Experiments for identical parallel machine scheduling with bee algorithm

Ing. George Cristian Gruia

Supervisor: Doc. Ing. Michal Kavan, CSc.

Abstract
In this paper a case of identical parallel machine scheduling is presented and based on the swarm intelligence of honey bees, a Bee Preventive Quality Assurance System is introduced. Our goal is to maintain the quality of work and consequently of the product, within certain limits, prior defined, according to the total costs and makespan of the foraging operations. Our purpose is to see if a manufacturing process can be managed by a computer programme with only some input data from user-defined intervals, and when we can rely on such a situation, where quality of work, time and productivity maximization play an important and strategic role in the company. The complex issue of scheduling is designed through a mathematical model and its viability is successfully tested using a 2^4 full factorial experiment. The paper shows partial results of the author's research and was elaborated within the project SGS13/191/OHK2/3T/12.

Keywords: *identical, parallel machine scheduling, bee algorithm, manufacturing*

1. INTRODUCTION

The automated machines and robots must answer the productivity level imposed by globalization and serial production. Accordingly we can optimize the production process by minimizing the production time of n products or semi-products on m parallel machines/ industrial robots, but with a predefined level of quality imposed by the customers.
We can define a classical parallel machine scheduling as a class of problems of scheduling j jobs J = {1, 2, … , j} on m identical, uniform, or unrelated parallel machines M = {1, 2, … , m}with an objective of minimizing an objective function. Our target is to minimize the time and consecutively the operational costs related to the operations on these machines, but also to produce our goods at the required quality level. This can be translated in a problem of improving the makespan with the additional increasing of the quality level. We further denote this problem as MAKEMAX.
Each job $k \in J$ and each machine $l \in M$ has p processing times t_{kl}, a weight w_k, a due date d_k, a release date r_k and a required level of quality q_k which today in most of the companies must be within quality limits of 6σ, i.e. only 3.4 defects are allowed per 1 million products. A job cannot start before its release date and pre-emption is not allowed.
When studying the parallel machine scheduling we should divide the main problem in three sub-problems:

- **A.** For the parallel identical machine case, all the machines have the same processing speed and thus the processing times of a job are identical on different machines, i.e. $t_{kl} \equiv t_k$, t_{kl} is the actual processing time of job *k* if processed on machine *l*.
- **B.** For the uniform machines, we can have different speed; hence the processing times of a job may differ by speed factors, i.e. $t_{kl} \equiv \frac{t_k}{s_k}$, where s_k is the speed of machine *k*.

C. In the case of unrelated machines, t_{kl} is considered arbitrary without any characteristics.

In order to fulfil the required level of quality, an inspection should be made to the products or semi-finished products, but taking in consideration the serial production, the quality inspection is automated, made by sensors and an inspection time is considered t_{ik} for each job $k \in J$.

The tardiness of a job $k \in J$ is defined as $T_k = \max\{0, C_k - d_k\}$ and the earliness is defined as $E_k = \max\{0, d_k - C_k\}$, where C_k is completion time.

In this paper we consider the first case, i.e. the case of identical parallel machine scheduling, where the processing speed and processing times are the same. Even if, this is the simplest case, it is a NP-hard problem and cannot be solved in polynomial time. We develop a heuristic approach inspired from the honey bees' life and way of foraging nectar in the sunny summer days.

Bees, like ants and other insects are social insects and have an instinct ability known as swarm intelligence, which enables them to solve complex problems of the group, beyond capability of individual members by functioning collectively and interacting with each other amongst members of the group ((Nakrani and Tovey 2004, Teodorovic and Dell'orco, 2005). For the honey bees this intelligence is crucial due to the complexity of finding flowers and collecting nectar and pollen, in a relatively short period of time, related to the life of a honey bee.

Honey bee algorithm has been used in other scheduling problems like Job Shop Scheduling, but we are not aware of being used for identical parallel machine scheduling.

An analogy is made between the working bees of a hive, which have to get out of the hive and search for new flowers and parallel machine scheduling. Let H be the set of honey bees searching for flowers and F the set of possible flowers and food sources where a bee can start and finish her job of foraging nectar.

The hive must be productive, thus the goal is to find a way of scheduling the set of identical bees H (with the same processing speed) to the set of foraging jobs on F flowers in the fastest possible way, to maximize the output by minimizing the input. An analogy is made in this way to the "identical parallel machines", but the quality of the pollen and nectar is a function of the bee ability to forage and the path taken, i.e. the shortest path ensures the biggest amount of pollen brought back to the hive in a certain time period, but not necessarily the highest quality of the foraged pollen and nectar. The colony's main goal is to produce honey with the highest quality and an optimum solution is to maximize the quantity of this type of honey.

This will be also our goal.

In this way we define a mathematical model where the bees are identical parallel machines and the flowers represent jobs which should be processed by the bees.

2. MATHEMATICAL MODEL

Before defining the mathematical model some restrictions are applied due to the analogy with the honeybees, as follows:

- Each working bee has a random defined loading capacity lc_i of pollen and nectar (further referred as "food");
- Each flower in the flower patch, should be visited by the working honey bee only once and when returning to the hive will carry a certain amount of food between 0,01 and 1, where 1 is the maximum loading capacity of the bee equal to the maximum amount of available food in the visited flower;
- A flower patch consists of a random number of flowers from minimum 1 to a maximum of the number of all the given flowers.

- The colony is divided in scouts (which is a defined number of 10% from the total number of bees), which randomly search for food, working bees (80% of the population) which wait in the hive for scouts to return and after getting to the flower, they also become scouts for the flower patch where were sent initially by the scouts; and onlookers (10% of the population) which stay in the hive and perform maintenance and “final” operations to the final product (honey);
- The scouts after finding a flower, return to the hive with foraged nectar and pollen and using waggle dance recruit the working bees, until all the flower patches are discovered. After that they transform into working bees and according to their visited flowers remain to harvest the flower patch which is the richest in food;
- After unloading the food in the hive, a recovery time *tr* is considered as a sum of: the prepare time *tp*, the bee prepares itself for the second flight (fixed input costs are related with this) and waggle dance time *td*, it shows its findings to the other colleagues by waggle dance, for other bees to follow and forage the flower patch as soon as possible, where *tr =tp + td*
- A bee can fly only on a maximum 3 km radius around the hive, with an average speed of 30 km/h, that is a forage trip can last at most (720 + t_f) seconds, where t_f is the foraging time.
- Each flower within the visited flower patch is assessed by the onlooker bees with a certain priority or weight w_k according to the duration of the waggle dance of the working bee returned to the hive.
- The jobs are done without pre-emption or re-assignment.

We will further consider only the case of the scouts with working bees in their search for food.

We can characterize each bee by a flying time, defined as eq. (7) and a weight w_k as eq. (5). Each flower is foraged by a certain bee within a certain period of time, i.e. a processing time t_f which is the same time for all the bees in this particular case, a tardy weight tw_k , an early weight ew_k and an optimistic due date d_k^o as well as a pessimistic due date d_k^p and a minimal flying cost fc_i (the amount of pollen and nectar foraged is lower than the actual amount brought in the hive because a part is consumed by the bee according to eq.(14)).

Rebai [3] introduces a function for total costs but nothing is mentioned about the quality of the work. We will further consider the same cost function but in a different environment. According to our MAKEMAX model, we want to improve the quality of the product within the earliest processing time, i.e. the optimistic due date, but due to different factors we have another pessimistic due date which we want to minimize. From the nectar and pollen foraged, honey bees produce the final product, the honey, and assuming that the available resources are of the best quality in different amounts, and that each bee is “trained” to produce the honey at the beekeeper’s quality requirements, we want to measure the quality of the work of the honey bees, which should be at the highest levels and thus we start to measure it from the time t = 0. Thus we introduce a Bee Preventive Quality Assurance System (BPQAS), so that counteractive measures can be taken in due time in order to solve possible problems.

Generally the costs are divided in direct and indirect costs. We consider in this paper, the costs related to the individuals’ work (honey bees) and we will work under the assumption that the time–cost trade-offs for project activities are linear (Swink [4]).

When BPQAS is put to work, within an optimistic d_k^o and pessimistic due date d_k^p , the operational cost associated is minimum and is the same as the flying cost fc_i. On the other hand, if the BPQAS is put to work before the optimistic due date d_k^o (which is the smallest amount of time in which the job is fulfilled), we will have a new total cost as follows:

$tc_i = fc_i + TR(d_k^o - d_s)$,where d_s is the point from when we start implementing the BPQAS. An explication for this expression can be given, that the bee begins to consume honey before arriving to the flowers.
If the foraging time is greater than the pessimistic due date, the total costs are increasing according to the following relationship:
$tc_i = fc_i + TR(d_f - d_k^p)$, where d_f is the actual finish time of the forage and TR is the time rate of the job with the unit CZK/s..

Definition 1 – A foraging operation is equivalent with a manufacturing operation and is considered complete when the input (nectar and pollen) is transformed to output (honey).
In order to be able to measure the efficiency of the honeybee's work we introduce the following coefficient:
$k_i = \frac{quality\ of\ the\ i-th\ bee}{(processing\ time+recovery\ time) of\ the\ i-th\ bee}$, where processing time is the time taken for flying away from the hive to the flower, foraging the pollen and nectar and returning to the hive, and together with the recovery time form the makespan for the manufacturing of the final product, i.e. honey. In this manner we want to obtain big values of the coefficient which can be done in three ways:

- By increasing the quality of the work (assuming that the available nectar and pollen is of the best quality, i.e. the raw input materials are according to the standards)
- By decreasing the makespan, i.e. the completion time of the last job in the schedule, but which must correspond to the working conditions and standards (the recovery time cannot be reduced to zero).
- By increasing the quality with the additional decreasing of the processing time.

In this article we focus on the third way of increasing the value of coefficient k_i.
Our model is a system of equations where we should increase quality of the work, decrease costs and decrease total makespan of the colony and can be translated as a minimizing criterion of function F:

$$F = \frac{\sum_{i=1}^{h}\sum_{k=1}^{f}(ew_{ki}E_{ki}y_{ik} + tw_{ki}T_{ki}z_{ik} + w_{ki}C_i)}{\sum_{i=1}^{h}\sum_{k=1}^{f}\frac{q_i}{tc_i * (E_{ki} + T_{ki} + C_i)}} = min$$

Subject to:

$$E_{ki} = \begin{cases} d_k^o - d_f, d_k^o > d_f \\ 0,\ otherwise \end{cases} \quad (1)$$

$$T_{ki} = \begin{cases} d_f - d_k^p,\ d_f > d_k^p \\ 0,\ otherwise \end{cases} \quad (2)$$

$$ew_{ki} = \begin{cases} (priority\ according\ to\ rule\ X) + 1,\ C_i < d_k^o - d_s \\ 0, C_i \in [d_k^o; d_k^p] \end{cases} \quad (3)$$

$$tw_{ki} = \begin{cases} (priority\ according\ to\ rule\ X) - 1, C_i > d_k^p - d_s \\ 0,\ C_i \in [d_k^o; d_k^p] \end{cases} \quad (4)$$

$$w_{ki} = \begin{cases} priority\ according\ to\ rule\ X, C_i \in [d_k^o; d_k^p] \\ 0,\ otherwise \end{cases} \quad (5)$$

$$C_i = f_i + t_p + t_d, where\ C_i \in [d_s; d_f],\ \ t_p \in \left[\frac{f_i}{4}; \frac{f_i}{3}\right],\ \ t_d = \frac{f_i}{10}, t_f \in [d_s; d_f] \quad (6)$$

$$f_i \in [t_f, 720 + t_f] \quad (7)$$

$$y_{ik} + z_{ik} = 1, \forall\ i = 1,2, \dots, h\ and\ k = 1,2, \dots, f, i \neq k \quad (8)$$

Taking in consideration the experience from the workers production lines we define the quality as a quadratic function, with the variable x_i, taking the form as follows:

$q_i = {x_i}^2 + x_i + 1$, where $x_i = \sum_{k=1}^{f} priority_k \; lc_i \; \frac{1}{c_i}$ (9)

and $i \in \{1, 2, \dots, h\} is\ the\ set\ of\ honeybees\ and$

$j \in \{1, 2, \dots, f\} is\ the\ set\ of\ available\ flowers\ on\ a\ radius\ of\ 3km\ around\ the\ hive$

The loading capacity of i-th bee is defined as $lc_i \in (0; 1]$ (10)

The overall quality of the work of the honeybees should not exceed 3.4 defects per million opportunities, according to the 6σ level. Thus after computing the minimum of the stated above function, different quality levels with corresponding total costs and makespan are analyzed using graphical representation and using regression and correlation analysis, we find the trend of our data with the purpose to see how the distribution looks like within the upper and lower limits, which in our case are the optimistic respectively the pessimistic due dates of the foraging jobs.

The priority of each flower is function of the waggle dance, which at its turn is a function of the dancing time on the dance floor. But also prioritizing the flowers which are to be foraged increases the chance of profitability of the individual and of the colony. In this consideration, according to the priority from the waggle dance, each bee assesses the flower visited before the dance and after looking at the new source of food, showed by its colleagues. Thus the weight of a foraging job is divided into the normal, early and tardy weight, which are assessed together and correspondingly a new set of priority is set for the bee, before engaging in another foraging expedition.

We use two binary variables y_i and z_i:

$$y_{ik} = \begin{cases} 1, for\ E_k > 0 \\ 0, otherwise \end{cases} \quad (11)$$

$$z_{ik} = \begin{cases} 1, T_k > 0 \\ 0, otherwise \end{cases} \quad (12)$$

$$\begin{cases} tc_i = fc_i + TR(d_k^o - d_s), for\ t_f \in [d_s; d_k^o] \\ tc_i = fc_i, \qquad for\ t_f \in [d_k^o; d_k^p] \\ tc_i = fc_i + TR(d_f - d_k^p), for\ t_f \in [d_k^p; d_f] \end{cases} \quad (13)$$

$$fc_i = 1\%\ lc_i \quad (14)$$

Eq. (1) and (2) define the earliness and tardiness, as a function of the finishing time and the optimistic and respectively pessimistic due date. Taking in consideration that the total costs decrease until the optimistic due date, remain constant until the pessimistic due date and then increase if the finishing time is bigger than the pessimistic due date, we assign tardy weights and early weight in such a way that we want to postpone as much as possible the finalization of a job with earliness, but on the other hand to speed up the finalization of the job with tardiness, as shown by eq. (3) and (4). However eq. (3), (4) and (5) must comply with the Corollary 1, i.e. if $ew_k = tw_k = w_k$, the jobs will be scheduled as follows: $w_k \succ tw_k \succ ew_k$.

According to Chong [5], a forager is more likely to randomly observe and follow a bee's waggle dance on the dance floor if the profitability rating is low as compared to the colony's profitability. We adapt in this manner the following Table 1 from Nakrani and Tovey [2]:

Table 1. *– Priority for the forager bee adapted from Nakrani and Tovey [2]*

Profitability rating	Probability of following the waggle dance	Priority for the forager bee
$Pf_i < 0.9Pf_{colony}$	0.60	4th
$0.9Pf_{colony} \leq Pf_i < 0.95Pf_{colony}$	0.20	3rd
$0.95Pf_{colony} \leq Pf_i < 1.15Pf_{colony}$	0.02	2nd
$1.15Pf_{colony} \leq Pf_i$	0.00	1st

The bee can choose only from a list of 4 possible flower patches according to the initial memorized source food, which found itself and after following the waggle dances of its colleagues, other bee workers. In case when the probability of following the waggle dance is equal to zero, the bee doesn't stay to dance for another round of the waggle bee and continues with the foraging of the initial food source, which found.

However in our case, we aim at introducing and maintaining quality of the produced product, thus we define the profitability index for a bee as follows:

$Pf_i = \frac{1}{tc_i * C_i}$ and the profitability index of the colony as $Pf_{colony} = \frac{1}{h}\sum_{i=1}^{h} Pf_i$, where C_i is the completion time of one bee measured between two consecutive flights, assuming that every time it performs the waggle dance.

It was considered the case when the working bees are working in a single 12 hours shift and accordingly the minimum for flying time is 1second and maximum for completion time is 43200 seconds.

Lemma 1: For a given number of m jobs which are to be sorted, according to n constraints, there exists a schedule where the 1st job has the nth constraint the same or belonging to the same set of similar vectors with the rest of the (m-1) jobs; the 2nd job has the (n-1) constraint the same or belonging to the same set of similar vectors with the rest of the (m-2) jobs, etc.

Proof: The proof is rather trivial and we will try to show it in a simple example. There are given m=4 jobs and n=3 constraints, i.e. total costs, completion time and quality of the job. Then for random values of these constraints between 1 and 10, we can choose a way of job sequencing according to our constraints as follows:

1st job – TC=4, C=2, Q=7;
2nd job – TC=3, C=4, Q=5;
3rd job – TC=7, C=3, Q=8;
4th job – TC=1, C=1, Q=10.

If we sort after the first constraint, minimum total costs, we have the following sequence:

$$4 \rightarrow 2 \rightarrow 1 \rightarrow 3$$

If we sort after the minimum completion time we have:

$$4 \rightarrow 1 \rightarrow 3 \rightarrow 2$$

If we sort after the maximum quality level of the jobs we have:

$$4 \rightarrow 3 \rightarrow 1 \rightarrow 2$$

We can see that the 4th job should be done first, then we can see that in 2 out of 3 existent cases 1st initial job should be done 3rd and 2nd initial job should be done last. Finally the 3rd initial job can be successfully made 2nd and the sequence after our rule X will be:

$$4 \rightarrow 3 \rightarrow 1 \rightarrow 2$$

Corollary 1: If the weight of a job J_i with finishing time between the optimistic and pessimistic due date, is the same as the early weight or tardy weight of different previous or later job, priority will have the job J_i, followed by the job with tardiness and finally the one with earliness.

3. COMPUTATIONAL RESULTS USING DOE

Due to the fact that the program takes a lot of time to compute the multiple combinations of solutions, we have decided to test our quality defined function, using Design of Experiments method in order to show how the quality of a bee is influenced by other external factors. We will further show initial computational results using the technique of Design of Experiments. In order to compute the quality of the i-th bee (qi), we need the following factors: priority (A), loading capacity (B), foraging time (C) and prepare time (D), while dancing time can be

easily computed if we know the foraging time, according to eq. (6). We can write the quality as following:
qi = A x B x C x D , and we can use a single replicate of $2^k=2^4$ factorial design, that is k=4 factors each at two levels.
The design matrix and the response data obtained from a single replicate of the 2^4 experiment are shown in Table 2 and Figure 1. The 16 runs are made in a random order, according to Table 3. We are interested in maximizing the quality level of the individual bee, and consequently of the colony, but also reduce the completion time and accordingly the costs related to the foraging job. We want to examine the magnitude and direction of the factor effects to determine which variables are likely to be important. We will begin the analysis of this data by constructing a normal probability plot of the effect estimates and to try to explain to what extent the model explains the variability (Fig.2) The table of plus and minus signs for the contrast constants for the 2^4 design are shown in Table 4. From these contrasts we can estimate the 15 factorial effects and the sum of squares as shown in Table 5.

Table 2. *– The quality of i-th bee experiment*

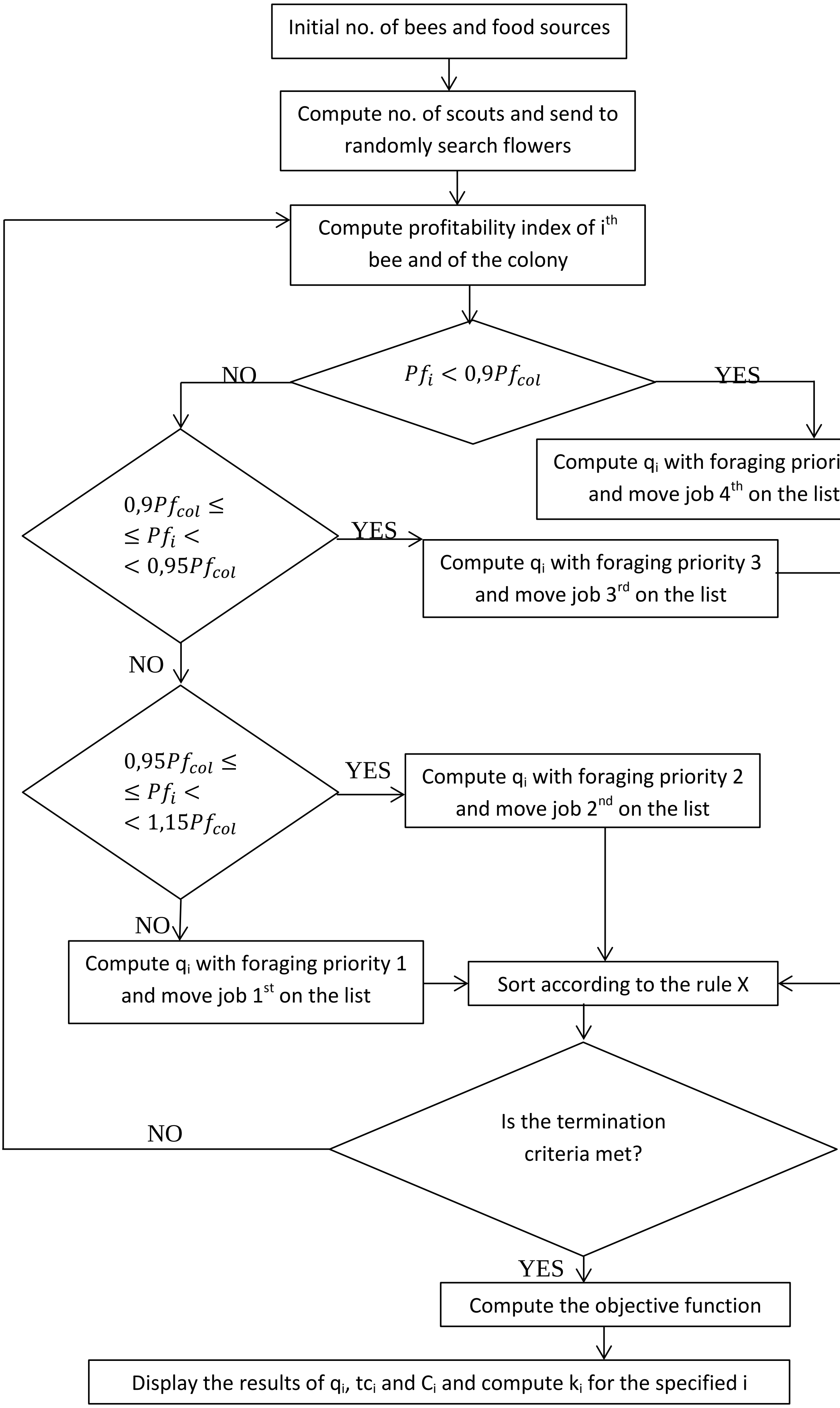
Run Number	Factor				Run label	Quality qi	xi
	A	B	C	D			
1	-	-	-	-	(1)	1,030507545	0,02962963
2	+	-	-	-	a	1,007462277	0,007407407
3	-	+	-	-	b	12,74211248	2,962962963
4	+	+	-	-	ab	2,289437586	0,740740741
5	-	-	+	-	c	1,000001207	1,2065E-06
6	+	-	+	-	ac	1,000000302	3,01625E-07
7	-	+	+	-	bc	1,000120665	0,00012065
8	+	+	+	-	abc	1,000030163	3,01625E-05
9	-	-	-	+	d	1,000003981	3,98105E-06
10	+	-	-	+	ad	1,000000995	9,95263E-07
11	-	+	-	+	bd	1,000398264	0,000398105
12	+	+	-	+	abd	1,000099536	9,95263E-05
13	-	-	+	+	cd	1,000000926	9,25927E-07
14	+	-	+	+	acd	1,000000231	2,31482E-07
15	-	+	+	+	bcd	1,000092601	9,25927E-05
16	+	+	+	+	abcd	1,000023149	2,31482E-05

Table 3. *– Data for the experiment*

Factors	Low(-)	High(+)
A=priority	4	1
B=load.cap.	1%	100%
C=fly time	1	30139,5
D=prepare time	0,25	10046,5

There are 15 degrees of freedom between the 16 combinations in the 2^4 design. Four degrees of freedom are associated with the main effects of A, B, C and D. Six degrees of freedom are associated with AB, AC, BC, AD, BD, CD interactions, four with ABC, ABD, ACD, BCD interactions and one with ABCD. We further consider estimating the main effects.

The flowchart of our algorithm is presented below:



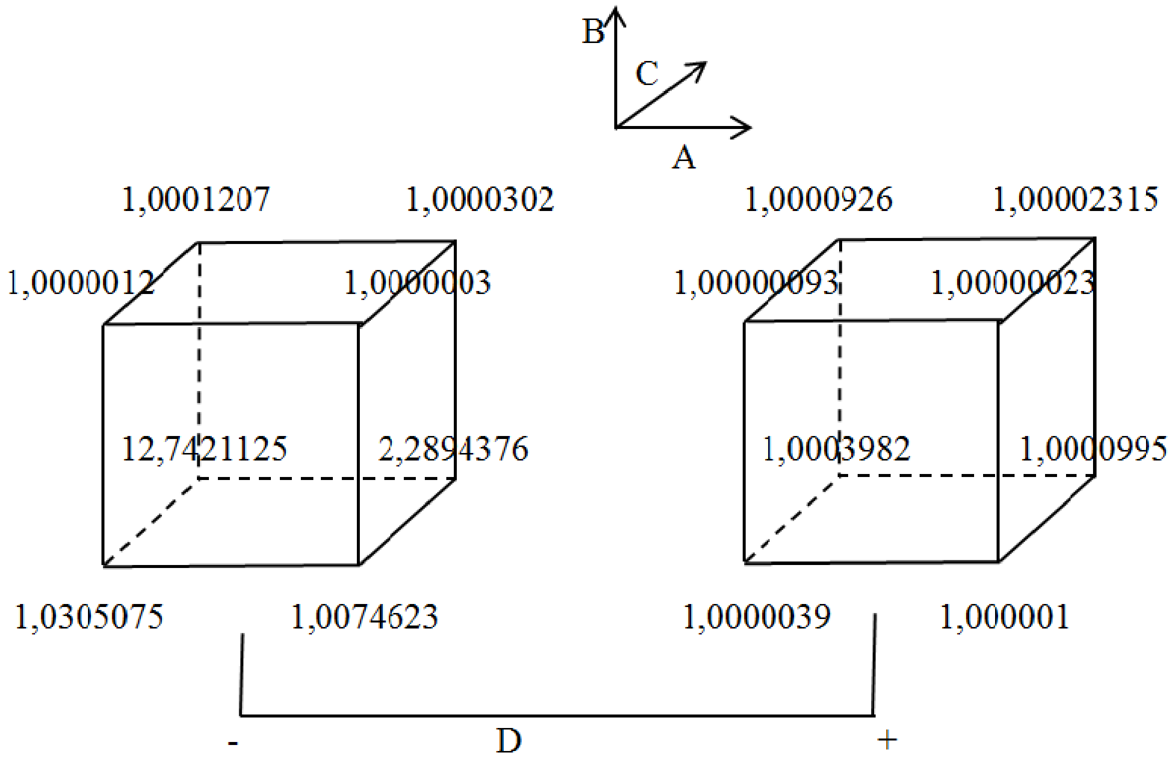


Fig. 1. *Data from our experiment*

Table 4. *– Contrast constants for the 2^4 design*

	A	B	AB	C	AC	BC	ABC	D	AD	BD	ABD	CD	ACD	BCD	ABCD
(1)	-	-	+	-	+	+	-	-	+	+	-	+	-	-	+
a	+	-	-	-	-	+	+	-	-	+	+	+	+	-	-
b	-	+	-	-	+	-	+	-	+	-	+	+	-	+	-
ab	+	+	+	-	-	-	-	-	-	-	-	+	+	+	+
c	-	-	+	+	-	-	+	-	+	+	-	-	+	+	-
ac	+	-	-	+	+	-	-	-	-	+	+	-	-	+	+
bc	-	+	-	+	-	+	-	-	+	-	+	-	+	-	+
abc	+	+	+	+	+	+	+	-	-	-	-	-	-	-	-
d	-	-	+	-	+	+	-	+	-	-	+	-	+	+	-
ad	+	-	-	-	-	+	+	+	+	-	-	-	-	+	+
bd	-	+	-	-	+	-	+	+	-	+	-	-	+	-	+
abd	+	+	+	-	-	-	-	+	+	+	+	-	-	-	-
cd	-	-	+	+	-	-	+	+	-	-	+	+	-	-	+
acd	+	-	-	+	+	-	-	+	+	-	-	+	+	-	-
bcd	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-
abcd	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

The effect of A when B, C and D are at the low level is [a – (1)]/n; the effect of A when B is at the high level and C and D at low level is [ab – b]/n; the effect of A when C is at high level and B and D at low level is [ac – c]/n; effect of A when D is at high level and B and C at low level is [ad – d]/n; effect of A when B and C are at high level and D at low level is [abc – bc]/n; effect of A when C and D are at high level and B at low level is [acd – cd]/n; effect of A when B and D are at high level and C at low level is [abd – bd]/n and finally effect of A when B,C and D are at high level is [abcd – bcd]/n. Thus the average effect of priority on the quality level of the work of the bee is the average as follows:

$A = \frac{1}{8n}[a - (1) + ab - b + ac - c + ad - d + abc - bc + acd - cd + abd - bd + abcd - bcd]$, where n is the number of replications

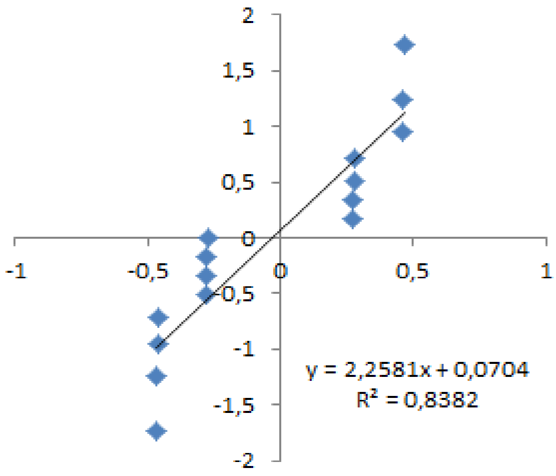
In a similar way we compute all the estimated effects and put them in Table 5.

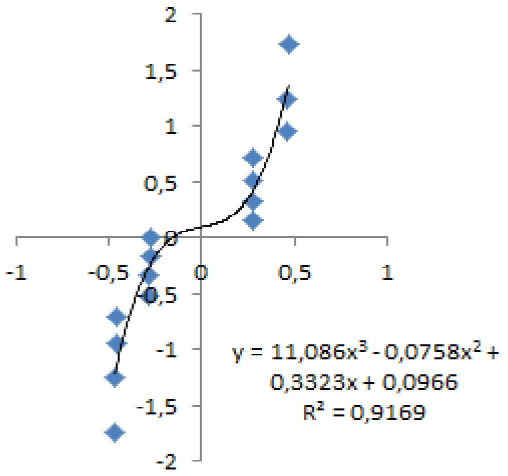
The column labeled “percent contribution” measures the percentage contribution of each model term to the total sum of squares (which is 8,453847). This is a rough but effective guide to the relative importance of each model term. We can see that factors C, D and B are the factors which influence the most the variable xi of the quality function of the bee accounting for 10,3% respectively 9,9% of the total variability.

We further analyze the results of the experiment in terms of a regression model and try to find to what extend our model explains the variability in the variable x_i of the quality function.

Table 5. *– Factor effect estimates and sums of squares for our 2^4 design*

Model term	Effect estimate	Sum of squares	Percent contribution
A	-0,28061	0,314976	3,726%
B	0,458428	0,840625	9,944%
C	-0,46762	0,87468	10,347%
D	-0,46753	0,874353	10,343%
AB	-0,27506	0,302625	3,580%
AC	0,280573	0,314885	3,725%
AD	0,280521	0,314767	3,723%
BC	-0,45836	0,840383	9,941%
BD	-0,45828	0,840068	9,937%
CD	0,467525	0,87432	10,342%
ABC	0,275017	0,302538	3,579%
ABD	0,274966	0,302424	3,577%
ACD	-0,28052	0,314755	3,723%
BCD	0,458267	0,840036	9,937%
ABCD	-0,27496	0,302413	3,577%





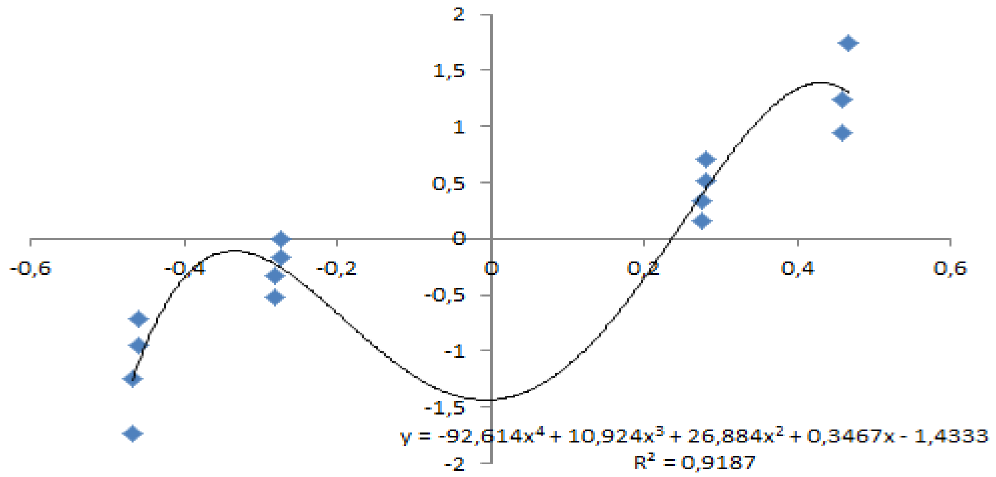


Fig. 2. *Normal probability plots of the effects for the 2^4 factorial with different regression functions of the expected values of x_i*

4. CONCLUSION

From the first plot of the Fig. 2 we can state that if we choose to use a straight line as a regression model for our experimental x's, we get that our model explains about 83,8% of the possible variability of x's when computing all possible solutions for finding the maximum quality of the honey bees' work. However if we choose a polynomial type function (as shown in the second and third plot) for x's we see that the new model explains about 91,7% to 91,9% of the variability in our process.

Also using Design of Experiments method we have showed that the factors influencing negatively the most our variable x_i within the quality function are C, D and their combination CD, which can also be seen from the graphs above.

To sum up we can say that the quality of the worker, assuming that the raw materials are according to our technical and economical specifications, is negatively influenced by the time spent for completing its job (including the recovery time). On the other hand if we want to increase the quality of the bee, we should choose only those bees which are carrying the highest amount of food back to the hive. This can be translated back in our daily business as a solution for increasing the productivity and optimizing the loading capacity in a logistical problem, which is, where our model can be successfully applied, inspired from the bees' swarm intelligence.

Accordingly our designed model is viable, as we have showed using a 2^4 full factorial experiment and we can continue in our research in implementing it in a programming language, where based on a competitive analysis we can decide which of the working bees from the population can work with a quality within 6σ limits. This paper represents partial results of the author's yet unpublished research.

Symbols

tc_i	total costs	(CZK)
fc_i	fixed costs	(CZK)
C_i	completion time	(s)
ew_{ki}	early weight	(-)
tw_{ki}	tardy weight	(-)
w_{ki}	weight within optimistic and pessimistic due date	(-)
lc_i	loading capacity of the bee	(-)

Pf_i	profitability index of the bee	($s^{-1}.CZK^{-1}$)
E_{ki}	earliness of the job	(s)
T_{ki}	tardiness of the job	(s)
d_k^o	optimistic due date	(s)
d_k^p	pessimistic due date	(s)
d_s	start time of the job	(s)
d_f	finish time of the job	(s)
t_d	waggle dance time	(s)
t_f	foraging time	(s)
t_p	prepare time for new job	(s)
f_i	flying time	(s)
q_i	quality of the work of the bee defined as a quadratic function of x_i	(s^{-2})
x_i	variable of the quality	(s^{-1})
TR	time rate of the job	($CZK . s^{-1}$)

References

[1] Teodorovic, D, Dell'orco, M, Bee colony optimization - A cooperative learning approach to complex transportation problems, In *Advanced OR and AI Methods in Transportation*, 2005, p. 51-60.

[2] Nakrani, S., Tovey, C., On honey bees and dynamic allocation in an internet server colony. In *Adaptive Behavior*, 2004, vol. 12, no.3-4, p. 223-240.

[3] Rebai, M., Kacem, I., Adjallah K.H., Scheduling jobs and maintenance activities on parallel machines. In *Journal of Operational Research*, 2012, [online], p.1-21.

[4] Swink, M., Talluri, S., Pandejpong, T., Faster, better, cheaper: a study of NPD project efficiency and performance tradeoffs. In *Journal of Operation Management*, 2006, vol. 24, p.542–562.

[5] Chong, C.S., Low, M.Y.H., Sivakumar, A.I., Gay, K.L., A bee colony optimization algorithm to job shop scheduling. In *Proceedings of the 2006 Winter Simulation Conference*, p.1954-1961.

[6] Montgomery, C.D., Design and analysis of experiments, 6th edition, John Wiley & Sons, 2005, pp.160-254, ISBN 0-471-48735-X.