Electrical Energy Cost Minimization of Yeast Production at Lallemand Iberia S.A. Company Using a Multi-Objective Optimization Approach Considering Time-dependent Electricity Cost

Marcos André Marques Policarpo
marcos.policarpo@tecnico.ulisboa.pt
Instituto Superior Técnico, Universidade de Lisboa, Portugal
July 2017

***Abstract* - In this study, a single objective Genetic Algorithm (GA) optimization model and a multi-objective GA (NSGA-II) optimization model are created to determine the best schedule of production of a fermentation facility that minimizes the energetic costs, considering a set of constraints imposed by the factory. It's also studied the behavior of the various parameters of both algorithm in both performance (quality of the results obtained in each case) and velocity, so that it may be possible to compare and conclude which of the models adapts better to the case study**

Introduction

As it is known, the main expense in most of the industries is associated to the energetic sector as it can be seen in [1] and [2]. Regarding that, there are certain methods to reduce the cost associated to it, such as, replacing old machines with improved new ones (solution normally used in complex production processes), and some of this methods and others applied to the food sector can be seen in [3] and [4]. Inside the food industry, the baker's yeast industry has high electric energy costs that influence its competitiveness. Besides, this industry faces rising feedstock and energy prices, as Portugal is a clearly example. To address this, it is crucial to improve the energy efficiency and also energy costs of the overall yeast production process to the highest possible extent [5]. Energy efficiency encompasses all activities that lead to a decrease in electric energy consumption. Applying this concept today to the industry demands a complete different meaning than simply replacing equipment by one more energetically efficient. As it is known, the main expense in most of the industries is associated to the energetic sector. Regarding that, there are certain methods to reduce the cost associated to it, such as, replacing old machines with improved new ones (solution normally used in complex production processes). Another method is to analyze the production process with the objective of finding inefficiencies in the process that can be improved. In this case, it will be used optimization algorithms to minimize the energetic costs associated to the production process of a fermentation industry. Due to the high complexity of the production process of the fabric, it will be used heuristic methods of optimization, specifically, it will be used genetic algorithms. These kinds of algorithms are based in various procedures with the objective of finding feasible and good solutions in specific problems, with a specific set of constraints. These procedures do not guarantee always optimal solutions, but often requires less time of process. Also, once the genetic algorithm is based on natural evolutionary process, it adjust well in problems with a wide range of solutions. The main objective of this model will be to reduce the electrical bill by adjusting the scheduled production plan of the fabric, taking into account that the weekly production plan must be ensured.

In section 2, it will be explained the process that is involved in a fermentation facility, the scope used for this model and the relevant information from the electrical revenue. Section 3 will display all the mathematical part of the model. Section 4 will explain what are GA and how they are used. Section 5 will display all the results and tests made to this model and section 6 will resume the conclusions. Section 7 will enumerate all the references used in this paper.

1. Problem definition

As it was referred, the objective of the factory will be to turn the production process of the fabric more efficient. To achieve that, since it is not always possible to invest in the fabric, the approach used to achieve this goal was to adjust the production plan to reduce the electrical bill of the fabric. In order to understand how this objective will be obtained, it is necessary to understand how the fermentation process of the fabric is made. In the first stage, a pure culture of the lineage Saccharomyces cerevisiae non-modified genetically is subjected to series of early fermentations so that the ferment can develop (most of these fermentations are made in a laboratory). After it achieves a certain development, this ferment will be transferred into the industrial fermenters (which will be the part of the process that will be studied). From this fermentation, it is possible to obtain 2 types of ferment: the mother yeast and the commercial yeast. From this point on, the ferment can be submitted into a separator to obtain fresh yeast and, later, if desired, submitted into a drying process to obtain the dry yeast. In the case study, the fermentation facility will be equipped with 5 fermenters (*Ferm*) that will produce both mother yeast and commercial yeast. Also, for each fermenter, it will be defined a fermentation cycle divided in 3 processes: Fermentation (*Ferment*), Separation (*Sep*) and Initialization (*Inic*)). In tables 1 and 2, it can be observed the time distribution (hours) and the total production (Ton) made by each cycle in each one of the 5 fermenters.

Ferm	***Ferment*[h]**	***SEP*[h]**	***INIC*[h]**	***Total*[h]**
F1	15	3,5	2,5	21
F2	14	2,5	3,5	20
F3	14	2,5	3,5	20
F5	14	2,5	3,5	20
F4	24	2,5	3,5	30

Table 1 – Fermentation duration

Ferm	***Ferment Type***	***Output* [Ton]**
F1	Commercial	30
F2	Commercial	24
F3	Commercial	20,5
F5	Commercial	20,5
F4	Mother	20,5

Table 2 – Fermentation type and output

Also, for each process, it is necessary to know how much air is necessary to feed to each fermenter to get the fermentation process done. Taking that into account, the figures from 1 to 5 will show the air consumption needed to feed in each fermenter so they can execute a complete fermentation cycle represented in intervals of half hours.

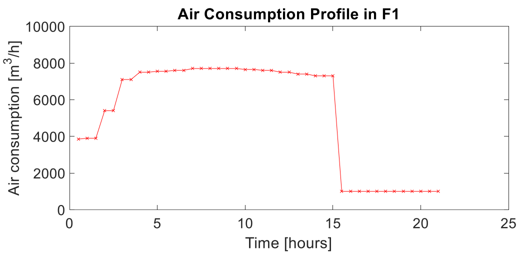

Figure 1 – Average air consumption profile in F1

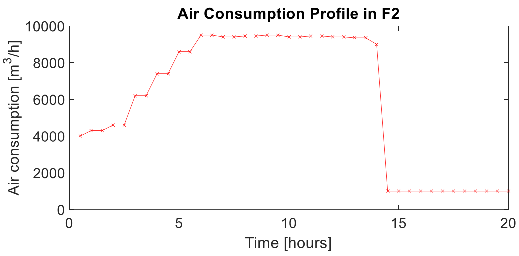

Figure 2 – Average air consumption profile in F2

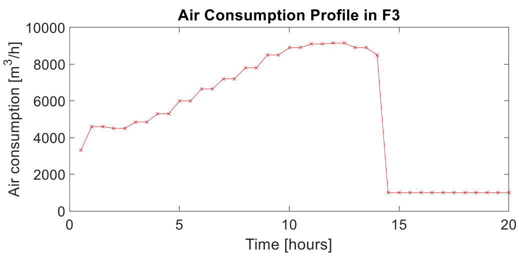

Figure 3 – Average air consumption profile in F3

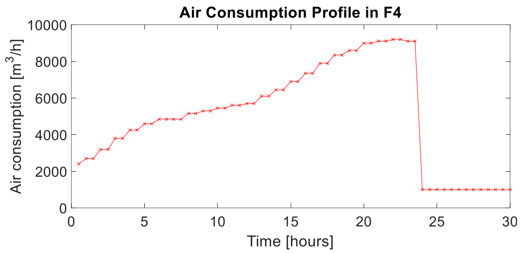

Figure 4 – Average air consumption profile in F4

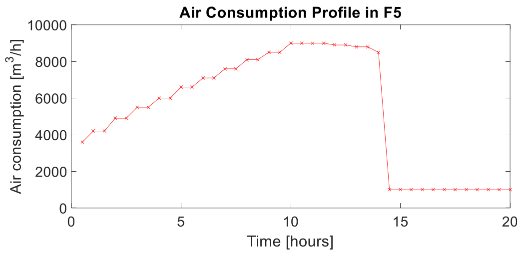

Figure 5 – Average air consumption profile in F5

Since it is now known the air consumption for each of the fermenters, it is necessary to know how the air is fed to each one. For that, it is necessary to know which compressors connected to which fermenters, which is displayed in figure 6.

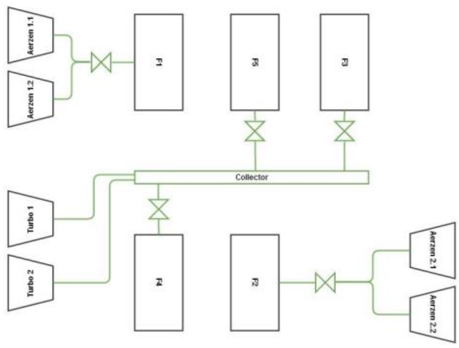

Figure 6 – Fermenters and compressors connection scheme

As it can be observed in figure 6, Fermenters 3, 4 and 5 are fed by 2 Turbo compressors and fermenters 1 and 2 are fed by 4 Aerzen compressors. It is not represented in figure 6 but there also exists another set of compressors (the 2 Holmes compressors) will be responsible to feed the air to fermenter 1 and/or fermenter 2 every time that the air consumption in either one of them is lower than 1000 m^3/h.

It is now necessary to define the relation between the air consumption of the compressors and their energetic costs. In figure 7, it is possible to observe the relation between power and the volume of air in each one of the Aerzen compressors.

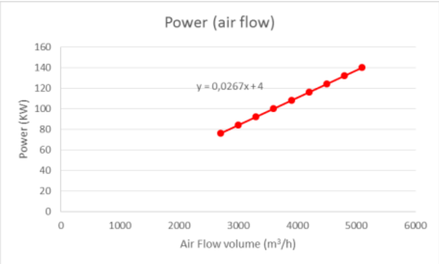

Figure 7 – Power consumption as a function of the air flow volume in each Aerzen compressor

The turbo compressors have a different way of working. When the air consumption is lower than 10500 m³/h, the air flow is provided by the Turbo 1 (T1); when the air consumption is between 10500 m³/h and 15000 m³/h, the air flow is provided by the Turbo 2 (T2); when the air consumption is higher than 15000 m³/h, Turbo 2 is used in its maximum potency and, the remaining air, is provided by Turbo 1. In figure 8 and 9, it is possible to observe the relation between power and the volume of air in each one of the Turbo compressors.

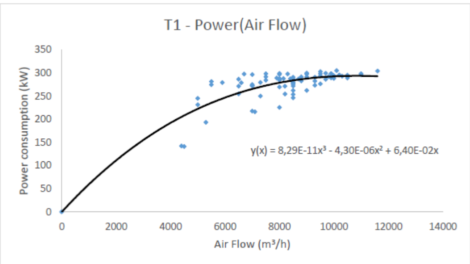

Figure 8 – Power consumption as a function of the air flow volume in Turbo 1

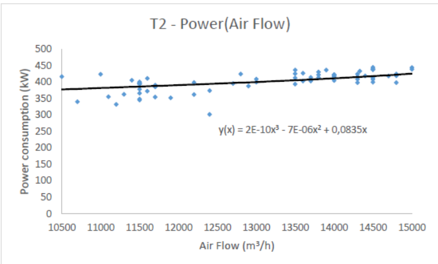

Figure 9 – Power consumption as a function of the air flow volume in Turbo 2

As it was referred earlier, the two Holmes compressors will be responsible to feed the air to fermenter 1 and/or fermenter 2 every time the air consumption in either one of them is lower than 1000m³/h. These compressors have a nominal power of 22 kW and they always work at nominal power, since they always ensure a constant air flow of 1000 m³/h when they are working. The consumption of the separators will not be considered in the model since their value wouldn't cause enough impact in the results. Also, due to the low number of separators in the fabric, it is only possible to do one separation process at a time.

Finally, to finish the analysis of the fabric, it is only necessary to define the tariff that is applied to the fabric. The tariff has 4 hourly periods during every week which are represented in table 3.

Monday to Friday	
Super Peak	09:15 – 12:15
Peak	07:00 – 09:15 ; 12:15 – 24:00
Super Off-Peak	02:00 – 06:00
Off-Peak	00:00 – 02:00 ; 06:00 – 07:00
Saturday	
Peak	09:00 – 14:00 ; 20:00 – 22:00
Super Off-Peak	02:00 – 06:00
Off-Peak	00:00 – 02:00 ; 06:00 – 09:00 14:00 – 20:00 ; 22:00 – 24:00
Sunday	
Super Off-Peak	02:00 – 06:00
Off-Peak	00:00 – 02:00 ; 06:00 – 24:00

Table 3 – Weekly hour tariff

To calculate the electric invoice, it is necessary to know the term that it contains and are going to be enumerated now:

- **Active energy invoiced (€/kWh)**

Active energy consumed in each time period of the contracted tariff. Its value is negotiated between the company and its retailer.

The values used for this constants (that change according to the tariff being applied) are:

$$active\ energy\ costs \rightarrow \begin{cases} Super\ peak\ hour - c_{Sp}^{a} = 0.0335 \\ Peak\ hour - c_{p}^{a} = 0.0291 \\ Super\ off\ peak\ hour - c_{Soff}^{a} = 0.0156 \\ Off\ peak\ hour - c_{off}^{a} = 0.0162 \end{cases} (€/kWh)$$

- **Energy network access term (€/kWh)**

This is relative to the usage of the transition and distribution power lines (grid usage) and also varies with the hourly periods. Its value is regulated and fixed by ERSE (it is independent from the retailer).

The values used for these constants (that change according to the tariff being applied) are:

$$energy\ network\ access\ term \rightarrow \begin{cases} Super\ peak\ hour - c_{Sp}^{n} = 0.0702 \\ Peak\ hour - c_{p}^{n} = 0.0642 \\ Super\ off\ peak\ hour - c_{Soff}^{n} = 0.0442 \\ Off\ peak\ hour - c_{off}^{n} = 0.0548 \end{cases} (€/kWh)$$

- **Power network term**
  - **Contracted Power (€/kWday)**

Maximum average power registered in any uninterrupted interval of 15 minutes over the last 12 months, including the one concerning the invoice

Contracted power won't be considered in the program since its value remains almost constant during the months.

- **Power consumed in super peak hours (€/kWday)**

Total energy consumed in peak hours, divided by the number of peak hours existent in the period relative to the invoice

The values used for this constant is:

$$c_{spH} = 0.2945\ (€/kW\ day)$$

- **Reactive Power (€/kWArh)**

Total reactive power consumed. This term will not be considered in this thesis, because Lallemand has a capacitor bank connected to the grid.

Because of that, the values of this variable are very low and, at most of the times, are even zero.

- **Special tax over electricity consumption (€/kWh)**

This last value is a special tax applied to electricity (similar to other taxes like the ones that are applied to tobacco, alcohol and hydrocarbons). This tax is calculated by multiplying what you will pay for the consumption and power contracted. As such it will not have that much effect on the end result of the model since this will be presented in percentage numbers.

2. Mathematical formulation

In this section the mathematical formulation will be defined. It will be explained all the formulations and calculations that are necessary to go from our decision variable to the objective function and also to the constraints that are necessary to take in count in the problem presented.

2.1. Decision variable

The first thing that is needed to do when creating a mathematical optimization model is to define what will be the decision variables. Our decision variables will be composed by 36 integer variables (that can vary between 1 and 336 = 7days*24 hours*2 half hours) and 36 binary variables. These variables represent the maximum number of fermentations that are possible to execute with the 5 fermenters during a week period. The table 4 summarizes the representation.

Fermenter	Integer Variables	Binary variables
F1	x_1 to x_7	x_{37} to x_{43}
F2	x_8 to x_{15}	x_{44} to x_{51}
F3	x_{16} to x_{23}	x_{52} to x_{59}
F5	x_{24} to x_{31}	x_{60} to x_{67}
F4	x_{32} to x_{36}	x_{68} to x_{72}

Table 4 – Integer and binary variables for each fermenter

2.2. Fermenters air consumption

Knowing the decision variables, it is now possible to start defining the model. From the information of the decision variable and the air consumption curves of the fermenters, it is possible to define 10 variables ($x1(t)$ to $x5(t)$ and $y1(t)$ to $y5(t)$) that, when combined, will give us all the air needed over the time t that is necessary to give to each fermenter to execute their fermentations. The variables $x1(t)$ to $x5(t)$ will be binary variables, that will indicate if there is a fermentation occurring or not and if there is any superposition along the week plan (it will give the value 0 if there aren´t any fermentation cycle occurring, it will give value 1 if there is 1 fermentation cycle occurring and if it gives a value a bigger than 1 than that means that, in that time, there are a fermentations going on in that time t – superposition), from fermenter F1 to fermenter F5. The variables $y1(t)$ to $y5(t)$ will give us the integer value of the air being consumed at time t form fermenter F1 to F5.

2.3. Energy model

With the air consumption of each fermenter defined, it is possible to calculate the energy consumed by the compressors.

As it can be seen in figure 6, the 4 Aerzen compressors are divided between the fermenters 1 and 2. Knowing that and the relation between the power curve and the air consumption, it is possible to determine the energy consumed by the Aerzen compressors by applying:

$$P_{A1}(t) = 2 * 0.027653 * x1(t) * y1(t) \quad (1)$$

$$P_{A2}(t) = 2 * 0.027653 * x2(t) * y2(t) \quad (2)$$

$$E_{A1}(t) = P_{A1}(t) * \Delta t \quad (3)$$

$$E_{A2}(t) = P_{A2}(t) * \Delta t \quad (4)$$

$$E_{Aerzen}(t) = E_{A1}(t) + E_{A2}(t) \quad (5)$$

As referred earlier, the Turbo compressors have a different way of working. They depend directly from the instantaneous air consumption in the collector, and, the usage of the compressors is different than normal as it can be seen in table 5.

$air_{collector}\ (t)\ =\ a_c\ (t)$ [m³/h]	Turbo compressors
$air_{collector}$ <= 10500	T1 – ON T2 – OFF
10500 < $air_{collector}$ <=15000	T1 – OFF T2 – ON
15000 < $air_{collector}$	T1 – ON T2 – ON

Table 5 – Approximate mode of turning ON and OFF T1 and T2 depending on the air volume consumption

Knowing the amount of air consumption in the collector it is possible to calculate the energy consumed by the Turbo compressors. Using the information from figures 8 and 9:

If $a_c(t)$ <= 10500:

$$P_{T1}(t) = 8{,}29 * 10^{-11} * a_c(t)^3 - 4{,}30 * 10^{-6} * a_c(t)^2 + 6{,}40 * 10^{-2} * a_c(t) \quad (6)$$

$$P_{T2}(t) = 0 \quad (7)$$

If 10500 < a_c (t) <=15000:

$$P_{T1}(t) = 0 \quad (8)$$

$$P_{T2}(t) = 2 * 10^{-10} * a_c(t)^3 - 7 * 10^{-6} * a_c(t)^2 + 0.0835 * a_c(t) \quad (9)$$

If 15000 < $a_c(t)$:

$$P_{T1}(t) = 8{,}29 * 10^{-11} * (a_c(t) - 1500)^3 - 4{,}3010^{-6} * (a_c(t) - 1500)^2 + 6{,}40 * 10^{-2} * (a_c(t) - 1500) \quad (10)$$

$$P_{T2}(t) = 2 * 10^{-10} * 15000^3 - 7 * 10^{-6} * 15000^2 + 0.0835 * 15000 \quad (11)$$

$$E_{T1}(t) = P_{T1}(t) * \Delta t \quad (12)$$

$$E_{T2}(t) = P_{T2}(t) * \Delta t \quad (13)$$

$$E_{Turbo}(t) = E_{T1}(t) + E_{T2}(t) \quad (14)$$

Also, as it was stated earlier the two Holmes compressor will be responsible to feed the air to fermenter 1 and/or fermenter 2 every time that the air consumption in either one of them is lower than 1000m^3/h. Knowing that their nominal power is 22kW, the energy consumed by the Holmes compressors can be calculated by doing:

$$E_{H1} = P_{H1} * \Delta t \quad (15)$$

$$E_{H2} = P_{H2} * \Delta t \quad (16)$$

Every time the respective compressor is working obtaining finally:

$$E_{Holmes} = E_{H1} + E_{H2} \quad (17)$$

After all this, it is possible to obtain the total energy consumed by doing:

$$E_{Total}(t) = E_{Turbo}(t) + E_{Holmes}(t) + E_{Aerzen}(t) \quad (18)$$

2.4. Objective function

Knowing the total energy consumed and the tariffs, it is possible to calculate the amount of energy consumed in each of the 4 hourly periods:

$$E_{SpH}(t) = E_{Total}(t) * H_{Sp}(t) \quad (19)$$

$$E_{pH}(t) = E_{Total}(t) * H_p(t) \quad (20)$$

$$E_{SoffH}(t) = E_{Total}(t) * H_{Soff}(t) \quad (21)$$

$$E_{offpH}(t) = E_{Total}(t) * H_{off}(t) \quad (22)$$

With this, it is possible to calculate the electric cost by applying the cost formula:

$$c(t) = \left(c_{Sp}^a + c_{Sp}^n + 7 * \frac{1}{N_{SpH}} c_{SpH}\right) * E_{SpH}(t) + (c_p^a + c_p^n) * E_{pH}(t) + (c_{Soff}^a + c_{Soff}^n) * E_{SoffH}(t) + (c_{offp}^a + c_{offp}^n) * E_{offH}(t) \quad (23)$$

$$C(x) = \sum_{t=1}^{336} c(t) \quad (24)$$

Our objective function is a ratio between the cost and the total production:

$$F_{objective}(x) = \frac{C(x)}{Y(x)} \quad (25)$$

It is now necessary to calculate the total production. Using the binary variables from the decision variable and the information from table 2:

$$Y(x) = 30 * N_{F1} + 24 * N_{F2} + 20{,}5 * (N_{F3} + N_{F4} + N_{F5}) \; with \begin{cases} N_{F1} = \sum_{i=37}^{43} x(i) \\ N_{F2} = \sum_{i=44}^{51} x(i) \\ N_{F3} = \sum_{i=52}^{59} x(i) \\ N_{F5} = \sum_{i=60}^{67} x(i) \\ N_{F4} = \sum_{i=68}^{72} x(i) \end{cases} \quad (26)$$

Obtaining the objective function:

$$MIN\left(F_{objective}(x)\right) = MIN\left(\frac{C(x)}{Y(x)}\right) \quad (27)$$

2.5. Constraints

With the objective function calculated, it is now necessary to define the constraints of the model.

It is relevant to create constraints to the possible solutions to remove possible mathematical solutions that cannot be applied in real life. For this problem, 4 constraints were defined.

The first constraint is related to the separators of the fabric. As it was mentioned earlier, it is only possible to execute one separation process at a time. To compute this constraint, it is necessary to sum, in one variable, all the times the separation process of all the fermenters in a week period. Since it is only possible to execute one separation at a time, it is necessary to guarantee that this variable is always lower or equal to 1:

$$U_{sep} \leq 1 \quad (28)$$

The second constraint is applied to the production goal. It is necessary to guarantee that the production goals are achieved. For that, it is applied the following formulas:

$$Y_A(x) = 30 * N_{F1} + 24 * N_{F2} + 20{,} * (N_{F3} + N_{F5}) \geq A \quad (29)$$

$$Y_B(x) = 20{,}5 * N_{F4} \geq B \quad (30)$$

A and B being the production goals that are expected to achieve in that week period.

The third constraint is a time constraint. It is important to certify that the production plan obtained, doesn't surpass the week period time. To ensure that, it is necessary to apply the following constraints for each fermenter process:

$$
\begin{cases}
for\ i = [1-7]\ (T_{F1} + x_i) * x_{i+36} \leq 168 \\
for\ i = [8-15]\ (T_{F2} + x_i) * x_{i+36} \leq 168 \\
for\ i = [16-23]\ (T_{F3} + x_i) * x_{i+36} \leq 168 \\
for\ i = [24-31]\ (T_{F5} + x_i) * x_{i+36} \leq 168 \\
for\ i = [32-36] (T_{F4} + x_i) * x_{i+36} \leq 168
\end{cases} \quad (31)
$$

The fourth and last constraint is the superposition constraint. Since it is impossible, for each fermenter, to execute two fermentation processes at the same time, it is relevant to secure that this does not happen. To ensure that, it is necessary to apply the following formulas:

$$
\begin{cases}
sep(i) = x_{i+1} * x_{i+1+N} - x_i * x_{i+N} - 42 \\
\quad for\ i = [1-6] \\
sep(i) = x_{i+1} * x_{i+1+N} - x_i * x_{i+N} - 40 \\
\quad for\ i = [8-14] \\
sep(i) = x_{i+1} * x_{i+1+N} - x_i * x_{i+N} - 40 \\
\quad for\ i = [16-22] \\
sep(i) = x_{i+1} * x_{i+1+N} - x_i * x_{i+N} - 40 \\
\quad for\ i = [24-30] \\
sep(i) = x_{i+1} * x_{i+1+N} - x_i * x_{i+N} - 60 \\
\quad for\ i = [32-36]
\end{cases} \quad (32)
$$

It is not necessary to verify the other combinations of the fermentation cycles because the upper and lower bound are defined in a way that can secure that.

The upper and lower bound were define with a mathematical algorithm that locks the tasks in intervals. Also this algorithms gives the possibility to start the first fermentation only after a certain amount of time (a characteristic that takes into account possible delays from previous weeks). For the case study all this information is summed in tables 5and 6. For the binary variables the upper bound is always 1 and the lower bound is always 0.

Fermenter	Possible start
F1	1
F2	16
F3	22
F5	1
F4	12

Table 1 – Integer values of the minimal initiation of the first tasks of the fermenters

	Bound	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Task 8
F 1	UB	43	85	127	169	211	253	294	
	LB	1	43	85	127	169	211	253	
F 2	UB	56	96	136	176	216	256	296	0
	LB	16	56	96	136	176	216	256	0
F 3	UB	62	102	142	182	222	262	296	0
	LB	22	62	102	142	182	222	262	0
F 4	UB	41	81	121	161	201	241	281	296
	LB	1	41	81	121	161	201	241	281
F 5	UB	72	132	192	252	276			
	LB	12	72	132	192	252			

Table 2 – Upper and lower bound of the integer part of the decision variable

With this information it is possible to compute the total possible cases for the decision variable, which is around 2.70*10^65 cases. As it is possible to conclude it is a gigantic amount of cases to analyze the problem using a mathematical deterministic program.

3. Implementation of the genetic algorithm

Genetic algorithms (GA) are evolutionary randomized search heuristic algorithms based on natural selection processes presented in biological evolution with the objective of finding solutions in optimization problems [6]. Genetic algorithms were introduced by John Holland in 1975 [7] and they apply the principle of “survival of the fittest”. Instead of searching just one solution at a time, this type of algorithm works with a population of individuals (group of solutions).

As it was said earlier, in this article two types of GA will be applied: the single objective and the multi-objective (NSGA-II). Since the model was already define, it will now be explained what the variables of the GA are and how they will influence the algorithm.

Firstly, it is necessary to know how both algorithms are going to work. The GA is an iterative algorithm of random search, as it was said earlier, and to explain how their work cycle it is only necessary to analyze the figure 9:

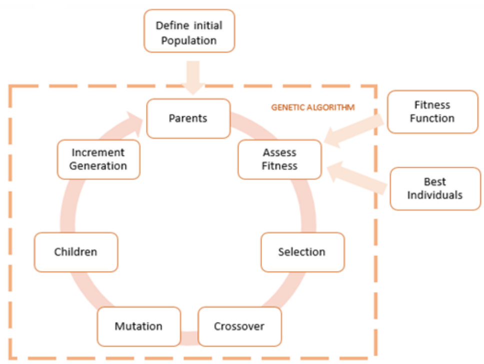

Figure 9 – Genetic Algorithm basic cycle

The multi-objective system is a little more complex but still can be summarized in 6 steps [9] [10]:

- Step 1 :Population initialization

In this step, the population is initialized based on the desired range (number of individuals) and constraint (lower and upper bounds of the decision variable).

- Step 2 :Non dominated sort

The initial population is sorted based on a non-domination criteria.

- Step 3 :Crowding distance

After sorting the population, the crowding distance is assigned front wise.

- Step 4 :Selection

The individuals in the population are selected based on their rank and crowding distance and it's carried out using a binary selection with a crowded-comparison operator (mechanism of selection).

- Step 5 :Genetic Operators

In this step, the population is crossed and mutated. For the crossover, it's used a simulated binary crossover and, for the mutation, it's used a polynomial mutation.

- Step 6 :Recombination and selection

The new population and the current generation population are now combined and the individuals of the next generation are selected after this combination. The new generation is filled by each front subsequently until the population size exceeds the current population size. After this, the steps 4 to 6 are repeated until it achieves the stopping criteria (which is the number of desired generations).

Since this algorithm is a multi-objective algorithm, this means that it has 2 or more objectives at the same time. For a result to be feasible, it has to verify all the constraints. But how can it be confirmed that a certain feasible result is better than another feasible result? To determine that, it must be identified the non-dominated region with the best results of the program (the Pareto-optimal front) [10]. (in figure 10 it can be seen an example of a Pareto-optimal front for a 2-objective function).

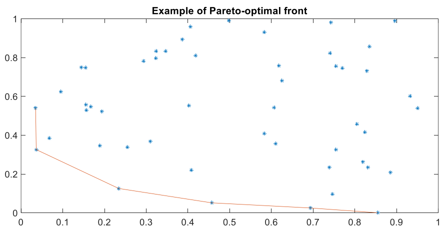

Figure 10 – Example of Pareto-optimal front

Since it is now known how both algorithms work, it will be enumerated the various variables that influence the algorithm performance and how they vary.

- Selection

Selection is a function of the genetic algorithm where the parents are chosen to generate the offspring of the next generations through crossover and mutation. For the selection to be made, firstly, the individuals are "ranked" regarding their fitness value. After that, the selection is made, according to the desired method. Both the Global Optimization Toolbox and the NSGA-II use the tournament selection (both programs used a size of 2 participants)

- Elitism

Carrying over the best individuals from one generation to another without any change is called elitism. For the Global Optimization Toolbox, used in Matlab, it is possible to customize the elitism property adapted to integer problems (Elite Count), by choosing the number of best individuals that pass from one generation to another unaltered. This parameter does not exist in the NSGA-II algorithm, since it is difficult to define the best individuals of the generation.

- Crossover

Crossover is a genetic operator responsible for the commutation of genes between the chromosomes.

The Global Optimization Toolbox uses a non-customizable special crossover function, when applied with integer variables. Still, it is possible to customize (in both programs) the fraction of individual subjected to crossover (crossover fraction). In NSGA – II, it is possible to customize the crossover function, i.e. change the type of function and the value of the child will be calculated as such:

$$child1 = parent1 + randn*ratio*(parent2 - parent1) \quad (1)$$

$$child2 = parent2 - randn*ratio*(parent2 - parent1) \quad (2)$$

- Mutation

Mutation is another genetic operator and its objective is to randomly change genes in chromosomes. Since mutation can create new solutions non-related to their parents (contrarily to what happens with crossover), its solutions are considered to be global exploration. Due to this characteristic, mutation ensures that the population does not get stuck in a permanent local solution [8].

As it was said earlier, the Global Optimization Toolbox uses a non-customizable special mutation function, when applied with integer variables. Still, it is possible to customize (in both programs) the fraction of individual subjected to mutation (mutation fraction). In NSGA-II, besides the mutation, it is possible to change the mutation function and the value of the child will be calculated as such:

$$child = parent + S*randn*(UB-LB) \quad (3)$$

$$S = scale*(1-shrink*currGen/maxGen) \quad (4)$$

- Creation

Creation is the last parameter that will be referenced in this section. This parameter defines how the first population of the program is created. Since the dimension of this problem is large and has a lot of constraints, it is very difficult for the program to create and/or find feasible solutions from nothing. Knowing that, it will be used in both programs a function that will load an initial.

4. Results and discussion

After a quick analysis of the GA and the model, it is now possible to start running the program. In this case, the first thing that was done was run the program and change each one of the parameters of both algorithms. In summary, it was done 10 to 20 points per test (in this work, a test is an experiment where it was frozen all the parameters expect one) and each experiment was run 30 times (so that it was achieved a good sampling). The tests done are summed in tables 6 and 7 (for the GA the time test was executed twice: the first with the graphics of the GA running and the second time without the graphics so that it was possible to see the influence of the graphics on the program).

Single Objective GA 2D test				
Test done	Generation	Population	Elitism	Crossover value
Time	1000	Var (14X)	0.05	0.75
Time	Var (11X)	150	0.05	0.75
Time and Profit	600	150	0.05	Var (11X)
Time	600	150	Var (11X)	0.75
Single Objective GA 2D test				
Profit	Var (11X)	Var (14X)	0.15	0.75
Profit	600	Var (14X)	Var (11X)	0.75

Table 6 – Tests done to the single objective GA

Multi-Objective GA (NSGA-II) 2D test							
Test done	Generation	Population	Crossover fraction	Mutation Fraction	Crossover Value	Mutation scale	Mutation shrink
Time	1000	Var (14X)	0.75	2/72	1.7	0.9	0.7
Ti m	Var (11X)	150	0.75	2/72	1.7	0.9	0.7
Ti m	600	150	Var (11X)	2/72	1.7	0.9	0.7
Ti m	600	150	0.75	Var (11X)	1.7	0.9	0.7
Ti m	600	150	0.75	0.15	1.7	Var (11X)	0.7
Ti m	600	150	0.75	0.15	1.7	1	Var (11X)
Time and Profit	600	150	0.75	0.15	Var (11X)	1	0.1
Multi-Objective GA (NSGA-II) 3D test							
Profit	Var (11X)	Var (14X)	0.75	2/72	1.7	0.9	0.7
Profit	600	150	Var (11X)	Var (11X)	1.7	0.9	0.7
Profit	600	150	0.75	0.15	1.7	Var (11X)	Var (11X)

Table 7 – Tests done to the single objective GA

The results of the tests were analyzed considering two important characteristics: the Cost Reduction and the Time needed to execute the program. For every test the variables were all fixed except one or two (the variables that were changing where represented by, for example Var (14X), which means that this parameter was a variable for a specific test and it was tested 14 different points).It can be seen from figures 11 to 20 the result obtained from this tests

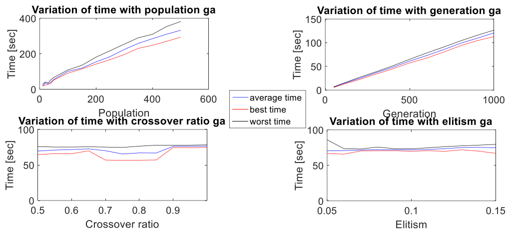

Figure 11 – Time results of the single objective GA

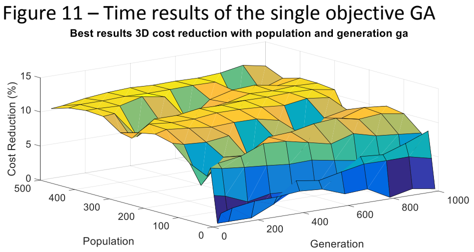

Figure 12 – Profit results of 3D population and Generation parameters of the single objective GA

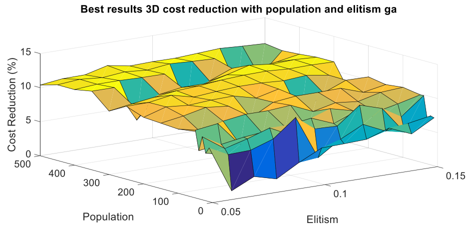

Figure 13 – Profit results of 3D population and elitism parameters of the single objective GA

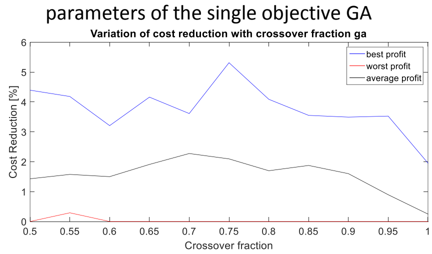

Figure 14 – Profit results of 2D crossover value parameter of the single objective GA

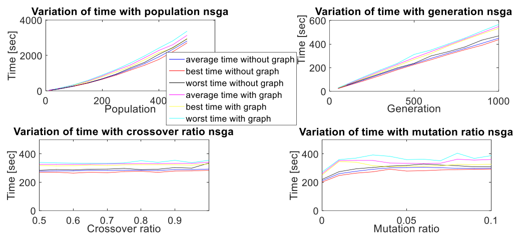

Figure 15 – Time results of the multi-objective NSGA-II (part A)

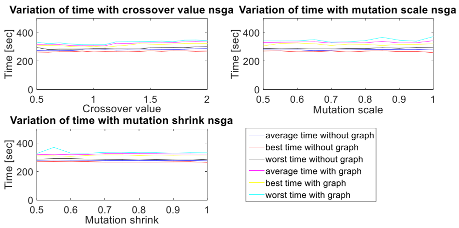

Figure 16 – Time results of the multi-objective NSGA-II (part B)

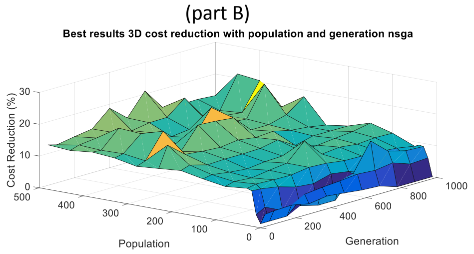

Figure 17 – Profit results of 3D population and Generation parameters of the multi-objective NSGA-II

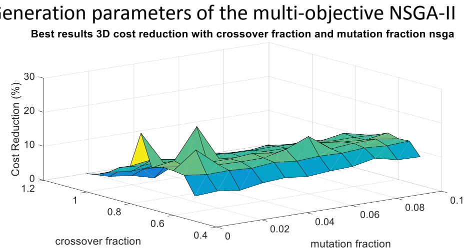

Figure 18 – Profit results of 3D crossover and mutation fraction parameters of the multi-objective NSGA-II

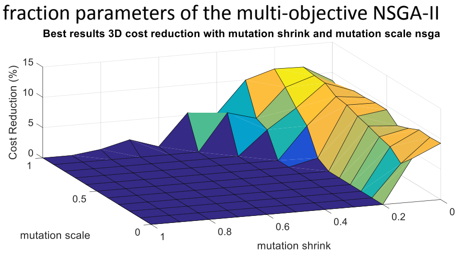

Figure 19 – Profit results of 3D mutation scale and mutation shrink parameters of the multi-objective NSGA-II

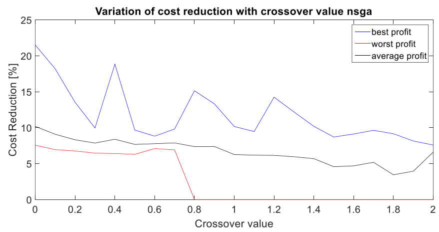

Figure 20 – Profit results of 2D crossover value parameter of the multi-objective NSGA-II

As it can be observed in the graphics, the single objective GA is much faster than the NSGA-II, almost 10 times. It can also be observed that the NSGA-II can have, in most of the cases, twice the profit from the initial plan, when compared with the single objective GA. In terms of the algorithm itself, the crossover and mutations variables do not vary the time result obtained (except the crossover and mutation ratio when equal to 0, because the program will run a lesser function). The generation and the population vary linearly with the time. In terms of results, the parameter that is more sensible in the algorithm is the mutation (if it's too high, the program will never converge and if it's to low, the program will not have the capacity to get out of local minimums). The generation and the population vary linearly with the profit. The crossover and the mutation variables have a certain parabolic behavior with the profit (there is a certain value or group of values where these parameters give a higher profit).

Now it will be compared the highest profit situation of the NSGA-II with the initial condition. Tables 8 and 9 represent the plans for the initial fabric plan and the plan created by the NSGA-II.

With these tables, it is almost impossible to understand what the changes made from the initial plan to the new plan influenced the electricity tariff. The figures from 21 to 29 will aid in understanding it.

Fabric Plan				
F1	F2	F3	F5	F4
00:30 Mon	10:30 Mon	13:30 Mon	08:00 Mon	06:30 Mon
21:30 Mon	06:30 Tue	10:30 Tue	04:00 Tue	15:30 Tue
06:30 Tue	07:30 Wed	10:30 Wed	05:00 Wed	00:30 Thu
16:30 Wed	03:30 Thu	06:30 Thu	01:00 Thu	07:30 Fry
13:30 Thu	01:30 Fry	04:30 Fry	23:00 Thu	14:30 Sat
10:30 Fry	22:30 Fry	01:30 Sat	20:00 Fry	
07:30 Sat	18:30 Sat		16:00 Sat	

Table 8 – Initial plan

NSGA-II plan				
00:30 Mon	15:00 Mon	17:30 Mon	05:00 Mon	10:30 Mon
21:30 Mon	12:00 Tue	14:30 Tue	05:00 Tue	20:00 Tue
18:30 Tue	08:00 Wed	13:30 Wed	01:00 Wed	02:30 Thu
16:30 Wed	04:00 Thu	09:30 Thu	21:00 Wed	08:30 Fry
16:00 Thu	00:00 Fry	05:30 Fry	20:00 Thu	17:00 Sat
13:30 Fry	22:30 Fry	06:00 Sat	20:00 Fry	
15:30 Sat	22:30 Sat		19:30 Sat	

Table 9 –NSGA-II plan

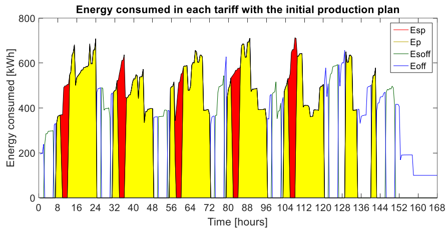

Figure 21 – Energy consumed in each tariff period of the initial plan

	Esp	Ep	Eoff	Esoff
Maximum Value [kWh]	712,53	710,29	656,29	592,96
Mean Value [kWh]	545,95	511,03	328,22	431,30
NSD	0,1441	0,1960	0,4964	0,2096
Total Energy [kWh]	16378,57	78698,65	31509,45	24152,96

Table 10 – Relevant information removed from figure 21

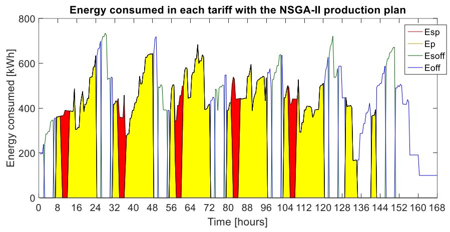

Figure 22 – Energy consumed in each tariff period of the NSGA-II plan

	Esp	Ep	Eoff	Esoff
Maximum Value [kWh]	538,31	683,54	718,82	734,25
Mean Value [kWh]	417,34	451,80	382,90	530,09
NSD	0,1195	0,2263	0,4837	0,2447
Total Energy [kWh]	12520,27	69576,67	36758,49	29685,29

Table 11 – Relevant information removed from figure 22

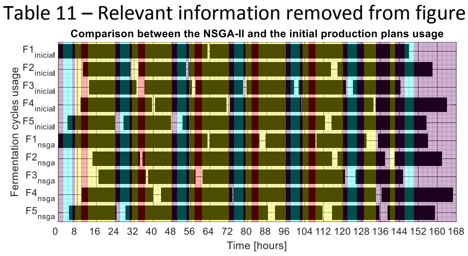

Figure 23 – Comparison of fermenters usage between the NSGA-II plan and the initial plan

		Hsp	Hp	Hoff	Hsoff
Initial plan	F1	10,20%	51,70%	20,41%	17,69%
	F2	10,18%	51,96%	23,57%	14,29%
	F3	8,75%	58,75%	17,50%	15,00%
	F4	9,83%	46,83%	27,33%	16,00%
	F5	10,71%	52,14%	21,79%	15,36%
	All fermenters	9,97%	52,01%	22,31%	15,71%
GA plan	F1	10,20%	46,94%	23,81%	19,05%
	F2	7,86%	48,57%	26,43%	17,14%
	F3	8,12%	56,88%	21,25%	13,75%
	F4	9,50%	47,50%	27,00%	16,00%
	F5	10,71%	50,00%	22,86%	16,43%
	All fermenters	9,33%	49,71%	24,39%	16,57%

Table 12 – Comparison of distribution of the fermentation cycles, in the tariffs, in percentage

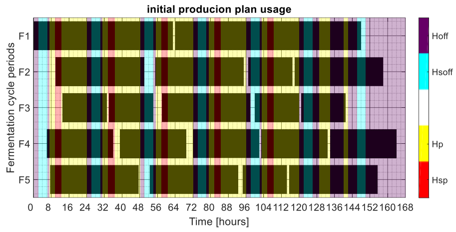

Figure 24 – Fermentation cycles plans of the initial plan

	Hsp	Hp	Hoff	Hsoff
F1	10,20%	51,70%	20,41%	17,69%
F2	10,18%	51,96%	23,57%	14,29%
F3	8,75%	58,75%	17,50%	15,00%
F4	9,83%	46,83%	27,33%	16,00%
F5	10,71%	52,14%	21,79%	15,36%
All fermenters	9,97%	52,01%	22,31%	15,71%

Table 13 – Distribution of the fermentation cycles, in the tariffs, in percentage

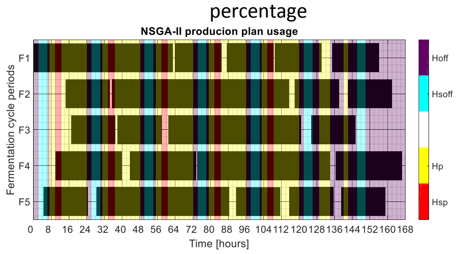

Figure 25 – Fermentation cycles plans of the NSGA-II plan

	Hsp	Hp	Hoff	Hsoff
F1	10,20%	46,94%	23,81%	19,05%
F2	7,86%	48,57%	26,43%	17,14%
F3	8,12%	56,88%	21,25%	13,75%
F4	9,50%	47,50%	27,00%	16,00%
F5	10,71%	50,00%	22,86%	16,43%
All fermenters	9,33%	49,71%	24,39%	16,57%

Table 13 - Distribution of the fermentation cycles, in the tariffs, in percentage

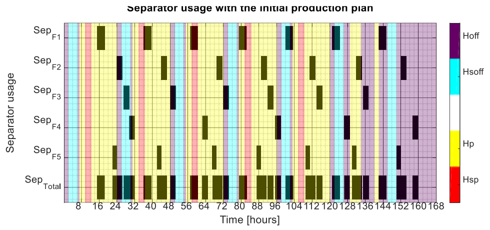

Figure 26 – Separators usage of the initial plan

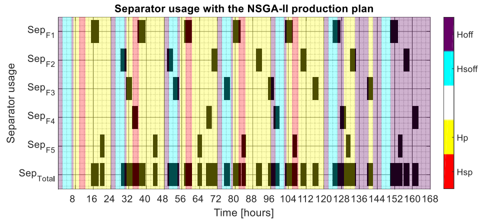

Figure 27 – Separators usage of the NSGA-II plan

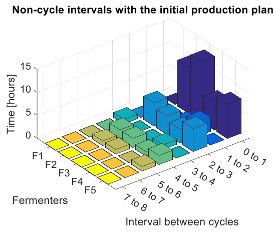

Figure 28 – Off-cycles of the initial plan

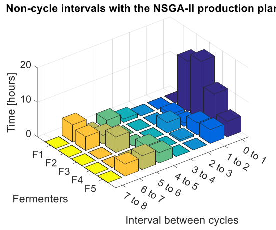

Figure 29 – Off-cycles of the NSGA-II plan

As it's observed in the tables 8 and 9 and also in the figures 23, 24, 25 and also in figures 28 and 29 (indirectly), two of the main differences between the initial plan and the NSGA-II plan are the fact that, in the initial fabric plan, there are more fermentation cycles being executed to close between each other (which will generate higher energy consumption in certain periods when compared to lower periods); when compared to the NSGA-II plan, and also in the initial plan that Sunday is practically not used, which is the contrary to what can be seen in the NSGA-II plan. Also, it can be observed that most of the allocations done in the NSGA-II were done with the objective of reducing at the maximum the energy consumed in the super-peak and peak hours, which will result in a lower tariff in the end of the week por this case it resulted in a cost reduction of 22,20%, the equivalent to over 3000€. It can also be observed in pictures 26 and 27 that as it was expected, in any period of the week was used 2 separators at the same time. It also can be observed that the program maintained the same number of fermentation cycles done by each fermenter, which can be related to the fact that the initial plan given to the program was the only feasible solution that the program had in the first set of solutions, and in the following generations, the program converged to a set of solutions that had the same behavior.

5. Conclusions

The main conclusions that were obtained with these tests were that it is possible to optimize the fabric plans developed by this factory with both of the algorithms used. Also, it was possible to conclude that the single objective GA is much faster to execute than the multi-objective NSGA-II program, most likely due to the higher complexity of the NSGA-II program. Although the single objective GA is faster, the best results of the simulation were found in the multi-objective NSGA-II program which can be explained, not only because of its higher complexity but also due to the fact that the single objective GA is more limited in the definition of its variables (a limitation that exists with his library when inserted in the algorithm binary variables).

6. References

1. Mazinan, A. H., and A. R. Khalaji. "A comparative study on applications of artificial intelligence-based multiple models predictive control schemes to a class of industrial complicated systems." *Energy Systems* 7.2 (2016): 237-269.

2. Guo, Z. C., and Z. X. Fu. "Current situation of energy consumption and measures taken for energy saving in the iron and steel industry in China." *Energy* 35.11 (2010): 4356-4360.

3. Rodriguez-Gonzalez, Oscar, et al. "Energy requirements for alternative food processing technologies—principles, assumptions, and evaluation of efficiency." *Comprehensive Reviews in Food Science and Food Safety* 14.5 (2015): 536-554.

4. Roos, Y. H., Fryer, P. J., Knorr, D., Schuchmann, H. P., Schroën, K., Schutyser, M. A., and Windhab, E. J. (2015). Food Engineering at Multiple Scales: Case Studies, Challenges and the Future—A European Perspective. *Food Engineering Reviews*, 1-25.

5. Therkelsen, P., Masanet, E., & Worrell, E. (2014). Energy efficiency opportunities in the US commercial baking industry. *Journal of Food Engineering*, 130, 14-22.

6. The MathWorks Inc. Global Optimization Toolbox. http://www.mathworks.com/products/global-optimization [12/05/2014].

7. J. Holland. "*Adaptation in Natural and Artificial Systems*". MIT Press, 1992.

8. M. Mitchell. "*An Introduction to Genetic Algorithms*", 5th edition, A Bradford Book The MIT Press, 1999.

9. Yusliza Yusoff, Mohd Salihin Ngadiman, Azlan Mohd Zain "Overview of NSGA-II for Optimizing Machining Process Parameters" 2011 Published by Elsevier Ltd

10. Srinivas, N. and Deb K, Sundar. "Multi-Objective function optimization using non-dominated sorting

genetic algorithms", Evolutionary Computation, vol. 2, no. 3, pp. 221-248, 19