ORESTA

# APPLICATION SOLUTION TO THE STAGE OF AGGREGATION METHOD FOR ASSESSING THE QUALITY OF SERVICE PROVIDED

## Milovan Tomašević

Faculty of Information Studies in Novo Mesto, Slovenia

*Original scientific paper*

**Abstract.** *Formed FAM4QS (Fuzzy Aggregation Method for Quality Service (software), FAM4QS), has been created with modification of LSP (Logical scoring of preferences) methods and SSSI (Six-Step Service Improvement method used LSP (Logical scoring of preferences) algorithms. This had imposed a need for the support of the appropriate software Given that FAM4QS is a new and unique approach to this issue, the proposed software provides unique computer support for this method. With the support of FAM4QS, it is possible for the decision-maker to better demonstrate its own subjective preferences in multi-criteria decision-making. An overview of a large number of results allows numerous analyzes of the application of this decision-making method, as well as an increase in the efficiency of the decision-making process itself. This makes it easier to analyze and consider their solutions, while at the same time it provides managers to use this method in deciding in easy way.*

**Key words**: *FAM4QS, software quality, software services*

## 1. Introduction

Companies in the modern market face global competition, but also with demands for increasing profits, which is happening in times of constant change. In order for companies to accomplish their task in relation to the mentioned challenges, it is necessary to invest certain resources in reliable information technologies and in specific software solutions for the given industrial branch. This also means reorganizing own business with minimal changes when it comes to profitability and transparency. (Gajic, 2013). End-user expectations for software services are increasing.

It is common knowledge that today's society is developing into an information society. This technology becomes an instrument in the service of information, so information is knowledge, power, and money.

* Corresponding author.
milovan.tomasevic@fis.unm.si

The speed and success of the application of information technology will become the basic factor of the strength and usability of today's managers (Ilić et al. 2017). Managers who decide on a day-to-day basis determine and choose ways to solve the problems they face, which will be in line with the set aims of the organization, but also taking into account the circumstances in which the business takes place. On that case, managers use all available sources of information and quality-processed data on the problem or the conditions in which they need to be addressed, while in the absence of the necessary information they rely on intuition and experience.

Professor Oldcorn in his book states that "managers must make decisions - that is their responsibility", according to which the decision-making phases are the following (Oldcorn, 1998):

1. Identify the problem that needs to be solved
2. Discover the facts and find the cause
3. Develop some of the possible solutions to the problem
4. Narrow the choice of the alternative direction of action
5. Make a decision
6. Implement the decision made
7. Analyze the consequences of this solution.

The rapid development of information systems and computer technology, introducing the decision making in the presence of a larger number of the most often conflicting criteria. The specific approach to the application of information systems in decision-making has imposed a decision support system that, together with expert systems, provides support for decision-making.

Practical managerial problems set different and diverse requirements, often with different relative significance, differently sensitive to changes in input and output sizes. Therefore, managerial decision-making requires the application of multi-criteria decision-making methods. A number of different and diverse criteria provide a more comprehensive and objective picture in accordance with the requirements that the decision-maker sets. Criteria can appear in different units, often with different relative significance and different requirements for maximizing or minimizing. This method makes it possible to better understand the underlying causes of specific service behavior. Understanding the behavior of service is a key prerequisite for improving services. The method encapsulates a systematic approach in a comparative analysis of the defined parameters of each service, with the same parameters of other services that belong to the same ranking.

In order to remain competitive, it is very important to constantly improve the quality of software services and be able to meet new needs faster (to be more agile) (Tomašević, 2017). Below is proposed a software solution FAM4QS that can be used in continuous improvement of quality.

## 2. The Method

Formed FAM4QS (Fuzzy Aggregation Method for Quality Service (software), FAM4QS), has been created with modification of LSP (Logical scoring of preferences) methods (Dujmović and Dujmović, 2016; Dujmović, 2018) and SSSI (Six-Step Service Improvement method used LSP (Logical scoring of preferences) algoritama (Marković and Maksimović, 2012). This had imposed a need for the support of the appropriate software (Tomaseivć et al. 2018). The FAM4QS mathematical model for

assessing the quality of the service provided is based on operations with fuzzy numbers (Tomašević, 2017; Tomaseivć et al. 2018). By formulating the FAM4QS method, a more accurate assessment of the quality of the service is done, choosing different values for degrees in the aggregation used to estimate the parameters, or groups of system parameters, and the service itself. Also contributing to a better assessment, which is conditioned by the different nature of the parameters. That difference implies more or less disjunctively, that is, the conjugacy of the form of the chosen aggregation function (greater r disjunctive form, less r more conjunctive form).

Instead of standard real numbers, the model looks at fuzzy numbers and corresponding operations defined over them shown in (Stević, 2017; Puška et al. 2018; Stević et al. 2018; Stević et al. 2019; Chatterjee et al. 2019). Justification for the introduction, i.e. replacing crisp numbers with fuzzy numbers consists of the fact that the estimates of the parameters considered in the system are either vague (imprecise) or can range in a range.

Below is a shown in detail of how to calculate the quality of service using FAM4QS.

In experimental data processing, the use of the fuzzy method includes the following steps:

- • data fuzzification;
- • processing the fuzzy data;
- • defuzzification the results.

The first step shows that data which is vague, for example, about 20% fuzzification, i.e. we present the fuzzy set (fuzzy number) (Klement et al. 2000). The second step is to work with these fuzzy objects, for example, the addition of two fuzzy numbers (Klir et al. 1995). The result of the second step is the fuzzy number, and it is usually required to answer to the solution of problem be a crisp number, and in the third step, it is performed defuzzification of that number, that is, assigned a crisp value.

Depending on the nature of the data, i.e. professional assessments (whether precisely determined or not) are applied fuzzy numbers and the fuzzy operations of them, for imprecisely determined or not accurately estimated weights $w_i$ on the following way:

$$\hat{E} = \left( \hat{w}_1 \cdot (\hat{e}_1)^r + \ldots + \hat{w}_n \cdot (\hat{e}_n)^r \right)^{\frac{1}{r}}. \tag{1}$$

The parameter estimation is first calculated $P_j, j = 1 \ldots m$ using equation (2):

$$\hat{e}_j = \left( \sum_{q=1}^{k} \hat{w}_{jq} \cdot \left( \hat{e}_{jq} \right)^{r_j} \right)^{\frac{1}{r_j}}. \tag{2}$$

Application solution to the stage of aggregation method for assessing the quality of service provided

by each of the fuzzy numbers $\hat{w}_{jq} = (l_{w_{jq}}, m_{w_{jq}}, r_{w_{jq}})$ joins its $\alpha$ section according

to (Tomašević, 2017): $\left[w^{*}_{jq}, w^{**}_{jq}\right]$, and each of fuzzy number $\hat{e}_{jq} = (l_{e_{jq}}, m_{e_{jq}}, r_{e_{jq}})$

joins its section according to (Tomašević, 2017): $\left[e^{*}_{jq}, e^{**}_{jq}\right]$.

Now $\alpha$ section of $\hat{e}_j$ can be calculated by the following way:

$$^{\alpha}\hat{e}_j = (\sum_{q=1}^{k}\left[w^{*}_{jq}, w^{**}_{jq}\right]\cdot\left[e^{*}_{jq}, e^{**}_{jq}\right]^{r_j})^{\frac{1}{r_j}}, \tag{3}$$

After that we using equation (4):

$$^{\alpha}\hat{e}_j = \left[e^{*}_j, e^{**}_j\right] = \left[(\sum_{q=1}^{k}w^{*}_{jq},(e^{*}_{jq})^{r_j})^{\frac{1}{r_j}}, \sum_{q=1}^{k}w^{**}_{jq},(e^{**}_{jq})^{r_j})^{\frac{1}{r_j}}\right]. \tag{4}$$

By applying previous equations formule (1) assessment of service is calculated as fuzzy value, so the final assessment of service in form of section is:

$$^{\alpha}\hat{E} = (\sum_{j=1}^{k}\left[w^{*}_j, w^{**}_j\right]\cdot\left[e^{*}_j, e^{**}_j\right]^{r})^{\frac{1}{r}} \tag{5}$$

In analogy to the previous use of the rules for working with intervals, we get:

$$^{\alpha}\hat{E} = \left[E^{*}, E^{**}\right] = \left[(\sum_{j=1}^{k}w^{*}_j \cdot \left(e^{*}_j\right)^{r})^{\frac{1}{r}}, (\sum_{j=1}^{k}w^{**}_j \cdot \left(e^{**}_j\right)^{r})^{\frac{1}{r}}\right]. \tag{6}$$

Especially if $\hat{e}_j$ is crisp value equal to $e_j$ :

$$^{\alpha}\hat{E} = \left[E^{*}, E^{**}\right] = \left[(\sum_{j=1}^{k}w^{*}_j \cdot e^{r}_j)^{\frac{1}{r}}, (\sum_{j=1}^{k}w^{**}_j \cdot e^{r}_j)^{\frac{1}{r}}\right]. \tag{7}$$

With the FAM4QS method, the ranking of the service from the lowest C, middle B and highest rank A was done according to the following criterion:
Observing the mean value interval $\left[E^{*}, E^{**}\right], i = 1, ... n$ :

$$E = \left[\frac{1}{n}\sum_{i=1}^{n}E^{*}_i, \frac{1}{n}\sum_{i=1}^{n}E^{**}_i\right], \tag{8}$$

Adding for example. $\pm 10\%$ or $\pm 5\%$ (UCL 1.05, LCL 0.95) on left and right border of interval:

$$UCL = \left[ 1.1 \cdot \frac{1}{n} \sum_{i=1}^{n} E_i^*, \ 1.1 \cdot \frac{1}{n} \sum_{i=1}^{n} E_i^{**} \right],$$

$$LCL = \left[ 0.9 \cdot \frac{1}{n} \sum_{i=1}^{n} E_i^*, \ 0.9 \cdot \frac{1}{n} \sum_{i=1}^{n} E_i^{**} \right], \tag{9}$$

getting the criterion of choosing whether a service belongs to the highest-ranking (A) or the lowest (C) is obtained. Those services that have a core (dots) (ie $\alpha$ section for $\alpha = 1$) higher than the right border UCL have the highest-ranking (A), and those services that have a core less than the left-hand LCL have the lowest ranking (C). Services with a core within the left-hand LCL and right-hand side of UCL are middle-level services (B).

## 3. FAM4QS Implementation

FAM4QS is written in the programming language C#. Defined operators facilitate basic operations with fuzzy numbers and alpha cross-sections (Figure 1).

```csharp
public static AlphaSection FromFuzzy(FuzzyNumber fuzzy)
{
    var alpha = new AlphaSection();
    alpha.Start = AlphaValue * fuzzy.Middle + (1 - AlphaValue) * fuzzy.Left;
    alpha.End = AlphaValue * fuzzy.Middle + (1 - AlphaValue) * fuzzy.Right;
    return alpha;
}
public static AlphaSection operator +(AlphaSection a1, AlphaSection a2)
{
    return new AlphaSection(a1.Start + a2.Start, a1.End + a2.End);
}
public static AlphaSection operator +(AlphaSection a, FuzzyNumber f)
{
    var alpha = FromFuzzy(f);
    return alpha + a;
}
public static AlphaSection operator +(FuzzyNumber f, AlphaSection a)
{
    return a + f;
}
public static AlphaSection operator -(AlphaSection a1, AlphaSection a2)
{
    return new AlphaSection(a1.Start - a2.Start, a1.End - a2.End);
}
public static AlphaSection operator *(AlphaSection a1, AlphaSection a2)
{
    return new AlphaSection(a1.Start * a2.Start, a1.End * a2.End);
}
public static AlphaSection operator *(AlphaSection a, FuzzyNumber f)
{
    var alpha = FromFuzzy(f);
    return a * alpha;
}
public static AlphaSection operator *(FuzzyNumber f, AlphaSection a)
{
    var alpha = FromFuzzy(f);
    return a * alpha;
}
public static AlphaSection operator *(AlphaSection a, double d)
{
    return new AlphaSection(a.Start * d, a.End * d);
}
public static FuzzyNumber operator *(FuzzyNumber fuzzy, double score)
{
    var res = new FuzzyNumber();
    res.Left = fuzzy.Left * score;
    res.Middle = fuzzy.Middle * score;
    res.Right = fuzzy.Right * score;
    return res;
}

public static FuzzyNumber operator +(FuzzyNumber f1, FuzzyNumber f2)
{
    return new FuzzyNumber(f1.Left + f2.Left, f1.Middle + f2.Middle, f1.Right +
                    f2.Right);
}
```

Figure 1. Basic operations with alpha sections and fuzzy numbers

Basic operations include: addition, subtraction and multiplication. For example, the addition of an alpha cross-section with a fuzzy number is done by converting the

first fuzzy number into an alpha cross-section, and then the operation of the addition of two alpha cross-sections is performed, which collects the initial boundaries of the interval with the initial, that is shown in Figure 2.

```csharp
public static AlphaSection operator *(FuzzyNumber f1, FuzzyNumber f2)
{
    AlphaSection res = new AlphaSection();
    res.Start = f1.Left * f2.Left + AlphaSection.AlphaValue * (f1.Left * f2.Middle - 2.0
    * f1.Left * f2.Left + f2.Left * f1.Middle) + Math.Pow(AlphaSection.AlphaValue, 2.0)
    * (f1.Middle - f1.Left) * (f2.Middle - f2.Left);
    res.End = f1.Right * f2.Right + AlphaSection.AlphaValue * (f1.Middle * f2.Right -
    2.0 * f1.Right * f2.Right + f1.Right * f2.Middle) +
    Math.Pow(AlphaSection.AlphaValue, 2.0) * (f1.Middle - f1.Right) * (f2.Middle -
    f2.Right);
    return res;
}
public static AlphaSection operator ^(AlphaSection alpha, double pow)
{
    return new AlphaSection(Math.Pow(alpha.Start, pow), Math.Pow(alpha.End, pow));
}

public static FuzzyNumber operator ^(FuzzyNumber fuzzy, double pow)
{
    var res = new FuzzyNumber();
    res.Left = Math.Pow(fuzzy.Left, pow);
    res.Middle = Math.Pow(fuzzy.Middle, pow);
    res.Right = Math.Pow(fuzzy.Right, pow);

    return res;
}
```

Figure 2. Complex operations with alpha cross-sections and fuzzy numbers

```csharp
private void CalculateFAM4QS()
{
    CollectScores();
    var subgroups = Groups.Keys.Where(x => !x.Equals(Const.MainGroupName)).ToList();
    foreach (var subgroupName in subgroups)
    {
        var subgroup = Groups[subgroupName];
        var selectedR = subgroup.RList.Where(x => x.IsSelected).ToList();
        if (selectedR.Count <= 0)
        {
            ...
            return;
        }

        var services = subgroup.Services;
        foreach (var service in services.Values)
        {
            service.KrispCalculation.Clear();
            service.FuzzyCalculations.Clear();
            foreach (var r in selectedR)
            {
                if (subgroup.WeightType == NumberType.Krisp)
                {
                    KrispWeightCalc(service, r, subgroup.Weights);
                }
                else
                {
                    FuzzyWeightCalc(service, r, subgroup.Weights, subgroup.ScoreType);
                }
            }
        }
    }
    List<List<string>> combinations = new List<List<string>>();
    foreach (var group in Groups.Values)
    {
        combinations.Add(group.RList.Where(x => x.IsSelected)
                                    .Select(x => x.DisplayName)
                                    .ToList());
    }
    Console.WriteLine(combinations);
    var product = combinations.CartesianProduct();
}
```

Figure 3. Calculation of FAM4QS method

Method Calculate FAM4QS () calculating imprecise data in a way that *CollectScores ()* collecting the values of the ratings entered by the user in the rating table (Figure 3). *var subgroups = Groups.Keys.Where(x => !x.Equals(Const.MainGroupName)). ToList();* is a code that filters all the groups and only names the subgroups. With these names, *for* loops, it goes through all subgroups. First, check whether the user has selected $r$ values for each subgroup. If not, the error message is printed and the code execution is stopping.

If the check $r$ value is passed, the code continues by going through all the services, using the other *for* the loops. First, all previous calculations are canceled. Then, the third *for* loop goes through all the selected $r$ values.

Depending on the selected weight type (Crisp or Fuzzy), the estimation of each subgroup is calculated. By calling the *CartesianProduct method ()*, combinations of selected $r$ values are obtained for all groups.

```csharp
using (SaveFileDialog saveDialog = new SaveFileDialog())
{
    saveDialog.Filter = "FAM4QS Data|*.tfcd";
    saveDialog.Title = "Save data to file";
    saveDialog.ShowDialog();

    //if (saveDialog.FileName != null)
    if (!string.IsNullOrEmpty(saveDialog.FileName))
    {
        //if location is valid, save data as blob
        var fs = (FileStream)saveDialog.OpenFile();
        Tools.SaveObject(Data, fs);
        fs.Close();
    }
}
                        ...

Stream myStream = null;
OpenFileDialog openDialog = new OpenFileDialog();

openDialog.InitialDirectory = "c:\\";
openDialog.Filter = " FAM4QS Data|*.tfcd";
openDialog.FilterIndex = 2;
openDialog.RestoreDirectory = true;
if (openDialog.ShowDialog() == DialogResult.OK)
{
    try
    {
        if ((myStream = openDialog.OpenFile()) != null)
        {
            using (myStream)
            {
                Data = Tools.LoadObject<DataHolder>(myStream);
            }
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Error: Could not read file from disk. Original error: " +
                    ex.Message);
    }
}
                        ...
```
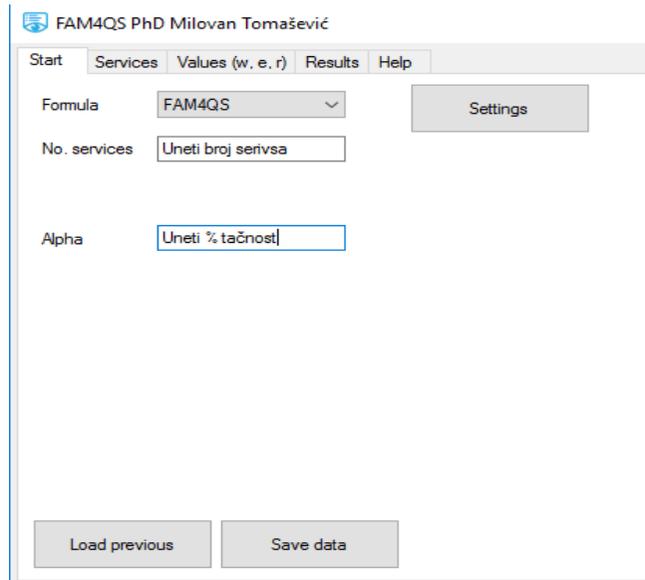
Figure 4. Save and loading data

The standard way of storing objects in a file in the C # programming language is specified in the Windows Forms environment, as well as loading them from files (Figure 4).
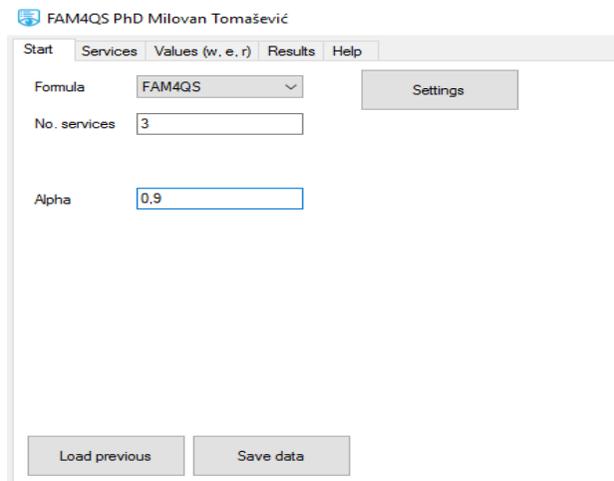
The first step defines the number of services that are evaluated and the percentage accuracy of the results is entered (Figure 5).

Figure 5. Entering initial data

In the example, it is shown case for three services and a probability of accuracy of 90% (Figure 6).

Figure 6. Example of entering data

In step 2 (Figure 7) we enter the names of the services we want to compare.
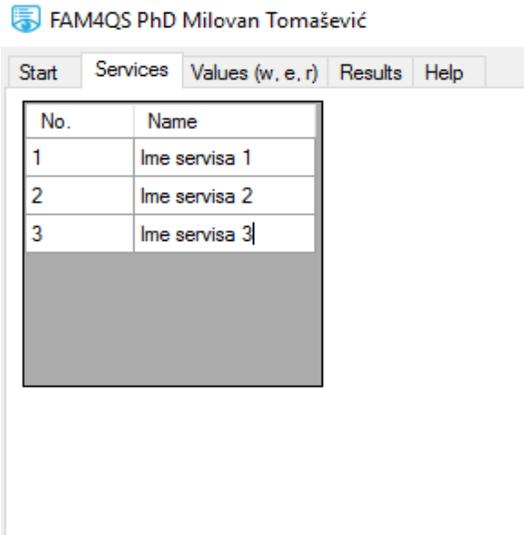


Figure 7. Entering the names of the services

In step 3 (Figure 8), the parameter values for the groups are selected, whether the correct value or the fuzzy number.
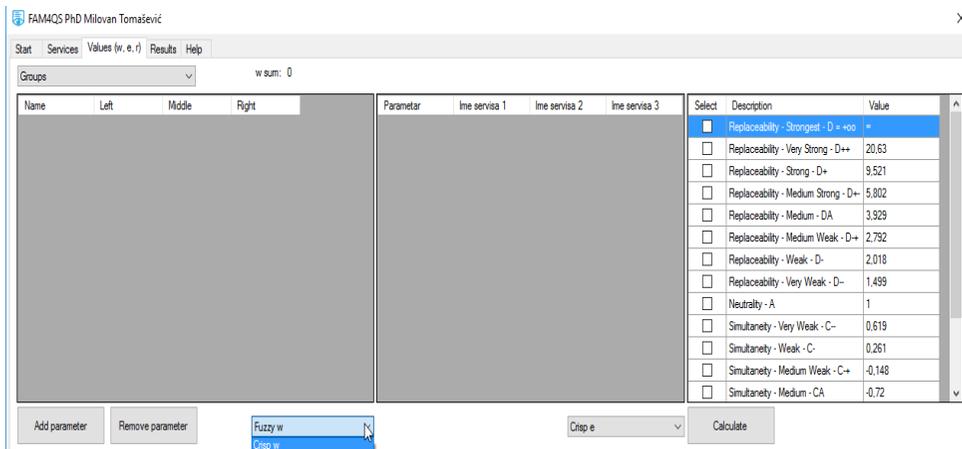


Figure 8. Selection of the parameter type

In step 4 (Figure 9), group parameters are entered.

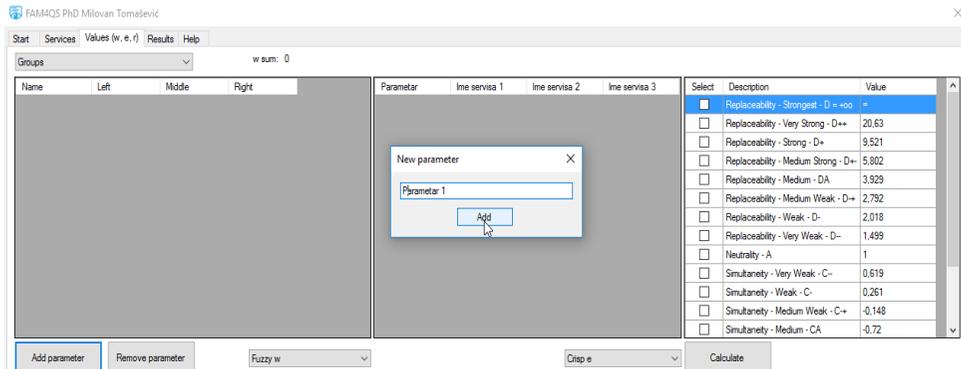Application solution to the stage of aggregation method for assessing the quality of service provided



Figure 9. Entering parameters for groups

In step 5 (Figure 10), values for the parameters are entered.
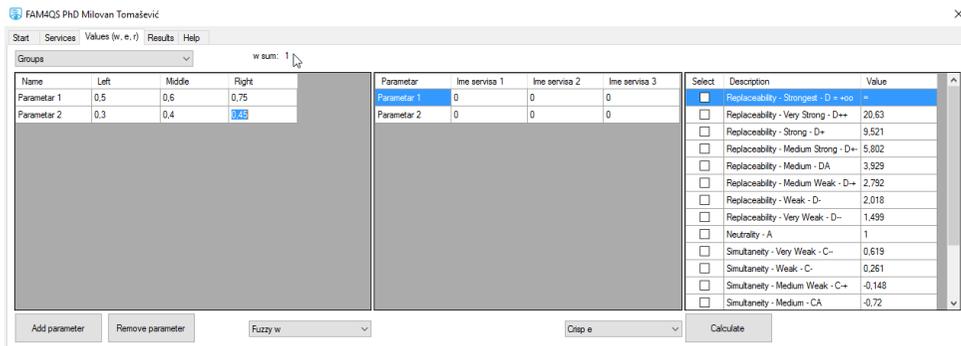


Figure 10. Entering values for parameters

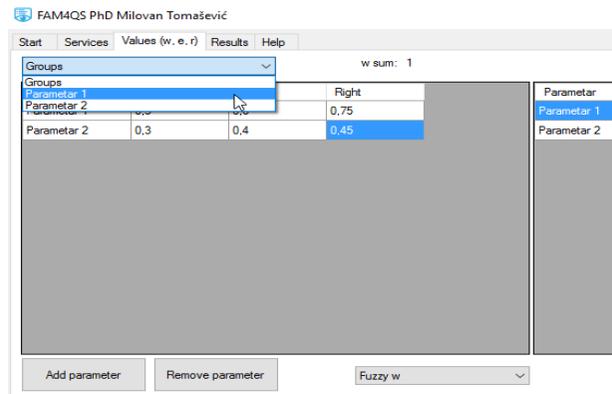In step 6 (Figure 11), a parameter is selected for entering data into subgroups.



Figure 11. Selection a parameter for entering data into subgroups

In the seventh step (Figure 12), the selection of the type of parameter in the subgroup is made.
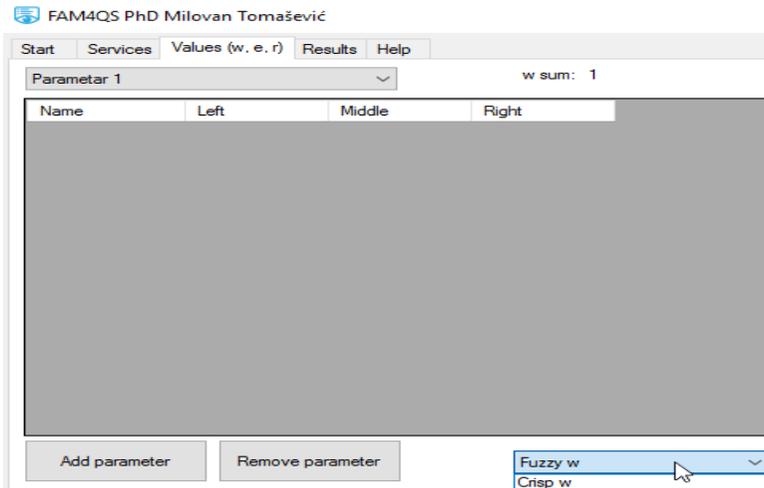


Figure 12. Selection the type of parameters in the subgroup

In the eighth step (Figure 13) subgroup parameters and assign values are entered (Figure 14):

- • weight coefficients for parameters,
- • each service is evaluated by this parameter and
- • values for r are selecting.

The values for r are taken from the nature of the data, which is explained in more detail in the previous section.
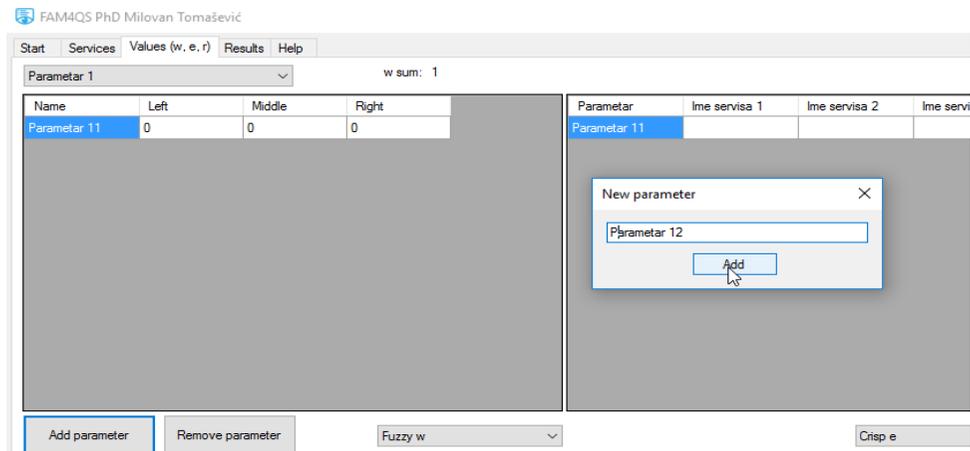


Figure 13. Entering parameters in a subgroup

Application solution to the stage of aggregation method for assessing the quality of service provided
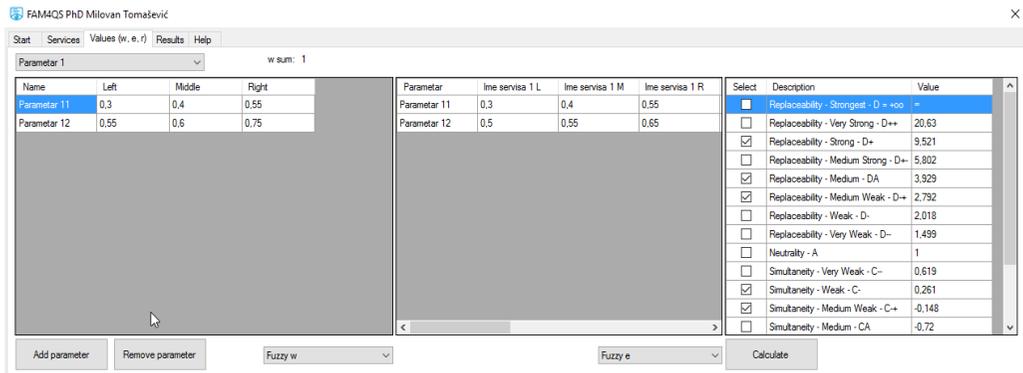


Figure 14. Assigning values to parameters from a subgroup

This process (step seven and eight) is repeated for other groups of parameters. When finished, click the *Calculate* button. In the ninth step, results are obtained as all combinations of selected values for r. In relation to the data type, results, fuzzy numbers (Figure 15) or intervals are obtained (Figure 16). The mean value of the fuzzy number is the mean value - crisp.



Figure 15. Results – fuzzy number



Figure 16. Results - interval

In step ten:
- • Ranking services by quality, showing them graphically,
- • Services are evaluated and
- • Analyze the best and the worst.

## 4. Discussion

Unlike LSP (Logical scoring of preferences), in the newly introduced model here, instead of the standard real numbers, the fuzzy numbers and the corresponding operations defined over them are observed. Justification for the introduction, i.e. replacing crisp numbers with fuzzy numbers consists of the fact that the estimates of the parameters considered in the system are either vague (imprecise) or can range in a range.

Similar to the LSP, where individual parameters are evaluated, as well as the entire service, and here it is taken into account that each of the individual parameters does not participate equally in the overall assessment, and therefore assign different weights which are in this model triangular fuzzy numbers. And for the estimates for individual parameters, the triangular fuzzy numbers are taken, which finally gives the assessment of the service that is the fuzzy number (it does not have to be necessarily triangular). For the purpose of this calculation, the apparatus fuzzy arithmetic was used, ie the display of the results as an alpha cross-section (closed interval), rather than a crisp number.

By selecting an alpha the degree of confidence in the assessment of the experts for a particular parameter is chosen, and depending on this, the result obtained is vague. The final result is mainly corrected by $\pm 5\%$ (or 10% depends on the nature of the parameters themselves) and the obtained interval values (UCL and LCL) that determine the rank (quality) of the service.

The nature of the results determines the choice of the rank of the service, that is, if the core is the fuzzy number corresponding to the estimated service, less than the left limit UCL is assigned the worst rank, and if it is higher than the right limit, the LCL is the highest ranking. All observed values of service ranges that are between these borders are of the middle rank.

Number $r$ which occurs in the formula for estimating parameters as well as the entire service and determines whether the given rating is more or less pessimistic or optimistic, which is determined by the nature of the parameters.

In a model developed for parameter estimation, a fuzzy-aggregation function is actually used, which in its nature works with imprecise data and generates a new average value from more than one value. Therefore, this apparatus can be used to model a decision that represents some sort of averaged value from several individual imprecision decisions made, in any similar decision making where imprecise data enters. The disadvantage of this model is that the result is not precise, but it is also a result of the imprecision of the experts judgment.

## 5. Conclusions

Forming FAM4QS, imposed a need for the implementation of the appropriate software for its support, which was the goal of the software application of this work as well as the improvement of the support system for multi-criteria decision-making.

The software developed in this paper had the basic goal of automating the FAM4QS-based calculation based on aggregation functions. In this way, a qualitatively new approach to the FAM4QS budget is provided, and at the same time an analysis of the solutions obtained by this software solution. In addition to the calculation, the software provides a display of comparative results obtained by changing the parameters $r$ over certain groups and subgroups of the given service. This enables the analysis obtained based on different parameter values r (whether the given rating of a particular service is more or less pessimistic or optimistic, which is also determined by the nature of the parameters).

The contribution of this application solution is reflected in a more faithful reflection of reality and increasing the quality of decisions made, making this process faster and more efficient. Also, the software solution is reflected in the possibility of direct application of the developed software and providing new information for the scientific and professional public, which can represent a quality basis for further development of the model for decision making.

The presented solution is general and with certain settings and a higher level of integration can be applied in different decision areas.

Research has shown that there are certain constraints that require attention and should be the subject of further research in the future:

1) Extension and testing of methods with a larger number of parameters

2) Application of neural networks

3) Developing a web system that would make it easier to make a decision, or whose result would not be a number, but a clear report based on a large projection of parameters and knowledge base (the current state - the standard).

In a mathematical view, the given model can be changed in several directions:
- By changing the aggregation function - one can observe a function that depends not only on one parameter $r$, but more than that of which we can adjust the criteria for evaluating the service.
- Presenting values not as triangular but as trapezoidal fuzzy numbers.

## References

Chatterjee, P., & Stević, Ž. (2019). A two-phase fuzzy AHP-fuzzy TOPSIS model for supplier evaluation in manufacturing environment. Operational Research in Engineering Sciences: Theory and Applications, 2(1), 72-90.

Dujmović J., (2018). Soft Computing Evaluation Logic: The LSP Decision Method and Its Applications. John Wiley & Sons, Inc., (Chapter 3).

Dujmović, J. J., & Nagashima, H. (2006). LSP method and its use for evaluation of Java IDEs. International journal of approximate reasoning, 41(1), 3-22.

Gajic G., (2013). Unapredenje upravljanja naftno-gasnim sistemima, Doktorska disertacija, Univerzitet u Novom Sadu Fakultet tehnickih nauka, Novi Sad

Graovac B., (2009). Elektronska identifikaciona dokumenta na bazi SMART kartica, Master rad, Univerzitet Singidunum, Beograd

Ilić, Đ., Andrejić, M., & Janošević, M. (2017). Attachment to improvement of communication and business in logistical operations of system. Vojno delo, 69(7), 424-436.

Klement E. P., Mesiar R., Pap E., (2000). Triangular Norms, Springer

Klir G. J., Yuan B., (1995). Fuzzy Sets and Fuzzy Logic: Theory and Applications, 1st edition, Prentice-Hall, Inc. Upper Saddle River, NJ, USA

Marković V., Maksimović R., (2012). A contribution to continual software service improvement based on the six-step service improvement method. International Journal of Software Engineering and Knowledge Engineering, 22(04), 549-569.

Oldcorn, R. (1998). Management, Hampshire: Macmillan Education Ltd.

Puška, L. A., Kozarević, S., Stević, Ž., & Stovrag, J. (2018). A new way of applying interval fuzzy logic in group decision making for supplier selection. Economic Computation & Economic Cybernetics Studies & Research, 52(2).

Sistem za učenje na daljinu, VIŠER, Proces odlučivanja, http://lectio.viser.edu.rs (Accessed 10 April 2019)

Stević, Ž., Vasiljević, M., Puška, A., Tanackov, I., Junevičius, R., & Vesković, S. (2019). Evaluation of suppliers under uncertainty: a multiphase approach based on fuzzy AHP and fuzzy EDAS. Transport, 34(1), 52-66.

Stević, Ž. (2017). Modeling performance of logistics subsystems using fuzzy approach. The International Journal of Transport & Logistics, 17(42), 30-39.

Stevic, Z., Vasiljevic, M., Zavadskas, E. K., Sremac, S., & Turskis, Z. (2018). Selection of carpenter manufacturer using fuzzy EDAS method. Eng. Econ, 29, 281-290.

Stević, Ž., Stjepanović, Ž., Božičković, Z., Das, D., & Stanujkić, D. (2018). Assessment of Conditions for Implementing Information Technology in a Warehouse System: A Novel Fuzzy PIPRECIA Method. Symmetry, 10(11), 586.

Tehnike donošenja odluka, https://www.link-elearning.com (Accessed 10 April 2019).

Tomašević M., (2017). Adaptivni model za upravljanje lancima snabdevanja u malim i srednjim preduzećima, Doktorska disertacija, Univerzitet u Novom Sadu Fakultet tehnickih nauka

Tomašević, M., Ralević, N., Stević, Ž., Marković, V., & Tešić, Z. (2018). Adaptive Fuzzy Model for Determining Quality Assessment Services in the Supply Chain. Tehnički vjesnik, 25(6), 1690-1698.