Neural network for minimizing tricriteria objective function for machine scheduling problem

Faez Hassan Ali, Hanan Ali Chachan and Sharmeen Bakr Abdulkareem
Department of Mathematics, College of Science, Al-Mustansiryah University, Baghdad, Iraq
E-mail: faezhassan@uomustansiriyah.edu.iq

Abstract. In this paper, neural networks (NN) are implemented to manipulate a problem of machine scheduling for a single machine to minimize multicriteria objective function: maximum tardiness, maximum late work and total late work simultaneously ($T_{max}, V_{max}, \sum V_j$). Also the results of applying NN are compared with some exact methods, some heuristic methods and local search methods [7,14]. The used NN is learned by pack propagation algorithm for $n=3\times500$ jobs. NN technique was be found to be effective and used to select the best efficient and optimal schedule which minimizes the objective function.

1. Introduction
When the single Machine Scheduling Problem (MSP) concerns Non Polynomial-hard (NP-hard) problem, the computational time requirements are enormous for large sized problem, to avoid these drawbacks we can appeal to heuristics methods. In recent years, the improvement in heuristic methods has become under the name “local search methods (LSM)”. The local search provides approach high quality solutions to NP-hard problems of a realistic size in reasonable time. The LSM's start with an initial solution and then continually try to add better solution by searching neighborhoods [1].

The concept of Neural Networks (NN) came into being when the first model of neuron was created by two physiologists, Warren S. McCulloch and Walter H. Pitts in 1943 [2]. Willems and Rooda (1994) [3] looked at first formulating a job shop scheduling problem as an integer programming problem, and using a NN to solve the resultant integer programming problem. Foo and Takefuji (1998) [4], employed integer linear programming NNs to minimizing the total starting times of all jobs with a precedence constraint. Hamad et al., (2002) [5], presented a method for machine scheduling with common due date on single machine which depended on artificial NN. The objective of this problem is to minimize the total earliness and the total tardiness cost. The paper of Muralidhar and Alwarsamy (2013) [6] considers the problem of scheduling jobs on parallel machines with the combined objective to minimize the makespan, total tardiness and total earliness using NN technique. Abdul-Razaq and Ali (2014) [2] presented an approach for scheduling problem on single machines with multi objective to decrease the total completion of time and the total tardiness by using NN.

One of the important local search methods is Particle swarm optimization (PSO) which introduced by R. Eberhart and J. Kennedy in 1995. This method is obtained from the experiments with some algorithms that depend on the “flocking behavior” which are seen in some kinds of birds. PSO is an
optimization algorithm that lies under the soft computing umbrella which covers genetic algorithm (GA) and some evolutionary algorithms [7].

The Bees Algorithm (BA) is an optimization algorithm inspired by the natural foraging behavior of honey bees to find the optimal solution. The pseudo code for the BA in its simplest form is mentioned in [7].

In section 2 we discuss some important terminologies and notations for MSP. The NN is discussed in details in section 3. The MSP formulation of this paper is detailed in section 4, while the experimental results of applying NN, for the problems which will be discussed in this paper, are shown in section 5. In section 6 we discuss the analysis of the results of applying NN to solve MSP. Lastly, the conclusions are shown in section 7.

2. Machine Scheduling Terminologies and Notations
Most single MSPs (SMSP’s) assume that the processing times of jobs are known and have fixed values throughout the process, and SMSP’s has good learning effects form a new subfield in many areas like the scheduling [8].

In the classical scheduling models, most researchers consider that the job processing times are all constant numbers [9]. For decades, MSP researches will be more interest on one performance measures. The most real world with applications of scheduling, the researchers will be more interest in the multiobjectives which have received more attention in recent years [10].

Definition (1) [10]: The "optimize" term in a multicriteria decision making problem means that a solution with some objective which we can't improve it without worsening the other objectives.

Definition (2) [10]: Any feasible schedule $\sigma$ is called Pareto optimal, or nondominated (or efficient) with respect to the objective functions $f_1$ and $f_2$ if there is no different feasible schedule $\pi$ such that each $f_1(\pi) \leq f_1(\sigma)$ and $f_2(\pi) \leq f_2(\sigma)$. take in consider that at least one of the inequalities is strict.

The more common notations which used in scheduling are:

- $n$: number of jobs.
- $p_i$: Processing time of job $i$.
- $d_i$: Due date of job $i$.
- $C_i$: Completion time of job $i$.
- $T_i$: Tardiness of job $i$.
- $T_{\text{max}}$: max $\{T_i\}$.
- $V_i$: $\min\{T_i, p_i\}$.
- $V_{\text{max}}$: $\max\{V_i\}$.
- $\Sigma V_i$: Sum of $V_i$.

3. Neural Network
The most important neural networks is the Feedforward neural networks (FNNs) which have been used widly in regressions and classification applications. To approximate any target continuous function we can use a single hidden layer feedforward networks with additive models. Single hidden layer feedforward networks have good role in many applications and have been implemented widly in both sides, the theory and application sides [11].

3.1 Back Propagation Algorithm
Back Propagation (BP) algorithm consists of two important stages, the forward stage and revers propagation. The NN contains hidden layer with output layer and lastly input layer, in which the states of used neurons in every layer can influence the neurons under them. As initial state, starting with the process which acts in forward propagation, the signal will be transmitted from the input layer to the hidden layer and calculated at the hidden layer. The results of calculation is recalculated in hidden layer
which are send to the output layer and will be considered as output. The final results are compared
with the some values called the expected valu, then the error which is the absolute differences will be
corrected by the reverse state propagation, which is consider as backtrack. In the hidden layer, a
function which is used in this process is called the activation function. This process still be repeated.
The random weights will be changed depending on the results in output layer through every reverse
calculation propagation to reduce the error. When the error reach some spesfied value, stop the
calculations [12].

The learning rate (η) determines the portion of weight needed to be adjusted. However, the optimum
value of η depends on the problem. The momentum (α) determines the fraction of the previous weight
adjustment that is added to current weight adjustment. It accelerates the network convergence process.

During the training process, the learning rate and the momentum are adjusted to bring the network
out of its local minima, and to accelerate the convergence of the network [2]. The details of BP
Algorithm are described in [2].

3.2 Fitness Criterion of NN
One of the stopping criterions is the fitness value. Since the BP algorithm is chosen to be a supervised
learning algorithm, then there are observed values (Oi) and desired output values (Gi). These two values
have to be compared, if they are closed to each other, then the fitness is good, else the algorithm must
continue its calculations until this condition is satisfied or the specified number of iterations is finished
[2].

The Mean Squared Error (MSE) is one of the tests for the comparison process:

\[ \text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (O_i - G_i)^2 \]  

where \( n \) is the number of the compared categories.

4. Problem Formulation
The problem of scheduling \( N = \{1, 2, \ldots, n\} \) the set of \( n \) jobs, these jobs are processed on a one machine
to minimize the multicriteria which may be act as follows. Each job \( j \in N \) has is to be processed on an
one machine which can process only one job at a time, job \( j \) has a processing time \( p_j \) with due date \( d_j \).
All jobs are ready for processing at a time zero. If a schedule \( \sigma = (1, 2, \ldots, n) \) is given, then the
completion time \( C_j = \sum_{k=1}^{j} p_j \) for every job \( j \) can be calculated and then the tardiness of job \( j \), \( T_j = \max\{C_j - d_j, 0\} \) is easy to compute. We aim to find a schedule \( \sigma \in S \) (where \( S \) is the set of all possible
feasible schedules) that minimizes the tricriteria problem. This problem belongs to simultaneous
optimization.

4.1 The 1//\( T_{max}, V_{max}, \Sigma V_j \) Problem
We aim to find the order of the jobs on one machine to minimize the multicriteria \( (T_{max}, V_{max}, \Sigma V_j) \).
This problem, which is denoting by (P), can be formulted mathematically as follows:

\[ Z = \min(T_{max}(\sigma), V_{max}(\sigma), \Sigma V_j(\sigma)) \]

Subject to

\[
\begin{align*}
T_{\sigma(j)} & \geq 0, \quad j = 1, \ldots, n \\
T_{\sigma(j)} & \geq C_j - d_{\sigma(j)}, \quad j = 1, \ldots, n \\
V_{\sigma(j)} & \leq T_{\sigma(j)}, \quad j = 1, \ldots, n \\
V_{\sigma(j)} & \leq p_{\sigma(j)}, \quad j = 1, \ldots, n \\
T_{\sigma(j)} - V_{\sigma(j)} & \geq 0, \quad j = 1, \ldots, n 
\end{align*}
\]  

where \( \sigma(j) \) represents the position of job \( j \) in the schedule \( \sigma \). The discussed P-problem is \( NP \)-hard
because the problem 1//\( \Sigma_{j=1}^{n} V_j \) is \( NP \)-hard [13].
4.2 The $1/(T_{max} + V_{max} + \sum V_j)$ Problem

We aim to find the order of the jobs on one machine to minimize the sum of maximum tardiness, maximum late work and total late work. This problem, which is denoting by ($P_1$), can be formulated mathematically as follows:

$$Z = \min(T_{max}(\sigma) + V_{max}(\sigma) + \sum V_j(\sigma))$$

Subject to

$$T_{\sigma(j)} \geq 0, \quad j = 1, \ldots, n$$

$$T_{\sigma(j)} \geq C_j - d_{\sigma(j)}, \quad j = 1, \ldots, n$$

$$V_{\sigma(j)} \leq T_{\sigma(j)}, \quad j = 1, \ldots, n$$

$$V_{\sigma(j)} \leq p_{\sigma(j)}(\sigma), \quad j = 1, \ldots, n$$

$$T_{\sigma(j)} V_{\sigma(j)} \geq 0, \quad j = 1, \ldots, n$$

5. Experimental Results of Applying NN for $P$ and $P_1$-Problems

For the two problems ($P$) and ($P_1$), a simulation has been constructed in order to apply the NN. For practical comparison, the Branch and Bound (BAB), Hybrid Method (HM) and Modified BAB (MBAB) details results are obtained from [14], and PSO and BA details results are obtained from [7].

5.1 Constructing of Multi-Layer NN to Solve $P$ and $P_1$-Problem

The NN which is proposed for the single machine to minimize the multiple criteria functions ($T_{max}, V_{max}, \sum V_j$) for MSP, is arranged into three layers of the processing units. First we have an input layer consists of (10) nodes, then the hidden layer consists of (8) nodes, Lastly the output layer which include a single node. The number of units in each, the input and output layers depended on to the representation adopted for the MSP [15]. In the suggested representation of the proposed NN, the input layer contains important information interpreted the MSP in the form of a vector or string of continuous values. In table (1) we show the 10 input nodes which are designed to contain the important information for each job that have to be scheduled [5].

| Input units | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-------------|---|---|---|---|---|---|---|---|---|----|
| $p_j$       |   |   |   |   |   |   |   |   |   |    |
| $M_p$       |   |   |   |   |   |   |   |   |   |    |
| $d_j$       |   |   |   |   |   |   |   |   |   |    |
| $M_d$       |   |   |   |   |   |   |   |   |   |    |
| $S_{L_j}$   |   |   |   |   |   |   |   |   |   |    |
| $M_{st}$    |   |   |   |   |   |   |   |   |   |    |
| $\alpha_j$  | 10|   |   |   |   |   |   |   |   |    |
| $\beta_j$  | 10|   |   |   |   |   |   |   |   |    |
| $\bar{p}$  |   |   |   |   |   |   | 1 |   |   |    |
| $\bar{S}_{L}$ |   |   |   |   |   |   |   |   |   |    |
| $\frac{\sum (p_j - \bar{p})^2}{n \times \bar{p}^2}$ |   |   |   |   |   |   |   |   |   |    |
| $\frac{\sum (S_{L_j} - \bar{S}_{L})^2}{n \times S_{L}^2}$ |   |   |   |   |   |   |   |   |   |    |

where $S_{L_j}$ slack for job $j$ such that $S_{L_j} = d_j - p_j$.

$M_p$: longest processing time among the $n$ jobs $= max\{p_j\}$.

$M_d$: longest due date among the $n$ jobs $= max\{d_j\}$.

$M_{st}$: largest slack time for the $n$ jobs $= max\{S_{L_j}\}$.

$\alpha_j = \beta_j = 1, j \in N, N = \{1, 2, ..., n\}$.

$\bar{p} = \sum_{j=1}^{n} \frac{p_j}{n}$ and $\bar{S}_{L} = \sum_{j=1}^{n} \frac{S_{L_j}}{n}$ are the mean of processing times and slack time respectively.
Thus, 10-input vectors represent each job, which includes information related to that job and the relation of this job to the other jobs in the MSP. The output unit includes values that are in the range of [0.10-0.90], the value being an pointer of where this job represented at the input layer should where its position lie in this schedule. Low values should put in the beginning positions in this schedule, while higher values which has less priority and the job position put in the end of this schedule. The objective value related for each input training pattern is a target value that assigns the position determines the best or optimal schedule. The objective value $G_j$ for the job has the $j^{th}$ position in the best or optimal schedule is calculated as in equation (2) [5];

$$G_j = \{0.1 + 0.8 \times \left(\frac{j-1}{n-1}\right)\}, j = 1,2,\ldots, n$$  \hspace{1cm} \text{...(2)}$$

Equation (2) ensures that the n target values are distributed evenly between 0.1–0.9. The number of units in the hidden layer is selected by trial and error during the training phase. The suggested final network for single machine with multi objective has 8 units in its hidden layer and 1 unit of output layer, therefore known as 10–8–1 network.

5.2 Neural Network BP -Training Results
To train the proposed NN, each vector with their output is presented individually at the input layer and output layer of the NN. Before we discuss the NN training we will describe the proposed NN-Scheduling algorithm using BP as a learning algorithm.

NN-Scheduling Algorithm for P and P$_1$-problem

Step(1):READ (n) \hspace{1cm} \{ n=number of jobs\}

READ ($p_j$, $d_j$) \hspace{1cm} \{ processing time and due date\}

Step(2):DEFINE input units \hspace{1cm} \{ as input equations in Table (1) \}

Calculate $G_j$ \hspace{1cm} \{ actual value as in equation (2) \}

Step(3):INITIALIZATION

$W_{jk}\ell$ =random($-1,1$) \hspace{1cm} \{ weight in layer j, node k connected node l \}

Step(4):IMSE = Maxint; Error= 0.0001; Threshold=0.01 \{specified by experiance\}

Iter=0; NI=10000; BSC=Mxint; BST=Maxint;

WHILE (Iter $\leq$ NI) AND (IMSE $>$ Error) DO

Iter = Iter $+$ 1;

Compute MSE \hspace{1cm} \{ using equation (1) \}

$i$ =SORT($Q_i$);

IF($T(\sigma) < BT$) $OR (V(\sigma) < BV)OR (SV(\sigma) < BSV)$THEN

{IF ($T(\sigma) + V(\sigma) + SV(\sigma) < BT + BV + BSV$)THEN for P$_1$}

$\sigma' = \sigma$; $BT = T(\sigma)$; $BV = V(\sigma)$; $BSV = SV(\sigma)$; IMSE=MSE;

END; \{ENDIF\}

END; \{ENDWHILE \}

Step(5):Best sequence $= \sigma'$, with minimum BT, BV and BSV;

{with minimum $BT + BV + BSV$ for P$_1$}

Step(6): END;

Training is considered completed after an average10000 cycles or more using a 10-8-1 configuration. A cycle is concluded after the network has been exposed once, in the course of the BP algorithm, to each one of the available training patterns. The trained NN is used to find job schedule for our problem.

Table (2) shows the comparison results of Effecinit Points (EP), consuming time, Average of MAE (AMAE) and MAER with EP of applying NN-BP for P-problem with CEM for $n = 3:10$. 

\[5\]
MET : Method={CEM, HM, MBAB}.
NI : Number of iterations.
MSE : Mean square error.
MAER : MAE Relative =\( \frac{\text{abs}(\text{best}(\text{EP of MET})-\text{best}(\text{EP of BP}))}{\text{best}(\text{EP of MET})} \)
R : \( 0 \leq \text{Real value} \leq 1 \).
F : Objective or target function of Problem (P).
F_1 : Objective or target function of Problem (P_1).
BS : Best Solution.
CT : Complete Time.
BT : Best Time to obtain best value(s) of each experiment.
NI : Number of Iterations.

**Table 2.** Applying NN-BP for (P) for \( n = 3:10 \) and compared with CEM.

| \( n \) | CEM | NN-BP | AMAE | MAER |
|-------|-----|-------|------|------|
|       | EP  | CT    | EP   | CT   | NI   |
| 3     | (4.4,5), (18.2,4) | R | (4.4,5) | 2 | \( 1.75, R, R \) | (0,0,0) |
| 4     | (14.7,15), (15.6,15), (15.7,14) | R | (15.6,15) | 12 | (R,R,R) | (0,0,0) |
| 5     | (23.9,24), (24.9,23) | R | (23.9,24) | 4 | (R,0,R) | (0,0,0) |
| 6     | (20.9, 24), (24.10,20), (26.8,20) | R | (20.9,24), (26.10,20) | 7 | 3 | (R,R,R) | (0,0,0) |
| 7     | (27.10,28), (28.7,27) | R | (27.10,30), (28.7,27) | 6 | 5 | (0,0,R) | (0,0,0) |
| 8     | (34.10,39), (39.8,34) | 3.1 | (39.8,34) | 6 | 2387 | (R,R,R) | (0,0,0) |
| 9     | (30.10,33), (33.9,30), (37.7,30) | 32.5 | (33.9,30) | 6 | 3 | (R,R,R) | (0,0,0) |
| 10    | (35.7,35) | 232 | (35.7,35) | 7 | 1165 | 0 | (0,0,0) |

Table (3) shows the comparison results of EP, consuming time, AMAE and MAER with EP of applying NN-BP for P-problem with HM for \( n = 11:19 \).

**Table 3.** Applying NN-BP for (P) for \( n = 11:19 \) and compared with HM.

| \( n \) | HM | NN-BP | AMAE | MAER |
|-------|----|-------|------|------|
|       | EP  | CT    | EP   | CT   | NI   |
| 11    | (42.9,45), (48.9,42) | R | (42.9,45) | 7 | 384 | (R,0,R) | (0,0,0) |
| 12    | (59.9,59), (60.8,59) | R | (60.8,59) | 10 | 3194 | (R,R,0) | (0,0,0) |
| 13    | (58.10,65), (63.8,58) | R | (58.10,65) | 10 | 25 | (R,R,R) | (0,0,0) |
| 14    | (65.10,65) | R | (65.10,65) | 6 | 5 | (0.0,0) | (0,0,0) |
| 15    | (77.10,79), (81.9,77) | R | (81.9,77) | 7 | 8771 | (R,R,R) | (0,0,0) |
| 16    | (76.10,78), (83.10,76) | R | (78.10,87), (83.10,76) | 7 | 3251 | (R,0,R) | (0,0,0) |
| 17    | (83.10,90), (86.9,83) | R | (83.10,90) | 10 | 3036 | (R,R,R) | (0,0,0) |
| 18    | (100,9,100) | R | (102.9,100) | 7 | 113 | (R,0,0) | (R,0,0) |
| 19    | (91.9,93), (93.10,91) | R | (91.10,97), (93.10,91) | 11 | 10697 | (0,0,R) | (0,0,0) |

The shaded calls are similar value for the compared methods. Table (4) shows the comparison results of EP, consuming time, AMAE and MAER with EP of applying NN-BP for P-problem with HM for \( n = 20:100 \).
Table 4. Applying NN-BP for (P) for n = 20:100 and compared with HM.

| n   | HM         | NN-BP        | AMAE | MAER |
|-----|------------|--------------|------|------|
|     | EP         | CT           | EP   | CT   | NI   |       |
| 20  | (118,10,120)(120,10,118) | 2.0 | (118,10,123) | 7 | 2 | (R,0,R) | (0,0,R) |
| 30  | (150,10,155)(152,10,150) | 9.1 | (152,10,150) | 11 | 10975 | (R,0,R) | (0,0,0) |
| 40  | (198,10,198) | 2.5 | (198,10,202)(202,10,201) | 12 | 1890 | (R,0,R) | (0,0,R) |
| 50  | (259,10,264)(262,10,259) | 97.7 | (259,10,264) | 13 | 251 | (R,0,R) | (0,0,0) |
| 60  | (310,10,311)(312,10,310) | 242.6 | (312,10,310) | 13 | 1296 | (R,0,R) | (0,0,0) |
| 70  | (367,10,367) | 9.9 | (367,10,367) | 15 | 1 | (0,0,0) | (0,0,0) |
| 80  | (425,10,425) | 20.7 | (425,10,430) | 15 | 2093 | (0,0,R) | (0,0,0) |
| 90  | (446,10,446) | 38.2 | (452,10,446) | 16 | 7041 | (R,0,R) | (0,0,0) |
| 100 | (542,10,548)(543,10,542) | 1214.4 | (542,10,548) | 17 | 4763 | (R,0,R) | (0,0,0) |

Table 5 shows the comparison results of EP, consuming time, AMAE and MAER with EP of applying NN-BP for P-problem with MBAB for n = 20:90.

Table 5. Applying NN-BP for (P) for n = 20:90 and compared with MBAB.

| n   | MBAB        | NN-BP        | AMAE | MAER |
|-----|-------------|--------------|------|------|
|     | EP          | CT           | EP   | CT   | NI   |       |
| 20  | (118,10,123)(123,10,118) | R  | (118,10,123) | 7 | 2 | (R,0,R) | (0,0,0) |
| 30  | (150,10,152)(152,10,150) | R  | (152,10,150) | 11 | 10975 | (R,0,R) | (0,0,0) |
| 40  | (198,10,198) | R  | (198,10,202)(202,10,201) | 12 | 1890 | (R,0,R) | (0,0,R) |
| 50  | (259,10,262)(262,10,259) | 2.0 | (259,10,264) | 13 | 251 | (R,0,R) | (0,0,0) |
| 60  | (310,10,311)(311,10,310) | 3.8 | (312,10,310) | 13 | 1296 | (R,0,R) | (0,0,0) |
| 70  | (367,10,368)(368,10,367) | 5.5 | (376,10,367) | 15 | 1 | (R,0,R) | (0,0,R) |
| 80  | (425,10,425) | 3.9 | (425,10,430) | 15 | 2093 | (0,0,R) | (0,0,0) |
| 90  | (446,10,447)(447,10,446) | 10.9 | (452,10,446) | 16 | 7041 | (R,0,R) | (0,0,0) |

Table 6 shows the comparison results of EP, consuming time, AMAE and MAER with EP of applying NN-BP for P-problem with MBAB for n = 100:500.

Table 6. Applying NN-BP for (P), n = 100:500 and compared with MBAB.

| n   | MBAB        | NN-BP        | AMAE | MAER |
|-----|-------------|--------------|------|------|
|     | EP          | CT           | EP   | CT   | NI   |       |
| 100 | (542,10,543)(543,10,542) | 14.7 | (542,10,548) | 17 | 4763 | (R,0,R) | (0,0,R) |
| 200 | (1085,10,1086)(1086,10,1085) | 38.8 | (1087,10,1090) | 25 | 3 | (R,0,R) | (0,0,0) |
| 300 | (1647,10,1647) | 65.5 | (1656,10,1647) | 45 | 8 | (R,0,0) | (R,0,0) |
| 400 | (2218,10,2218) | 163.8 | (2220,10,2222) | 42 | 10 | (R,0,R) | (R,0,R) |
| 500 | (2629,10,2629) | 351.4 | (2636,10,2629) | 52 | 2 | (R,0,0) | (R,0,0) |

Table 7 shows the comparison results of CEM and NN-BP for P_1-problem using n = 3:10.

Table 7. The results of comparison of CEM and NN-BP for P_1-problem, n = 3:10.

| n   | CEM | NN-BP | AMAE | MAE |
|-----|-----|-------|------|-----|
|     | OP of F_1 | BS of F_1 | CT | NI | MSE | MAE |
| 3   | 13 | R | 13 | R | 2 | 0.0000 | 0 |
| 4   | 36 | R | 36 | 2 | 12 | 0.0000 | 0 |
| 5   | 56 | R | 56 | 3 | 0 | 0.0000 | 0 |
| 6   | 53 | R | 53 | 7 | 3 | 0.0000 | 0 |
Table (8) shows the comparison results of BAB and NN-BP for $P_1$-problem using $n = 11:15$.

**Table 8. The results of comparison of BAB and NN-BP for $P_1$-problem, $n = 11:15$.**

| $n$ | BAB | NN-BP | MAE |
|-----|-----|-------|-----|
|     | OP of $F_1$ | CT | BS of $F_1$ | CT | NI | MSE |     |
| 11  | 96  | 31.4 | 96  | 7  | 384 | 0.0006 | 0   |
| 12  | 127 | 3.1  | 127 | 10 | 3194| 0.0053 | 0   |
| 13  | 129 | 10.4 | 133 | 10 | 25  | 0.0041 | 0.03 |
| 14  | 140 | 5.2  | 140 | 6  | 5   | 0.0208 | 0   |
| 15  | 166 | 267.2| 167 | 7  | 8771| 0.0538 | 0.006|
|     | 658 | 131.6| 663 | 132.6|     | AMAE=1.0|     |

The comparison results of BAB with NN-BP for $P_1$-problem, using $n = 11:15$ are shown in Figure (1).

![Figure 1. Comparison of NN-BP with BAB for $P_1$-problem using $n = 11:15$.](image)

Table (9) shows the comparison results of HM and NN-BP for $P_1$-problem using $n = 20:100$.

**Table 9. The results of comparison of HM and NN-BP for $P_1$-problem, $n = 20:100$.**

| $n$ | HM | NN-BP | MAE |
|-----|----|-------|-----|
|     | OP of $F_1$ | CT | BS of $F_1$ | CT | NI | MSE |     |
| 20  | 248 | 2    | 251 | 7  | 2  | 0.0372 | R   |
| 30  | 312 | 9    | 312 | 11 | 10975 | 0.0328 | 0   |
In figure 2, the comparison results between HM and NN-BP for P1-problem for n = 20:10:100 are shown.

Table 10 shows the results of NN-BP, these results are compared with BA and PSO for P1-problem using n = 100:100:500.

Table 10. The results of comparison of NN-BP with BA and PSO for P1-problem, n = 100:500.

| n    | BA       | PSO      | NN-BP    | MAE |
|------|----------|----------|----------|-----|
|      | BS       | CT       | BS       | CT  | NI | MSE | BA | PSO |
| 100  | 1097     | 20.3     | 1095     | 37.7| 1100| 4763| 0.0461|R   | R   |
| 200  | 2188     | 39.4     | 2182     | 73.7| 2187| 25  | 3   | 0.0237|R   | R   |
| 300  | 3313     | 60.7     | 3304     | 111.5| 3315| 33 | 3   | 0.0174|R   | R   |
| 400  | 4452     | 82.9     | 4447     | 149.9| 4452| 42 | 10  | 0.0156|0   | R   |
| 500  | 5274     | 105.8    | 5268     | 198.6| 5275| 52 | 2   | 0.0122|R   | R   |
| sum  | 16324    | -----    | 16296    | -----| 16329|     |     |       |
| mean | 3264.8   | 3259.2   | 3265.8   |     |     |     |     |     |
In figure 3, the comparison of the results between NN-BP, BA, PSO for $P_1$-problem for $n = 100: 500$ are shown.

6. Discussion and Analysis of Table Results
1. This paper discussed a various number of jobs ($n$), starting from $n = 3: 10$, $n = 11: 19$, $n = 20: 10: 90$ and $n = 100: 100: 500$, with number of iterations ($NI$) which is suitable to specific $n$ to solve the two problems ($P$) and ($P_1$).
2. The parameters of testing the efficiency of LSM (BA and PSO) and NN-BP are calculated, these parameters represented by, the exact efficient solutions of $F$ and $F_1$ (are calculated using CEM for $n \leq 10$), the local search and NN-BP best efficient solutions (EP) and their averages, the AMAE, MAER the consuming time which calculated for (5) experiments, the CT and BT and their averages and lastly, the number of iterations which found the corresponding BS or EP.
3. For problem ($P$):
   - Table (2): NN has exact results compared with CEM for $n = 3: 10$.
   - Table (3): NN has MAER, near 100% similar to HM results for $n = 11: 19$.
   - Table (5): MBAB is near to NN by (75%) for $n = 20: 10: 90$.
   - Table (4): HM is better than NN, in 5 similar from 9 solutions, for $n = 20: 10: 100$.
   - Table (6): MBAB is near to NN by (100%) for $n = 100: 100: 500$.
   - NN has little difference in time with MBAB for $n = 20: 10: 90$.
   - NN is the best from BA, PSO and MBAB for $n = 100: 100: 500$.
4. For problem ($P_1$):
   - Table (7): NN has exact results compared with CEM for $n = 3: 10$.
   - Table (8): NN has AMAE=1.0, so its very closed to BAB for $n = 11: 15$ (see figure (1)).
   - Table (9): NN has AMEA=3.8 so its very closed to HM (see figure (2)) for $n = 20: 10: 100$.
   - Table (10): PSO has mean of BS= 3259.2, BA has mean of BS= 3264.8, while NN has mean of BS= 3265.8, but they are very closed to each other for $n = 100: 100: 500$ (see figure (3)).
7. Conclusions
1. An NN applied to solve the problems (P) and (P₁), were found that the NN gave an efficient and optimal solutions for n ≤ 500.
2. NN is best in consuming time than BA and PSO, for all n for all common tables.
3. Some local search methods can be used as learning algorithms (like PSO or genetic algorithm) for NN to solve MSP.
4. An interesting future research topic would involve experimentation with exact, local Search and neural network algorithms for the following problems.
   a. 1/\tau_j/(T_{max}, V_{max}, \sum V_j).
   b. 1/\text{lex}(V_{max}, T_{max}, \sum V_j).
   c. 1/\text{lex}(\sum V_j, T_{max}, V_{max}).

References
[1] Abdul-Razzaq K F 2014 Solving Multi-Criteria Scheduling Problems M.Sc. Thesis, Univ. of Al-Mustansiriyah, College of Science Dept. of Mathematics
[2] Abdul-Razaq T S and Ali F H 2014 Constructing of an Artificial Neural Network to Minimize Total Completion Time and Total Tardiness IOSR Journal of Mathematics 10(2) 25-37
[3] Willems T T and Rooja J E 1994 Neural Network for Job-Shop Scheduling Control Engineering Practice 2 (1) 31-39
[4] Foo Y P S and Takefuji Y 1998 Integer Linear Programming Neural Networks for Job-Shop Scheduling IEEE International Conference on Neural Networks 2 341-348
[5] Hamad A, Sanugi B and Salleh S 2002 Single Machine Common Due Data Scheduling Problems using Neural Network Journal Teknologi Universiti Teknologi Malaysia 36 75-82
[6] Muralidhar A and Alwarsamy T 2013 Multi-Criteria Optimization of Parallel Machine Scheduling Using Neural Networks International Journal of Latest Trends in Engineering and Technology (IJLTET) 2 (2)
[7] Ali F H and Abdul-Kareem S B 2017 April Local Search Methods to Solve Multi-Criteria Machine Scheduling Problems the 23 th Scientific Conference Al-Mustansiriyah University, College of Education, 26-27
[8] Kuo-Ching Y, Chung-Cheng L, Shih-Wei L and Jie-Ning C 2017 Single-Machine Scheduling with Learning Effects and Maintenance: A Methodological Note on Some Polynomial-Time Solvable Cases Hindawi Mathematical Problems in Engineering Article ID 7532174 6 pages https://doi.org/10.1155/2017/7532174
[9] Shang-Chia L, Wei-Ling H, and Chin-Chia W 2015 Note on a Single-Machine Scheduling Problem with Sum of Processing Times Based Learning and Ready Times Hindawi Publishing Corporation, Mathematical Problems in Engineering Article ID 452602 9 http://dx.doi.org/10.1155/2015/452602
[10] Abdul-Razaq T S and Ali F H 2016 Algorithms for Scheduling a Single Machine to Minimize Total Completion Time and Total Tardiness The 2nd International Conference on Mathematical Sciences-ICMA, 23-24 Oct. 2013, Basrah Journal of Science (A) 34 (2) 113-132
[11] Hai-Feng Ke, Cheng-Bo Lu, Xiao-Bo Li, Gao-Yan Zhang, Ying Mei and Xue-Wen Shen 2018 An Incremental Optimal Weight Learning Machine of Single-Layer Neural Networks Hindawi Scientific Programming Article ID 3732120 7 https://doi.org/10.1155/2018/3732120
[12] Bo Y, Yao Cui, Lin Z, Chao Zhang, Yongzhi Y, Zhenming B, and Guobao N 2014 Beam Structure Damage Identification Based on BP Neural Network and Support Vector Machine Hindawi Publishing Corporation Mathematical Problems in Engineering Article ID 850141 http://dx.doi.org/10.1155/2014/850141
[13] Potts C N and Van Wassenhove L N 1991 Single Machine Scheduling to Minimize Total Late Work Operations Research 40(3) 586-595
[14] Ali F H and Abdul-Kareem S B 2017 Scheduling a Single Machine to Minimize Max Tardiness Max Late Work and Total Late Work *Mathematics and Statistics Journal* 3(1)1-17
[15] El-Bourni A, Balakrishnan S and Popplewell N 2002 Sequencing Jobs on a Single Machine: A neural Network Approach *Euro Journal of Op. Res* 126 474-490