University of Oklahoma, Norman, OK 73019.

[22] Sergio Medina-González, C. P.-G. (2017). Using Pareto filters to support risk management in optimization under uncertainty: Application to the strategic planning of chemical supply chains.

[23] Ahmed Aseeri, M. J. (2004). New measures and procedures to manage financial risk with applications to the planning of gas commercialization in Asia. Elsevier, Computers & Chemical Engineering.

7. APPENDIXES

7.1 Complementary theory

7.1.1 Identifying the order of ARIMA model

As it was said before, the first (and most important) step in fitting an ARIMA model is the determination of the order of differencing needed to stationarize the series. Normally, the correct amount of differencing is the lowest order of differencing that yields a time series which fluctuates around a well-defined mean value.

The next step in fitting an ARIMA model is to determine whether AR or MA terms are needed to correct any autocorrelation that remains in the differenced series. It could just be tried some different combinations of terms and see what works best. But there is a more systematic way to do this. By looking at the autocorrelation function (ACF) and partial autocorrelation (PACF) plots explained previously of the differenced series, you can tentatively identify the numbers of AR and/or MA terms that are needed.

Here it is presented an illustrative example (see *Figure 38*) to explain the functioning. *Figure 39* and *Figure 40* shows the autocorrelation function (ACF) of the series, before any differencing is performed.

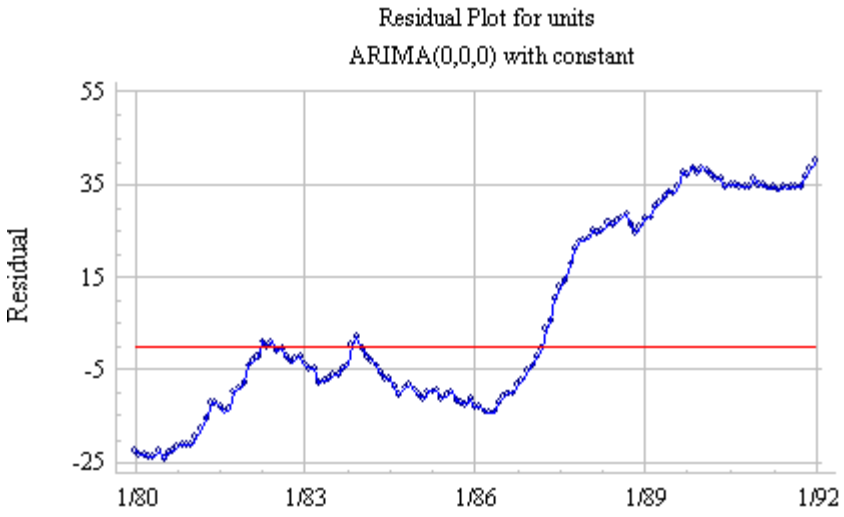


Figure 38. Example of nonseasonal time series.

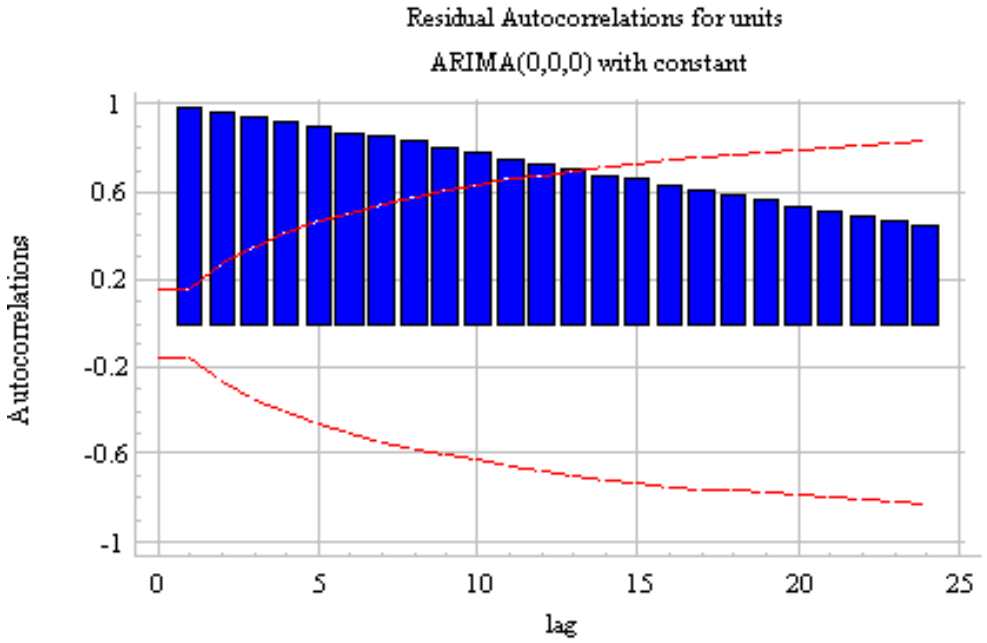


Figure 39. ACF without differencing.

The autocorrelations are significant for a large number of lags (spikes exceed the red line). However, it could be thought that perhaps the autocorrelations at lags 2 and above are merely due to the propagation of the autocorrelation at lag 1. This is confirmed by the PACF plot in *Figure 40*.

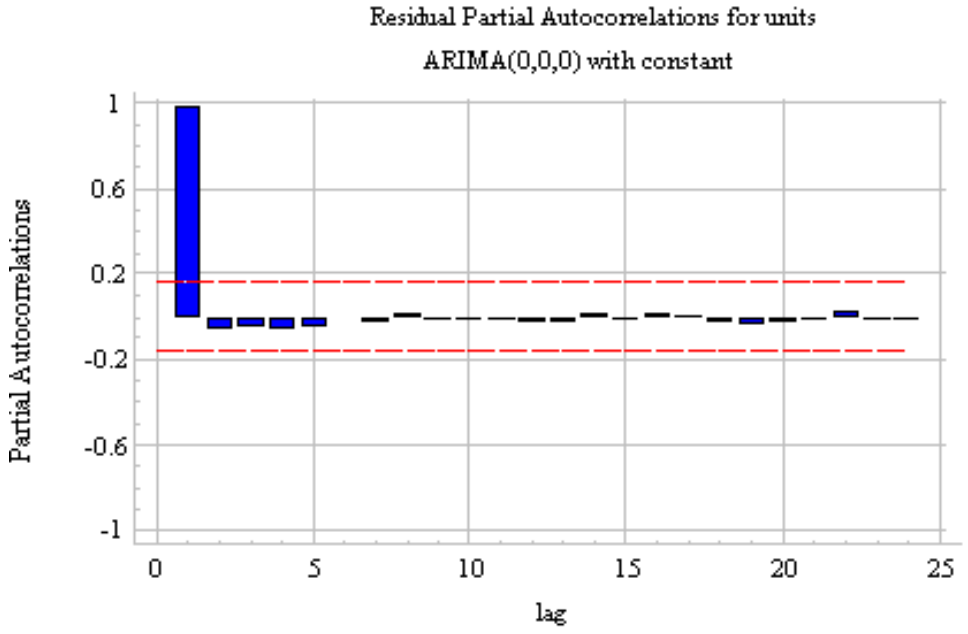


Figure 40. PACF wihout differencing.

Note that the PACF plot has a significant spike only at lag 1, meaning that all the higher-order autocorrelations are effectively explained by the lag-1 autocorrelation. By mere inspection of the

PACF you can determine how many AR terms you need to use to explain the autocorrelation pattern in a time series: if the partial autocorrelation is significant at lag k and not significant at any higher order lags (i.e., if the PACF cuts off at lag k). Then this suggests that you should try fitting an autoregressive model of order k. In this particular case of *Figure 40* there is very large spike at lag 1 and no other significant spikes, indicating that in the absence of differencing an AR(1) model should be used.

AR(1) or ARIMA(1,0,0) is equivalent to equation (5).

$$\hat{Y}_t = \mu + \alpha_1 Y_{t-1}$$

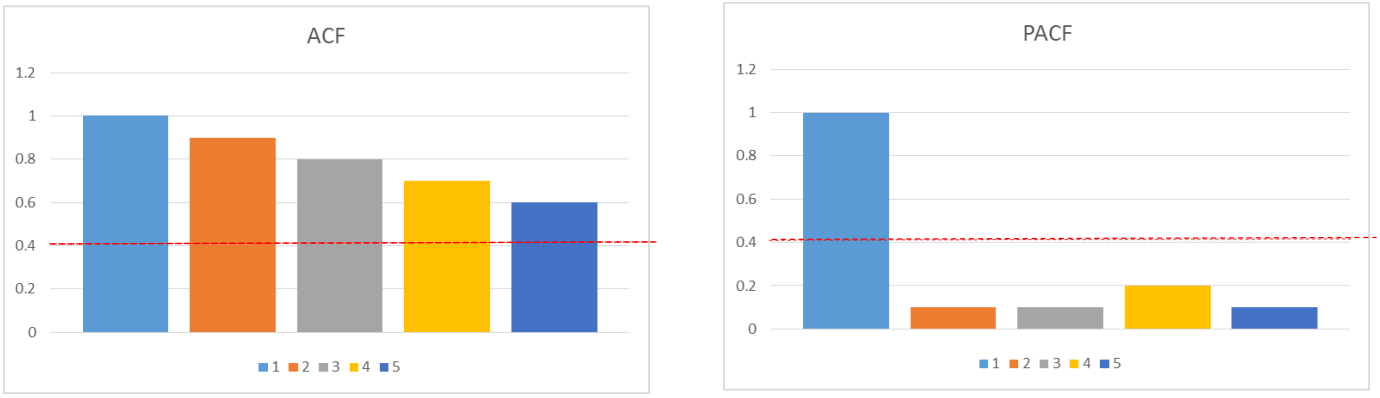
If $\alpha_1 = 1$, then

$$\hat{Y}_t = \mu + Y_{t-1}$$

what is equivalent to predicting that the first difference, an ARIMA(0,1,0) (random walk). See equation (7). That is to say, PACF is showing that this series needs an order of differencing to be stationarized. That is something it should be known with only look at *Figure 40*. It can be seen the differenced data in *Figure 45*.

Once we perform the difference, two cases are possible:

- **Case 1.** If the PACF of the differenced series displays a sharp cutoff and/or the lag-1 autocorrelation is positive (i.e., if the series appears slightly "underdifferenced"), then consider adding an AR term to the model. The lag at which the PACF cuts off is the indicated number of AR terms. *Figure 41* shows a theoretical graphical example.

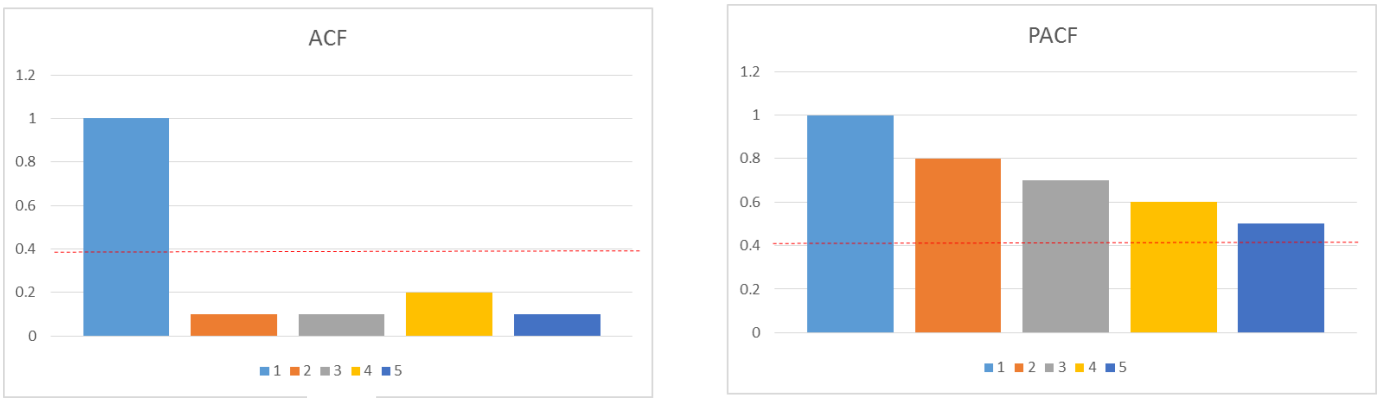


a) b)

Figure 41. Cutoff in PACF (Case 1, adding AR terms): a) ACF; b) PACF.

- **Case 2.** If the ACF of the differenced series displays a sharp cutoff and/or the lag-1 autocorrelation is negative (i.e., if the series appears slightly "overdifferenced"), then

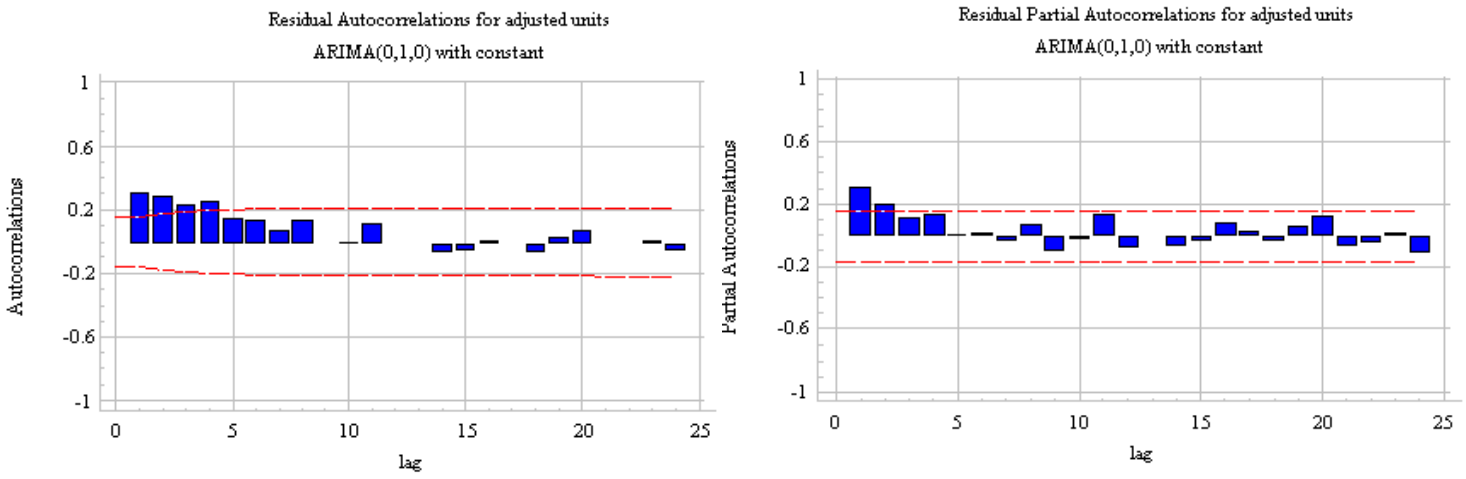
consider adding an MA term to the model. The lag at which the ACF cuts off is the indicated number of MA terms. *2* shows a theoretical graphical example.



a) b)

Figure 42. Cutoff in ACF (Case 2, adding MA terms) a) ACF; b) PACF.

Continuing with the same example of *Figure 41* and *Figure 42*. After taking one nonseasonal difference, or in other words, fitting an ARIMA(0,1,0) model with constant, the ACF and PACF plots look like in *Figure 44* below.



a) b)

Figure 43. Example of: a) ACF after differencing ; b) PACF.after differencing.

Notice that, on the one hand, the correlation at lag 1 is significant and positive, and on the another hand, the PACF shows a sharper "cutoff" than the ACF. In particular, the PACF has only two significant spikes, while the ACF has four. Therefore, *Figure 43* match case 1. According to above

definition, the differenced series displays an AR(2) signature (ARIMA(2,1,0) model). In *Figure 44*, the ACF and PACF of ARIMA(2,1,0) model is shown.

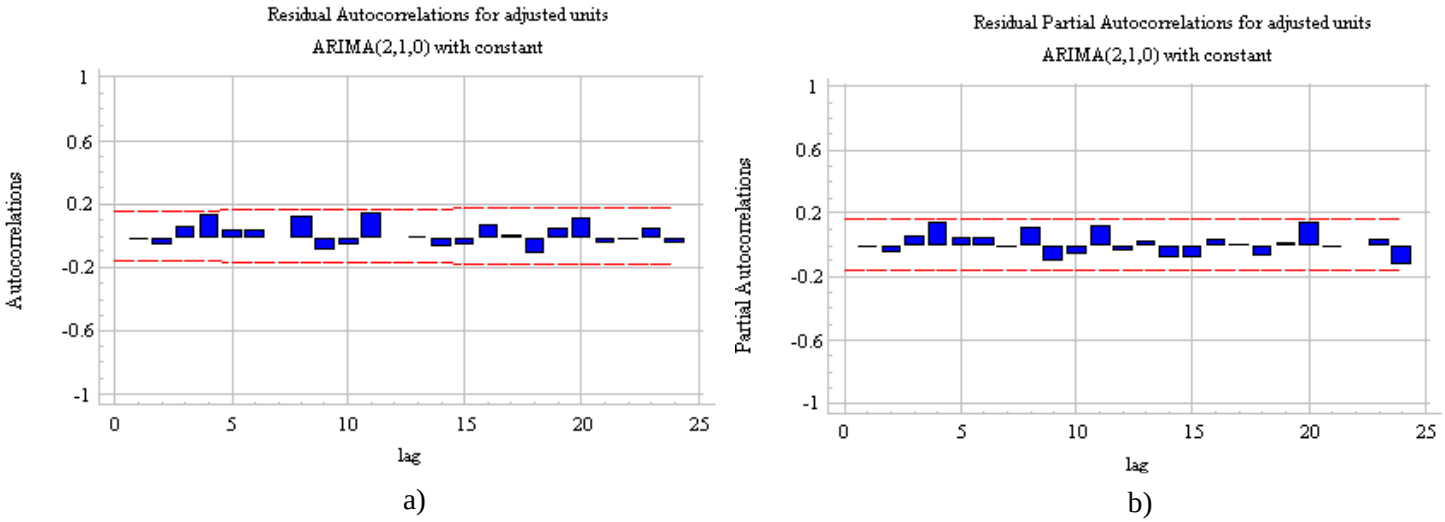


Figure 44. ARIMA(2,1,0). Example of a) ACF after differencing ; b) PACF.after differencing.

The autocorrelation at the crucial lags, lags 1 and 2, has been eliminated, and there is no discernible pattern in higher-order lags. Considering the difference, the final model should fit the time series showed in *Figure 45*.

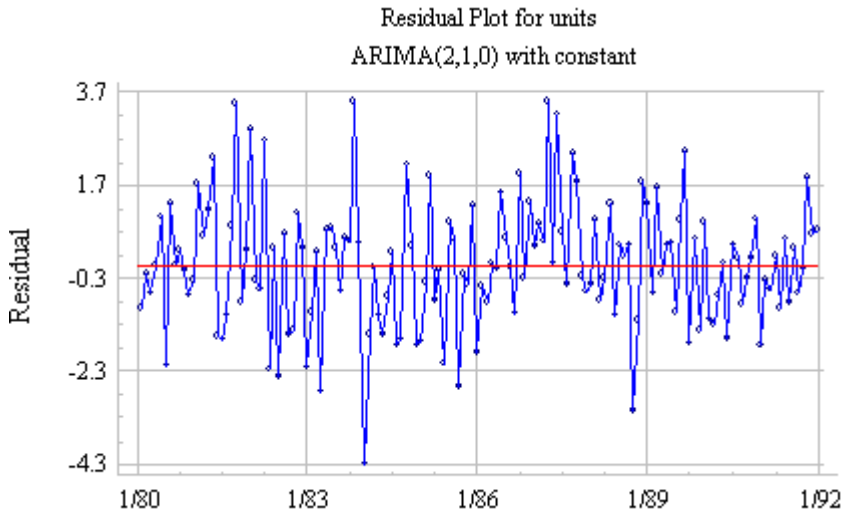


Figure 45. Time series plot after differencing that ARIMA(2,1,0) model could fit.

7.1.2 Additional information of nonlinear autoregressive neural network (NAR)

7.1.2.1 Network training function (NAR)

The most common learning rule for the NAR network is the Levenberg-Marquardt backpropagation procedure (LMBP). This training function is often the fastest backpropagation-type algorithm. The LMBP algorithm was designed to approximate the second-order derivative with no need to compute the Hessian matrix, therefore increasing the training speed. When the performance function has the form of a sum of squares (frequently in feedforward network training), then the Hessian matrix can be approximated as shown in Equation (25) and the gradient can be computed as described in Equation (26).

$$H = J^T J \tag{25}$$

$$g = J^T e \tag{26}$$

In Equations (25) and (26), J is the Jacobian matrix which contains the first derivatives of the network errors with respect to the weights and biases, and e is a vector of network errors in all training samples. To estimate the Jacobian matrix, it is used a standard backpropagation algorithm to approximate the Hessian matrix. This approach is simpler than computing the Hessian matrix [8]. The LMBP algorithm uses this approach in the Newton-like update described in Equation (27).

$$x_{k+1} = x_k - [\, J^T J + \mu I \,]^{-1} J^T e \tag{27}$$

It should be noted that this method uses the Jacobian matrix for calculations, assuming that the performance function is the mean of the sum of the squared errors. Hence, networks must use either the mean square error (MSE) or error sum of squares (SSE). Like it was show in Equation (22).

In general, there are two different architecture options for NAR. Both these architectures include a time delay line.

- Serial architecture: lacks any feedback property.
- Parallel architecture: includes a feedback loop that sends the data form the output directly back to the input.

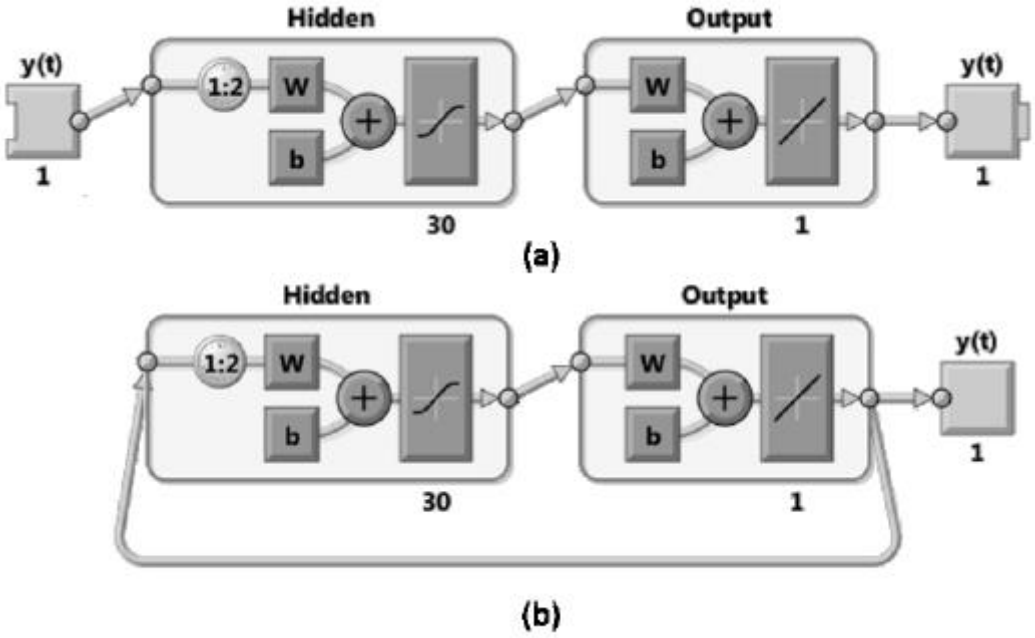


*Figure 46. .Architecture of the open (**a**) and closed loop (**b**) of nonlinear autoregressive neural network given by MATLAB.*

7.1.2.2 Network architecture

The main difference between these two options relates to the training procedure. The accuracy of the training is higher in the second case, because through the serial feed-forward architecture the network is fed only with real data. On the contrary the parallel NAR combines both feedback and real data. This often has a negative effect at the network training accuracy as long as the output data are already processed. Another advantage of the serial model over the parallel one is its simplicity. The serial architecture produces more responsive models that are easier to implement and train faster. This is considered as a very significant feature when the NAR model is meant to be used in real time applications [9]. In general, feedback loop is open for training and closed for predictions.

It is important to adjust the delay parameters of the ANN for, in order to obtain an accurate prediction model. Delay parameters concern the number of days the model is going to use to perform the prediction. In other words, the model is trained with the last p days as delays. To find the best delay, it must be set all parameters to a fixed value and modified the delay in a trial-and-error procedure to find the value that could provide the network with the best performance. Once the delays have been set, the next step is to adjust the number of neurons needed to train the NAR network. In this case, the delay parameters have been set to a fixed value, obtained from the best MSE.

7.1.3 Model selection

7.1.3.1 Akaike information criterion (AIC)

The Akaike information criterion (AIC) is an estimator of the relative quality of statistical models for a given set of data. Given a collection of models for the data, AIC estimates the quality of each model, relative to each of the other models. Thus, AIC provides a means for model selection. When a statistical model is used to represent the process that generated the data, the model will almost never be exact; so some information will be lost by using the model to represent the process. AIC estimates the relative information lost by a given model: the less information a model loses, the higher the quality of that model.

Suppose that we have a statistical model of some data. Let k be the number of estimated parameters in the model and $\hat{L}$ the maximum value of the likelihood function for the model. Then the AIC value of the model is the following:

$$AIC = 2k - 2\ln(\hat{L})$$

(28)

Given a set of candidate models for the data, the preferred model is the one with the minimum AIC value.

7.1.3.2 Bayesian information criterion (BIC)

The formula for the Bayesian information criterion (BIC) is similar to the formula for AIC, but with a different penalty for the number of parameters. As well as AIC, the model with the lowest BIC is preferred. When fitting models, it is possible to increase the likelihood by adding parameters, but doing so may result in overfitting. Both BIC and AIC attempt to resolve this problem by introducing a penalty term for the number of parameters in the model; the penalty term is larger in BIC than in AIC.

$$BIC = \ln(n)\, k - 2\ln(\hat{L})$$

(29)

where,

- $\hat{L}$ is the maximized value of the likelihood function of the model
- n is the number of data points, the number of observations
- k is the number of parameters estimated by the model

7.1.3.3 Likelihood (-2LogLH)

Many statistical models are fit using a technique called maximum likelihood. Rather than maximize the likelihood function, it is more convenient to work with the negative of the natural logarithm of the likelihood function, $-Log\ L(\beta)$. The problem of maximizing $L(\beta)$ is reformulated as a minimization problem where you seek to minimize the negative log-likelihood:

$$-LogLikelihood\ =\ -Log\ L(\beta) \tag{30}$$

Therefore, smaller values of the negative log-likelihood or twice the negative log-likelihood indicate better model fits.

7.1.3.4 Rsquare

RSquare is computed as follows:

$$R^2 = 1 - \frac{SSE}{SST}$$

$$SST = \sum_{I=1}^{N} (y_i - \bar{y}_i)^2$$

$$SSE = \sum_{I=1}^{N} (y_i - \hat{y}_i)^2 \tag{31}$$

where,

- $\bar{y}_i$ are the mean y_i
- $\hat{y}_i$ are the one-step-ahead forecast

If the model does not fit the series well, the model error sum of squares, SSE, might be larger than the total sum of squares, SST. As a result, R^2 can be negative.

7.1.3.5 Weights

In *Table 7* appears a parameter called weight. It is a parameter that is used for model comparison. This fit statistic is the normalized AIC Weight. The AIC Weight for a model is calculated as follows:

$$Weight = \frac{\exp[-0.5(AIC - BestAIC)]}{\sum_{k=1}^{K} \exp[-0.5(AIC_k - BestAIC)]} \tag{32}$$

7.1.3.6 MAPE

The mean absolute percentage error is computed as follows:

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{33}$$

7.1.3.7 MAE

The mean absolute error is computed as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{34}$$

7.1.4 Simulation

7.1.4.1 MMSE Forecast (ARIMA Time Series Models)

The primary objective of ARIMA modeling is to forecast future values. There are two types of forecast:

- In sample (prediction): The expected value of the in-sample, given the estimates of the parameters.
- Out of sample (forecasting): The value of a future that is not observed by the sample

Minimum squared error (MMSE) forecast is a method for obtain the second type of forecast, which considers the following. Given a time series data set $I_T = \{Y_1, Y_2, \dots, Y_T\}$, the forecast for $T + l$ made at time T can be written as:

$$\hat{Y}_{T+l} \text{ or } \hat{Y}_{T+l|T} \text{ or } \hat{Y}_T(l) \tag{35}$$

On the other hand, the forecast error at $T + l$:

$$e_{T+l} = Y_{T+l} - \hat{Y}_{T+l} \tag{36}$$

And the mean squared error (MSE):

$$MSE(e_{T+l}) = E\left[e_{T+l}^2\right] = E\left[\left(Y_{T+l} - \hat{Y}_{T+l}\right)^2\right] \tag{37}$$

To get a point estimate, $\hat{Y}_{T+l}$, it is needed a cost function to judge various alternatives. This cost function is call loss function. Since it is a forecast, it is worked with an expected loss function. A popular loss functions is the MSE, which is quadratic and symmetric. This MSE is wanted to be minimized:

$$\min E[e_{T+l}^2] = E[(Y_{T+l} - \hat{Y}_{T+l})^2] = E\left[Y_{T+l}^2 - 2Y_{T+l}\hat{Y}_{T+l} + \hat{Y}_{T+l}^2\right] \tag{38}$$

Differentitating $\hat{Y}_{T+l}$ to obtain the minimum value:

$$E\left[-2Y_{T+l} + 2\hat{Y}_{T+l}\right] = 0 \rightarrow E\left[2Y_{T+l}\right] = E\left[2\hat{Y}_{T+l}\right]$$

(39)

What implies:

$$E[Y_{T+l}] = \hat{Y}_{T+l}$$

(40)

That is to say, the optimal point forecast under MSE is the (conditional) mean:

$$\hat{Y}_T(l) = E[Y_{T+l}|I_T] = E[Y_{T+l}|Y_1, Y_2, \dots, Y_T]$$

(41)

The ARIMA model for Y_{T+l} is given by:

$$Y_{T+l} = \mu + \alpha_1 Y_{T+l-1} + \cdots + \alpha_p Y_{T+l-p} + \varepsilon_{T+l} - \beta_1 e_{T+l-1} - \cdots - \beta_q e_{T+l-q}$$

(42)

This can be also represented with other notation as:

$$Y_{T+l} = \mu + \psi(B)\varepsilon_T = \mu + \frac{\beta_q(B)}{\alpha_p(B)}\varepsilon_T$$

$$Y_{T+l} = \mu + \varepsilon_{T+l} + \psi_1 \varepsilon_{T+l-1} + \psi_2 \varepsilon_{T+l-2} + \cdots + \psi_l \varepsilon_T + \cdots$$

(43)

Taking the expectation of Y_{T+l}:

$$\hat{Y}_T(l) = E(Y_{T+l}|Y_T, Y_{T-1}, \dots, Y_1) = \psi_l \varepsilon_T + \psi_{l+1}\varepsilon_{T-1} + \cdots$$

(44)

Where

$$E\left(\varepsilon_{T+j}|Y_T, \dots, Y_1\right) = \begin{cases} 0, & j > 0 \\ \varepsilon_{T+j}, & j \leq 0 \end{cases}$$

(45)

Then, the forecast error:

$$e_T(l) = Y_{T+l} - \hat{Y}_T(l) = \varepsilon_{T+l} + \psi_1 \varepsilon_{T+l-1} + \cdots + \psi_{l-1}\varepsilon_{T+1} = \sum_{i=0}^{l-1} \psi_i \, \varepsilon_{T+l-i}$$

(46)

The variance of the forecast error:

$$Var\big(e_T(l)\big) = Var\left(\sum_{i=0}^{l-1} \psi_i\, \varepsilon_{T+l-i}\right) = \sigma^2 \sum_{i=0}^{l-1} {\psi_i}^2$$

(47)

Therefore, one-step ahead ($l = 1$) the forecast is given by:

$$Y_{T+1} = \mu + \varepsilon_{T+1} + \psi_1\varepsilon_T + \psi_2\varepsilon_{T-1} + \cdots$$

$$\hat{Y}_T(1) = \mu + \psi_1\varepsilon_T + \psi_2\varepsilon_{T-1} + \cdots$$

$$\varepsilon_T(1) = Y_{T+1} - \hat{Y}_T(1) = \varepsilon_{T+1}$$

$$Var\big(e_T(1)\big) = \sigma^2$$

(48)

When the time horizon is 2 ($l = 2$):

$$Y_{T+2} = \mu + \varepsilon_{T+2} + \psi_1\varepsilon_{T+1} + \psi_2\varepsilon_T + \cdots$$

$$\hat{Y}_T(2) = \mu + \psi_2\varepsilon_T + \cdots$$

$$\varepsilon_T(2) = Y_{T+2} - \hat{Y}_T(2) = \varepsilon_{T+2} + \psi_1\varepsilon_{T+1}$$

$$Var\big(e_T(2)\big) = \sigma^2(1 + \psi_1^2)$$

(49)

The confidence interval ($100(1-\delta)\%$) l −steps ahead is given by:

$$\hat{Y}_T(l) \pm z_{\delta/2}\sqrt{Var\big(e_T(l)\big)}$$

$$\hat{Y}_T(l) \pm z_{\delta/2}\sqrt{\sigma^2 \sum_{i=0}^{l-1} {\psi_i}^2}$$

(50)

7.1.4.2 Monte Carlo Forecast (ARIMA Time Series Models)

Monte Carlo technique consists of repeating many times a random experiment, through the generation of pseudo-random numbers that later will be used to compute statistics measurements (based on results) that allow adopted conclusions to make decision.

When simulating time series models, one draw (or realization) is an entire sample path of specified length $N, y_1, y_2, \ldots, y_N$. When you generate a large number of draws, say M, you generate M sample paths, each of length N. In other words, it is a way of forecasting future events and estimating the probability of future events.

The method simulates the behaviour of a system by taking repeated sets of random numbers from the underlying probability distribution of the process under investigation

For example, consider simulating N responses from the model ARMA(2,1):

$$\hat{Y}_t = \alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} - \beta_1 e_{t-1} + \epsilon_t$$

(51)

The Monte Carlo simulation takes into account the ARMA(2,1) model and a random variable as a future predictor:

$$u_t = \hat{Y} + X_t \sigma$$

(52)

Generate *N* independent innovations from the Gaussian distribution:

$$\{\hat{\epsilon}_1, \hat{\epsilon}_2, \dots, \hat{\epsilon}_N\}$$

(53)

Filter the innovations recursively to obtain the unconditional disturbances:

$$\hat{Y}_1 = \alpha_1 Y_0 + \alpha_2 Y_{-1} + \hat{\epsilon}_1 + \epsilon_0$$

$$\hat{Y}_2 = \alpha_1 Y_1 + \alpha_2 Y_0 + \hat{\epsilon}_2 + \hat{\epsilon}_1$$

$$\hat{Y}_3 = \alpha_1 Y_2 + \alpha_2 Y_1 + \hat{\epsilon}_3 + \hat{\epsilon}_2$$

$$\vdots$$

$$\hat{Y}_N = \alpha_1 Y_{N-1} + \alpha_2 Y_{N-2} + \hat{\epsilon}_N + \hat{\epsilon}_{N-1}$$

(54)

7.1.5 Resolution method: Sample Average Approximation

The resolution of the MILP model (Mixed Integer Linear Programming) previously described in section 7.2.1.2, is carried out following a decomposition strategy based on the Sample Average Approximation algorithm. This is based on obtaining a deterministic version of the stochastic model when considering a single scenario in a first stage. After the optimization of this first deterministic problem, the values of the first stage variables (structural decisions) are obtained, which will be fixed and the model will be solved again but now considering all the scenarios (see *Figure 47*). This process is repeated iteratively changing the values assigned to the uncertain parameters by those corresponding to the scenario that defines the deterministic problem (step 1). Finally, we will eliminate repeated or sub-optimal solutions (dominated) by applying a filtering based on the elimination of solutions.

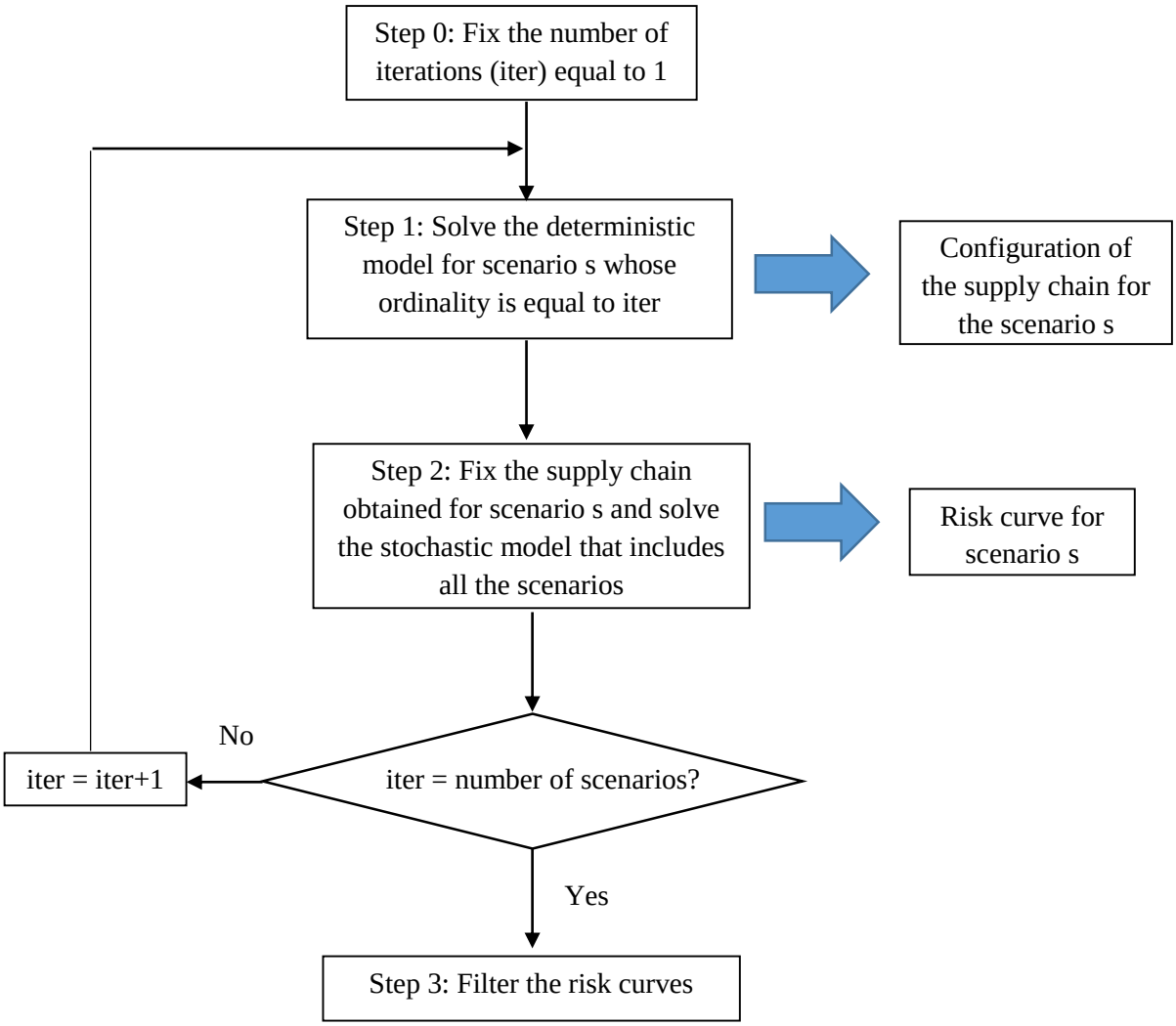


Figure 47. Sample Average Algorithm (SAA).

7.1.6 Financial Risk metrics

There are different methodologies used in financial risk management in the framework of two-stage stochastic programming under uncertainty. These methodologies use a well-known definition of risk based on cumulative probability distributions. This section introduces several definitions of the metrics used in this work.

7.1.6.1 Downside Risk (DRisk)

To present the concept of downside risk, let us first define $\delta(x,\Omega)$ as the positive deviation from a profit target Ω for design x and scenario s, that is

$$\delta(x,\Omega)\begin{cases}\Omega - Profit(x) & \text{If Profit(x)} < \Omega \\ 0 & \text{otherwise}\end{cases} \qquad \forall\, s \in S \tag{55}$$

Because the scenarios are probabilistically independent, the downside risk can be expressed as the following linear function of δ

$$DRisk(x,\Omega) = \sum_{s\in S} p_s \delta_s(x,\Omega) \tag{56}$$

Downside risk graphically represented is given by the blue area given by the cumulative probability distribution as shown in *Figure 48.*

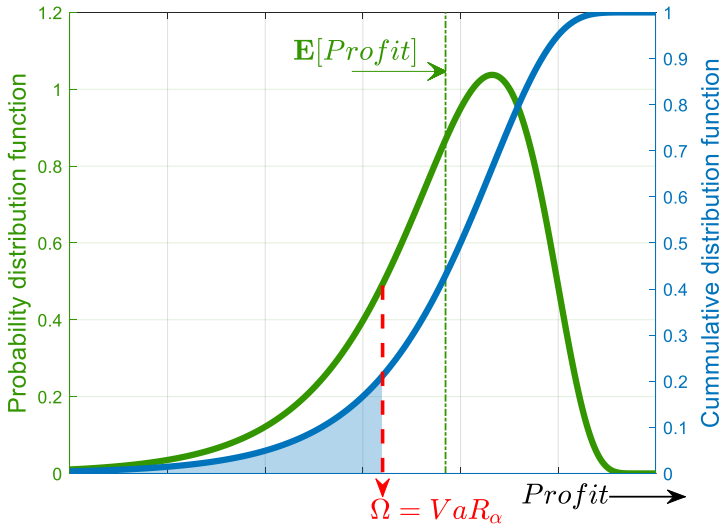


Figure 48. Expected downside risk representation.

7.1.6.2 Value at Risk (VaR)

It is defined as the profit value corresponding to the α −quantile

$$VaR(x,\alpha) = Profit_{s_\alpha}(x) \quad \text{where} \quad s_\alpha = \{s \mid \textstyle\sum_{k=1}^{s} p_k = \alpha\} \tag{57}$$

That is to say, it is computed by sorting the scenarios in ascending profit order and taking the profit value of the scenario for which the cumulative probability equals the specified confidence level. *Figure 49* shows a graphical representation of VaR.

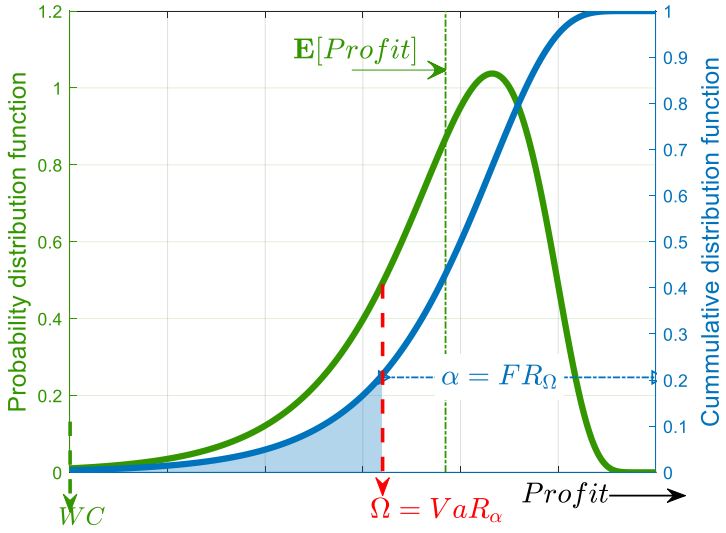


Figure 49. Value at Risk representation.

7.1.6.3 Opportunity value (OV)

The opportunity value is defined in a similar way to VaR but at the other end of the risk curve with a quantile of (1−p). That is translated into the profit obtained in scenario s when the cumulative probability curve is at 0.95 (typical value).

7.1.6.4 Worst Case (WC)

It is also used the Worst Case metric to control the occurrence of unfavorable scenarios. Its formulation is very simple and requires few computational resources:

$$WC(x) \leq Profit_w(x) \;\; \forall w \in \Omega \tag{58}$$

In other words, worst case is represented by first point of each scenario s.

7.1.6.5 Value at Risk difference (VaRD) and Opportunity value difference (OVD)

It is also interesting to compute the difference between the VaR and OV defined previously with respect the net expected value. In the case of OVD, it is calculated as the difference between the net present value corresponding to a risk of (1−p) and the expected value.

$$OVD(x,\alpha) = OV(x,\alpha) - E[NPV(x)] \tag{59}$$

This modification is exactly the same for VaR:

$$VaRD(x,\alpha) = E[NPV(x)] - VaR(x,\alpha) \tag{60}$$

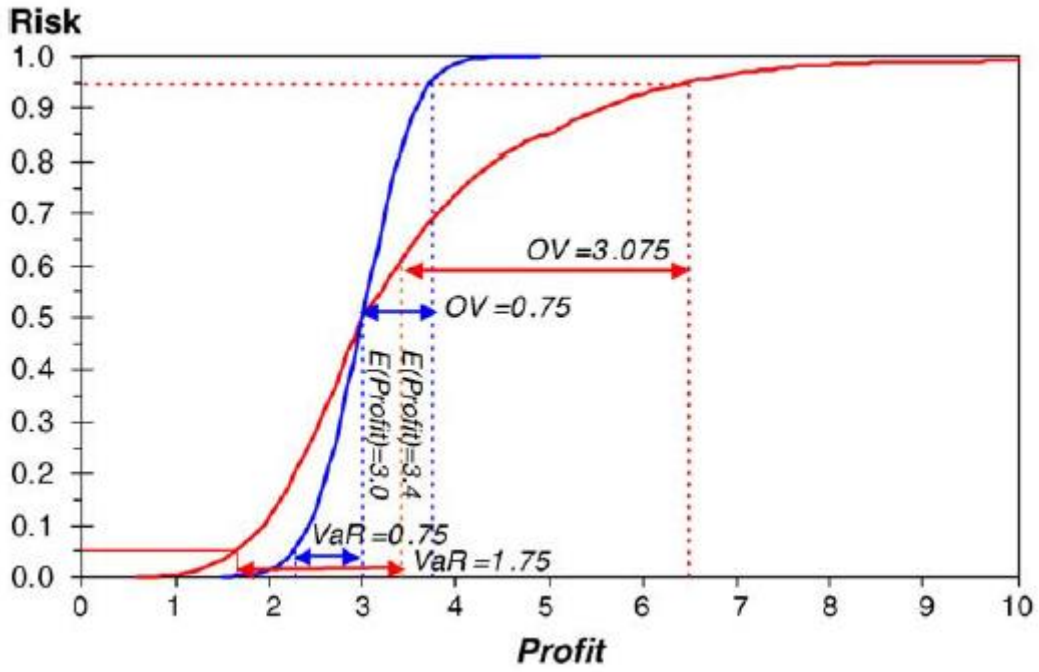


Figure 50. OVD and VaRD representation.

7.2 Additional optimization model information

7.2.1 Optimization model reformulation

7.2.1.1 Notation

Sets

i	Process technologies
j	Production Plants
k	Warehouses
l	Markets
p	Chemical products
s	Scenarios
t	Time periods

Subsets

IN(p)	Set of process technologies that consume p
$MP(i)$	Set of main products p of technology i
OUT(p)	Set of process technologies that produce p

Parameters

$\overline{CE_{l,j,t}^{PL}}$	Upper bound on the capacity expansion of manufacturing technology i at plant j in time period t
$\underline{CE_{l,j,t}^{PL}}$	Lower bound on the capacity expansion of manufacturing technology i at plant j in time period t
$\overline{CE_{k,t}^{WH}}$	Upper bound on the capacity expansion of warehouse k in time t
$\underline{CE_{k,t}^{WH}}$	Lower bound on the capacity expansion of warehouse k in time t
$\overline{D_{l,p,t}^{MK}}$	Upper bound on the demand of product p sold at market l of product p in time period t
$\underline{D_{l,p,t}^{MK}}$	Lower bound on the demand of product p sold at market l of product p in time period t
IMP^{EN}	Cumulative LCIA result for the GWP indicator associated with the consumption of 1 MJ of energy in Kg CO2-eq / MJ
IMP_P^{RM}	Cumulative LCIA result for the GWP indicator associated with the consumption of 1 kg of raw material p in Kg CO2-eq / Kg

IMP^{TR}	Accumulated LCIA result for the GWP indicator associated with the transport of 1 km of a ton of product p in Kg CO2-eq / Tn Km
ir	Interest rate
$\overline{FCI}$	Upper bound of total capital invested
$nexp_{i,j}^{PL}$	Maximum number of expansions of the capacity of the technology i available in the plant j
$nexp_{k}^{WH}$	Maximum number of capacity expansions of the warehouse k
NT	Number of time periods
$\overline{PU_{j,p,t}}$	upper bound on the purchases of product p at plant j in period t
$\underline{PU_{j,p,t}}$	lower bound on the purchases of product p at plant j in period t
$\overline{Q_{j,k,p,t}^{PL}}$	Upper bound on product p transported from plant j to warehouse k in period t and scenario s
$\underline{Q_{j,k,p,t}^{PL}}$	Lower bound on product p transported from plant j to warehouse k in period t and scenario s
$\overline{Q_{k,l,p,t}^{WH}}$	Upper bound on product p transported from warehouse k to market l in period t and scenario s
$\underline{Q_{k,l,p,t}^{WH}}$	Upper bound on product p transported from warehouse k to market l in period t and scenario s
sv	Salvage value fraction of the network
tor_{k}	turnover ratio of warehouse k
$\mu_{i,p}$	Coefficient of mass balance associated with technology i and product p
φ	Tax rate
$\gamma_{l,p,t}^{FP}$	Price of final product p sold at market l in time period t
$\gamma_{j,p,t}^{RM}$	Price of raw material p purchased at plant j in time period t
$v_{i,j,p,t}$	Production cost per unit of main product p manufactured with technology i at plant j in period t
$\pi_{p,k,t}$	Inventory cost per unit of product stored p in warehouse k during period t
$\omega_{j,k,p,t}^{PL}$	Unit cost of transportation of product p from plant j to warehouse k in period t
$\omega_{k,l,p,t}^{WH}$	Unit cost of transportation of product p from warehouse k to market l in period t
$\alpha_{i,j,t}^{PL}$	Variable investment term associated with technology i and plant j in period t in $ year/ton
$\beta_{i,j,t}^{PL}$	Fixed investment term associated with technology i and plant j in period t in k$
$\alpha_{k,t}^{WH}$	Variable investment term associated with warehouse k in period t

$\beta_{k,t}^{WH}$	Fixed investment term associated with warehouse k in period t
$\beta_{j,k,t}^{TPL}$	Fixed investment term associated with the establishment of a transport connection between plant j and warehouse k in period t
$\beta_{k,l,t}^{TWH}$	Fixed investment term associated with the establishment of a transport connection between warehouse k and market l in period t
$\eta_{i,p}^{EN}$	Energy consumption per unit of manufactured product p with technology i in TFOE / Tn
$\lambda_{j,k}^{PL}$	Distance between plant j and warehouse k in Km
$\lambda_{k,l}^{WH}$	Distance between the warehouse k and the market l in Km
τ	The minimum desired percentage of the available installed capacity that should be used

Variables

$Buy_{t,s}^{CO_2}$	Emission allowance purchased [$]
$C_{i,j,t}^{PL}$	Capacity of manufacturing technology i at plant j in time period t [tons]
$CE_{i,j,t}^{PL}$	Expansion capacity of technology i in plant j in period t [tons]
$C_{k,t}^{WH}$	Capacity of warehouse k in time period t [tons]
$CE_{k,t}^{WH}$	Capacity expansion of manufacturing technology i at plant j in time period t [tons]
$CF_{t,s}$	Cash flow in period t in scenario s [$]
$Cost_{t,s}^{CO_2}$	CO_2 emission allowance cost [$]
$DEP_{t,s}$	Depreciation term in period t [$]
FCI	Total fixed capital investment [$]
$FTDC_{t,s}$	Fraction of the total depreciable capital tha tmust be paid in period t [$]
$E[NPV]$	Expected value of the distribution of net present values
GWP_s^{EN}	Contribution to the total GWP due to the energy consumed by the utilities in scenario s [kg CO2-Eq]
GWP_s^{total}	Global Warming Potential for the whole horizon time in scenario s.

GWP^{TR}	Contribution to the total GWP due to the transportation of the materials between the nodes of the SC
$GWP_{t,s}$	Total CO_2 emissions that take place in the supply chain in period t and scenario s
$IL_{k,t}$	Average inventory level at warehouse k in time period t [tons]
$INV_{k,p,t,s}$	Inventory of product p kept at warehouse k in period t in scenario s [tons]
$Max_t^{CO_2}$	Limit of Free- CO_2 emissions in period t
$Ne_{t,s}$	Net earnings in period t and scenario s (profit after taxes) [$]
$Net_{t,s}^{CO_2}$	Net income obtained by trading emissions [$]
NPV_s	Net present value in scenario s [$]
$Price_{t,s}^{CO_2}$	CO_2 emission rights price [$]
$prob_s$	Probability of scenario s occurrence
$PU_{j,p,t,s}$	Purchases of product p made by plant j in period t in scenario s [tons]
$Q_{j,k,p,t,s}^{PL}$	Flow of product p sent from plant j to warehouse k in period t in scenario s [tons]
$Q_{k,l,p,t,s}^{WH}$	Flow of product p sent from warehouse k to market l in period t in scenario s [tons]
$SA_{l,p,t,s}$	Sales of product p at market l in time period t in scenario s [tons]
$W_{i,j,p,t,s}$	Input/output flow of product p associated with technology i at plant j in period t in scenario s [tons]
$X_{i,j,t}^{PL}$	binary variable(1ifthecapacityofmanufacturing technology i at plant j is expanded in time period t, 0 otherwise) [dimensionless]
$X_{k,t}^{WH}$	Binary variable (1 if the capacity of warehouse k is expanded in time period t , 0 otherwise) [dimensionless]
$Y_{j,k,t}^{PL}$	Binary variable (1 if a transportation link between plant j and warehouse k is established in time period t, 0 otherwise) [dimensionless]

7.2.1.2 Complete optimization model

The general structure of the mathematical model is based on that presented by Guillén-Gosálbez and Grossmann [11]. The model has been formulated as a linear mixed integer program (MILP) taking into account all the characteristics of the supply chain explained previously.

The model can be divided into two blocks, the first block would include the structural variables (they are not affected by uncertainty), and they are related to strategic or structural decisions. The

second block consist of the operational variables (they can be affected by uncertainty). These ones are related to planning decisions.

The model equations are classified into three main blocks: mass balance equations, capacity constraints and objective function equations.

7.2.1.2.1 Mass balance constraints

The mass balance must be satisfied for each node embedded in the network. Thus, for each plant j and chemical p the purchases (PU_{jpts}) made during period t plus the amount produced must equal the amount transported from the plant to the ware houses (QPL^{PL}_{jkpts}) plus the amount consumed in every scenario s:

$$PU_{j,p,t,s} + \sum_{i \in OUT(p)} W_{i,j,p,t,s} = \sum_{k} Q^{PL}_{j,k,p,t,s} + \sum_{i \in OIN(p)} W_{i,j,p,t,s} \; \forall j, p, t, s \tag{61}$$

In equation (61), $PU_{j,p,t,s}$ represents the quantity of product p purchased by plant j in time period t and scenario s, $W_{i,j,p,t,s}$ denotes the input / output flow of p associated with the technology i in the plant j in the time period t and scenario s; and finally, $Q^{PL}_{j,k,p,t,s}$ is the amount of p transported between plant j and warehouse k in the period of time t and scenario s. As for the purchases of products, it can be said that they can be raw materials or final products; and for each product, the total purchases are restricted between a minimum value ($\underline{PU_{j,p,t}}$) and a maximum value ($\overline{PU_{j,p,t}}$), which are given by their availability in the market:

$$\underline{PU_{j,p,t}} \leq PU_{j,p,t,s} \leq \overline{PU_{j,p,t}} \quad \forall j, p, t, s \tag{62}$$

In addition, a material balance must be conformed for each technology i exploited in plant j, which is represented by equation (63).

$$W_{i,j,p,t,s} = \mu_{i,p} W_{i,j,p',t,s} \; \forall j, p, t, s \; \forall p' \in MP(i) \tag{63}$$

Where the material balance coefficient $\mu_{i,}$ for the technology i and the product p is indicated, while MP(i) is the set of products that corresponds to each technology.

Eq. (64) represents the mass balance for the warehouses.

$$INV_{k,p,t-1,s} + \sum_{j} Q^{PL}_{j,k,p,t,s} = \sum_{l} Q^{WH}_{k,l,p,t,s} \; \forall \, \mathrm{k, p, t} > 1, \mathrm{s} \tag{64}$$

Where the initial inventory $INV_{k,p,t-1,s}$, s plus the amount of product transported from the plants j to the warehouse k should be equal to the material flow transported from the warehouses to the markets ($Q^{WH}_{k,l,p,t,s}$) plus the final inventory at time period t.

Products sales at the markets ($SA_{l,p,t,s}$) are determines the sales of products in each market as can be seen in equation (65).

$$SA_{l,p,t,s} = \sum_{k} Q^{WH}_{k,l,p,t,s} \; \forall p, l, t, s$$

(65)

Finally, the constraint in equation (66) forces total sales of product p on the market l in the period of time t to be greater than the objective of minimum demand target level $\left(\underline{D_{l,p,t,s}}\right)$ and lower than the maximum demand $\left(\overline{D_{l,p,t,s}}\right)$.

$$\underline{D_{l,p,t,s}} \leq SA_{l,p,t,s} \leq \overline{D_{l,p,t,s}} \;\; \forall p, l, t, s$$

(66)

7.2.1.2.2 Capacity constraints

The production rate of each technology i in plant j, for each time period t and scenario s, must be lower than the existing capacity, $C^{PL}_{i,j,t}$, and higher than a minimum desired percentage, t, of this existing capacity.

$$\tau C^{PL}_{i,j,t} \leq W_{i,j,p,t,s} \leq C^{PL}_{i,j,t} \;\; \forall \mathrm{i, j, t, s} \;\; \forall \mathrm{p} \in \mathrm{MP_i}$$

(67)

The capacity of plant j in time period t is calculated from the existing capacity at the end of the previous period plus the expansion in capacity carried out in t, $CE^{PL}_{i,j,t}$:

$$C^{PL}_{i,j,t} = C^{PL}_{i,j,t-1} + CE^{PL}_{i,j,t} \;\; \forall \mathrm{i, j, t}$$

(68)

The capacity expansions are constrained within lower and upper bounds, which are denoted by:

$$\underline{CE^{PL}_{i,j,t}} X^{PL}_{i,j,t} \geq CE^{PL}_{i,j,t} \leq \overline{CE^{PL}_{i,j,t}} X^{PL}_{i,j,t} \; \forall \mathrm{i, j, t}$$

(69)

In equation. (69), binary variable $X^{PL}_{i,j,t}$ indicates the occurrence of the capacity expansion. This variable takes the value of 1 if technology i at plant j is expanded in capacity in time period t, and 0 otherwise.

Similarly, as with the plants, we define a continuous variable to represent the capacity of the warehouses, $C_{k,t}^{WH}$. Equation (70) forces the total inventory kept at warehouse k at the end of time period t in each scenario s ($INV_{k,p,t,s}$) to be less than or equal to the available capacity:

$$\sum_{p} INV_{k,p,t,s} \leq C_{k,t}^{WH} \quad \forall k,t,s$$

(70)

In order to cope with fluctuations in demand, the storage capacity $\left(C_{k,t}^{WH}\right)$ must be twice the summation of the average storage inventory level kept at the ware house in each scenario s $\left(IL_{k,t,s}\right)$:

$$2IL_{k,t,s} \leq C_{k,t}^{WH} \; \forall k,t$$

(71)

The value of $IL_{k,t,s}$ is calculated from the output flow of materials and the turnover ratio of the warehouse (tor_k), which represents the number of times that the stock is completely replaced per time period:

$$IL_{k,t,s} = \frac{\sum_{l} \sum_{p} Q_{k,l,p,t,s}^{WH}}{tor_k} \quad \forall k,t,s$$

(72)

The capacity of the warehouse at any time period is determined from the previous one and the expansion in capacity executed in the same period:

$$C_{k,t}^{WH} = C_{k,t-1}^{WH} + CE_{k,t}^{WH} \; \forall k,t$$

(73)

The capacity expansion is also bounded within lower and upper bounds.

$$\underline{CE_{k,t}^{WH}} X_{k,t}^{WH} \leq CE_{k,t}^{WH} \leq \overline{CE_{k,t}^{WH}} X_{k,t}^{WH} \quad \forall k,t$$

(74)

This constraint makes use of the binary variable $X_{k,t}^{WH}$, which equals 1 if an expansion in the capacity of warehouse k takes place in time period t and 0 otherwise.

The existence of a transportation link between plant j and warehouse k in period t and scenario s is represented by the binary variable $Y_{j,k,t}^{PL}$. A zero value of this variable prevents the flow of materials form taking place, whereas a value of 1 allows it within some lower and upper bounds:

$$\underline{Q_{j,k,t,s}^{PL}} Y_{j,k,t}^{PL} \leq \sum_{p} Q_{j,k,p,t,s}^{PL} \leq \overline{Q_{j,k,p,t,s}^{PL}} Y_{j,k,t}^{PL} \quad \forall j,k,t,s$$

(75)

7.2.1.2.3 Objective functions

The designed model has been built to optimize the supply chain described above, in which the objective is maximize the economic performance, measured through the economic indicator NPV (Net Present Value). That is to say, the objective of the mathematical model is maximize the average value of the resulting NPV distribution. Moreover, the GWP 100 (Global Warming Potential) indicator, based on the principles of Life Cycle Assessment (LCA), is implement to measure the environmental performance. The calculation of each of these metrics is described in detail below.

1. Expected NPV

At the end of the time horizon, different NPV values are obtained for each scenario, NPVs, once the demand uncertainty is resolved. The expected value of the resulting distribution is determined from these values as follows:

$$E[NPV] = \sum_{s} prob_s NPV_s \tag{76}$$

where $prob_s$ is the probability of scenario s.

The NPVs is calculated as the summation of the discounted cash flows (CFts) generated in each of the time periods t in which the time horizon is divided:

$$NPV_s = \sum_{t} \frac{CF_{t,s}}{(1+ir)^{t-1}} \quad \forall s \tag{77}$$

In this equation, ir represents the interest rate. The cash flow in each time period is determined from the net earnings (i.e., profit after taxes), and the fraction of the total depreciable capital ($FTDC_t$) that corresponds to the period:

$$\mathrm{CF_{t,s}} = NE_{t,s} - FTDC_t \quad t = 1, \dots, NT-1, \forall \mathrm{s} \tag{78}$$

In Eq. (79), we consider that in the cash flow of the last time period ($t = NT$), part of the total fixed capital investment (FCI) will be recovered. This amount, which represents the salvage value of the network, may vary from one type of industry to another.

$$CF_{t,s} = NE_{t,s} - FTDC_t + svFCI \quad t = NT, \forall s \tag{79}$$

where sv is the salvage value fraction of the network.

The net earnings are obtained by subtracting costs and taxes from total incomes. Taxes accrued in period t are determined from the tax rate (φ) and gross profit (i.e., difference between incomes, total cost and depreciation, DEP_t):

$$NE_{t,s} = incomes - costs - \varphi\big(incomes - (costs + DEP_t)\big) \;\; \forall t, s$$

(80)

The total cost accounts for purchases of raw materials, operating cost, inventory costs, and transportation cost, as shown in equation (81):

$$NE_{t,s} = (1 - \varphi)\left[\sum_l \sum_p \gamma^{FP}_{l,p,t} SA_{l,p,t,s} - \sum_j \sum_p \gamma^{RM}_{j,p,t} PU_{j,p,t,s} \sum_i \sum_j \sum_{p \in MP(i)} v_{i,j,p,t} W_{i,j,p,t,s} - \sum_k \pi_{k,t} IL_{k,t,s} + \sum_j \sum_k \sum_p \psi_{j,k,p,t} \lambda^{PL}_{j,k} Q^{PL}_{j,k,p,t,s} - \sum_k \sum_l \sum_p \psi_{k,l,p,t} \lambda^{WH}_{k,l} Q^{WH}_{k,l,p,t,s}\right] + \varphi DEP_t$$

$$\forall t, s$$

(81)

In this equation, $\gamma^{FP}_{l,p,t}$ and $\gamma^{RM}_{j,p,t}$ denote the prices of final products and raw materials, respectively. Furthermore, $v_{i,j,p,t}$ denotes the production cost per unit of main product p manufactured with technology i at plant j in period t, $\pi_{k,t}$ represents the inventory cost per unit of product stored in warehouse k during period t, and $\psi_{j,k,p,t}$ and $\psi_{k,l,p,t}$ are the unitary transports cost. The depreciation term is calculated with the straight-line method.

$$DEP_t = \frac{(1 - sv)FCI}{NT} \quad \forall t$$

(82)

where the total fixed cost investment (FCI) is determined from the capacity expansions made in plants and warehouses as well as the establishment of transportation links during the entire time horizon as follows:

$$FCI = \sum_i \sum_j \sum_k \alpha^{PL}_{i,j,t} CE^{PL}_{i,j,t} + \beta^{PL}_{i,j,t} X^{PL}_{i,j,t} + \sum_k \sum_t \alpha^{WH}_{k,t} CE^{WH}_{k,t} \beta^{WH}_{k,t} X^{WH}_{k,t} + \sum_j \sum_k \sum_t \beta^{TR}_{i,k,t} Y^{PL}_{j,k,t}$$

$$FCI = \sum_i \sum_j \sum_k \alpha^{PL}_{i,j,t} CE^{PL}_{i,j,t} + \beta^{PL}_{i,j,t} X^{PL}_{i,j,t} + \sum_k \sum_t \alpha^{WH}_{k,t} CE^{WH}_{k,t} + \beta^{WH}_{k,t} X^{WH}_{k,t} + \sum_j \sum_k \sum_t \beta^{TR}_{i,k,t} Y^{PL}_{j,k,t}$$

(83)

Here, parameters $\alpha_{i,j,t}^{PL}$, $\beta_{i,j,t}^{PL}$ and $\alpha_{k,t}^{WH}$, $\beta_{k,t}^{WH}$ are the variable and fixed investment terms corresponding to plants and warehouses, respectively. $\beta_{i,k,t}^{TR}$ is the fixed investment term associated with the establishment of transportation links between plants and warehouses.

The total capital investment is constrained to be lower than an upper bound, as stated in equation (84):

$$FCI \leq \overline{FCI}$$

(84)

Finally, the model assumes that the payments of the fixed capital investment are divided into equal amounts distributed over the entire planning horizon. Hence, variable $FTDC_t$ is calculated as follows:

$$FTDC_t = \frac{FCI}{NT} \quad \forall t$$

(85)

2. Environmental impacts assessment

To evaluate the environmental performance of the supply chain, an approach is used that combines the Life Cycle Assessment (LCA) with the principles of mathematical programming [12].

The Life Cycle analysis is a quantitative tool to measure the potential environmental impact derived from the use of a certain industrial technology [13]. This tool is characterized by mass and energy balances applied to the entire system under study, instead of applying it only to the process of interest [14].

In this work, it is measured the environmental impact of the supply chain with the GWP100 indicator, as described in the IPCC 2013 (The Intergovernmental Panel on Climate Change). The GWP 100a (Global Warming Potential) indicator is an index that is used to estimate the relative contribution to global warming by the emission of 1 kg of a particular greenhouse gas compared to the emission of 1 kg of carbon dioxide in a 100 year period. The unit of measurement for this indicator is the kg CO2-eq or kilograms of carbon dioxide equivalents.

To determine the total amount of greenhouse gases emitted into the atmosphere during the entire life of the supply chain, a life cycle analysis is carried out, which takes into account three main emission sources that contribute to warming global, and therefore, to the GWP indicator: the consumption of raw materials (GWP_s^{RM}), the energy requirements (GWP_s^{TR}). Hence, the total GWP for each scenario$\left(GWP_s^{total}\right)$ is determined as follows:

$$GWP_s^{total} = GWP_s^{RM} + GWP_s^{EN} + GWP_s^{TR} \quad \forall s$$

(86)

Mathematically, the impact is determined from the purchases of raw materials $(PU_{j,p,t,s})$, production rates at the manufacturing plants ($W_{i,j,p,t,s}$) and transport flows ($Q^{PL}_{j,k,p,t,s}$ and $Q^{WH}_{k,l,p,t,s}$), as stated in equation (87).

$$GWP_s^{total} = \sum_i \sum_p \sum_t IMP_p^{RM} PU_{j,p,t,s} + \sum_i \sum_j \sum_{p \in MP(i)} \sum_t IMP^{EN} \eta^{EN}_{i,j,p} W_{i,j,p,t,s} + \sum_j \sum_k \sum_p \sum_t IMP^{EN} \eta^{EN}_{i,j,p} W_{i,j,p,t,s} + \sum_j \sum_k \sum_p \sum_t IMP^{TR} \lambda^{PL}_{j,k} Q^{PL}_{j,k,p,t,s} + \sum_k \sum_l \sum_p \sum_t IMP^{TR} \lambda^{WH}_{k,l} Q^{WH}_{k,l,p,t} \quad \forall s$$

(87)

In this equation, IMP_p^{RM} , IMP^{EN}, IMP^{TR} denote the cumulative life cycle impact assessment (LCIA) results associated with the consumption of 1 kg of raw material p, 1 MJ of energy, and transportation of 1 ton 1 km of distance, respectively. These LCIA values are available in environmental database such as Ecoinvent [15]. In equation (87), $\lambda^{PL}_{j,k}$ and $\lambda^{WH}_{k,l}$ denote the distance between plants and warehouses and warehouses and markets, respectively. Finally, $\eta^{EN}_{i,j,p}$ represents the consumption of energy per unit of product p manufactured with technology i at plant j. This includes utilities such as electricity, steam, fuel, and cooling water.

Finally, the trade of CO_2 emissions in the model is introduced through equation (88).

$$Sales^{CO_2}_{t,s} + GWP_{t,s} = Max^{CO_2}_t + Buy^{CO_2}_{t,s} \quad \forall t,s$$

(88)

Thus, according to equation (88), the total CO_2 emissions that take place in the supply chain in period t and scenario s must be equal to the limit of free CO_2 emissions $\left(Max^{CO_2}_t\right)$ plus the extra emission rights purchased $\left(Buy^{CO_2}_{t,s}\right)$ minus the emission rights sold $\left(Sales^{CO_2}_{t,s}\right)$ in the period of time t and scenario s.

Therefore, the net income $\left(Net^{CO_2}_{t,s}\right)$ obtained through emissions trading is calculated with equation (89).

$$Net^{CO_2}_{t,s} = Price^{CO_2}_{t,s} Sales^{CO_2}_{t,s} - Cost^{CO_2}_{t,s} Buy^{CO_2}_{t,s} \quad \forall t,s$$

(89)

Where $Cost^{CO_2}_{t,s}$ and $Price^{CO_2}_{t,s}$ represents the cost and price of CO2 emission rights, respectively.

7.2.1.3 Case study

The case study that is used to illustrate the application of the mathematical model proposed in Section 7.2.1.2 has been taken from the work presented by Guillén-Gosálbez and Grossmann [11] and it is included in the European Union Emission Trading Scheme or EU ETS.

The superstructure of this case study is shown in *Figure 51*. The SC under study comprises 1 plant and 1 warehouse that are both located in Tarragona (Spain), and 4 final markets that are located in the following European cities: Leuna (Germany), Neratovice (Czech Republic), Sines (Portugal) and Tarragona. There are 6 different technologies available to manufacture 6 main products: acetaldehyde, acetone, acrylonitrile, cumene, isopropanol and phenol from 9 potential raw materials: oxygen, ethylene, hydrogen cyanide, hydrochloric acid, ammonium, sulfuric acid, propylene and benzene. (*Figure 52*).

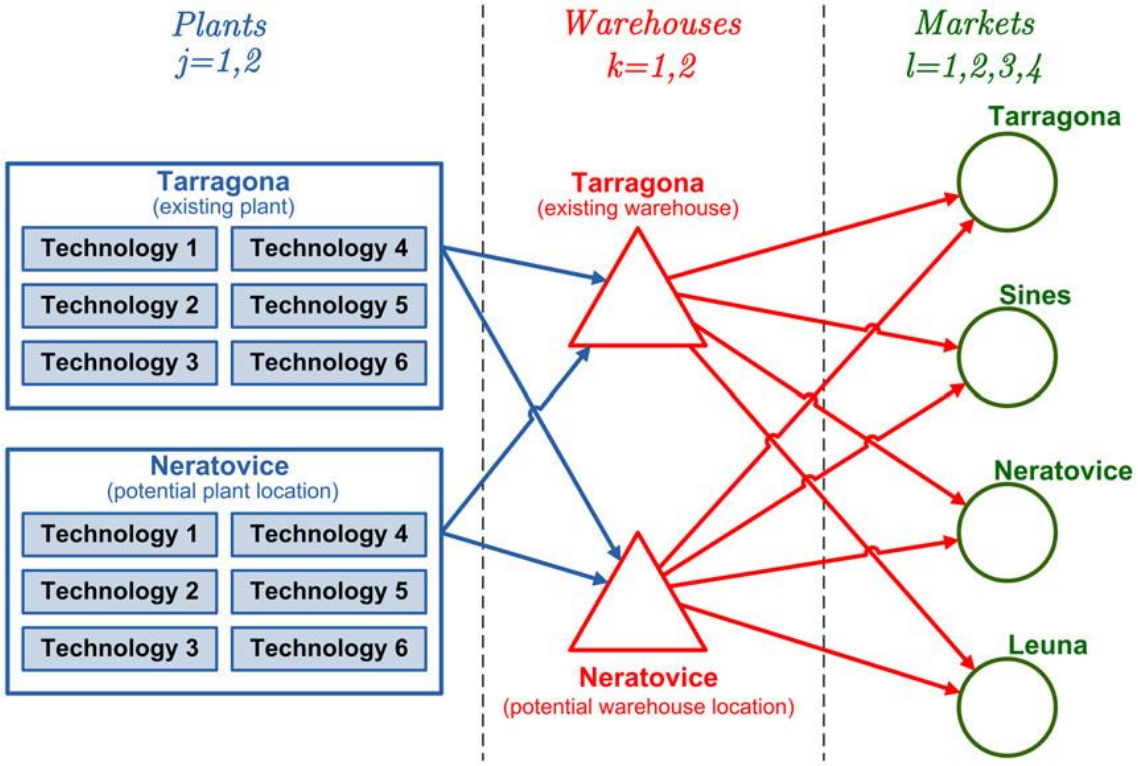


Figure 51. Superstructure of the case study.

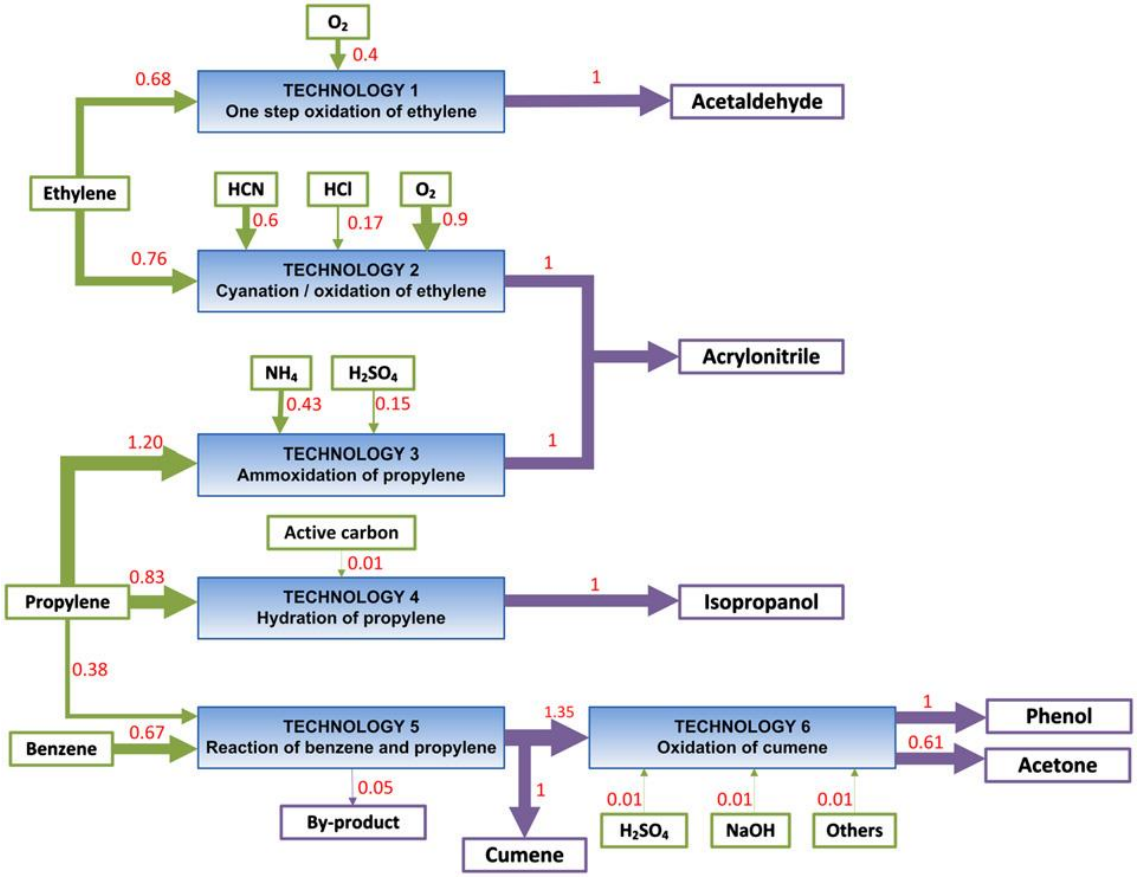


Figure 52. Avaliable technologies at each plant.

The production capacity of the existing plant (Tarragona) is 100000 Tn/year for each of the available technologies, and the capacity of the existing warehouse, also in Tarragona, is 100000 Tn. The expansion capacity of the plants should be between 50000 and 400000 Tn/year, and the capacity of expansion of the warehouses between 50000 and 400000 Tn. The products transported between the different nodes of the supply chain are bounded by 5000 tons/year and 500000 tons/year. In addition, it is considered that the initial inventory of the warehouses is zero.

The interest rate, the redemption value and the tax rate are equal to 10%, 20% and 30% respectively. Besides, the unit transport cost, a low value of 0.4 cent/Tn Km is assumed. The fixed costs associated with the establishment of transport connections are considered zero.

The emission inventories used to illustrate the case study were obtained from the Ecoinvent database. Direct emissions associated with process technologies have not been taken into account.

As it is indicated in Equation (87), transport costs between each node of the supply chain can be calculated directly from the transport unit cost ($\omega^{PL}_{j,k,p,t}$ and $\omega^{WH}_{k,l,p,t}$) and the distances between the different nodes showed in *Table 16.*

Table 16. Distances between the different locations of the supply chain.

	$\lambda_{j,k}^{PL}/\lambda_{k,l}^{WH}\ (Km)$			
Product/Market	**Leuna**	**Neratovice**	**Sines**	**Tarragona**
Neratovice	295.45	0	2970.72	1855.47
Tarragona	1781.36	1855.47	1212.82	0

The raw materials costs of each process plant are shown in *Table 17*.

Table 17 Raw materials costs of each process plant.

	$\lambda_{j,p,t}^{RM}\ (\$/Tn)$	
Product/Plant	**Neratovice**	**Tarragona**
Ammonium	140.54	148.81
Benzene	200.51	212.30
Ethylene	233.68	247.42
Hydrochloric acid	116.18	123.02
Hydrogen cyanide	468.47	496.03
Oxygen	29.98	31.75
Propylene	159.28	168.65
Sodium hydroxide	140.54	148.81
Sulfuric acid	42.16	44.64

The variable and fixed investment costs associated with the 6 process technologies are shown in *Table 18.*

Table 18. Variable and fixed investment costs associated with process technologies

	$\alpha_{i,j,t}^{PL}\ (\$\ year/Tn)$		$\beta_{i,j,t}^{PL}\ (k\$)$	
Technology/Plant	**Neratovice**	**Tarragona**	**Neratovice**	**Tarragona**
T1	91.28	109.53	8306.45	9967.74
T2	93.43	112.12	8502.82	10203.38
T3	235.81	282.97	21459.49	25751.38
T4	104.73	125.68	9530.80	11436.97
T5	46.34	55.60	4216.72	5060.06
T6	165.59	198.70	15069.01	18082.81

The demand for final products in each markets is shown in *Table 19.*

Table 19. Final products demand in each market.

	$\overline{D_{l,p,t}^{MK}}\ (Tn/year)$			
Product/Market	**Leuna**	**Neratovice**	**Sines**	**Tarragona**
Acetaldehyde	13500	37500	12000	7500
Acetone	10800	30000	9600	6000
Acrylonitrile	18000	50000	16000	10000
Cumene	13500	37500	12000	7500
Isopropanol	9000	25000	8000	5000
Phenol	12600	35000	11200	7000

Table 20 shows the price of the final products in each of the markets in which they are sold in $/Tn.

Table 20. Final products price.

	$\gamma_{l,p,t}^{FP}\ (\$/Tn)$			
Product/Market	**Leuna**	**Neratovice**	**Sines**	**Tarragona**
Acetaldehyde	509.26	487.43	491.07	500.17
Acetone	432.87	414.32	417.41	425.14
Acrylonitrile	36.40	34.84	35.10	35.75
Cumene	401.23	384.04	386.90	394.07
Isopropanol	401.23	384.04	386.90	394.07
Phenol	709.88	679.45	684.52	697.20

The energy consumption of each technology is expressed in TFOE per ton produced, and its values are shown in *Table 21.*

Table 21. Energy consumption associated with each technology,

Technology	$\eta_{i,p}^{EN}\ (FOET/Tn)$
T1	0.22
T2	0.60
T3	0.15
T4	0.38
T5	0.06
T6	0.38

Finally, *Table 22* shows the LCIA results according to the IPCC 2013 associated with the raw materials processed, the energy consumed by the facilities and the transport in Kg CO2-eq / Kg; Kg CO2-eq / MJ and Kg CO2-eq / Tn Km.

Table 22. LCIA results for the GWP according to the 2013 IPCC.

Environmental Aspect	**Raw Material**	**LCIA Value**	**Units**	**Name of the data set according to the database of Ecoinvent**
Raw Material consumption (IMP_p^{RM})	Ammonium	2.9424	Kg CO2-eq/Kg	Ammonia, partial oxidation, liquid, at plant
	Oxygen	0.7040		Oxygen, liquid, at plant
	Sulfuric acid	0.1066		Sulphuric acid, liquid, at plant
	Hydrogen cyanide	7.7834		Hydrogen cyanide, at plant
	Ethylene	1.4547		Ethylene, average, at plant
	Propylene	1.4967		Propylene, at plant
	Hydrocloric acid	1.1562		Hydrochloric acid, from the reaction of hydrogen with chlorine, at plant
	Benzene	18563		Benzene, at plant
	Sodium hydroxide	0.9582		Sodium hydroxide, 50% in H2O, production mix, at plant
Energy consumption (IMP^{EN})		0.0882	Kg CO2-eq/MJ	Heavy fuel oil, burned in refinery furnace
Transport (IMP^{TR})		0.092	Kg CO2-eq/Tn Km	Transport, lorry >32t, EURO3

7.2.2 Code implementation

7.2.2.4 Pseudo-code risk curve filter

Once the process has been solved, the solution includes a set of curves equal to the number of scenarios. Each curve also consists of a number of points equal to the number of scenarios with a probability of occurrence. This leads to a great quantity of solutions. For this reason, it is appropriate exclude the sub-optimal solutions, that is to say, those curves that are dominated by other.

The filtering process discards those curves that are dominated by at least one other curve. A solution A is dominated by another solution B when its risk curve is entirely above B. This implies that, for any level of probability, solution A will always lead to lower benefits than solution B. In other words, Solution B will be better considering the entire range of probability levels.

The implemented pseudo-code to identify the dominated curves is shown below. Further information can be found in [21].

Input: NPV risk curves which consist on a matrix of s' points and scenario s ($NPV_{s',s}$) Output: Dominated scenarios s
1: Initializate scenarios s to 1
2: **while** scenario s is less than or equal to the total number of scenarios s
3: $Exit_{ss} \leftarrow 0$ and $ss \leftarrow s + 1$
4: **while** ss is less than or equal to the total number of scenarios s
5: **if** all s' points of $NPV_{s,s'}$ are greater than or equal to $NPV_{ss,s'}$
6: add the scenario s as dominated
7: **elseif** all s' points of $NPV_{s,s'}$ are lower than or equal to $NPV_{ss,s'}$
8: add the scenario ss as dominated
9: **end if**
10: Compute $ss \leftarrow ss + 1$
11: **end while**
12: Compute $s \leftarrow s + 1$
13: **end while**

7.3 Additional Results

Table 23 shows the forecasted CO_2 allowance prices that are obtained with ARIMA(4,1,2) model during the first 3 years.

Table 23. Forecast CO_2 allowance prices given by ARIMA(4,1,2) during 3 first years. .

Date	Y	Date	Y	Date	Y	Date	Y	Date	Y
'30-Sep-2018'	20.727	'24-Nov-2018'	21.246	'14-Jan-2019'	21.062	'06-Mar-2019'	21.176	'28-Apr-2019'	21.355
'02-Oct-2018'	21.251	'25-Nov-2018'	21.046	'15-Jan-2019'	20.996	'08-Mar-2019'	21.214	'29-Apr-2019'	21.376
'03-Oct-2018'	21.506	'27-Nov-2018'	20.822	'17-Jan-2019'	21.021	'10-Mar-2019'	21.266	'01-May-2019'	21.381
'05-Oct-2018'	21.297	'28-Nov-2018'	20.740	'19-Jan-2019'	21.117	'11-Mar-2019'	21.297	'03-May-2019'	21.369
'09-Oct-2018'	20.396	'30-Nov-2018'	20.854	'20-Jan-2019'	21.214	'12-Mar-2019'	21.288	'04-May-2019'	21.353
'10-Oct-2018'	20.367	'02-Dec-2018'	21.075	'21-Jan-2019'	21.246	'14-Mar-2019'	21.251	'06-May-2019'	21.347
'12-Oct-2018'	20.720	'04-Dec-2018'	21.241	'23-Jan-2019'	21.196	'15-Mar-2019'	21.216	'08-May-2019'	21.358
'14-Oct-2018'	21.184	'05-Dec-2018'	21.240	'25-Jan-2019'	21.106	'17-Mar-2019'	21.212	'10-May-2019'	21.380
'16-Oct-2018'	21.424	'06-Dec-2018'	21.083	'26-Jan-2019'	21.047	'19-Mar-2019'	21.242	'12-May-2019'	21.399
'18-Oct-2018'	21.282	'08-Dec-2018'	20.894	'28-Jan-2019'	21.061	'21-Mar-2019'	21.286	'14-May-2019'	21.405
'20-Oct-2018'	20.882	'09-Dec-2018'	20.814	'29-Jan-2019'	21.138	'23-Mar-2019'	21.314	'16-May-2019'	21.397
'22-Oct-2018'	20.528	'11-Dec-2018'	20.896	'30-Jan-2019'	21.222	'25-Mar-2019'	21.310	'17-May-2019'	21.384
'23-Oct-2018'	20.474	'13-Dec-2018'	21.076	'01-Feb-2019'	21.255	'27-Mar-2019'	21.282	'19-May-2019'	21.379
'25-Oct-2018'	20.745	'15-Dec-2018'	21.223	'03-Feb-2019'	21.219	'28-Mar-2019'	21.253	'21-May-2019'	21.387
'27-Oct-2018'	21.134	'17-Dec-2018'	21.236	'05-Feb-2019'	21.146	'30-Mar-2019'	21.247	'23-May-2019'	21.406
'29-Oct-2018'	21.358	'19-Dec-2018'	21.116	'06-Feb-2019'	21.094	'01-Apr-2019'	21.271	'25-May-2019'	21.423
'31-Oct-2018'	21.268	'21-Dec-2018'	20.957	'08-Feb-2019'	21.101	'03-Apr-2019'	21.308	'27-May-2019'	21.430
'02-Nov-2018'	20.948	'22-Dec-2018'	20.880	'10-Feb-2019'	21.162	'05-Apr-2019'	21.334	'29-May-2019'	21.424
'04-Nov-2018'	20.642	'24-Dec-2018'	20.937	'12-Feb-2019'	21.233	'07-Apr-2019'	21.334	'30-May-2019'	21.414
'05-Nov-2018'	20.571	'26-Dec-2018'	21.085	'14-Feb-2019'	21.267	'09-Apr-2019'	21.312	'01-Jun-2019'	21.410
'07-Nov-2018'	20.778	'28-Dec-2018'	21.214	'16-Feb-2019'	21.242	'10-Apr-2019'	21.287	'03-Jun-2019'	21.417
'09-Nov-2018'	21.101	'30-Dec-2018'	21.236	'18-Feb-2019'	21.184	'12-Apr-2019'	21.281	'05-Jun-2019'	21.432
'11-Nov-2018'	21.307	'01-Jan-2019'	21.145	'19-Feb-2019'	21.137	'14-Apr-2019'	21.300	'07-Jun-2019'	21.448
'13-Nov-2018'	21.256	'03-Jan-2019'	21.013	'21-Feb-2019'	21.139	'15-Apr-2019'	21.331	'09-Jun-2019'	21.455
'14-Nov-2018'	21.001	'04-Jan-2019'	20.941	'23-Feb-2019'	21.187	'17-Apr-2019'	21.355	'11-Jun-2019'	21.452
'16-Nov-2018'	20.739	'06-Jan-2019'	20.979	'25-Feb-2019'	21.249	'19-Apr-2019'	21.357	'12-Jun-2019'	21.444
'17-Nov-2018'	20.660	'08-Jan-2019'	21.098	'27-Feb-2019'	21.281	'21-Apr-2019'	21.341	'14-Jun-2019'	21.440
'19-Nov-2018'	20.815	'09-Jan-2019'	21.211	'01-Mar-2019'	21.265	'22-Apr-2019'	21.321	'16-Jun-2019'	21.446
'21-Nov-2018'	21.082	'10-Jan-2019'	21.240	'03-Mar-2019'	21.218	'24-Apr-2019'	21.315	'18-Jun-2019'	21.459
'23-Nov-2018'	21.268	'12-Jan-2019'	21.171	'04-Mar-2019'	21.178	'26-Apr-2019'	21.329	'19-Jun-2019'	21.473

Date	Y	Date	Y	Date	Y	Date	Y	Date	Y
'21-Jun-2019'	21.480	'13-Aug-2019'	21.587	'05-Oct-2019'	21.703	'30-Nov-2019'	21.823	'25-Jan-2020'	21.942
'23-Jun-2019'	21.479	'15-Aug-2019'	21.587	'07-Oct-2019'	21.707	'02-Dec-2019'	21.828	'26-Jan-2020'	21.946
'24-Jun-2019'	21.473	'17-Aug-2019'	21.590	'09-Oct-2019'	21.713	'04-Dec-2019'	21.832	'28-Jan-2020'	21.950
'26-Jun-2019'	21.470	'19-Aug-2019'	21.596	'11-Oct-2019'	21.719	'05-Dec-2019'	21.836	'30-Jan-2020'	21.953
'28-Jun-2019'	21.475	'21-Aug-2019'	21.604	'13-Oct-2019'	21.723	'07-Dec-2019'	21.839	'01-Feb-2020'	21.957
'30-Jun-2019'	21.486	'23-Aug-2019'	21.611	'14-Oct-2019'	21.726	'09-Dec-2019'	21.842	'03-Feb-2020'	21.961
'01-Jul-2019'	21.498	'25-Aug-2019'	21.614	'16-Oct-2019'	21.728	'11-Dec-2019'	21.846	'05-Feb-2020'	21.965
'03-Jul-2019'	21.506	'26-Aug-2019'	21.615	'18-Oct-2019'	21.731	'13-Dec-2019'	21.851	'07-Feb-2020'	21.969
'05-Jul-2019'	21.506	'28-Aug-2019'	21.615	'20-Oct-2019'	21.735	'15-Dec-2019'	21.855	'08-Feb-2020'	21.973
'06-Jul-2019'	21.502	'30-Aug-2019'	21.618	'22-Oct-2019'	21.740	'17-Dec-2019'	21.860	'10-Feb-2020'	21.977
'08-Jul-2019'	21.500	'01-Sep-2019'	21.624	'24-Oct-2019'	21.746	'18-Dec-2019'	21.863	'12-Feb-2020'	21.981
'10-Jul-2019'	21.504	'03-Sep-2019'	21.631	'26-Oct-2019'	21.750	'20-Dec-2019'	21.867	'14-Feb-2020'	21.985
'12-Jul-2019'	21.513	'04-Sep-2019'	21.638	'27-Oct-2019'	21.753	'22-Dec-2019'	21.870	'16-Feb-2020'	21.989
'14-Jul-2019'	21.524	'06-Sep-2019'	21.641	'29-Oct-2019'	21.756	'24-Dec-2019'	21.874	'17-Feb-2020'	21.993
'15-Jul-2019'	21.532	'07-Sep-2019'	21.643	'31-Oct-2019'	21.759	'26-Dec-2019'	21.878	'19-Feb-2020'	21.997
'17-Jul-2019'	21.533	'09-Sep-2019'	21.644	'02-Nov-2019'	21.763	'28-Dec-2019'	21.883	'20-Feb-2020'	22.001
'18-Jul-2019'	21.531	'11-Sep-2019'	21.646	'04-Nov-2019'	21.768	'30-Dec-2019'	21.887	'22-Feb-2020'	22.005
'20-Jul-2019'	21.529	'12-Sep-2019'	21.652	'06-Nov-2019'	21.773	'31-Dec-2019'	21.891	'24-Feb-2020'	22.008
'22-Jul-2019'	21.532	'13-Sep-2019'	21.658	'08-Nov-2019'	21.777	'02-Jan-2020'	21.894	'26-Feb-2020'	22.012
'24-Jul-2019'	21.541	'15-Sep-2019'	21.664	'09-Nov-2019'	21.781	'04-Jan-2020'	21.898	'28-Feb-2020'	22.016
'26-Jul-2019'	21.551	'17-Sep-2019'	21.668	'11-Nov-2019'	21.783	'06-Jan-2020'	21.902	'01-Mar-2020'	22.020
'28-Jul-2019'	21.558	'18-Sep-2019'	21.670	'13-Nov-2019'	21.787	'08-Jan-2020'	21.906	'03-Mar-2020'	22.025
'30-Jul-2019'	21.560	'20-Sep-2019'	21.672	'15-Nov-2019'	21.791	'10-Jan-2020'	21.910	'04-Mar-2020'	22.029
'31-Jul-2019'	21.559	'22-Sep-2019'	21.675	'17-Nov-2019'	21.795	'12-Jan-2020'	21.914	'06-Mar-2020'	22.032
'02-Aug-2019'	21.558	'24-Sep-2019'	21.679	'19-Nov-2019'	21.800	'13-Jan-2020'	21.918	'08-Mar-2020'	22.036
'04-Aug-2019'	21.561	'26-Sep-2019'	21.686	'21-Nov-2019'	21.805	'15-Jan-2020'	21.922	'10-Mar-2020'	22.040
'06-Aug-2019'	21.568	'28-Sep-2019'	21.691	'22-Nov-2019'	21.808	'17-Jan-2020'	21.925	'12-Mar-2020'	22.044
'08-Aug-2019'	21.577	'30-Sep-2019'	21.696	'24-Nov-2019'	21.811	'19-Jan-2020'	21.929	'14-Mar-2020'	22.048
'10-Aug-2019'	21.584	'01-Oct-2019'	21.698	'26-Nov-2019'	21.814	'21-Jan-2020'	21.933	'16-Mar-2020'	22.052
'12-Aug-2019'	21.587	'03-Oct-2019'	21.700	'28-Nov-2019'	21.818	'23-Jan-2020'	21.938	'17-Mar-2020'	22.056

Date	Y	Date	Y	Date	Y	Date	Y	Date	Y
'19-Mar-2020'	22.060	'14-May-2020'	22.178	'09-Jul-2020'	22.296	'01-Sep-2020'	22.415	'27-Oct-2020'	22.533
'21-Mar-2020'	22.064	'16-May-2020'	22.182	'11-Jul-2020'	22.300	'03-Sep-2020'	22.419	'29-Oct-2020'	22.537
'23-Mar-2020'	22.068	'18-May-2020'	22.186	'12-Jul-2020'	22.304	'05-Sep-2020'	22.423	'31-Oct-2020'	22.541
'25-Mar-2020'	22.071	'20-May-2020'	22.190	'14-Jul-2020'	22.308	'07-Sep-2020'	22.427	'02-Nov-2020'	22.545
'27-Mar-2020'	22.076	'21-May-2020'	22.194	'16-Jul-2020'	22.312	'09-Sep-2020'	22.430	'04-Nov-2020'	22.549
'29-Mar-2020'	22.080	'23-May-2020'	22.198	'18-Jul-2020'	22.316	'11-Sep-2020'	22.434	'05-Nov-2020'	22.553
'30-Mar-2020'	22.084	'25-May-2020'	22.202	'20-Jul-2020'	22.320	'13-Sep-2020'	22.438	'07-Nov-2020'	22.557
'01-Apr-2020'	22.088	'27-May-2020'	22.206	'22-Jul-2020'	22.324	'14-Sep-2020'	22.442	'09-Nov-2020'	22.561
'03-Apr-2020'	22.091	'29-May-2020'	22.210	'24-Jul-2020'	22.328	'16-Sep-2020'	22.446	'11-Nov-2020'	22.565
'05-Apr-2020'	22.095	'31-May-2020'	22.214	'25-Jul-2020'	22.332	'18-Sep-2020'	22.450	'13-Nov-2020'	22.568
'07-Apr-2020'	22.099	'02-Jun-2020'	22.218	'27-Jul-2020'	22.336	'20-Sep-2020'	22.454	'15-Nov-2020'	22.572
'09-Apr-2020'	22.103	'03-Jun-2020'	22.222	'29-Jul-2020'	22.340	'22-Sep-2020'	22.458	'17-Nov-2020'	22.576
'11-Apr-2020'	22.107	'05-Jun-2020'	22.225	'31-Jul-2020'	22.344	'24-Sep-2020'	22.462	'18-Nov-2020'	22.580
'12-Apr-2020'	22.111	'07-Jun-2020'	22.229	'02-Aug-2020'	22.348	'26-Sep-2020'	22.466	'20-Nov-2020'	22.584
'14-Apr-2020'	22.115	'09-Jun-2020'	22.233	'03-Aug-2020'	22.352	'27-Sep-2020'	22.470	'22-Nov-2020'	22.588
'16-Apr-2020'	22.119	'11-Jun-2020'	22.237	'05-Aug-2020'	22.356	'29-Sep-2020'	22.474	'24-Nov-2020'	22.592
'18-Apr-2020'	22.123	'13-Jun-2020'	22.241	'06-Aug-2020'	22.360	'01-Oct-2020'	22.478	'26-Nov-2020'	22.596
'20-Apr-2020'	22.127	'15-Jun-2020'	22.245	'08-Aug-2020'	22.363	'03-Oct-2020'	22.482	'28-Nov-2020'	22.600
'22-Apr-2020'	22.131	'16-Jun-2020'	22.249	'10-Aug-2020'	22.367	'05-Oct-2020'	22.486	'30-Nov-2020'	22.604
'24-Apr-2020'	22.135	'18-Jun-2020'	22.253	'12-Aug-2020'	22.371	'07-Oct-2020'	22.490	'01-Dec-2020'	22.608
'25-Apr-2020'	22.139	'20-Jun-2020'	22.257	'14-Aug-2020'	22.375	'09-Oct-2020'	22.494	'03-Dec-2020'	22.612
'27-Apr-2020'	22.143	'22-Jun-2020'	22.261	'16-Aug-2020'	22.379	'10-Oct-2020'	22.497	'05-Dec-2020'	22.616
'29-Apr-2020'	22.147	'24-Jun-2020'	22.265	'18-Aug-2020'	22.383	'12-Oct-2020'	22.501	'07-Dec-2020'	22.620
'01-May-2020'	22.150	'26-Jun-2020'	22.269	'19-Aug-2020'	22.387	'14-Oct-2020'	22.505	'08-Dec-2020'	22.624
'03-May-2020'	22.154	'28-Jun-2020'	22.273	'21-Aug-2020'	22.391	'16-Oct-2020'	22.509	'09-Dec-2020'	22.628
'05-May-2020'	22.158	'29-Jun-2020'	22.277	'23-Aug-2020'	22.395	'18-Oct-2020'	22.513	'10-Dec-2020'	22.632
'07-May-2020'	22.162	'01-Jul-2020'	22.281	'25-Aug-2020'	22.399	'20-Oct-2020'	22.517	'11-Dec-2020'	22.635
'08-May-2020'	22.166	'03-Jul-2020'	22.285	'27-Aug-2020'	22.403	'22-Oct-2020'	22.521	'13-Dec-2020'	22.639
'10-May-2020'	22.170	'05-Jul-2020'	22.288	'29-Aug-2020'	22.407	'23-Oct-2020'	22.525	'15-Dec-2020'	22.643
'12-May-2020'	22.174	'07-Jul-2020'	22.292	'31-Aug-2020'	22.411	'25-Oct-2020'	22.529	'17-Dec-2020'	22.647

Date	Y	Date	Y
'19-Dec-2020'	22.651	'11-Feb-2021'	22.770
'21-Dec-2020'	22.655	'12-Feb-2021'	22.773
'23-Dec-2020'	22.659	'14-Feb-2021'	22.777
'24-Dec-2020'	22.663	'16-Feb-2021'	22.781
'26-Dec-2020'	22.667	'17-Feb-2021'	22.785
'28-Dec-2020'	22.671	'18-Feb-2021'	22.789
'30-Dec-2020'	22.675	'20-Feb-2021'	22.793
'01-Jan-2021'	22.679	'21-Feb-2021'	22.797
'03-Jan-2021'	22.683	'22-Feb-2021'	22.801
'05-Jan-2021'	22.687	'24-Feb-2021'	22.805
'06-Jan-2021'	22.691	'26-Feb-2021'	22.809
'08-Jan-2021'	22.695	'27-Feb-2021'	22.813
'10-Jan-2021'	22.699	'01-Mar-2021'	22.817
'12-Jan-2021'	22.703	'03-Mar-2021'	22.821
'14-Jan-2021'	22.706	'04-Mar-2021'	22.825
'16-Jan-2021'	22.710	'05-Mar-2021'	22.829
'18-Jan-2021'	22.714	'07-Mar-2021'	22.833
'19-Jan-2021'	22.718	'08-Mar-2021'	22.837
'21-Jan-2021'	22.722	'09-Mar-2021'	22.841
'23-Jan-2021'	22.726		
'25-Jan-2021'	22.730		
'26-Jan-2021'	22.734		
'28-Jan-2021'	22.738		
'30-Jan-2021'	22.742		
'31-Jan-2021'	22.746		
'02-Feb-2021'	22.750		
'04-Feb-2021'	22.754		
'06-Feb-2021'	22.758		
'07-Feb-2021'	22.762		
'09-Feb-2021'	22.766		

In *Table 24* are presented the ARIMA(4,1,2) model parameters obtained from the CO_2 allowance prices and *Table 25* the average mean used as parameters in the deterministic model.

Table 24. ARIMA(4,1,2) Model parameters

Parameter	**Value**	**Standard Error**	t **Statistic**
Constant	0.00323827	0.00496789	0.65184
AR{1}	115.614	0.0381616	30.296
AR{2}	-0.926418	0.0353956	-261.732
AR{3}	0.0159244	0.0252981	0.629468
AR{4}	-0.0669539	0.0211635	-316.366
MA{1}	-116.982	0.039005	-299.916
MA{2}	0.87485	0.0331871	263.612
Variance	0.0862777	0.00100673	857.008

Table 25. Mean of ARIMA(4,1,2) forecasted values used in deterministic model.

Time Period	Forecasted value (€/ton)
1	21,686
2	22,466
3	23,195
4	23,893
5	24,611
6	25,332
7	26,054
8	26,772
9	27,493
10	28,932