Development of fuzzy decision making system to quantify the Project Management Efficiency using Interval Type-2 Fuzzy Logic

Shweta Shukla
Department of Commerce, Govt. College Bhaisma, Chhattisgarh-India

Nivritti James
Department of Commerce, Govt. College Pendra, Chhattisgarh-India

Abstract
The objective of this paper is to present an approach that utilizes an interval type-2 fuzzy decision making system (IT2FDMS) to quantify the Project Management Efficiency (PME). The algorithm developed in this paper is based upon interval type-2 fuzzy logic, giving it the ability to solve complex problems plagued with uncertainty and vagueness. A interval type-2 fuzzy decision making system is designed and implemented using the MATLAB Fuzzy Logic tool box for the evaluation of the PME. This algorithm once refined to each area under the industry of software development can be used for subsequent projects, saving large percentages of time, money, and effort, without sacrificing quality.

Key Words: Project management efficiency; Fuzzy decision making system; Fuzzy sets; Project time delay; Project time delay gradient.

I. Introduction
Effective management of projects is crucial for the development and survival of any economy because development is about growth and growth is about a series of successfully managed projects. A project may be viewed as “the entire process required to produce a new product, new plant, new system, or other specific results”. In this fiercely competitive world, project organizations are forced to look for scientific tool that assist them in the evaluation of their projects. The project management team is responsible for producing the project output and hence the project management team must be constantly aware of the project goal, project purpose and project management efficiency. Project effectiveness, which is a synonym of project success, is measured or assessed in terms of the degree of achievement of project objectives.

For the successful completion and maintenance of software projects, there is a need for effective project managers. The long proclaimed ineffectiveness of software projects to maintain their schedule, cost, and quality, continues to plague most development projects [6, 2, 5]. It has been observed that over half of all software development projects are considered a failure with respect to their cost and schedule [6, 3]. The software crisis must be addressed and, to the extent possible, resolved. To do so require more accurate schedule and cost estimates, better products, and higher productivity. All these can be achieved through more effective software management. These factors include, but are not limited to, time constraints, tangible costs, and projects. Extensive research has been done to develop sophisticated tools that can analyze and provide accurate information for the choice of investments and projects. Many of these researchers though, have faced the dilemma that much of their data is plagued by uncertainty, vagueness and approximation. This paper provides a good example and guide to processing vaguely defined variables, and variables whose relationships cannot be defined by mathematical relationships. There is continuing interest by academics and practitioners alike in measuring and coping with project schedule uncertainty. Interval Type-2 Fuzzy logic has been proposed as an alternate approach to probability theory for quantifying uncertain related to activity duration. However, the interval type-2 fuzzy logic approach is not widely understood, and generally accepted computational
approaches are not available. The objective of this paper is to present an approach that utilizes an interval type-2 fuzzy decision making system (IT2FDMS) to quantify the Project Management Efficiency (PME). The evaluation of PME can serve for project managers and for project organizations as an indicator for the level of achievement of the project management objectives. PME may help in the evaluation of the performance of project teams.

Interval Type-2 fuzzy sets, characterized by membership grades that are themselves fuzzy, were introduced by Zadeh in 1975 to better handle uncertainties. As illustrated in Fig.1, the membership function (MF) of a type-2 set has a footprint of uncertainty (FOU), which represents the uncertainties in the shape and position of the type-1 fuzzy set. The FOU is bounded by an upper MF and a lower MF, both of which are type-1 MFs. Fuzzy logic systems constructed using rule bases that utilize at least one interval type-2 fuzzy sets are called interval type-2 FLSs. Since the FOU of a type-2 fuzzy set provides an extra mathematical dimension, type-2 FLSs can better handle system uncertainties and have the potential to outperform their type-1 counterparts.

![Fig.1. Interval type-2 fuzzy sets (a) Triangular MF, (b) Trapezoidal MF.](image)

(a) A type-2 fuzzy set obtained by blurring the width of a triangular type-1 fuzzy set and (b) A type-2 fuzzy set obtained by blurring the apex of a triangular type-1 fuzzy set, thus trapezoidal fuzzy set is obtained.

II. Interval Type-2 Fuzzy Logic Systems (IT2FLS)

Fuzzy Logic Systems (FLS) are known as the universal-approximators and have various applications in identification and control designs. A type-1 fuzzy system consists of four major parts: fuzzifier, rule base, inference engine and defuzzifier. A type-2 fuzzy system has a similar structure, but one of the major differences can be seen in the rule base part, where a type-2 rule base has antecedents and consequents using Type-2 Fuzzy Sets (T2FS). In a T2FS, we consider a Gaussian function with a known standard deviation, while the mean (m) varies between m1 and m2. Therefore, a uniform weighting is assumed to represent a footprint of uncertainty as shaded in Fig.2. Because of using such a uniform weighting, we name the T2FS as an Interval Type-2 Fuzzy Set (IT2FS). Utilizing a rule base which consists of IT2FSs, the output of the inference engine will also be a T2FS and hence we need a type-reducer to convert it to a type-1 fuzzy set before defuzzification can be carried out. Fig.3 shows the main structure of type-2 FLS.

By using singleton fuzzification, the singleton inputs are fed into the inference engine. Combining the fuzzy if-then rules, the inference engine maps the singleton input \( x = [x_1, x_2, \ldots, x_n] \) into a type-2 fuzzy set as the output. A typical form of an if-then rule can be written as:
\[ R_i = \text{if } x_1 \text{ is } \tilde{F}^i_1 \text{ and } x_2 \text{ is } \tilde{F}^i_2 \text{ and } \ldots \text{ and } x_k \text{ is } \tilde{F}^i_k \text{ then } \tilde{G}^i \]  

(1)

Fig. 2: Interval type 2 fuzzy set (IT2FS)

Where \( F_k \) are the antecedents \((k = 1, 2, \ldots, n)\) and \( G^i \) is the consequent of the \( i^{th} \) rule. We use sup-star method as one of the various inference methods. The first step is to evaluate the firing set for \( i^{th} \) rule as following:

\[
F^i(x) = \prod_{i=1}^{n} \mu_{F^i} (x_i) 
\]

(2)

As all of the \( F^i \) are IT2FSs, so \( F^i(x) \) can be written as \( F^i(x) = \left[ \tilde{F}^i(x), \overline{F}^i(x) \right] \)

Where:

\[
\tilde{F}^i(x) = \prod_{i=1}^{n} \mu_{F^i} (x_i) 
\]

(3)

\[
\overline{F}^i(x) = \prod_{i=1}^{n} \overline{\mu}_{F^i} (x_i) 
\]

(4)

The terms \( \mu_{F^i} \) and \( \overline{\mu}_{F^i} \) are the lower and upper membership functions, respectively (Fig.2). In the next step, the firing set \( F_i(x) \) is combined with the \( i^{th} \) consequent using the product t-norm to produce the type-2 output fuzzy set. The type-2 output fuzzy sets are then fed into the type reduction part. The structure of type reducing part is combined with the defuzzification procedure, which uses Center of Sets (COS) method. First, the left and right centroids of each rule consequent are computed using Karnik-Mendel (KM) algorithm. Let’s call them \( y_l \) and \( y_r \), respectively.

The firing sets \( F^i(x) = \left[ \tilde{F}^i(x), \overline{F}^i(x) \right] \) computed in the inference engine are combined with the left and right centroid of consequents and then the defuzzified output is evaluated by finding the solutions of following optimization problems:

\[
y_l(x) = \min_{y f \in \left[ \tilde{F}^i(x), \overline{F}^i(x) \right]} \left( \sum_{i=1}^{n} y_i f^i(x) / \sum_{i=1}^{n} f^i(x) \right) 
\]

(5)

\[
y_r(x) = \max_{y f \in \left[ \tilde{F}^i(x), \overline{F}^i(x) \right]} \left( \sum_{i=1}^{n} y_i f^i(x) / \sum_{i=1}^{n} f^i(x) \right) 
\]

(6)

Define \( f^i_l(x) \) and \( f^i_r(x) \) as a functions which are used to solve (5) and (6) respectively and let

\[
\xi^i_l(x) = f^i_l(x) / \sum_{i=1}^{n} f^i_l(x) 
\]

\[
\xi^i_r(x) = f^i_r(x) / \sum_{i=1}^{n} f^i_r(x) 
\]

And
Then we can write (5) and (6) as:

\[
y_1(\Delta) = \frac{\sum_{i=1}^{M} y_i f_i(\Delta)}{\sum_{i=1}^{M} f_i(\Delta)} = \theta_1^T \xi_1(\Delta) = y_2(\Delta) = \frac{\sum_{i=1}^{M} y_i r_i(\Delta)}{\sum_{i=1}^{M} r_i(\Delta)} = \theta_r^T \xi_r(\Delta)
\]  

(7)  

(8)  

Where

\[
\xi_i(\Delta) = [\xi_{i1}(\Delta), \xi_{i2}(\Delta), \ldots, \xi_{iM}(\Delta)]
\]

And \( \xi_r(\Delta) = [\xi_{r1}(\Delta), \xi_{r2}(\Delta), \ldots, \xi_{rM}(\Delta)] \) are the fuzzy basis functions and

\[
\theta_i(\Delta) = [y_{i1}(\Delta), y_{i2}(\Delta), \ldots, y_{iM}(\Delta)]
\]

And \( \theta_r(\Delta) = [y_{r1}(\Delta), y_{r2}(\Delta), \ldots, y_{rM}(\Delta)] \) are the adjustable parameters.

Finally, the crisp value is obtained by the defuzzification procedure as follows:

\[
y(\Delta) = \frac{1}{2} [y_1(\Delta) + y_r(\Delta)] = \frac{1}{2} [\theta_1^T \xi_1(\Delta) + \theta_r^T \xi_r(\Delta)] = \frac{1}{2} \theta^T \xi(\Delta)
\]  

(9)  

Where

\[
\theta = [\theta_1^T \theta_r^T]^T \text{ and } \xi = [\xi_1^T \xi_r^T]^T
\]

2.1 Fuzzifier

Fuzzifier maps the measured inputs into fuzzy linguistic values with the help of fuzzy reasoning mechanism. In the present study, singleton fuzzifier was used which its output is a single point of a unity membership grade [2].

2.2 Rule base

In this part which is a set of IF-THEN rules, the knowledge of experts will be placed. Jth rule in IT2FLS can be written as:

\[
R_j: \text{ If } x_1 \text{ is } \tilde{E}_1^j \text{ and } x_2 \text{ is } \tilde{E}_2^j \text{ and } \ldots x_n \text{ is } \tilde{E}_n^j \text{ then } y \text{ is } \tilde{O}^j
\]

(10)

Where \( x_i \) \((i=1,2,\ldots,n)\) and \( y \) are IT2FLS input and output, respectively and also show the type-1 or type-2 antecedent and consequent sets, respectively.

2.3 Inference engine
In IT2FLS, the inference engine combines rules and represents a mapping from input to output IT2FLS. Using input and antecedent operations, the firing set is obtained as:

$$F^j(X) = \bigcap_{j=1}^{n} \mu_{\mu_j}(x_j)$$  \hspace{1cm} (11)$$

Where, t-norm is assumed to be product. Since the present study discusses IT2FLS, the firing input sets are defined based on the upper and lower membership functions as:

$$F^j(X) = \left(\bar{f}^j(X), \overline{f}^j(X)\right)$$  \hspace{1cm} (12)$$

Where * shows the t-norm and $\bar{f}^j(X)$ and $\overline{f}^j(X)$ are the jth upper and lower membership functions, respectively and can be determined by:

$$\bar{f}^j(X) = \mu_{\bar{\mu}_j} \mu_{\bar{\mu}_j} \ldots \mu_{\bar{\mu}_j}$$  \hspace{1cm} (13)$$

$$\overline{f}^j(X) = \mu_{\overline{\mu}_j} \mu_{\overline{\mu}_j} \ldots \mu_{\overline{\mu}_j}$$  \hspace{1cm} (14)$$

2.4 Type reducer and Defuzzifier

Since the output of the inference engine is an IT2FS, a type reducer is needed before defuzzification to convert IT2FS into type-1 fuzzy set. Type reducer was first proposed by Karnik & Mendel. Five different methods of type reduction have been suggested. Among these methods, center of sets (COS) has been extensively used due to easy calculation with the help of Karnik & Mendel's iterative algorithm [3]. The COS type reducer is an interval set which is determined by left-end point ($y_l$) and right-end point ($y_r$) and can be written as:

$$y_{\text{COS}}[y_l, y_r] = \int_{\bar{y}_l}^{y_r} \int_{\bar{y}_r}^{y_r} \mu_{\sum_{j=1}^{n} f_j}(\sum_{j=1}^{n} y_j)$$  \hspace{1cm} (15)$$

Where $f_j \in F_j = \left(\bar{f}_j(X), \overline{f}_j(X)\right)$ and $f_j$ is the centroid of jth consequent set. In general, there is no closed-form formula for calculating $y_l$ and $y_r$. However, Karnik and Mendel [10] have proposed two algorithms for calculating end-points which are known as KM iterative algorithms. In case of using singleton fuzzifier, product inference engine and COS type reducer, $y_l$ and $y_r$ can be written as:

$$y_l = \left(\sum_{j=1}^{n} \bar{f}_j \cdot \Theta_r \right) / \left(\sum_{j=1}^{n} \Theta_r \right)$$  \hspace{1cm} (16)$$

$$y_r = \left(\sum_{j=1}^{n} \overline{f}_j \cdot \Theta_r \right) / \left(\sum_{j=1}^{n} \Theta_r \right)$$  \hspace{1cm} (17)$$

Where $\Theta_r$ and $\Theta_r$ are related to left-end point and right-end point of jth consequent set, respectively. Finally, the obtained set from type reducer can be defuzzified by using the average of $y_r$ and $y_l$, as below:

$$y = \left(y_l + y_r\right) / 2$$  \hspace{1cm} (18)$$

III. Focus of Study

As mentioned above, for the project management efficiency (PME), the major measure is project time delay (PTD) and an additional set of variable namely Project time delay gradient (PTDG) is also considered. Hence the combined impact of PTD and PTDG might be evaluated, for example, according to interval type-2 fuzzy decision rule like the following:

If PTD is Low [$\mu_L(L), \mu_U(L)$] and PTDG is high [$\mu_L(H), \mu_U(H)$] then PME is very high [$\mu_L(VH), \mu_U(VH)$].
Here $[\mu_L, \mu_U]$ is an interval of lower membership function $[\mu_L]$ to upper membership function $[\mu_U]$ of Footprint Of Uncertainty (FOU). However, the boundaries of Very High, High, Medium, and Low of any decision variable are determined by experts of the project organization. A interval type-2 fuzzy decision making system is a scientific tool that can be used to solve the problem. This means that information of expert knowledge and experience in a IT2FDMS system is used for determining PME. The development of such an interval type-2 fuzzy decision making system is easily implemented using the MATLAB software. MATLAB fuzzy tool box is a menu driven software that allows the implementation of interval type-2 fuzzy constructs like membership functions and a database of decision rules. The software is easy to use and it is user friendly. The Project Management Efficiency (PME) can be determined by entering the values of PTD and PTDG.

IV. Fuzzy Decision Making System

Interval type-2 Fuzzy inference systems (IT2FIS) are rule-based systems. It is based on interval type-2 fuzzy set theory and interval type-2 fuzzy logic. IT2FIS are mappings from an input space to an output space. IT2FIS allows constructing structures which are used to generate responses (outputs) for certain stimulations (inputs). Response of IT2FIS is based on interval type-2 fuzzy constructs like membership functions and a database of decision rules. Knowledge is stored in the form of a knowledge base. Rule base is a set of rules. Rule base expresses relations between inputs of system and its expected outputs. Knowledge is obtained by eliciting information from specialists. These systems are usually known as fuzzy expert systems. Another common denomination for IT2FIS is fuzzy knowledge-based systems. It is also called as data-driven fuzzy systems. A fuzzy decision making system (IT2FDMS) is comprised of four main components: a fuzzification interface, a knowledge base, decision making logic, and a defuzzification interface [10]. In essence, a IT2FDMS is a fuzzy expert system (FES). Fuzzy expert systems are oriented towards numerical processing where conventional expert systems are mainly symbolic reasoning engines [1,8,9]. Fig. 5 provides a framework for the interrelationships between the components that constitute an interval type-2 fuzzy decision making system (IT2FDMS).The four components are explained as in the following:

4.1 Fuzzy Inference System

The human knowledge is expressed in fuzzy rules with the following form:

IF $<\text{fuzzy proposition}>$ THEN $<\text{fuzzy proposition}>$

![Figure 5: A Framework for interval type-2 fuzzy decision making (IT2FDMS)](image-url)
The fuzzy propositions are divided in two types, the first one is called **atomic**: for example \( x \text{ is } A \), where \( x \) is a linguistic variable and \( A \) is a linguistic value; the second one is called **compounded**: for example \( x \text{ is } A \text{ AND } y \text{ is } B \text{ OR } z \text{ is } \neg C \), this is a compounded atomic fuzzy proposition with the “AND”, “OR” and “NOT” connectors, representing fuzzy intersection, union and complement respectively. The compounded fuzzy propositions are fuzzy relationships. The membership function of the rule IF-THEN is a fuzzy relation determined by a fuzzy implication operator. The fuzzy rules combine one or more fuzzy sets of entry, called antecedent, and are associated with one output fuzzy set, called consequents.

The Fuzzy Sets of the antecedent are associated by fuzzy operators AND, OR, NOT and linguistic modifiers. The fuzzy rules permit expressing the available knowledge about the relationship between antecedent and consequents. To express this knowledge completely we normally have several rules, grouped to form what it is known a rule base, that is, a set of rules that express the known relationships between antecedent and consequents.

**The fuzzy rules are basically IF <Antecedent> THEN <Consequent>**

In fuzzy logic the reasoning is imprecise or approximated, which means that we can infer from one rule a conclusion even if the antecedent doesn’t comply completely. We can count on Generalized Modus Ponens (GMP) and Generalized Modus Tollens (GMT), that represent the generalizations of classic reasoning. The GMP inference method is known as direct reasoning and is summarized as:

**Rule** IF \( x \text{ is } A \) THEN \( y \text{ is } B \)

**Fact** \( x \text{ is } A' \)

**Conclusion** \( y \text{ es } B' \)

Where \( A, A', B \) and \( B' \) are fuzzy sets of any kind. This relationship is expressed as \( B' = A' \circ (A \rightarrow B) \).

4.2 Interval Type-2 Fuzzy Inference System

An Inference Fuzzy System is a rule base system that uses fuzzy logic, instead of Boolean logic utilized in data analysis. Its basic structure includes 4 components (Fig.6):

- **Fuzzifier**. Translates inputs (real values) to fuzzy values.
- **Inference System**. Applies a fuzzy reasoning mechanism to obtain a fuzzy output.
- **Type Defuzzifier/Reducer**. The defuzzifier translates one output to precise values; the type reducer transforms a Type-2 Fuzzy Set into a Type-1 Fuzzy Set.
- **Knowledge Base**. Contains a set of fuzzy rules, and a membership functions set known as the database.

![Fig. 6. Interval Type-2 Fuzzy Reasoning.](image_url)
The decision process is a task that identifies parameters by the inference system using the rules of the rule base data. These fuzzy rules define the connection between the input and output fuzzy variables. A fuzzy rule has the form: IF <Antecedent> THEN <Consequent>, where antecedent is a compound fuzzy logic expression of one or more simple fuzzy expressions connected with fuzzy operators; and the consequent is an expression that assigns fuzzy values to output variables.

V. Interval Type-2 Fuzzy System Design

The Mamdani and Takagi-Sugeno-Kang (TSK) Interval Type-2 Fuzzy Inference Models [10] and the design of Interval Type-2 membership functions and operators are implemented in the IT2FLS Toolbox (Interval Type-2 Fuzzy Logic Systems) reused from the Matlab® commercial Fuzzy Logic Toolbox. The IT2FLS Toolbox includes a series of folders called dit2mf, it2fis, it2mf and it2op (Fig. 8).

Fig.7. Structure of the Type-2 Fuzzy Inference System.

This folders contain the functions to create Mamdani and TSK Interval Type-2 Fuzzy Inference Systems (newfistype-2.m), functions to add input-output variables and their ranges (addvartype-2.m), it has functions to add 22 types of Interval Type-2 Membership functions for input-outputs (addmftype-2.m), functions to add the rule matrix (addruletype-2.m), it can evaluate the Interval Type-2 Fuzzy Inference Systems (evalifistype-2.m), evaluate Interval Type-2 Membership functions (evalimftype-2.m), it can generate the initial parameters of the Interval Type-2 Membership functions (igenparamtype-2.m), it can plot the Interval Type-2 Membership functions with the input-output variables (plotimftype-2.m), it can generate the solution surface of the Fuzzy Inference System (gensurftype-2.m), it plots the Interval type-2 membership functions (plot2dtype-2.m, plot2dctype-2.m), a folder to evaluate the derivatives of the Interval type-2 Membership Functions (dit2mf) and a folder with different and generalized Type-2 Fuzzy operators (it2op, t2op).

The Interval Type-2 Fuzzy Inference Systems (IT2FIS) structure is the MATLAB object that contains all the interval type-2 fuzzy inference system information. This structure is stored inside each GUI tool. Access functions such as getifistype-2 and setifistype-2 make it easy to examine this structure. All the information for a given fuzzy inference system is contained in the IT2FIS structure, including variable names, membership function definitions, and so on. This structure can itself be thought of as a hierarchy of structures, as shown in the following diagram (Fig. 9).
The implementation of the IT2FLS GUI is analogous to the GUI used for Type-1 FLS in the Matlab® Fuzzy Logic Toolbox, thus permitting the experienced user to adapt easily to the use of IT2FLS GUI. Figures 10 and 11 show the main view of the Interval Type-2 Fuzzy Systems Structure Editor called IT2FIS (Interval Type-2 Fuzzy Inference Systems).
VI. Model Building

Once identified the objective of the model, it is possible to begin to define the rules that compose the model. These rules are very subjective and vary a lot from researches to researches and from area to area. Therefore, this model is composed of logic where most of the researchers could agree upon as well as through own experience & study. This model composes the proposed algorithm and was programmed using the MATLAB software, where Interval Type-2 Fuzzy Logic techniques were implemented. To understand the model it is important to define what the following terms mean:

6.1 Project Time Delay

This term refers to the time that the project will be delayed due to the inherent risks in the project contributing to a project failure. This term has a very close relationship to the project failure.

6.2 Project Time Delay Gradient

This term refers to the impact that a project delay will have upon a given project. This term has a very close relationship to the impact of a project failure.

6.3 Project Management Efficiency

This term refers to how desirable a project is based upon how high the failure is and how high is the impact of this failure. PME (Project management) efficiency is a vogue or uncertain quantity. Interval Type-2 Fuzzy logic is initially introduced to deal with the fuzzy or ill-defined problem. To use interval type-2 fuzzy decision making for the project management efficiency (PME) the following procedure is proposed.

a) Identify the inputs and outputs variables

b) Find the factors that affect the PME. According to this work approach, the major factor that is contributing to the PME is PTD i.e. project time delay and corresponding priority PTDG (Project time delay gradient).

c) Assign membership functions to the variables (interval type-2 fuzzy subsets).

d) Build an interval type-2 fuzzy rule base.

e) Determine decision rules: Expert knowledge experience and various sources in the literature are used here for the development of IF-TAN rule that governs the relationships between inputs and the output.
f) The first function that our interval type-2 fuzzy system is to perform is that of fuzzification. These to do this, we must define what is Low \([\mu_L(L), \mu_U(L)]\), Medium \([\mu_L(M), \mu_U(M)]\), High \([\mu_L(H), \mu_U(H)]\) and Very High \([\mu_L(VH), \mu_U(VH)]\) in terms of various variables like PTD, PTDG and PME.

g) To define this commonly used techniques one proposed by Wang and Mendel Approach. In this approach the input and output variable are divided into various regions using some clustering technique namely interval type-2 fuzzy C-means clustering (IT2FCM) and these are represented in Table1.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Inputs</th>
<th>Output(PME)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PTD(Project Time Delay)</td>
<td>(Project Management Efficiency)</td>
</tr>
<tr>
<td></td>
<td>PTDG(Project Time Delay Gradient)</td>
<td></td>
</tr>
<tr>
<td>Low ([\mu_L(L), \mu_U(L)])</td>
<td>[(0-30),(0-35)]</td>
<td>[(0-20),(0-25)]</td>
</tr>
<tr>
<td>Medium ([\mu_L(M), \mu_U(M)])</td>
<td>[(20-25),(50-55)]</td>
<td>[(15-20),(55-60)]</td>
</tr>
<tr>
<td>High ([\mu_L(H), \mu_U(H)])</td>
<td>[(45-50),(75-80)]</td>
<td>[(45-50),(80-85)]</td>
</tr>
<tr>
<td>Very High ([\mu_L(VH), \mu_U(VH)])</td>
<td>[(65-70),(95-100)]</td>
<td>[(75-80),(95-100)]</td>
</tr>
</tbody>
</table>

The model build using fuzzy logic tool box is represented below:

**6.4 Scenario**

A scenario have been introduced to represent the developed Interval Type-2 Fuzzy rule base Model and evaluated using MATLAB scripts, Rule viewer and 3Dgraphs for the Project Time Delay (PTD) and Project Time Delay Gradient(PTDG) to determine the Project Management Efficiency (PME).

The results of first four scenarios are shown in Table2.
VII. Conclusions

The results of first four scenarios are compatible as per reasoning to know the project management efficiency. It is safe to assume that the developed model is working correctly. These results show that it is easy to choose between a project that is of low, medium and high project management efficiency, rather it becomes difficult to choose the projects that have high project management efficiency. This can never be eliminated but it can be fine, tuned by adding more variables to the analysis. In terms of reasoning and hypothetical data, the algorithm is working correctly, but in order to test the integrity of the algorithm, it should be applied to test the projects (in the fields), past and present, failed and successful. The results in the interval type-2 fuzzy decision making in management has similar result to the type-1 fuzzy with moderate uncertain footprints. To better characterize the interval type-2 fuzzy models we need to generate more case studies with better knowledge bases for the proposed problems, therefore classify the interval type-2 fuzzy model application strengths. The design and implementation done in the IT2FLS Toolbox is potentially important for research in the interval type-2 fuzzy logic area, thus solving complex problems on the different applied areas.

References