Int J Syst Assur Eng Manag (December 2018) 9(6):1302–1314
https://doi.org/10.1007/s13198-018-0729-6

ORIGINAL ARTICLE

An adaptive neuro-fuzzy inference system for makespan estimation of flexible manufacturing system assembly shop: a case study

Vineet Jain[1] · **Tilak Raj**[2]

Received: 7 May 2016 / Revised: 4 April 2017 / Published online: 19 July 2018

Abstract This paper considers the use of combination of neural networks and fuzzy system i.e. adaptive neuro-fuzzy inference system (ANFIS) applied to the n job, m machine real flexible manufacturing system assembly shop problem with the objective of prediction of makespan. Assembly shop makespan is calculated by Nawaz, Enscor, and Ham (NEH) algorithm. On the basis of this algorithm, adaptive neuro-fuzzy inference system model is made to predict the makespan of the jobs. The purpose of this study is to find the makespan estimation in advance if processing time of machines is known. The purpose of this research is to gain the advantage of the capabilities of both Fuzzy systems, which is a rule-based approach and neural network which focus on the network training. This model has been verified by testing and actual data set with the average percentage accuracy achieved is 95.97%. Coefficient of determination and Correlation coefficient is 0.9310 and 0.9649 respectively. The derived values of ANFIS model output are found within the range after being verified practically. Therefore, it can be concluded that makespan calculation of the production system, by the proposed adaptive neuro-fuzzy inference system, can be used as a reliable approach in estimating the makespan of flexible manufacturing system assembly shop.

Keywords FMS assembly shop · NEH heuristic · Makespan estimation · ANFIS

✉ Vineet Jain
vjdj2004@gmail.com

Tilak Raj
tilakraj64@gmail.com

[1] Department of Mechanical Engineering, Amity University Haryana, Gurgaon, India

[2] Department of Mechanical Engineering, YMCA University of Science and Technology, Faridabad, India

1 Introduction

Many large industries have tried to introduce flexible manufacturing systems in today's manufacturing environment as their strategy. It enables them to adapt to the ever-changing competitive market requirements based on quality of machining products, and to reduce the machining costs and to enhance the productivity (Cus and Balic 2003). Flexible manufacturing systems (FMSs) have been developed with the hope that they will be able to tackle new challenges like reduced cost, improved quality, improve delivery speed to satisfy different market segments (Jain and Raj 2016a). A flexible manufacturing system assembly shop schedule is one in which all jobs must visit all machines in the same sequence. Processing of the job should not be started on a succeeding machine before completing processing of a job on a current machine. Although all jobs are available in the beginning but only one job can be performed at any particular time by a machine (Onwubolu 1996). The other machines are left idle queued by other jobs because the first machine has to visit first by each job. Although queuing of jobs is prohibited in just-in-time manufacturing environments, production flow-shop manufacturing continues to find applications in manufacturing (Wittrock 1985), and has attracted much research work (Campbell et al. 1970; Gupta 1972; Nawaz et al. 1983).

An important aspect of scheduling is sequencing. The sequencing is the process in which order jobs visit a machine. Johnson (1954) Johnson's algorithm is apt for a two-machine problems and can be applied on three-machines. A generalization of Johnson's algorithm is that proposed by Campbell et al. (1970) for solving general n-jobs m-machine problems in which $m - 1$ two-machine problems are solved and the sequence having the least makespan is selected. Nawaz et al. (1983) proposed a Nawaz, Enscor, and Ham (NEH) algorithm to construct a jobs sequence in an iterative manner. The production flow shop scheduling of assembly problem is the problem of defining order over a set of jobs as they proceed from one machine (processor) to another in minimum time i.e. makespan of the jobs or assembly.

Scheduling outputs are generally graphically displayed by Gantt charts. Machine processing times for each job is used to draw them. It is also ensured that delay times are taken into consideration.

A minimum makespan, which represents the minimum time required to complete all the jobs, if not found, this process is repeated for different sequences. The obtained sequence is considered to be optimal. The manual method for scheduling is tedious and prone to error. So, soft computing technique is used to find the makespan of the production flow shop. The makespan of the jobs can be calculated by neuro and fuzzy system.

An adaptive neuro-fuzzy inference system (ANFIS) for makespan estimation of flexible manufacturing system assembly shop for five to ten jobs and five machines is presented by this research work. The manufacturing sequences of parts are flexible. Alternative sets of resources may be selected for a manufacturing operation. The characteristics such as resource sharing, concurrency, routing flexibility, mutual exclusion, lot sizes, and synchronization which are difficult to study (Der Jeng et al. 1999).

The main objectives of this research work are as follows:

- To find the makespan of the FMS assembly shop.
- To make a model with the help of neural network and fuzzy rules i.e. ANFIS model.
- To discuss the ANFIS model verification.

In the remainder of this paper, literature review is presented in Sect. 2 for makespan estimation, NEH heuristic, and ANFIS methodology. In Sect. 3, problem description. The Sect. 4 gives the NEH heuristic. Makespan calculation by NEH Algorithm is discussed in Sect. 5. The Sect. 6 gives the ANFIS methodology. Model verification and Conclusion are followed in Sects. 7 and 8 respectively.

2 Literature review

The literature has been reviewed from the perspectives of makespan estimation with neural network and fuzzy rules, NEH heuristic and ANFIS modeling. Cheng and Gupta (1989) used neural networks approaches for estimating the makespan. Yih et al. (1991) proposed a hybrid model to solve a crane scheduling problem. Philipoom et al. (1994) compared a non-linear regression analysis with neural networks of job scheduling problem. Fransoo et al. (1995) compare a makespan estimation based on the analysis of a stochastic queueing network model of the FMS and a makespan minimizing algorithm based on a combinatorial algorithm. Sabuncuoglu and Gurgun (1996) combined neural network and algorithmic approaches to solve the job-shop scheduling problem with minimum makespan. Chen and Muraki (1997) used back-propagation neural network for online rescheduling.Sabuncuoglu (1998) presented a review of the literature and future directions of scheduling approaches using neural network mainly scheduling problems involving artificial neural network (ANN) applications. Ivanescu et al. (2002) used regression analysis to estimate makespan in a batch process shop. Raaymakers et al. (2001) also estimated models based on regression. Raaymakers and Weijters (2003) found that in batch process industries, estimation of makespan is difficult because jobs interact at the shop floor. So, used two different techniques for estimating the makespan of job i.e. regression models and neural networks. Wilson et al. (2004) estimated the minimum makespan for scheduling non similar groups of jobs on a two-stage flow line. Akyol (2004) used ANN models for the prediction of the completion times for each job processed on each machine. Li et al. (2007) proposed a back-propagation network model combined with genetic algorithms for estimation of makespan. Ahmadizar et al. (2010) found a job schedule which minimizes the expected makespan based on ant colony optimization algorithm and a heuristic algorithm. Shokrollahpour et al. (2011) discussed two-stage assembly flowshop scheduling problem with minimisation of weighted sum of makespan and mean completion time by imperialist competitive algorithm. Verma et al. (2012) designed a job schedule that minimizes the makespan. González et al. (2013) tackled the job shop scheduling problem with sequence dependent setup times and maximum lateness minimization by means of a tabu search algorithm. Moradinasab et al. (2013) discussed no-wait two-stage flexible flow shop scheduling problem with setup times aiming to minimize the total completion time by adaptive imperialist competitive algorithm and genetic algorithm. NEH heuristics review as Taillard (1990) Compare the NEH heuristic with taboo search algorithm. Zheng and Wang (2003) used NEH algorithm for flow shop scheduling. Kalczynski and Kamburowski (2007) used NEH algorithm for minimizing the makespan in permutation flow shops. Kalczynski and Kamburowski (2008) used improved NEH algorithm to minimize makespan in permutation flow shops. Dong et al. (2008) also used improved

NEH algorithm to minimize makespan in the permutation flow shops. Yagmahan and Yenisey (2008) used NEH algorithm to compare ant colony optimization for multi-objective flow shop scheduling problem. Shafaei et al. (2011) used NEH algorithm with an adaptive neuro fuzzy inference system for estimating the makespan.

In this section, the study of ANFIS work are summarize as the used by other researchers. Some of review as Mar and Lin (2001) defined an ANFIS controller for the car-following collision prevention system. Ho et al. (2009) used an ANFIS to predict the work piece R_a for the end milling process with the hybrid Taguchi-genetic learning algorithm. Samanta (2009) used adaptive neuro-fuzzy inference system for a surface roughness model in end milling with genetic algorithms. Talei et al. (2010) evaluated rainfall by ANFIS in rainfall–runoff modeling. Güneri et al. (2011) used ANFIS model for supplier selection. Mellit and Kalogirou (2011) used ANFIS model for photovoltaic power supply system. Shafaei et al. (2011) used an adaptive neuro fuzzy inference system to solve a no-wait two stage flexible flow shop for minimizing makespan. Heddam et al. (2012) studied an adaptive neuro fuzzy inference system based modelling for coagulant dosage in drinking water treatment plant. Pousinho et al. (2012) proposed an adaptive neuro fuzzy inference system approach for electricity prices forecasting in a competitive market. Chen (2013) developed a hybrid ANFIS model for business failure prediction by utilizing particle swarm optimization and subtractive clustering. Heddam (2014) made a ANFIS model for hourly dissolved oxygen concentration by using two different adaptive neuro-fuzzy inference systems. Chen et al. (2014) proposed ANFIS for an active magnetic bearing system with unbalance mass. Ay and Kisi (2014) used modelling of chemical oxygen demand by using ANNs, ANFIS and k-means clustering techniques. Özkan and İnal (2014) determined that ANFIS algorithm can be used in multi-criteria decision making problems for supplier evaluation and selection with more precise and reliable results. Maher et al. (2014) Investigated the effect of machining parameters on the surface quality of machined brass (60/40) in CNC end milling by ANFIS modeling. Çevik and Çunkaş (2015) presented a short-term load forecasting models, which was developed by using fuzzy logic and ANFIS. Vasileva-Stojanovska et al. (2015) presented a Quality of Experience prediction model in a student-centered blended learning environment, equipped with appropriate technologically enriched classroom.Framinan and Perez-Gonzalez (2015) used heuristic solutions for the stochastic flowshop scheduling problem. Maher et al. (2015) made a ANFIS model based on cutting force for accurate surface roughness prediction in end milling operation for intelligent machining.Azadeh et al. (2015) used a hybrid computer simulation-adaptive neuro-fuzzy inference system algorithm for optimization of dispatching rule selection in job shop scheduling problems under uncertainty. Abdulshahed et al. (2015) applied ANFIS as a prediction models for thermal error compensation on CNC machine tools. Jung and Choi (2015) ANFIS method was used to predict the composite suitability index for the physical habitat simulation of a 2.5 km long reach of the Dal river in Korea. Jain and Raj (2016b) used ANFIS for tool life management for unmanned production system.

From the literature we have found that researchers focused on optimization or minimizing the makespan and have not discussed estimation of makespan which is necessary for good scheduling, product delivery. So, a model for prediction of makespan is developed which is helpful to any manufacturing system to maintain good scheduling system internally to get reliable product delivery.

3 Problem description

The production shop of flexible manufacturing system assembly shop problem formulated as given below. Each of n jobs from the jobs set i = [1,2….,n], for $n > 1$, has to processed on m machine j = [1,2,….m] in the order given by the indexing of the machine being $t_{i,j}$ to find the minimum makespan and make a model to predict or estimate the makespan of the assembly jobs.

The following assumptions are considered in this problem:

1. All jobs are independent and available at zero time.
2. Machines are also available at zero time.
3. Processing time of jobs is formerly specified.
4. No job has priority over any other job.
5. The transportation time between machines and set up time are included in the processing time.
6. Assembly of parts is also included in the processing time.
7. One job can only be processed on one machine at a time.
8. One machine can only process one job at a time.
9. No preemption is allowed, i.e. the processing of a job i on a machine j cannot be interrupted.

In this study, the operations set-up times are assumed to be independent of the job sequences, and hence is added to the operation times. The performance of the proposed heuristic algorithm is studied in terms of minimum makespan.

Here, taking a case study of flexible manufacturing system assembly shop. This is the case of a large multi nation organization X engaged in the manufacture of a wide variety of automobile components in India, with an estimated turnover of Rs. 350 crores per year. That is one of the largest automobile component supplier in the country. The product range includes different car manufacturing company like Maruti Suzuki, Hyundia, Honda, Toyota etc. with different models. The organization has to increase the good quality and supply the product with variations of models with minimum time frame.

So, a model is prepared to predict the makespan of the components with different variants (i.e. five to ten jobs) on five machines or workstations including machining and assembly processes. A sample assembly shop line is shown in Fig. 1. The final assembly is completed to pass five machines or workstations including machining and assembly process.

In this research work, the framework of the proposed ANFIS-based soft computing intelligent system is described in the ANFIS methodology section for consisting of five machines which are capable of handling a five to ten numbers of jobs.

4 NEH algorithm

An overview of the NEH algorithm can be stated as follows.

Step 1 Calculate total process times for each job i

$$T_i = \sum_{j=1}^{j=m} t_{i,j} \tag{1}$$

where $t_{i,j}$ is the process time of job i on machine j.

Step 2 The jobs are arranged according to descending order of total processing time T_i.

Step 3 The two jobs are picked from the first and second position of the list of Step 2, and the best sequence is found for these two jobs by calculating makespan for the two possible sequences. The relative positions of these two jobs should remain same with respect to each other in the remaining steps of the algorithm. Set i = 3.

Step 4 Next the job is picked in the *i*th position of the list generated in Step 2 and the best sequence is found by placing it at all possible i positions in the partial sequence found in the previous step without changing the relative positions to each other of the already assigned jobs. The number of enumerations at this step equals i.

Step 5 If $n = i$, then STOP, otherwise set $i = i + 1$ *and go to Step* 4.

5 Makespan calculation by NEH algorithm

Considering 5 machine and 5 jobs for calculation of makespan by NEH algorithm (see Table 1).

Step 1 Calculate total process times for each job i (see Table 2)

Step 2 Sort in the decreasing order of processing times (see Table 3)

Step 3 Take J_4 & J_3

Iteration 1

Possible combinations: J_4–J_3 & J_3–J_4.

For J_4–J_3 (see Table 4):

where c is makespan

For J_3–J_4 (see Table 5):

C_{max} for J_4–J_3 < J_3–J_4, therefore we choose J_4–J_3.

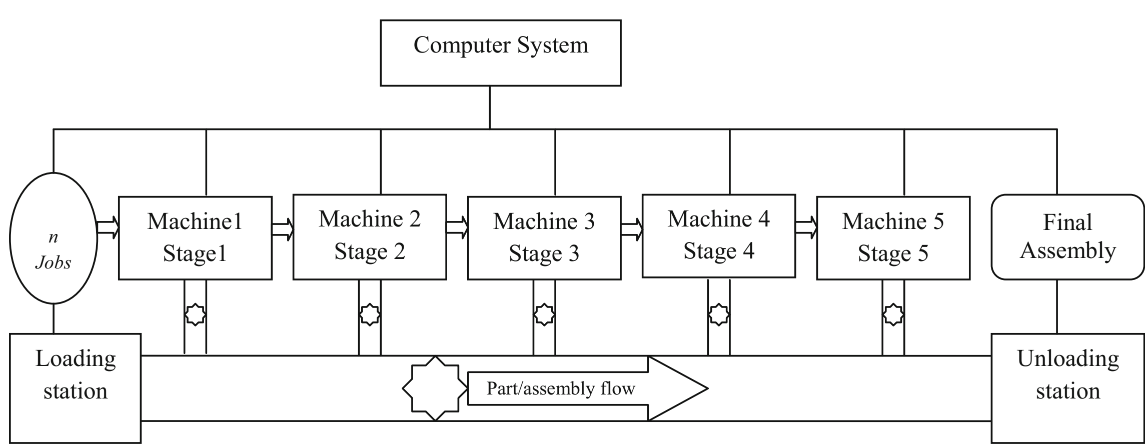

Fig. 1 Five machine FMS assembly shops

Table 1 Processing time of Jobs

	J_1	J_2	J_3	J_4	J_5
M_1	66	52	98	65	81
M_2	46	44	83	9	14
M_3	18	40	84	81	7
M_4	40	53	42	66	63
M_5	30	44	2	99	17

Table 2 Total processing time of individual Jobs

	J_1	J_2	J_3	J_4	J_5
M_1	66	52	98	65	81
M_2	46	44	83	9	14
M_3	18	40	84	81	7
M_4	40	53	42	66	63
M_5	30	44	2	99	17
Processing time	200	233	309	320	182

Table 3 Descending order of Jobs based on total processing time

J_4	J_3	J_2	J_1	J_5
320	309	233	200	182

Table 4 Makespan for partial sequence of 4-3 Jobs

	J_4	J_3	C_4	C_3	C_{max}
M_1	65	98	65	163	
M_2	9	83	74	246	
M_3	81	84	155	330	
M_4	66	42	221	372	
M_5	99	2	320	374	374

Table 5 Makespan for partial sequence of 3-4 Jobs

	J_3	J_4	C_3	C_4	C_{max}
M_1	98	65	98	163	
M_2	83	9	181	172	
M_3	84	81	265	253	
M_4	42	66	307	319	
M_5	2	99	309	418	418

Table 6 Makespan for partial sequence of 2-4-3 Jobs

	J_2	J_4	J_3	C_2	C_4	C_3	C_{max}
M_1	52	65	98	52	117	215	
M_2	44	9	83	96	126	298	
M_3	40	81	84	84	207	382	
M_4	53	66	42	93	273	424	
M_5	44	99	2	97	372	426	426

Table 7 Makespan for partial sequence of 4-2-3 Jobs

	J_4	J_2	J_3	C_4	C_2	C_3	C_{max}
M_1	65	52	98	65	117	215	
M_2	9	44	83	74	161	298	
M_3	81	40	84	90	201	382	
M_4	66	53	42	147	254	424	
M_5	99	44	2	165	298	426	426

Table 8 Makespan for partial sequence of 4-3-2 Jobs

	J_4	J_3	J_2	C_4	C_3	C_2	C_{max}
M_1	65	98	52	65	163	215	
M_2	9	83	44	74	246	259	
M_3	81	84	40	90	330	299	
M_4	66	42	53	147	372	352	
M_5	99	2	44	165	374	396	396

Step 4 Then we take the next job in the sequence i.e., J_2 Now J_2 can be squeezed in three ways i.e., J_2–J_4–J_3, J_4–J_2–J_3, J_4–J_3–J_2

Iteration 2

For J_2–J_4–J_3 (see Table 6):

For J_4–J_2–J_3 (see Table 7):

For J_4–J_3–J_2 (see Table 8):

C_{max} for J_4–J_3–J_2 < J_4–J_2–J_3, J_2–J_4–J_3 J_3–J_4, therefore we choose J_4–J_3–J_2.

Step 5 Then we take the next job in the sequence i.e., J_1 Now J1 can be squeezed in 4 ways i.e., J_1–J_4–J_3–J_2, J_4–J_1–J_3–J_2, J_4–J_3–J_1–J_2, J_4–J_3–J_2–J_1.

Iteration 3

For J_1–J_4–J_3–J_2 (see Table 9):

For J_4–J_1–J_3–J_2 (see Table 10):

For J_4–J_3–J_1–J_2 (see Table 11):

For J_4–J_3–J_2–J_1 (see Table 12):

C_{max} for J_4–J_3–J_2–J_1 < J_1–J_4–J_3–J_2, J_4–J_1–J_3–J_2, J_4–J_3–J_1–J_2, therefore we choose J_4–J_3–J_2–J_1.

Table 9 Makespan for partial sequence of 1-4-3-2 Jobs

	J_1	J_4	J_3	J_2	C_1	C_4	C_3	C_2	C_{max}
M_1	66	65	98	52	66	131	229	281	
M_2	46	9	83	44	112	140	312	325	
M_3	18	81	84	40	130	221	396	365	
M_4	40	66	42	53	170	287	438	418	
M_5	30	99	2	44	200	386	440	462	462

Table 10 Makespan for partial sequence of 4-1-3-2 Jobs

	J_4	J_1	J_3	J_2	C_4	C_1	C_3	C_2	C_{max}
M_1	65	66	98	52	65	131	229	281	
M_2	9	46	83	44	74	177	312	325	
M_3	81	18	84	40	155	195	396	365	
M_4	66	40	42	53	221	235	438	418	
M_5	99	30	2	44	320	265	440	462	462

Table 11 Makespan for partial sequence of 4-3-1-2 Jobs

	J_4	J_3	J_1	J_2	C_4	C_3	C_1	C_2	C_{max}
M_1	65	98	66	52	65	163	229	281	
M_2	9	83	46	44	74	246	275	325	
M_3	81	84	18	40	155	330	293	365	
M_4	66	42	40	53	221	372	333	418	
M_5	99	2	30	44	320	374	363	462	462

Table 12 Makespan for partial sequence of 4-3-2-1 Jobs

	J_4	J_3	J_2	J_1	C_4	C_3	C_2	C_1	C_{max}
M_1	65	98	52	66	65	163	215	281	
M_2	9	83	44	46	74	246	259	327	
M_3	81	84	40	18	155	330	299	345	
M_4	66	42	53	40	221	372	352	385	
M_5	99	2	44	30	320	374	396	415	415

Step 6 Then we take the next job in the sequence i.e., J_5 Now J_5 can be squeezed in 5 ways i.e., J_5–J_4–J_3–J_2–J_1, J_4–J_5–J_3–J_2–J_1, J_4–J_3–J_5–J_2–J_1, J_4–J_3–J_2–J_5–J_1, J_4–J_3–J_2–J_1–J_5.

Iteration 4

For J_5–J_4–J_3–J_2–J_1 (see Table 13):

For J_4–J_5–J_3–J_2–J_1 (see Table 14):

For J_4–J_3–J_5–J_2–J_1 (see Table 15):

For J_4–J_3–J_2–J_5–J_1 (see Table 16):

For J_4–J_3–J_2–J_1–J_5 (see Table 17):

C_{max} for J_4–J_3–J_2–J_1–J_5 < J_5–J_4–J_3–J_2–J_1, J_4–J_5–J_3–J_2–J_1, J_4–J_3–J_5–J_2–J_1, J_4–J_3–J_2–J_5–J_1, therefore we choose J_4–J_3–J_2–J_1–J_5 and final makespan is 463.

Hence the makespan can be calculated for 5–10 jobs on 5 machines through NEH algorithm. We have taken five machine and jobs from five to ten according to the requirement of production schedule and makespan is shown in Table 18. In this table N stand for no. of jobs, M_1–M_5 are processing time on each machine.

Table 13 Makespan for partial sequence of 5-4-3-2-1 Jobs

	J_5	J_4	J_3	J_2	J_1	C_5	C_4	C_3	C_2	C_1	C_{max}
M_1	81	65	98	52	66	81	146	244	296	362	
M_2	14	9	83	44	46	95	155	327	340	408	
M_3	7	81	84	40	18	102	236	411	380	426	
M_4	63	66	42	53	40	165	302	453	433	466	
M_5	17	99	2	44	30	182	401	455	477	496	496

Table 14 Makespan for partial sequence of 4-5-3-2-1 Jobs

	J_4	J_5	J_3	J_2	J_1	C_4	C_5	C_3	C_2	C_1	C_{max}
M_1	65	81	98	52	66	65	146	244	296	362	
M_2	9	14	83	44	46	74	160	327	340	408	
M_3	81	7	84	40	18	155	167	411	380	426	
M_4	66	63	42	53	40	221	230	453	433	466	
M_5	99	17	2	44	30	320	247	455	477	496	496

Table 15 Makespan for partial sequence of 4-3-5-2-1 Jobs

	J_4	J_3	J_5	J_2	J_1	C_4	C_3	C_5	C_2	C_1	C_{max}
M_1	65	98	81	52	66	65	163	244	296	362	
M_2	9	83	14	44	46	74	246	258	340	408	
M_3	81	84	7	40	18	155	330	265	380	426	
M_4	66	42	63	53	40	221	372	328	433	466	
M_5	99	2	17	44	30	320	374	345	477	496	496

Table 16 Makespan for partial sequence of 4-3-2-5-1 Jobs

	J_4	J_3	J_2	J_5	J_1	C_4	C_3	C_2	C_5	C_1	C_{max}
M_1	65	98	52	81	66	65	163	215	296	362	
M_2	9	83	44	14	46	74	246	259	310	408	
M_3	81	84	40	7	18	155	330	299	317	426	
M_4	66	42	53	63	40	221	372	352	380	466	
M_5	99	2	44	17	30	320	374	396	397	496	496

Table 17 Makespan for partial sequence of 4-3-2-1-5 Jobs

	J_4	J_3	J_2	J_1	J_5	C_4	C_3	C_2	C_1	C_5	C_{max}
M_1	65	98	52	66	81	65	163	215	281	362	
M_2	9	83	44	46	14	74	246	259	327	376	
M_3	81	84	40	18	7	155	330	299	345	383	
M_4	66	42	53	40	63	221	372	352	385	446	
M_5	99	2	44	30	17	320	374	396	415	463	463

Table 18 Makespan for five machine and jobs from five to ten

N	M_1	M_2	M_3	M_4	M_5	Actual Makespan
10	478	704	454	440	458	946
10	529	541	432	402	389	901
10	518	410	594	488	618	941
10	576	417	520	508	420	834
10	491	494	394	429	562	905
10	445	396	420	380	590	820
10	503	524	461	632	520	924
10	493	596	654	570	536	932
10	624	432	523	511	388	888
10	606	388	494	561	434	925
10	543	581	431	541	533	907
10	421	532	509	500	463	835
10	612	456	751	536	405	932
10	359	475	609	524	445	926
10	524	586	673	423	493	948
10	673	466	460	605	554	998
10	468	509	478	574	517	852
10	471	517	601	369	613	887
10	419	319	539	418	487	814
10	486	472	678	619	611	985
9	447	596	342	399	593	916
9	492	302	491	454	429	816
9	547	453	348	363	471	813
9	518	504	630	442	557	904
9	517	482	510	410	361	767
9	407	476	548	609	350	858
9	515	445	348	432	517	907
9	380	343	465	560	437	844
9	528	530	522	500	425	821
9	350	560	481	548	401	799
9	535	409	406	585	526	894
9	467	581	282	298	308	769
9	454	437	395	441	362	779
9	381	663	414	576	540	990
9	414	430	499	478	461	935
9	399	531	485	280	361	798
9	507	551	499	455	465	792
9	301	404	491	411	455	839
9	433	366	296	580	493	850
9	362	551	412	514	521	817
8	525	455	422	388	249	748
8	343	278	294	503	417	741
8	444	424	470	315	266	740
8	345	367	461	415	345	698
8	444	366	499	426	326	741
8	347	399	453	478	399	739
8	416	419	190	273	436	672
8	303	339	337	339	415	635

Table 18 continued

N	M_1	M_2	M_3	M_4	M_5	Actual Makespan
8	349	254	577	493	303	738
8	275	314	421	390	273	652
8	442	266	423	211	305	659
8	540	295	315	443	495	747
8	295	418	547	446	481	767
8	468	614	241	484	455	828
8	438	354	502	384	322	699
8	569	223	413	377	445	822
8	447	341	370	501	461	822
8	334	455	331	365	401	718
8	381	510	506	459	373	870
8	614	320	404	407	311	905
7	371	375	381	261	358	692
7	324	252	393	452	351	664
7	444	277	329	261	298	637
7	400	330	402	328	406	787
7	394	459	175	229	494	688
7	420	150	413	293	331	669
7	430	446	374	438	381	806
7	226	341	304	480	387	658
7	375	356	443	453	291	678
7	200	326	431	311	256	646
7	324	321	349	432	434	765
7	595	309	310	244	305	791
7	408	199	412	382	333	653
7	327	300	325	486	316	696
7	469	299	357	361	343	734
7	341	414	334	368	406	736
7	410	251	434	352	286	666
7	393	305	371	255	440	718
7	307	389	377	303	324	658
7	402	227	352	357	391	689
6	187	324	420	235	157	551
6	350	184	296	314	406	753
6	239	383	260	341	272	643
6	337	342	246	298	310	600
6	202	344	254	371	370	649
6	309	180	386	382	346	601
6	333	180	319	198	261	548
6	405	413	327	220	389	676
6	243	189	311	293	340	549
6	312	411	406	331	351	737
6	355	288	463	297	355	703
6	271	345	197	390	412	632
6	251	347	224	363	186	597
6	229	145	368	252	321	536
6	358	406	204	351	297	712
6	341	362	262	334	339	630

Table 18 continued

N	M_1	M_2	M_3	M_4	M_5	Actual Makespan
6	191	370	306	372	413	721
6	258	323	263	279	234	547
6	255	390	388	234	194	612
6	350	293	283	289	212	578
5	362	196	230	264	192	594
5	272	162	290	313	214	502
5	234	392	290	255	222	634
5	305	252	194	348	198	545
5	144	338	306	180	297	581
5	292	304	181	328	265	608
5	202	329	339	251	209	519
5	290	270	319	232	328	618
5	336	260	197	190	190	519
5	217	218	206	238	255	491
5	292	249	341	200	234	573
5	137	327	263	392	230	627
5	262	206	293	266	231	533
5	297	216	282	229	321	587
5	263	263	265	304	308	618
5	237	165	149	247	183	411
5	172	311	244	298	279	573
5	247	238	310	282	176	629
5	327	222	255	192	274	615
5	187	252	217	238	307	512

6 ANFIS method

Jang (1993) proposed Adaptive neuro-fuzzy inference system (ANFIS) to construct an input–output mapping based on both i.e. human knowledge (in the form of fuzzy if–then rules) and stipulated input–output data pairs. It is known as an adaptive network, a network of nodes and directional links. This network is connected with a learning rule—for example back propagation or hybrid algorithm. ANFIS can predict data using Sugeno FIS (Fuzzy Inference System) to relate membership and tune it using either back propagation or hybrid method. ANFIS model will simulate the inputs to the outputs correctly. In this research, the various input variables are trained and tested by ANFIS method. They are evaluated on the base of testing performances.

ANFIS schematic diagram is shown in Fig. 2. There are five network layers which are used by ANFIS to perform the following fuzzy inference steps: (a) input fuzzification, (b) fuzzy set database construction, (c) fuzzy rule base

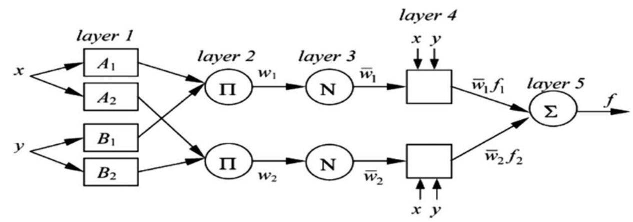

Fig. 2 Schematic diagram of ANFIS (Shafaei et al. 2011)

construction, (d) decision making, and (e) output defuzzification (Maher et al. 2014).

To explain this model simply, consists of five layers of adaptive network with two inputs (x and y) with two linguistic values and output f. Basically, inference system is constructed by five layers (Fig. 2) and each ANFIS layer consists of several nodes described by the node function. The present layers' inputs are derived from the nodes in the previous layers. The rule base of ANFIS contains fuzzy IF–THEN rules of the Sugeno type. For a first-order Sugeno fuzzy inference system, the two rules may be stated as:

$$\text{Rule 1}: \text{ IF x is } A_1 \text{ AND y is } B_1, \text{ THEN f is } f_1(x,y)$$
$$\text{Rule 2}: \text{ IF x is } A_2 \text{ AND y is } B_2, \text{ THEN f is } f_2(x,y),$$

where x and y are the inputs of ANFIS, A_i and B_i are the fuzzy sets, and f_i (x,y) is a first order polynomial and represents the outputs of the first order Sugeno fuzzy inference system. The structure of ANFIS is shown in Fig. 2, and the node function in each layer is described below. Represent the parameter sets that are adjustable in these nodes are presented by adaptive nodes, denoted by squares, whereas fixed nodes, denoted by circles, represent the parameter sets that are fixed in the system (Svalina et al. 2013).

Layer 1 this layer contains adaptive nodes with node functions like i explained as below:

$$Q_{1,i} = \mu_{Ai}(x) \quad \text{for} \quad i = 1, 2 \tag{2}$$

$$Q_{2,i} = \mu_{Bi-2}(y) \quad \text{for} \quad i = 3, 4 \tag{3}$$

where x and y are the input to node i, A_i and B_i are the linguistic labels such as small or large, μ (x) and μ (y) are the membership functions. Many sorts of the membership functions which are there can be used. However, a Gaussian membership function has been chosen to represent the linguistic terms because the relationship between the processing time and makespan is not linear, so this function assured a smooth transition between 0 and 1. It can be written as follows:

First parameter membership functions

$$\mu A_i(x) = \exp\left[-0.5\left(\frac{(x - a_{i1})}{b_{i1}}\right)^2\right] \tag{4}$$

Second parameter membership functions

$$\mu B_i(y) = \exp\left[-0.5\left(\frac{(y - a_{i2})}{b_{i2}}\right)^2\right] \tag{5}$$

where $a_{i,1}$, $a_{i,2}$, $b_{i,1}$, and $b_{i,2}$ are the parameter set. The bell-shaped functions vary while the values of this parameter are changing.

Layer 2 In this layer every node is a fixed node, which is marked by a circle and the node function has to be multiplied by input signals so that it can serve as output for every node. The nodes of this layer are called rule nodes. Each node computes the firing strength of the associated rule i.e.w_1.

$$Q_{2,i} = w_1 = \mu A_i(x) \times \mu B_i(y) \tag{6}$$

Layer 3 Every node in this layer is also a fixed node, marked by a circle and labeled N to show the normalization of the firing levels.

$$Q_{3,i} = \overline{w_i} = \frac{w_i}{\sum w_i} \quad \text{for} \quad i = 1, 2 \tag{7}$$

Layer 4 Every node i in this layer is an adaptive node with a node function and marked by a square:

$$Q_{4,i} = \overline{w_i} \times f_i \quad \text{for} \quad i = 1, 2 \tag{8}$$

Here f_1 and f_2 are the fuzzy IF–THEN rules as follows:

$$\begin{aligned}&\mathit{Rule}\,1: \text{ IF x is } A_1 \text{ AND y is } B_1, \text{THEN } f_1 \text{ is } = p_1x + q_1y + r_1\\ &\mathit{Rule}\,2: \text{ IF x is } A_2 \text{ AND y is } B_2, \text{THEN } f_2 \text{ is } = p_2x + q_2y + r_2,\end{aligned}$$

where $\overline{w_i}$ is normalised firing strength from layer 3 and $[p_i,q_i,r_i]$ is the parameter set of this node and marked as the consequent parameters.

Layer 5 One fixed node of this layer is marked by a circle. The node has to compute the overall output as the summation of all incoming signals:

$$Q_{5,i} = f_{out} = \sum \overline{w_i} \times f_i = \text{overall output.} \tag{9}$$

The first layer and the fourth layer are the two adaptive layers with square nodes in this ANFIS architecture. In the first layer, there are two modifiable parameters known as premise parameters $[a_i,b_i]$ which relates to the input membership functions. In the fourth layer, there are also three modifiable parameters known as consequent parameters $[p_i,q_i,r_i]$ pertaining to the first-order polynomial.

MATLAB is used for ANFIS model development. ANFIS command window is used for training and testing. Gaussian bell membership function was used in input and output. In ANFIS a hybrid learning method is applied for updating the FIS parameters. The training process continues till the desired number of training steps (epochs) or the desired root mean squared error (RMSE) between the desired and the generated output is achieved.

Steps of ANFIS model for makespan estimation of FMSAS are explained as follows:

Step 1 Normalize the training and test data
Because the range of data is different, so normalized the data as

$$x_i' = \frac{x_i - x_{i,\min}}{x_{i,\max} - x_{i,\min}} \tag{10}$$

where $x_{i,\min}$ and $x_{i,\max}$ are the minimum and maximum values of ith input data.

Step 2 Load Input training data and test data into the ANFIS model
Input data are a number of jobs, summation of processing times for one to five machines, whereas the output data is the makespan or the completion time of jobs.

Step 3 Set the input and output parameters and membership function
The output and input parameters for ANFIS are defined. Membership function i.e. Gaussian bell shape is defined and used evalfis command for this.

Step 4 Define the optimal parameter values for optimization
The parameters are optimized in which radii parameter is most important.

Step 5 Define the epochs of the FIS for training
The epochs are set for the training of the model.

Step 6 Trained the ANFIS model
The training of the model is started.

Step 7 Testing the ANFIS model
The model is tested after the training.

Step 8 Find the test output of the ANFIS model
Table 19 shows the parameter values used in testing with the output of the model. Finally, the obtained test output results with ANFIS model are compared with the measured values.

Table 19 Comparison of measured and predicted makespan

Sr. no.	Actual makespan	Calculated makespan	Error in %	Accuracy in %
1	901	894	− 0.78	99.22
2	941	923	− 1.91	98.09
3	834	904	8.39	91.61
4	905	884	− 2.32	97.68
5	916	912	− 0.44	99.56
6	816	879	7.72	92.28
7	813	857	5.41	94.59
8	904	956	5.75	94.25
9	767	824	7.43	92.57
10	698	751	7.59	92.41
11	741	742	0.13	99.87
12	672	691	2.83	97.17
13	696	665	− 4.45	95.55
14	734	679	− 7.49	92.51
15	736	729	− 0.95	99.05
16	666	657	− 1.35	98.65
17	718	724	0.84	99.16
18	548	552	0.73	99.27
19	676	633	− 6.36	93.64
20	549	558	1.64	98.36
21	703	652	− 7.25	92.75
22	502	506	0.80	99.20
23	545	521	− 4.40	95.60
24	608	549	− 9.70	90.30

Step 9 Plot correlation coefficient between measured and predicted makespan

Correlation coefficient is a statistical process for estimating the relationships among variables, i.e. prediction of ANFIS model and the measured data used for the testing. Correlation coefficient is widely used for prediction. After obtaining the output of ANFIS model, a plot is drawn between the predicted data of ANFIS model and measured data set. Correlation coefficient of ANFIS model is shown in Fig. 3.

7 Model verification

Twenty-four random readings were used as the testing data set (Table 19). The plot of 24 measured makespan values versus predicted makespan using the ANFIS model is shown in Fig. 4. This figure presents a comparison of the measured makespan and predicted makespan of the testing data set of 24 following training using ANFIS. Appropriate assent is evident between the measured and ANFIS-predicted makespan values. This close assent obviously displays that the ANFIS model can be used to predict the makespan under consideration. Thus, the proposed ANFIS model offers a promising solution to predicting makespan values in the specific range of parameters.

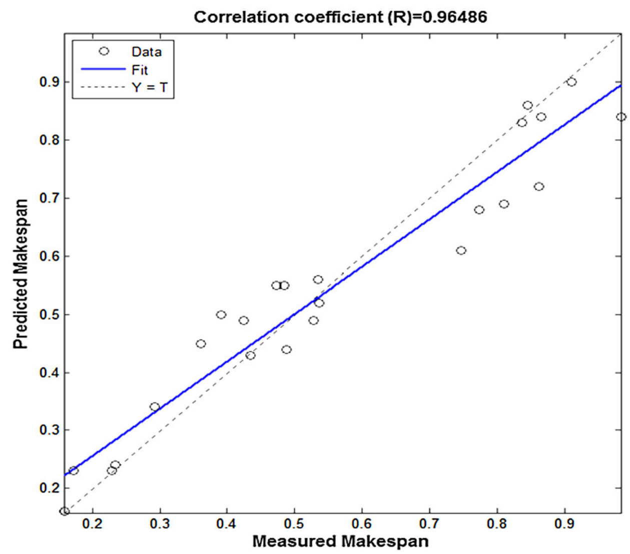

Fig. 3 Correlation Coefficient of ANFIS data

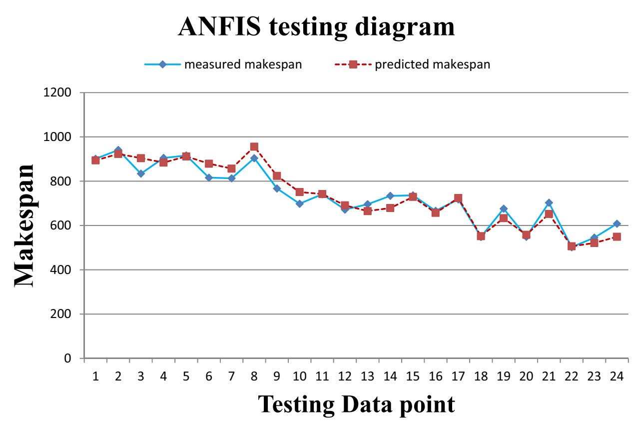

Fig. 4 Measured makespan versus predicted makespan

To assess the ANFIS model, the percentage error E_i and average percentage error E_{av} defined in Eqs. (11) and (12), respectively, were used.

$$E_i = \frac{|measured\ makespan - predicted\ makespan|}{measured\ makespan} \times 100 \tag{11}$$

$$E_{av} = \frac{1}{m}\sum_{i=1}^{m} E_i \tag{12}$$

where E_i is the percentage error of sample number i; and E_{av} is the average percentage error of m sample data.

From Table 19 and Fig. 5 show that the average percentage error for predicting makespan is 4.03%. Figure 5

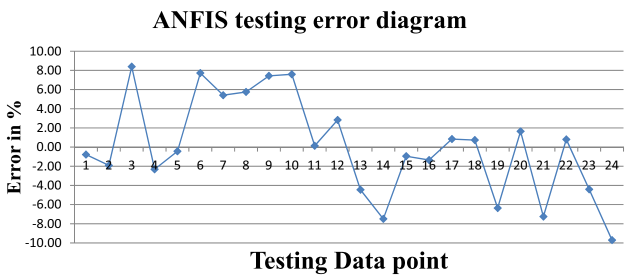

Fig. 5 The error percentage

presents the percentage error between the predicted and measured makespan. The highest percentage of error for ANFIS model prediction is 9.7%. The low error level signifies that the makespan results predicted by ANFIS are very close to the actual results. The error and accuracy values mean that the proposed model can predict makespan satisfactorily.

8 Conclusion

In this study, ANFIS was used to develop an empirical model for predicting the makespan of flexible manufacturing system assembly shop in a manufacturing plant. An ANFIS model was developed based on NEH heuristics for makespan calculation as a scheduling problem. The ANFIS model was developed into two phases, namely training phase and test phase. In the training phase, about 90 values, i.e. 79% of the problems were used and 24 values, i.e. 21% of the problems used for the testing phase. This model was verified by test data, and the 95.97 average percentage of accuracy was achieved. Therefore, it can be concluded that makespan calculation of the production system, by the proposed ANFIS with NEH heuristic rules can be used as a reliable approach in estimating the job completion time of the problem studied. ANFIS shows a good performance with a coefficient of determination is 0.9310 and root-mean-square error (RMSE) of 0.0731. The RMSE is a frequently used measure of the differences between values predicted by a model or an estimator and the values actually observed, and coefficient of determination, describes how much of the variance between the two variables is described by the linear fit. Coefficient of determination of 0.9310 means that 93.10% of the variance is predictable. Correlation coefficient between measured and predicted makespan is also shown in a graphical way (Fig. 3). The value of Correlation coefficient is 0.9649. The results mutually differ less than ± 10%. The correlation coefficient is close to 1 i.e. 0.9649, it would indicate that the variables are positively linearly related and the scatter plot falls almost along a straight line with positive slope. The derived values of ANFIS model output are found within the range after being verified practically. Therefore, it can be concluded that makespan calculation of the production system, by the proposed adaptive neuro-fuzzy inference system, can be used as a reliable approach in estimating the makespan of flexible manufacturing system assembly shop.

Acknowledgement We would like to thank everyone that participated in this research work, in particular Dr. Salim Heddam, Associate Professor, Faculty of Science, Agronomy Department, Hydraulic Division University 20 Août 1955 SKIKDA 21000 Route EL HADAIK, BP 26, SKIKDA, Algeria for the help in methodology. We express our gratitude all the anonymous reviewers of this paper for their valuable suggestions, who have helped to improve the quality of this paper.

References

Abdulshahed AM, Longstaff AP, Fletcher S (2015) The application of ANFIS prediction models for thermal error compensation on CNC machine tools. Appl Soft Comput 27:158–168

Ahmadizar F, Ghazanfari M, Ghomi SMTF (2010) Group shops scheduling with makespan criterion subject to random release dates and processing times. Comput Oper Res 37:152–162

Akyol DE (2004) Application of neural networks to heuristic scheduling algorithms. Comput Ind Eng 46:679–696. https://doi.org/10.1016/j.cie.2004.05.005

Ay M, Kisi O (2014) Modelling of chemical oxygen demand by using anns, anfis and k-means clustering techniques. J Hydrol 511:279–289. https://doi.org/10.1016/j.jhydrol.2014.01.054

Azadeh A, Hosseini N, Zadeh SA, Jalalvand F (2015) A hybrid computer simulation-adaptive neuro-fuzzy inference system algorithm for optimization of dispatching rule selection in job shop scheduling problems under uncertainty. Int J Adv Manuf Technol 79:135–145

Campbell HG, Dudek RA, Smith ML (1970) A heuristic algorithm for the n job, m machine sequencing problem. Manag Sci 16:630–637

Çevik HH, Çunkaş M (2015) Short-term load forecasting using fuzzy logic and ANFIS. Neural Comput Appl. https://doi.org/10.1007/s00521-014-1809-4

Chen M-Y (2013) A hybrid ANFIS model for business failure prediction utilizing particle swarm optimization and subtractive clustering. Inf Sci 220:180–195. https://doi.org/10.1016/j.ins.2011.09.013

Chen W, Muraki M (1997) An action strategy generation framework for an on-line scheduling and control system in batch processes with neural networks. Int J Prod Res 35:3483–3508

Chen SC, Le DK, Nguyen VS (2014) Adaptive network-based fuzzy inference system (ANFIS) controller for an active magnetic bearing system with unbalance mass. In: Zelinka I, Duy V, Cha J (eds) AETA 2013: recent advances in electrical engineering and related sciences. Lecture notes in electrical engineering, vol. 282, Springer, Berlin

Cheng T, Gupta M (1989) Survey of scheduling research involving due date determination decisions. Eur J Oper Res 38:156–166

Cus F, Balic J (2003) Optimization of cutting process by GA approach. Robot Comput Integr Manuf 19:113–121

Der Jeng M, Lin CS, Huang YS (1999) Petri net dynamics-based scheduling of flexible manufacturing systems with assembly. J Intell Manuf 10:541–555

Dong X, Huang H, Chen P (2008) An improved NEH-based heuristic for the permutation flowshop problem. Comput Oper Res 35:3962–3968

Framinan JM, Perez-Gonzalez P (2015) On heuristic solutions for the stochastic flowshop scheduling problem. Eur J Oper Res 246:413–420

Fransoo JC, de Kok TG, Paulli J (1995) Makespan estimations in flexible manufacturing systems working papers, Department of Mathematical Sciences, University of Aarhus

González MA, Vela CR, González-Rodríguez I, Varela R (2013) Lateness minimization with Tabu search for job shop scheduling problem with sequence dependent setup times. J Intell Manuf 24:741–754

Güneri AF, Ertay T, YüCel A (2011) An approach based on ANFIS input selection and modeling for supplier selection problem. Expert Syst Appl 38:14907–14917. https://doi.org/10.1016/j.eswa.2011.05.056

Gupta JN (1972) Heuristic algorithms for multistage flowshop scheduling problem. AIIE Trans 4:11–18

Heddam S (2014) Modeling hourly dissolved oxygen concentration (DO) using two different adaptive neuro-fuzzy inference systems (ANFIS): a comparative study. Environ Monit Assess 186:597–619

Heddam S, Bermad A, Dechemi N (2012) ANFIS-based modelling for coagulant dosage in drinking water treatment plant: a case study. Environ Monit Assess 184:1953–1971

Ho W-H, Tsai J-T, Lin B-T, Chou J-H (2009) Adaptive network-based fuzzy inference system for prediction of surface roughness in end milling process using hybrid Taguchi-genetic learning algorithm. Expert Syst Appl 36:3216–3222. https://doi.org/10.1016/j.eswa.2008.01.051

Ivanescu CV, Fransoo JC, Bertrand JWM (2002) Makespan estimation and order acceptance in batch process industries when processing times are uncertain. OR Spectr 24:467–495

Jain V, Raj T (2016a) Modeling and analysis of FMS performance variables by ISM, SEM and GTMA approach. Int J Prod Econ 171:84–96. https://doi.org/10.1016/j.ijpe.2015.10.024

Jain V, Raj T (2016b) Tool life management of unmanned production system based on surface roughness by ANFIS. Int J Syst Assur Eng Manag. https://doi.org/10.1007/s13198-016-0450-2

Jang JS (1993) ANFIS: adaptive-network-based fuzzy inference system. Syst Man Cybern IEEE Trans 23:665–685. https://doi.org/10.1109/21.256541

Johnson SM (1954) Optimal two-and three-stage production schedules with setup times included. Naval Res Logist Q 1:61–68

Jung SH, Choi S-U (2015) Prediction of composite suitability index for physical habitat simulations using the ANFIS method. Appl Soft Comput 34:502–512

Kalczynski PJ, Kamburowski J (2007) On the NEH heuristic for minimizing the makespan in permutation flow shops. Omega 35:53–60

Kalczynski PJ, Kamburowski J (2008) An improved NEH heuristic to minimize makespan in permutation flow shops. Comput Oper Res 35:3001–3008

Li S, Li Y, Liu Y, Xu Y (2007) A GA-based NN approach for makespan estimation. Appl Math Comput 185:1003–1014. https://doi.org/10.1016/j.amc.2006.07.024

Maher I, Eltaib M, Sarhan AA, El-Zahry R (2014) Investigation of the effect of machining parameters on the surface quality of machined brass (60/40) in CNC end milling—ANFIS modeling. Int J Adv Manuf Technol 74:531–537

Maher I, Eltaib M, Sarhan AA, El-Zahry R (2015) Cutting force-based adaptive neuro-fuzzy approach for accurate surface roughness prediction in end milling operation for intelligent machining. Int J Adv Manuf Technol 76:1459–1467

Mar J, Lin F-J (2001) An ANFIS controller for the car-following collision prevention system. Veh Technol IEEE Trans 50:1106–1113

Mellit A, Kalogirou SA (2011) ANFIS-based modelling for photovoltaic power supply system: a case study. Renew Energy 36:250–258. https://doi.org/10.1016/j.renene.2010.06.028

Moradinasab N, Shafaei R, Rabiee M, Ramezani P (2013) No-wait two stage hybrid flow shop scheduling with genetic and adaptive imperialist competitive algorithms. J Exp Theor Artif Intell 25:207–225

Nawaz M, Enscore EE, Ham I (1983) A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. Omega 11:91–95

Onwubolu GC (1996) A flow-shop manufacturing scheduling system with interactive computer graphics. Int J Oper Prod Manag 16:74–84

Özkan G, İnal M (2014) Comparison of neural network application for fuzzy and ANFIS approaches for multi-criteria decision making problems. Appl Soft Comput 24:232–238

Philipoom PR, Rees LP, Wiegmann L (1994) Using neural networks to determine internally-set due-date assignments for shop scheduling. Decis Sci 25:825–851

Pousinho H, Mendes V, Catalão J (2012) Short-term electricity prices forecasting in a competitive market by a hybrid PSO–ANFIS approach. Int J Electr Power Energy Syst 39:29–35. https://doi.org/10.1016/j.ijepes.2012.01.001

Raaymakers WH, Weijters A (2003) Makespan estimation in batch process industries: a comparison between regression analysis and neural networks. Eur J Oper Res 145:14–30

Raaymakers HM, Bertrand JWM, Fransoo JC (2001) Makespan estimation in batch process industries using aggregate resource and job set characteristics. Int J Prod Econ 70(2):145–161

Sabuncuoglu I (1998) Scheduling with neural networks: a review of the literature and new research directtions. Prod Plan Control 9:2–12

Sabuncuoglu I, Gurgun B (1996) A neural network model for scheduling problems. Eur J Oper Res 93:288–299

Samanta B (2009) Surface roughness prediction in machining using soft computing. Int J Comput Integr Manuf 22:257–266. https://doi.org/10.1080/09511920802287138

Shafaei R, Rabiee M, Mirzaeyan M (2011) An adaptive neuro fuzzy inference system for makespan estimation in multiprocessor no-wait two stage flow shop. Int J Comput Integr Manuf 24:888–899

Shokrollahpour E, Zandieh M, Dorri B (2011) A novel imperialist competitive algorithm for bi-criteria scheduling of the assembly flowshop problem. Int J Prod Res 49:3087–3103

Svalina I, Simunovic G, Simunovic K (2013) Machined surface roughness prediction using adaptive neurofuzzy inference system. Appl Artif Intell 27:803–817. https://doi.org/10.1080/08839514.2013.835233

Taillard E (1990) Some efficient heuristic methods for the flow shop sequencing problem. Eur J Oper Res 47:65–74

Talei A, Chua LHC, Wong TS (2010) Evaluation of rainfall and discharge inputs used by adaptive network-based fuzzy inference systems (ANFIS) in rainfall–runoff modeling. J Hydrol 391:248–262. https://doi.org/10.1016/j.jhydrol.2010.07.023

Vasileva-Stojanovska T, Vasileva M, Malinovski T, Trajkovik V (2015) An ANFIS model of quality of experience prediction in education. Appl Soft Comput 34:129–138. https://doi.org/10.1016/j.asoc.2015.04.047

Verma A, Cherkasova L, Campbell RH (2012) Two sides of a coin: optimizing the schedule of mapreduce jobs to minimize their makespan and improve cluster performance. In: 2012 IEEE 20th international symposium on modeling, analysis & simulation of computer and telecommunication systems (MASCOTS). IEEE, pp 11–18

Wilson AD, King RE, Wilson JR (2004) Case study on statistically estimating minimum makespan for flow line scheduling problems. Eur J Oper Res 155:439–454

Wittrock RJ (1985) Scheduling algorithms for flexible flow lines. IBM J Res Dev 29:401–412

Yagmahan B, Yenisey MM (2008) Ant colony optimization for multi-objective flow shop scheduling problem. Comput Ind Eng 54:411–420

Yih Y, Liang T-P, Moskowitz H (1991) A hybrid approach for crane scheduling problems. In: Dagli CH, Kumara SRT, Shin YC (eds) Intelligent engineering systems through artificial neural networks. ASME, New York, pp 867–872

Zheng D-Z, Wang L (2003) An effective hybrid heuristic for flow shop scheduling. Int J Adv Manuf Technol 21:38–44