

A modified branch and bound algorithm for a vague flow-shop scheduling problem

H. Gholizadeh¹, H. Fazlollahtabar² and R. Gholizadeh³

¹ *Department of Industrial Engineering, Mazandaran University of Science and Technology, Babol, Iran*

² *Department of Industrial Engineering, School of Engineering, Damghan University, Damghan, Iran*

³ *Department of Statistics, University of Campinas, R. Sérgio Buarque de Holanda, 651, Campinas (CEP 13083-859), Brazil.*

hadi.gh1988@gmail.com, hfazl@du.ac.ir, ramin.gholizadh@gmail.com

Abstract

Uncertainty plays a significant role in modeling and optimization of real world systems. Among uncertain approaches, fuzziness describes impreciseness while for ambiguity another definition is required. Vagueness is a probabilistic model of uncertainty being helpful to include ambiguity into modeling different processes especially in industrial systems. In this paper, a vague set based on distance is used to model a flow-shop scheduling problem being an important problem in assembly production systems. The vagueness being used as octagon numbers are employed to represent vague processes for the manufacturing system. As a modeling effort, first a flow-shop scheduling problem is handled with vagueness. Then, for solving and analyzing the proposed vague flow-shop scheduling model, a modified Branch and Bound algorithm is proposed. As an implementation, an example is used to explain the performance and to analyze the sensitivity of the proposed vague approach. The validity of the proposed model and modified algorithm is demonstrated through a robust ranking technique. The outputs help the decision makers to counteract the vagueness and handle operational decisions in flow-shop scheduling problems within dynamic environments.

Keywords: Flow-shop scheduling, branch and bound, octagonal vague numbers, ranking methods.

1 Introduction

Vague sets can be used to describe many processes and systems under uncertainty. From the combination of vague and random uncertainty factors, a new variable can be obtained as a random vague variable. In this paper, we use interval-based sets, which, instead of point-based membership in fuzzy sets, use interval-based membership in an obscure set. Several techniques are proposed for managing uncertainty. To solve vague situations in real problems, the first systemic approach related to vague set theory was successfully applied to many areas such as in scheduling problems. In recent studies, scheduling problems were fuzzified by using the concept of vague due date and processing times. Each job has the same routing through machines and the sequence of operations is fixed in a flow-shop. Branch- and- bound algorithms are the mostly used optimal technique to solve such type of problems excluding heuristic methods McCahon S. and lee E.S. [19], McCahon, C. S., & Lee, E. S [20], Ishibuchi H. and Lee K.H. [11], Martin L. and Roberto T. [18]. Branch and bound is an algorithm for separate and synthesis mathematical optimization problems. A branch-and-bound algorithm includes a systematic step by step solution space search. A set of candidate solutions is considered to form the root of the tree solution. The algorithm explores branches of the tree, which show subsets of the solution set. Before enumerating the solutions of a branch, the branch is checked against upper and lower estimated bounds on the optimal solution and is discarded if it cannot produce a better solution than the best one found so far by the algorithm. Ranking fuzzy numbers has a remarkable role in approximate reasoning, decision making, optimization, forecasting and some other fuzzy application systems. In decision-making problems, the value and importance of different options are commonly expressed as vague numbers. Robust ranking method satisfies the properties of compensation, linearity and

additively Nagarajan R. & Solairaju A. [22]. Therefore, by using the robust ranking technique the validity of the best output in comparison with others can be demonstrated.

The remainder of the paper is organized as follows. Next, a review of related literature is given. In Section 3, vagueness and vague environment are described and vague numbers are configured. In Section 4, the modified vague branch and bound method is proposed. Numerical example and the related analyses are presented in Section 5. Finally, conclusions are expressed in Section 6.

2 Literature review

Branch and bound is an algorithm depending on the efficient estimation of the lower and upper bounds of a region/branch of the search space and approaches exhaustive enumeration as the size (n -dimensional volume) of the region tends to zero. Branch and bound technique is an integer programming solving method. Combining branch and bound and fuzzy set theory to handle flow-shop scheduling problem was extensively investigated Cheng J. Kise H. & Matsumoto H. [4]. In Gupta D., Sharma S. and Aggarwal S. [9], flow shop scheduling on 2-machines with setup time and single transport facility under fuzzy environment was discussed. In Ambika G. & Uthra G. [1], branch and bound technique in flow-shop scheduling using fuzzy processing times was presented. In Dhanalakshmi, V. & Kennedy F. C. [6], some aggregation operations on octagonal fuzzy numbers and its applications to decision making was shown. In Temiz I. & Erol S. [27], fuzzy branch and bound algorithm for flow shop scheduling was shown. In Malini S. U., & Kennedy F. C. [17], an approach for solving fuzzy transportation problem using octagonal fuzzy numbers was developed.

McCahon S. and lee E.S. [19], discussed the job sequencing with fuzzy processing time. Ishibuchi H. and Lee, K.H. [11], addressed the formulation of fuzzy flow shop scheduling problem with fuzzy processing time. Hong, T. and Chuang, T. [10], developed a triangular Johnson algorithm. Martin, L. and Roberto T. [18], developed fuzzy scheduling with application to real time systems. Some of the noteworthy approaches are due to Johnson, S. M. [12], Bagga, P. C. [2], Baker K.R. (1974) [3], Yager R.R. [28], Cynthia, S. MacCahon [20], Maccarthy B.L. & Liu J. [16], Shukla C.S. & Chen F.F. (1996) [24], Cowling P.L. & Johanson M. [5], Sanuja P. & Xueyan S. [23], Singh, T. P., & Allawalia, P. [25, 26].

In Mehrabian, A., Tavakkoli-Moghaddam, R., & Khalili-Damaghani, K. [21], a two-objective mathematical programming model was presented to integrate flow shop scheduling and routing robots in a flexible manufacturing system considering fuzzy demand, due dates and processing times. The proposed fuzzy mathematical model was solved using Non-dominated Sorting Genetic Algorithm-II (NSGAI) and multi-objective particle swarm optimization (MOPSO).

In Khorram, E., & Nozari, V. [15], a new multi-objective fuzzy optimization problem in which each objective had a different level was studied. Since, the feasible domain was non-convex; the traditional methods cannot be applied and thus by specific techniques the model was transformed to a similar 0 – 1 integer programming and was solved by a branch and bound algorithm.

Essentially, in a fuzzy set each element is associated with a point value selected from the unit interval $[0, 1]$; A vague set, as well as an intuitionistic fuzzy set, is a further extension of a fuzzy set. Instead of using point-relying membership as in fuzzy sets, interval-based membership is used in a vague set Gau W.L. and Buehrer D.J. [7], Gonzalez-Lpez V. et al. [8], defined vague sets and described the optimization of queuing theory based on vague environment.

To sum up, by reviewing the related literature, mostly fuzzy set theory was employed to model uncertainty in flow-shop scheduling. Since, ambiguity is inevitable in real industrial cases, this paper propose to model a vague flow-shop scheduling problem. Then, as a generic and efficient solution approach vague branch and bound algorithm is developed.

3 Vague environment

Basic definitions and formulations of vague environment follow here.

Definition 3.1. [8, 14, 13] *A vague set \tilde{V} in a universe of discourse U is characterized by a true membership function, $t_{\tilde{V}}$, and a false membership function, $f_{\tilde{V}}$, as follows: $t_{\tilde{V}} : U \rightarrow [0, 1]$, $f_{\tilde{V}} : U \rightarrow [0, 1]$, and $t_{\tilde{V}}(u) + f_{\tilde{V}}(u) \leq 1$, where $t_{\tilde{V}}$ is a lower bound on the grade of membership of u derived from the evidence for u , and $f_{\tilde{V}}$ is a lower bound on the grade of membership of the negation of u derived from the evidence against u .*

Suppose $U = \{u_1, u_2, \dots, u_n\}$. A vague set \tilde{V} of the universe of discourse U can be represented by $\tilde{V} = \sum_{i=1}^n [\tilde{V}(u_i), 1 - f_{\tilde{V}}(u_i)] / u_i$, where $0 \leq t(u_i) \leq 1 - f(u_i) \leq 1$ and $1 \leq i \leq n$. In other words, the grade of membership of u_i is bounded to a subinterval $[t(u_i), 1 - f_{\tilde{V}}(u_i)]$ of $[0, 1]$. Thus, vague sets are generalization of Fuzzy sets, since the grade of membership $\mu_{\tilde{V}}(u)$ of u in Definition 3.1 may be inexact in a vague set. We now depict a vague set in Figure1.

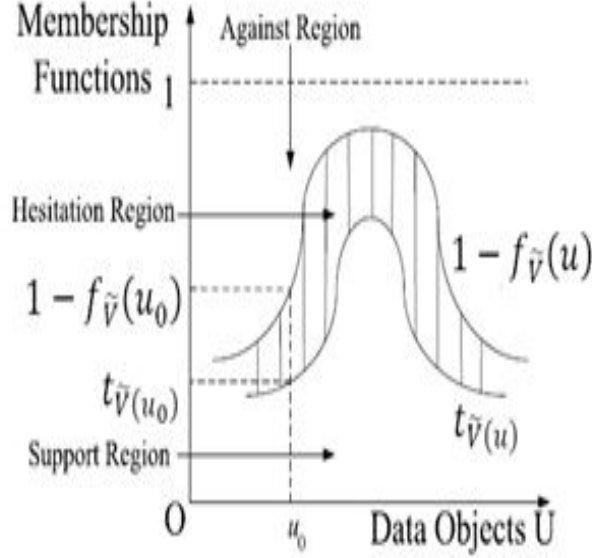


Figure 1: Membership Functions of a vague set.

Definition 3.2. [8, 14, 13] The complement of a vague set \tilde{V} is denoted by \tilde{V}' and is defined by

$$t_{\tilde{V}'}(u) = f_{\tilde{V}}(u), \quad 1 - f_{\tilde{V}'}(u) = 1 - f_{\tilde{V}}(u),$$

Definition 3.3. [8, 14, 13] Let $\tilde{V}_{t_{\tilde{V}}, f_{\tilde{V}}}$ be a vague set of U . Then, we define α_t -cuts and α_f -cuts of \tilde{V} as the crisp sets of U given by

$$\begin{aligned} \tilde{V}_{\alpha_t} &= \{u : t_{\tilde{V}}(u) \geq \alpha_t\}, & \alpha_t &\in [0, 1], \\ \tilde{V}_{\alpha_f} &= \{u : 1 - f_{\tilde{V}}(u) \geq \alpha_f\}, & \alpha_f &\in [0, 1]. \end{aligned}$$

Definition 3.4. [8, 14, 13] A vague set $\tilde{V}_{t_{\tilde{V}}, f_{\tilde{V}}}$ of \mathbb{R} with continuous membership functions $t_{\tilde{V}}$ and $f_{\tilde{V}}$ is called a vague number if and only if \tilde{V}_{α_t} and \tilde{V}_{α_f} , for all $\alpha_t, \alpha_f \in (0, 1]$, are bounded closed intervals; i.e.

$$\tilde{V}_{\alpha_t} = [\tilde{V}_{\alpha_t}^L, \tilde{V}_{\alpha_t}^U], \quad \tilde{V}_{\alpha_f} = [\tilde{V}_{\alpha_f}^L, \tilde{V}_{\alpha_f}^U].$$

We denote the class of all vague numbers by $V(\mathbb{R})$.

Definition 3.5. [8] The vague number $\tilde{V}_{t_{\tilde{V}}, f_{\tilde{V}}}$ is called a triangular vague number, if

$$t_{\tilde{V}}(u) = \begin{cases} \frac{u-a_1}{w(a_2-a_1)} & \text{if } a_1 \leq u \leq a_2 \\ \frac{a_3-u}{w(a_3-a_2)} & \text{if } a_2 \leq u \leq a_3 \\ 0 & \text{if otherwise} \end{cases} \quad . \quad 1 - f_{\tilde{V}}(u) = \begin{cases} \frac{u-a_1}{a_2-a_1} & \text{if } a_1 \leq u \leq a_2 \\ \frac{a_3-u}{a_3-a_2} & \text{if } a_2 \leq u \leq a_3 \\ 0 & \text{if otherwise} \end{cases} .$$

where $w \in [1, \infty)$. We denote such a vague number by $\tilde{V} = (a_1, a_2, a_3, w)_T$.

3.1 Octagonal vague numbers

An Octagonal vague number denoted by \tilde{A}_w is defined to be the ordered quadruple $\tilde{A}_w = (l_1(r), s_1(t), s_2(t), l_2(r))$ for $r \in [0, K]$ and $t \in [K, W]$ where

- $l_1(r)$ is a bounded left continuous non decreasing function over $[0, w_1]$, $0 \leq w_1 \leq K$
- $s_1(r)$ is a bounded left continuous non decreasing function over $[K, w_2]$, $K \leq w_2 \leq W$
- $s_2(r)$ is a bounded left continuous non decreasing function over $[K, w_2]$, $K \leq w_2 \leq W$
- $l_2(r)$ is a bounded left continuous non decreasing function over $[0, w_1]$, $0 \leq w_1 \leq K$

A fuzzy number \tilde{A} is a normal octagonal vague number denoted by $(a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8)$ where $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8$ are real numbers and its membership function is given below

$$t_{\tilde{V}}(u) = \begin{cases} 0 & a \leq a_1 \\ k \left(\frac{a-a_1}{w(a_2-a_1)} \right) & a_1 \leq a \leq a_2 \\ k & a_2 \leq a \leq a_3 \\ k + (1-k) \left(\frac{a-a_3}{w(a_4-a_3)} \right) & a_3 \leq a \leq a_4 \\ 1 & a_4 \leq a \leq a_5 \\ k + (1-k) \left(\frac{a_6-a}{w(a_6-a_5)} \right) & a_5 \leq a \leq a_6 \\ k & a_6 \leq a \leq a_7 \\ k \left(\frac{a_8-a}{w(a_8-a_7)} \right) & a_7 \leq a \leq a_8 \\ 0 & a > a_8 \end{cases} \quad 1 - f_{\tilde{V}}(u) = \begin{cases} 0 & a \leq a_1 \\ k \left(\frac{a-a_1}{a_2-a_1} \right) & a_1 \leq a \leq a_2 \\ k & a_2 \leq a \leq a_3 \\ k + (1-k) \left(\frac{a-a_3}{a_4-a_3} \right) & a_3 \leq a \leq a_4 \\ 1 & a_4 \leq a \leq a_5 \\ k + (1-k) \left(\frac{a_6-a}{a_6-a_5} \right) & a_5 \leq a \leq a_6 \\ k & a_6 \leq a \leq a_7 \\ k \left(\frac{a_8-a}{a_8-a_7} \right) & a_7 \leq a \leq a_8 \\ 0 & a > a_8 \end{cases} .$$

Where $0 < k < 1$. According to the above definition, the normal octagonal vague number \tilde{A} is the ordered quadruple $(l_1(r), s_1(t), s_2(t), l_2(r))$ for $r \in [0, K]$ and $t \in [K, 1]$ where

$$t_{\tilde{l}_1}(r) = K \left(\frac{r-a_1}{w(a_2-a_1)} \right), \quad 1 - f_{\tilde{l}_1}(r) = K \left(\frac{r-a_1}{a_2-a_1} \right), \quad t_{\tilde{s}_1}(t) = K + (1-K) \left(\frac{t-a_3}{w(a_4-a_3)} \right)$$

$$1 - f_{\tilde{s}_1}(t) = K + (1-K) \left(\frac{t-a_3}{a_4-a_3} \right), \quad t_{\tilde{s}_2}(t) = K + (1-K) \left(\frac{a_6-t}{w(a_6-a_5)} \right)$$

$$1 - f_{\tilde{s}_2}(t) = K + (1-K) \left(\frac{a_6-t}{a_6-a_5} \right), \quad t_{\tilde{l}_2}(r) = K \left(\frac{a_8-r}{w(a_8-a_7)} \right), \quad 1 - f_{\tilde{l}_2}(r) = K \left(\frac{a_8-r}{a_8-a_7} \right)$$

$$\widetilde{l_1(r)}_{\alpha_t} = \left[w \left(\frac{\alpha_t}{K} \right) (a_2 - a_1) + a_1, a_3 - w \left(\frac{\alpha_t}{K} \right) (a_3 - a_2) \right]$$

$$\widetilde{l_1(r)}_{\alpha_f} = \left[\left(\frac{\alpha_f}{K} \right) (a_2 - a_1) + a_1, a_3 - \left(\frac{\alpha_f}{K} \right) (a_3 - a_2) \right]$$

$$\widetilde{l_2(r)}_{\alpha_t} = \left[w \left(\frac{\alpha_t}{K} \right) (a_8 - a_7) + a_7, a_8 - w \left(\frac{\alpha_t}{K} \right) (a_8 - a_7) \right]$$

$$\widetilde{l_2(r)}_{\alpha_f} = \left[\left(\frac{\alpha_f}{K} \right) (a_8 - a_7) + a_7, a_8 - \left(\frac{\alpha_f}{K} \right) (a_8 - a_7) \right]$$

$$\widetilde{s_1(t)}_{\alpha_t} = \left[w \left(\frac{\alpha_t - K}{1 - K} \right) (a_4 - a_3) + a_3, a_4 - w \left(\frac{\alpha_t - K}{1 - K} \right) (a_4 - a_3) \right]$$

$$\widetilde{s_1(t)}_{\alpha_f} = \left[\left(\frac{\alpha_f - K}{1 - K} \right) (a_4 - a_3) + a_3, a_4 - \left(\frac{\alpha_f - K}{1 - K} \right) (a_4 - a_3) \right]$$

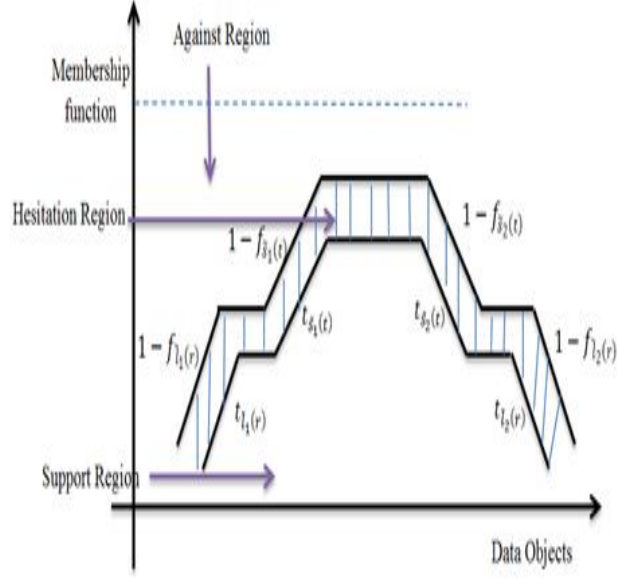


Figure 2: Membership functions of an octagonal vague number.

$$\begin{aligned}\widetilde{s}_2(t)_{\alpha_t} &= \left[w \left(\frac{\alpha_t - K}{1 - K} \right) (a_6 - a_5) + a_5, a_6 - w \left(\frac{\alpha_t - K}{1 - K} \right) (a_6 - a_5) \right] \\ \widetilde{s}_2(t)_{\alpha_f} &= \left[\left(\frac{\alpha_f - K}{1 - K} \right) (a_6 - a_5) + a_5, a_6 - \left(\frac{\alpha_f - K}{1 - K} \right) (a_6 - a_5) \right]\end{aligned}$$

Let $\tilde{A} = (a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8)$ and $\tilde{B} = (b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8)$ be two normal octagonal fuzzy numbers therefore algebraic operations using the alpha-cut are as follows:

$$\begin{aligned}\tilde{A}_{\alpha_t}^L &= \left[w \left(\frac{\alpha_t}{K} \right) (a_2 - a_1) + a_1, w \left(\frac{\alpha_t - K}{1 - K} \right) (a_4 - a_3) + a_3, w \left(\frac{\alpha_t - K}{1 - K} \right) (a_6 - a_5) + a_5, w \left(\frac{\alpha_t}{K} \right) (a_8 - a_7) + a_7 \right] \\ \tilde{B}_{\alpha_t}^L &= \left[w \left(\frac{\alpha_t}{K} \right) (b_2 - b_1) + b_1, w \left(\frac{\alpha_t - K}{1 - K} \right) (b_4 - b_3) + b_3, w \left(\frac{\alpha_t - K}{1 - K} \right) (b_6 - b_5) + b_5, w \left(\frac{\alpha_t}{K} \right) (b_8 - b_7) + b_7 \right] \\ \tilde{A}_{\alpha_t}^U &= \left[a_3 - w \left(\frac{\alpha_t}{K} \right) (a_3 - a_2), a_4 - w \left(\frac{\alpha_t - K}{1 - K} \right) (a_4 - a_3), a_6 - w \left(\frac{\alpha_t - K}{1 - K} \right) (a_6 - a_5), a_8 - w \left(\frac{\alpha_t}{K} \right) (a_8 - a_7) \right] \\ \tilde{B}_{\alpha_t}^U &= \left[b_3 - w \left(\frac{\alpha_t}{K} \right) (b_3 - b_2), b_4 - w \left(\frac{\alpha_t - K}{1 - K} \right) (b_4 - b_3), b_6 - w \left(\frac{\alpha_t - K}{1 - K} \right) (b_6 - b_5), b_8 - w \left(\frac{\alpha_t}{K} \right) (b_8 - b_7) \right] \\ \tilde{A}_{\alpha_f}^L &= \left[\left(\frac{\alpha_f}{K} \right) (a_2 - a_1) + a_1, w \left(\frac{\alpha_f - K}{1 - K} \right) (a_4 - a_3) + a_3, \left(\frac{\alpha_f - K}{1 - K} \right) (a_6 - a_5) + a_5, \left(\frac{\alpha_f}{K} \right) (a_8 - a_7) + a_7 \right] \\ \tilde{B}_{\alpha_f}^L &= \left[\left(\frac{\alpha_f}{K} \right) (b_2 - b_1) + b_1, \left(\frac{\alpha_f - K}{1 - K} \right) (b_4 - b_3) + b_3, \left(\frac{\alpha_f - K}{1 - K} \right) (b_6 - b_5) + b_5, \left(\frac{\alpha_f}{K} \right) (b_8 - b_7) + b_7 \right] \\ \tilde{A}_{\alpha_f}^U &= \left[a_3 - \left(\frac{\alpha_f}{K} \right) (a_3 - a_2), a_4 - \left(\frac{\alpha_f - K}{1 - K} \right) (a_4 - a_3), a_6 - \left(\frac{\alpha_f - K}{1 - K} \right) (a_6 - a_5), a_8 - \left(\frac{\alpha_f}{K} \right) (a_8 - a_7) \right] \\ \tilde{B}_{\alpha_f}^U &= \left[b_3 - \left(\frac{\alpha_f}{K} \right) (b_3 - b_2), b_4 - \left(\frac{\alpha_f - K}{1 - K} \right) (b_4 - b_3), b_6 - \left(\frac{\alpha_f - K}{1 - K} \right) (b_6 - b_5), b_8 - \left(\frac{\alpha_f}{K} \right) (b_8 - b_7) \right]\end{aligned}$$

3.2 Ranking of octagonal vague numbers

The characteristics of a system can be described by its membership function. Since input information is ambiguous, and in decision-making problems, the value and importance of different options are expressed as vague numbers. Therefore, by using the robust ranking technique, the validity relative to each of the mentioned alternatives is demonstrated.

Remark 3.6. The measure of an octagonal vague number is obtained by the average of the two vague side areas, left side area and right side area, from the membership function to axis. Robust ranking technique which satisfies compensation, linearity, and additively properties and provides results which are consistent with human intuition.

Definition 3.7. Let \tilde{A} be a normal octagonal vague number. The value $R(\tilde{A}_\alpha)$ called the measure of \tilde{A} is calculated as follows:

$$\begin{aligned} R(\tilde{A}_{\alpha_t}) &= \int_0^1 (0.5) [\tilde{l}_1(r)_{\alpha_t}^L, \tilde{l}_1(r)_{\alpha_t}^U] d\alpha_t + \int_0^1 (0.5) [\tilde{s}_1(r)_{\alpha_t}^L, \tilde{s}_1(r)_{\alpha_t}^U] d\alpha_t \\ R(\tilde{A}_{\alpha_t}) &= \int_0^1 (0.5) [\tilde{l}_2(r)_{\alpha_t}^L, \tilde{l}_2(r)_{\alpha_t}^U] d\alpha_t + \int_0^1 (0.5) [\tilde{s}_2(r)_{\alpha_t}^L, \tilde{s}_2(r)_{\alpha_t}^U] d\alpha_t \\ R(\tilde{A}_{\alpha_f}) &= \int_0^1 (0.5) [\tilde{l}_1(r)_{\alpha_f}^L, \tilde{l}_1(r)_{\alpha_f}^U] d\alpha_f + \int_0^1 (0.5) [\tilde{s}_1(r)_{\alpha_f}^L, \tilde{s}_1(r)_{\alpha_f}^U] d\alpha_f \\ R(\tilde{A}_{\alpha_f}) &= \int_0^1 (0.5) [\tilde{l}_2(r)_{\alpha_f}^L, \tilde{l}_2(r)_{\alpha_f}^U] d\alpha_f + \int_0^1 (0.5) [\tilde{s}_2(r)_{\alpha_f}^L, \tilde{s}_2(r)_{\alpha_f}^U] d\alpha_f \end{aligned}$$

4 Vague branch and bound algorithm for n jobs and 2 workstations

Branch and bound is an optimization algorithm for discrete and synthetic optimization problems. A branch-and-bound algorithm includes a material tab of entrant out by the remedy of state space search: the set of entrant solutions is the notion of as unifying a rooted tree with the full set at the root. The algorithm prospect branches of the tree, which represent subsets of the solution set. Before counting the candidate solutions of a branch, the branch is checked versus upper and lower assessment bounds on the optimal solution and is quit if it cannot produce a better solution than the best one found so far by the algorithm. To include vagueness in the algorithm, the vague processing times are used and their lower bounds are also expressed as vague numbers and a comparison of these vague numbers are done by finding the measure found in equation $R(\tilde{A}_\alpha)$ to be used as the performance criterion. Next, the modified vague branch and bound algorithm is composed.

4.1 Vague branch and bound algorithm

Step 1: Calculate vague lower bound for the 2 machine vague makespan problem;

$$\begin{aligned} \widetilde{LB}(S_r)_{\alpha_t} &= \text{Max} \begin{cases} \widetilde{CT1}_{S_r'}(S_r)_{\alpha_t}(+)_{S_r'}(+) \tilde{P}_{i1_{\alpha_t}}(+) \min \tilde{P}_{i2_{\alpha_t}} \\ \widetilde{CT1}_{S_r'}(S_r)_{\alpha_t}(+)(+) \tilde{P}_{i2_{\alpha_t}} \end{cases} \\ \widetilde{LB}(S_r)_{\alpha_f} &= \text{Max} \begin{cases} \widetilde{CT1}_{S_r'}(S_r)_{\alpha_f}(+)_{S_r'}(+) \tilde{P}_{i1_{\alpha_f}}(+) \min \tilde{P}_{i2_{\alpha_f}} \\ \widetilde{CT1}_{S_r'}(S_r)_{\alpha_f}(+)(+) \tilde{P}_{i2_{\alpha_f}} \end{cases} \end{aligned}$$

where

$\tilde{P}_{i,j}$: The vague processing time of the i^{th} job in the j^{th} machine.

S_r' : The set of $(n - r)$ jobs yet to be assigned to the machines.

\widetilde{CTK} : The vague completion time of the last job in the sequence S_r at the machine K .

(+) : Refers to fuzzy addition.

\widetilde{LB} : The lower bounds.

Step 2: After that the vague lower bound values for nodes are calculated, branching is done from the lowest bound to form new nodes for all unscheduled jobs. This process is continued till all the jobs are scheduled. Calculate the lower bounds $\widetilde{LB} = \{\widetilde{LB}(1), \widetilde{LB}(2), \widetilde{LB}(3)\}$.

Step 3: To find the minimum vague numbers, the generalized mean value (GMV) of these numbers are calculated and the vague number with the smallest GMV is considered as the smallest. If two vague numbers have the same

GMVs, to break the tie, the spread is calculated for each vague number as given in step 2 and the number with the smaller spread is adjudged to be the smallest.

Step 4: After calculating the maximum, now to determine which node to branch from, we have to find the minimum of $\widehat{LB}(1), \widehat{LB}(2), \widehat{LB}(3)$ using the GMVs.

Step 5: Calculate the new completion times of the partial sequence S_r .

Step 6: The vague completion time is computed for the last job in the sequence S_r at machine1, machine 2 and, $\widehat{CTK}(S_r)$.

Step 7: Using the new completion time, fuzzy lower bounds for partial sequences following the steps above is repeated and recalculated.

Step 8: The vague makespan must be calculated as the maximum of the job completion times using a ranking method. The vague waiting, processing and completion times for each of the job sequences are calculated. Negative portions of the vague numbers are deleted since it is not realistic.

5 Numerical example

Consider a system that includes two machines and three jobs. As we have seen, we use the vague numbers in this process, using the octagonal processing times. The data is as shown in the table below.

Job		Machine 1	Machine2
1	A	(12, 14, 16, 18, 20, 22, 24, 26)	(14, 15, 16, 17, 19, 20, 21, 22)
2	B	(10, 12, 14, 16, 18, 20, 22, 24)	(6, 7, 8, 9, 11, 12, 13, 14)
3	C	(7, 9, 11, 13, 15, 17, 19, 21)	(16, 17, 18, 19, 21, 22, 23, 24)

Table 1: Input Data

Given the input data, the vague data is presented in table 2, which is obtained from relations (4 and 6). Throughout this illustration, we have taken the value of $k = 0.75$ and $w = 2$ for octagonal numbers and the first level of vague lower bounds is calculated as shown in table 3.

α_t	M	$\widehat{LB}(1)_{\alpha_t}^L$		$\widehat{LB}(2)_{\alpha_t}^L$		$\widehat{LB}(3)_{\alpha_t}^L$	
0.15	1	$\widehat{LB}(1)_{\alpha_t}^L$ [42.8, 20.4, 35.4, 85.8]	$\widehat{LB}(1)_{\alpha_t}^U$ [51.2, 94.6, 109.6, 110.2]	$\widehat{LB}(2)_{\alpha_t}^L$ [45.5, 23.4, 38.4, 88.8]	$\widehat{LB}(2)_{\alpha_t}^U$ [54.2, 97.6, 112.6, 109.2]	$\widehat{LB}(3)_{\alpha_t}^L$ [48.2, 24.6, 40.6, 92.2]	$\widehat{LB}(3)_{\alpha_t}^U$ [58.5, 101.4, 115.4, 105.8]
	2	[38.5, 20.2, 34.6, 68.5]	[49.8, 54.2, 98.5, 109.6]	[33.4, 20.6, 35.4, 65.3]	[47.4, 50.4, 95.2, 108.8]	[30.8, 20.2, 34.2, 64.5]	[45.2, 49.8, 93.4, 105.4]
0.30	1	[45.6, 32.8, 43.8, 88.6]	[48.4, 86.2, 101.2, 84.4]	[47.4, 33.8, 46.8, 91.6]	[51.4, 89.2, 104.2, 87.4]	[53.5, 36.3, 50.4, 96.5]	[57.6, 91.4, 110.8, 91.5]
	2	[45.8, 33.4, 44.4, 89.8]	[68.2, 88.6, 108.6, 112.2]	[43.4, 30.8, 42.2, 71.6]	[65.6, 89.4, 104.8, 110.6]	[40.8, 30.2, 40.4, 70.2]	[62.2, 93.6, 110.8, 109.8]
0.45	1	[48.4, 41.2, 52.2, 91.4]	[48.4, 86.2, 101.2, 84.4]	[47.4, 31.8, 46.8, 90.6]	[48.3, 80.8, 95.8, 84.2]	[45.2, 30.9, 41.2, 88.4]	[47.8, 30.2, 39.8, 85.6]
	2	[51.6, 42.6, 53.8, 91.6]	[78.8, 96.4, 111.2, 116.8]	[48.8, 41.4, 52.6, 91.4]	[77.2, 94.8, 109.8, 115.5]	[45.2, 39.8, 50.4, 88.6]	[74.8, 92.2, 105.4, 113.8]
α_f	M	$\widehat{LB}(1)_{\alpha_f}^L$	$\widehat{LB}(1)_{\alpha_f}^U$	$\widehat{LB}(2)_{\alpha_f}^L$	$\widehat{LB}(2)_{\alpha_f}^U$	$\widehat{LB}(3)_{\alpha_f}^L$	$\widehat{LB}(3)_{\alpha_f}^U$
	0.55	1 [46.8, 52.8, 75.8, 83.8] [34.6, 42.8, 68.2, 74.4]	1 [52.8, 65.2, 78.2, 83.8] [45.6, 54.4, 62.4, 65.6]	1 [48.8, 56.4, 79.6, 88.2] [36.4, 45.5, 70.2, 78.6]	1 [53.6, 68.6, 80.2, 84.6] [48.4, 56.5, 65.8, 67.4]	1 [49.2, 58.4, 82.2, 95.6] [39.2, 49.7, 71.8, 83.4]	1 [55.1, 70.3, 83.5, 86.2] [53.1, 60.5, 69.4, 71.8]
0.75	1	[48, 56, 69, 85]	[52, 62, 75, 83]	[49, 58, 72, 87]	[55, 65, 79, 85]	[52, 66, 79, 91]	[59, 68, 84, 90]
	2	[35, 40, 50, 60]	[45, 52, 60, 65]	[40, 45, 55, 65]	[49, 56, 62, 67]	[45, 50, 60, 70]	[53, 58, 65, 72]
0.95	1	[48.8, 59.2, 72.2, 85.8]	[50.8, 58.8, 71.8, 81.8]	[49.2, 60.4, 73.8, 87.2]	[52.6, 60.2, 73.4, 83.5]	[50.4, 61.6, 75.2, 88.4]	[55.9, 64.7, 76.2, 85.8]
	2	[43.6, 50.4, 58.4, 67.6]	[44.1, 49.6, 57.6, 64.1]	[45.2, 51.6, 59.2, 69.4]	[46.8, 51.4, 59.2, 66.4]	[45.8, 52.4, 60.2, 71.5]	[50.2, 55.2, 65.2, 69.4]

Table 2: Vague data

α_t	$\widehat{LB}(1)_{\alpha_t}^L$		$\widehat{LB}(2)_{\alpha_t}^L$		$\widehat{LB}(3)_{\alpha_t}^L$	
0.15	$\widehat{LB}(1)_{\alpha_t}^L$ [42.8, 20.4, 35.4, 85.8]	$\widehat{LB}(1)_{\alpha_t}^U$ [51.2, 94.6, 109.6, 110.2]	$\widehat{LB}(2)_{\alpha_t}^L$ [45.5, 23.4, 38.4, 88.8]	$\widehat{LB}(2)_{\alpha_t}^U$ [54.2, 97.6, 112.6, 109.2]	$\widehat{LB}(3)_{\alpha_t}^L$ [48.2, 24.6, 40.6, 92.2]	$\widehat{LB}(3)_{\alpha_t}^U$ [58.5, 101.4, 115.4, 105.8]
	[38.5, 20.2, 34.6, 68.5]	[49.8, 54.2, 98.5, 109.6]	[33.4, 20.6, 35.4, 65.3]	[47.4, 50.4, 95.2, 108.8]	[30.8, 20.2, 34.2, 64.5]	[45.2, 49.8, 93.4, 105.4]
0.30	[45.8, 33.4, 44.4, 89.8]	[68.2, 88.6, 108.6, 112.2]	[43.4, 30.8, 42.2, 71.6]	[65.6, 89.4, 104.8, 110.6]	[40.8, 30.2, 40.4, 70.2]	[62.2, 93.6, 110.8, 109.8]
	[51.6, 42.6, 53.8, 91.6]	[78.8, 96.4, 111.2, 116.8]	[48.8, 41.4, 52.6, 91.4]	[77.2, 94.8, 109.8, 115.5]	[45.2, 39.8, 50.4, 88.6]	[74.8, 92.2, 105.4, 113.8]
α_f	$\widehat{LB}(1)_{\alpha_f}^L$	$\widehat{LB}(1)_{\alpha_f}^U$	$\widehat{LB}(2)_{\alpha_f}^L$	$\widehat{LB}(2)_{\alpha_f}^U$	$\widehat{LB}(3)_{\alpha_f}^L$	$\widehat{LB}(3)_{\alpha_f}^U$
	0.55 [46.8, 52.8, 75.8, 83.8] [34.6, 42.8, 68.2, 74.4]	1 [52.8, 65.2, 78.2, 83.8] [45.6, 54.4, 62.4, 65.6]	1 [48.8, 56.4, 79.6, 88.2] [36.4, 45.5, 70.2, 78.6]	1 [53.6, 68.6, 80.2, 84.6] [48.4, 56.5, 65.8, 67.4]	1 [49.2, 58.4, 82.2, 95.6] [39.2, 49.7, 71.8, 83.4]	1 [55.1, 70.3, 83.5, 86.2] [53.1, 60.5, 69.4, 71.8]
0.75	[48, 56, 69, 85]	[52, 62, 75, 83]	[49, 58, 72, 87]	[55, 65, 79, 85]	[52, 66, 79, 91]	[59, 68, 84, 90]
	[35, 40, 50, 60]	[45, 52, 60, 65]	[40, 45, 55, 65]	[49, 56, 62, 67]	[45, 50, 60, 70]	[53, 58, 65, 72]
0.95	[48.8, 59.2, 72.2, 85.8]	[50.8, 58.8, 71.8, 81.8]	[49.2, 60.4, 73.8, 87.2]	[52.6, 60.2, 73.4, 83.5]	[50.4, 61.6, 75.2, 88.4]	[55.9, 64.7, 76.2, 85.8]
	[43.6, 50.4, 58.4, 67.6]	[44.1, 49.6, 57.6, 64.1]	[45.2, 51.6, 59.2, 69.4]	[46.8, 51.4, 59.2, 66.4]	[45.8, 52.4, 60.2, 71.5]	[50.2, 55.2, 65.2, 69.4]

Table 3: Lower Bounds

The minimum values and the updated completion times are computed according to the steps of the modified branch and bound algorithm as presented in tables 4 and 5.

α_t	$\min \left\{ \widetilde{LB}(1)_{\alpha_t}^L, \widetilde{LB}(2)_{\alpha_t}^L, \widetilde{LB}(3)_{\alpha_t}^L \right\}$		
0.15	$\widetilde{LB}(1)$	-	-
0.30	-	$\widetilde{LB}(2)$	-
0.45	-	-	$\widetilde{LB}(3)$
α_f	$\min \left\{ \widetilde{LB}(1)_{\alpha_f}^L, \widetilde{LB}(2)_{\alpha_f}^L, \widetilde{LB}(3)_{\alpha_f}^L \right\}$		
0.15	$\widetilde{LB}(1)$	-	-
0.30	$\widetilde{LB}(1)$	-	-
0.45	$\widetilde{LB}(1)$	-	-

Table 4: Minimum Values

S_r	α_t	$\widetilde{LB}(S_r)_{\alpha_t}$	α_f	$\widetilde{LB}(S_r)_{\alpha_f}$
12	0.15	[40, 34.4, 34, 70, 41.1, 35.5, 36, 73.2]	0.55	[40.1, 45.6, 56.8, 70.1, 44.3, 49.8, 66, 79.4]
			0.75	[41, 48, 58, 71, 46, 50, 67, 74]
			0.95	[41.6, 40.4, 60.4, 71.6, 45.1, 45.6, 62.8, 75]
13	0.15	[33, 24.2, 26.2, 69.4, 35, 34.2, 37.1, 72]	0.55	[33.1, 40.6, 54.2, 69.1, 36.2, 46.3, 60.1, 73.6]
			0.75	[34, 43, 55, 70, 39, 48, 60, 76]
			0.95	[34.6, 45.4, 55.2, 71.1, 40.1, 49.5, 56.4, 75]
21	0.30	[23.4, 28.4, 37, 47.4, 25, 33, 41, 52.5]	0	0
23	0.30	[28.4, 35.6, 43.6, 48.4, 32.5, 39.6, 48.7, 51.4]	0	0
31	0.45	[21.4, 28.6, 36.6, 45.4, 27.6, 33, 40.8, 49.7]	0	0
32	0.45	[20.4, 25.4, 34, 44.4, 23, 30.2, 44, 51.2]	0	0
123	0.15	[46.4, 27.2, 40.2, 83.4, 52.3, 47, 49.2, 87.5]	0.55	[63.6, 100.8, 113.8, 94.6, 64.6, 99.8, 108.4, 98]
			0.75	[62, 72, 85, 93, 69, 79, 89, 99]
			0.95	[62.8, 75.2, 88.2, 93.8, 70, 80.4, 93.2, 99.7]
132	0.15	[46.8, 52.8, 67, 83.8, 50, 59.3, 71, 90.1]	0.55	[62.8, 75.2, 88.2, 93.8, 69.5, 79, 92.4, 98]
			0.75	[52, 62, 75, 83, 58, 68, 80, 88]
			0.95	[60.8, 68.8, 81.8, 91.8, 65.9, 71.2, 85, 96.3]
213	0.30	[53.6, 90.8, 103.8, 84.6, 66.1, 92.8, 105.7, 97]	0	0
231	0.30	[52.8, 65.2, 78.2, 83.8, 55, 67, 84.1, 92.4]	0	0
312	0.45	[50.8, 58.8, 71.8, 81.8, 52.8, 68.5, 76.8, 89.6]	0	0
321	0.45	[58.8, 69.2, 82.2, 95.8, 68.2, 74.1, 88, 101.7]	0	0

Table 5: New Completion Times

Thus, the optimal sequence is $S_r = 1 - 3 - 2$, the final solutions are shown in table 6.

Job	M_1	M_2
1	(50.6, 54.8, 58.5, 62.1, 80.6, 94.4, 108.6, 122.8)	(58.5, 63.6, 68.4, 73.3, 95.2, 128.1, 179.2, 212.7)
3	(25.5, 27.8, 29.7, 31.3, 35.6, 37.4, 39.5, 41.8)	(36.5, 39.5, 42.6, 45.1, 51.8, 54.8, 57.6, 60.7)
2	(52.4, 57.3, 62.2, 67.4, 99.6, 122.7, 145.8, 168.9)	(61.1, 65.2, 71.8, 77.5, 139.6, 201.4, 281.5, 343.1)

Table 6: The Optimal Solutions

Regarding the data analysis, we find that in the vague case, considering the weight factor, the selection conditions become harder and many states occur. Also, if the weighting factor is not considered, it only creates one choice, that is, in the branching a different selection in alpha cut is created. But, when the weight factor is not included, branching in alpha cut is differently created. Also, the CPU time for the calculations is about 495 seconds for the parameters setting of the proposed example. Managers take the most optimal decisions based on the corresponding conditions.

6 Conclusions

The branch and bound algorithm was discussed for flow-shop scheduling problem using vague octagon numbers. The processing time of the job was considered uncertain, which allows the decision maker to take into account unlimited information and make the best decisions. The proposed problem of a flow-shop scheduling was first modeled in vague environment. Then, as for a solution algorithm, branch and bound was modified so that vague computations were included into the algorithm. In past researches, parameters of flow-shop scheduling were mostly considered as fuzzy

numbers and to the best of our knowledge vague flow-shop scheduling using vague branch and bound is novel. The results of the implementation using the modified vague branch and band helped the decision makers to improve the decisions in uncertain environments by considering different setting of vague parameters choices. For further studies, one can handle larger sizes problems to analyze the efficiency of the algorithm. Also, including vague failure rate for machines in the model can be studied. Another direction is to develop the proposed vague branch and bound algorithm for 0/1 integer programming and multi-objective one.

References

- [1] G. Ambika, G. Uthra, *Branch and Bound Technique in flow shop scheduling using fuzzy processing times*, Annals of Pure and Applied Mathematics, **8**(2) (2014), 37–42.
- [2] P. C. Bagga, *N-job, 2-machine sequencing problem with stochastic service times*, Operation Research, **18** (7) (1970), 184–199.
- [3] K. R. Baker, *Introduction to sequencing and scheduling*, John Wiley & Sons, 1974.
- [4] J. Cheng, H. Kise, H. Matsumoto, *A branch-and-bound algorithm with fuzzy inference for a permutation flowshop scheduling problem*, European Journal of Operational Research, **96**(3) (1997), 578–590.
- [5] P. Cowling, M. Johansson, *Using real time information for effective dynamic scheduling*, European journal of operational research, **139**(2) (2002), 230–244.
- [6] V. Dhanalakshmi, F. C. Kennedy, *Some Aggregation Operations on Octagonal Fuzzy Numbers and its Application to Decision Making*, **5**(1) (2015), 145–160.
- [7] W. L. Gau, D. J. Buehrer, *Vague sets*, IEEE transactions on systems, man, and cybernetics, **23**(2) (1993), 610–614.
- [8] V. A. González-López, R. Gholizadeh, A. M. Shirazi, *Optimization of queuing theory based on vague environment*, International Journal of Fuzzy System Applications (IJFSA), **5**(1) (2016), 1–26.
- [9] D. Gupta, S. Sharma, S. Aggarwal, *Flow shop scheduling on 2-machines with setup time and single transport facility under fuzzy environment*, Opsearch, **50**(1) (2013), 14–24.
- [10] T. P. Hong, T. N. Chuang, *A new triangular fuzzy Johnson algorithm1*, Computers & Industrial Engineering, **36**(1) (1999), 179–200.
- [11] H. Ishibuchi, T. Murata, K. H. Lee, *Formulation of fuzzy flowshop scheduling problems with fuzzy processing time*, Proceedings of the Fifth IEEE International Conference on Fuzzy systems, **1**(2) (1996), 199–205.
- [12] S. M. Johnson, *Optimal two-and three-stage production schedules with setup times included*, Naval research logistics quarterly, **1**(1) (1954), 61–68.
- [13] C. Kao, C. C. Li, S. P. Chen, *Parametric programming to the analysis of fuzzy queues*, Fuzzy sets and systems, **107**(1) (2006), 93–100.
- [14] J. C. Ke, C. H. Lin, *Fuzzy analysis of queueing systems with an unreliable server: A nonlinear programming approach*, Applied Mathematics and Computation, **175**(1) (2006), 330–346.
- [15] E. Khorram, V. Nozari, *Multi-objective optimization with preemptive priority subject to fuzzy relation equation constraints*, Iranian Journal of Fuzzy Systems, **9**(3) (2012), 27–45.
- [16] B. L. Maccarthy, J. Liu, *Addressing the gap in scheduling research: a review of optimization and heuristic methods in production scheduling*, The International Journal of Production Research, **31**(1) (1993), 59–79.
- [17] S. U. Malini, F. C. Kennedy, *An approach for solving fuzzy transportation problem using octagonal fuzzy numbers*, Applied Mathematical Sciences, **7**(54) (2013), 2661–2673.
- [18] L. Martin, T. Roberto, *Fuzzy scheduling with application to real time system*, Fuzzy sets and Systems, **121**(3) (2001), 523–535.

- [19] C. S. McCahon, E. S. Lee, *Job sequencing with fuzzy processing times*, Computers & Mathematics with Applications, **19**(7) (1990), 31–41.
- [20] C. S. McCahon, E. S. Lee *Fuzzy job sequencing for a flow shop*, European Journal of Operational Research, **62**(3) (1992), 294–301.
- [21] A. Mehrabian, R. T. Moghaddam, K. K. Damaghani, *Multi-objective routing and scheduling in flexible manufacturing systems under uncertainty*, Iranian Journal of Fuzzy Systems, **14**(2) (2017), 45–77.
- [22] R. Nagarajan, A. Solairaju, *Computing improved fuzzy optimal Hungarian assignment problems with fuzzy costs under robust ranking techniques*, International journal of computer application, **6**(4) (2010), 263–276.
- [23] P. Sanuja, S. Xueyan, *A new approach to two machine flow shop problem with uncertain processing time*, Optimization and Engineering, **7**(3) (2006), 329–343.
- [24] C. S. Shukla, F. F. Chen, *The state of the art in intelligent real-time FMS control: a comprehensive survey*, Journal of intelligent Manufacturing, **7**(6) (1996), 441–455.
- [25] T. P. Singh, P. Allawalia, *Reformation of non fuzzy scheduling using the concept of fuzzy processing time under blocking*, International Conference on intelligence system & Networks, (2008), 322–324.
- [26] T. P. Singh, P. Sunita, *Allawalia Fuzzy flow shop problem on 2-machines with single transport facility-An heuristic approach*, Arya Bhatta Journal of Mathematics & Informatics, **1**(1-2) (2009), 38–46.
- [27] I. Temiz, S. Erol, *Fuzzy branch-and-bound algorithm for flow shop scheduling*, Journal of intelligent manufacturing, **15**(4) (2004), 449–454.
- [28] R. R. Yager, *A procedure for ordering fuzzy subsets of the unit interval*, Information sciences, **24**(2) (1981), 143–161.