Multi-criteria Analysis and Decision Methodology for the Selection of Internet-of-Things Hardware Platforms

Edgar M. Silva, Ricardo Jardim-Gonçalves

HAL Id: hal-01629562
https://inria.hal.science/hal-01629562

Submitted on 6 Nov 2017

Multi-Criteria Analysis and Decision Methodology for the Selection of Internet-of-Things Hardware Platforms

Edgar M. Silva and Ricardo Jardim-Goncalves,

Universidade Nova de Lisboa, Faculdade de Ciências e Tecnologia, Portugal
UNINOVA, Centro de Tecnologia e Sistemas, Portugal
{ems, rg}@uninova.pt

Abstract. The Internet-of-Things (IoT) is today a reality, and Smart Systems have taken advantage of this to improve its own sense, act and control capabilities. IoT is a highly heterogeneous environment composed by a vast number of "things" (sensors, smart objects, etc.). These "things" are based on hardware platforms which can differ widely since manufactures are being capable of develop new devices every day to tackle different application domains. Consequently, a problem emerges regarding which will be the suitable, proper hardware solution for an IoT deployment. Make a right decision is probably one of the toughest challenges for science and technology managers. This work proposes a novel methodology to analyze a set of hardware alternatives based on user's multi-criteria requirements, and advice on the more suitable hardware solution for a specific situation. For proof-of-concept it is used different Arduino boards as hardware alternatives, in which user requirements are based on hardware features. This methodology foresees its use during the development of Smart Systems (e.g.: Transportation, Healthcare) to optimize the selection of hardware platforms.

Keywords: IoT, Smart Systems, Decision Making, Analytic Hierarchy Process, Arduino.

1 Introduction

The emerging Internet-of-Things (IoT) technology is becoming more and more important in society. Devices are being used for environment awareness giving to people a more truly interaction with the surrounding world. With advances in microelectronic technologies, manufactures are being capable to develop small and cheap devices, to tackle different application domains and services based on IoT deployments [1]. Hence, these devices have limited resources, for example low power, small processing and storage capabilities, and for this reason, devices must be chosen properly accordingly to the task they will fulfil. Consequently, a research question arises: **"which is the more suitable device (hardware platform) to implement a certain Smart System solution in IoT?"**.

Multi-Criteria Decision-Making (MCDM) is one of the most widely used decision methodologies in sciences, business, government and engineering worlds. MCDM methods can help to improve the quality of decisions by making the decision-making

process more explicit, rational, and efficient [2, 3]. MCDM is an approach comprehensive of a set of techniques which provides ranking methods to classify criteria of a certain problem and assist on decision phase. MCDM can tear down complex problems into small, manageable problems, in such way that small sets of data can be analyzed from which is obtained reasonable judgements. Reassembling all pieces is then formed a coherent overall system view that will aid decision makers on their decisions [4]. From this emerges the following hypothesis that could respond to the question in place: "**If during a Smart System design, engineers could use a multi-criteria methodology to rank IoT hardware solutions, then more proper solutions could be selected, leading to a more suitable (i.e.: low-energy consumption, low cost, etc.) design of Smart-Systems.**" This work presents a methodology capable of analyse a set of hardware alternatives based on user's multi-criteria requisites and advise which is the more suitable hardware solution for a specific situation. Consequently, improve in overall, a Smart System deployment and operation (see Figure 1).
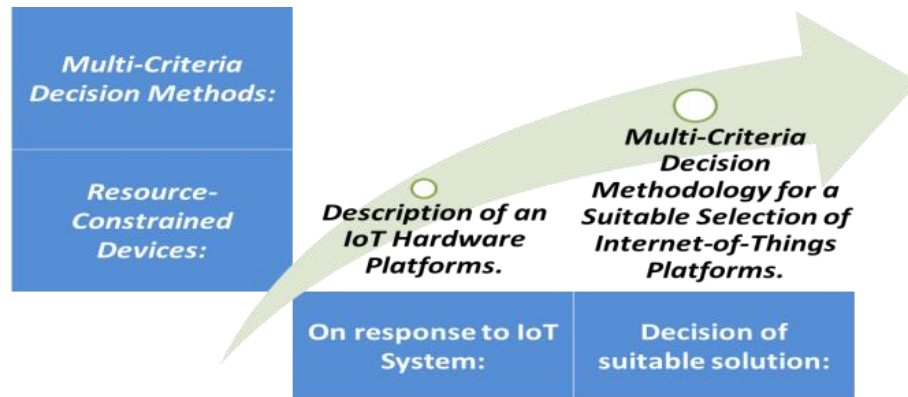

Fig. 1. Multi-Criteria Decision Methodology for the Selection of IoT Hardware Platforms.

2   Hardware Platforms of Smart Systems

People are living in an environment where Smart Systems are everywhere. These systems can be found in areas such as transportation, manufacturing, healthcare, energy, etc. In a succinct definition, Smart Systems can be seen as a system capable to sense, act and control a certain situation. Smart Systems present unique capabilities (cognitive as physical) to optimize their performance in response to external (sensed) information. With the technological advances in the microelectronic these capabilities have improved widely [5]. Deployment of a Smart Systems can vary in many ways, in scale, functionality, implementations and technologies. Although, developers struggle with the same challenges during the deployments design phase. One of problems arises with the need to choose a suitable hardware platform (device) or platforms between a set of alternatives. Even more when manufacturers are being capable of develop new devices to tackle different application domains. These devices, also

known as motes or sensor nodes, have some particular characteristics such as small size, low-power, sensing units, wireless communication and they are cheap [6]. As a result of such features they are categorized as Resource-Constrained Devices (RCD). Multi-Criteria Decision Methods (MCDM) can assist on the development of Smart Systems, in a way that a decision could be provided in terms of what will be a more proper hardware solution for a certain task. For example, sense the environment, perform a system action or apply/compute a control function.

3   Literature Review

3.1   Hardware Platforms for IoT: Resource-Constrained Devices

With the growing interest of major technology players on Internet-of-Things (IoT), it has been seen a constant evolution of microelectronic technologies, bringing to this domain a series of devices which enables the realization of the IoT concept. These devices, also known as motes or sensor nodes, have some particular features such as small size, low-power, and processing, storage, sensing and wireless communication capabilities. These characteristics, from which comes the categorization as Resource-Constrained Devices (RCD), allow Wireless Sensor Networks (WSN)s to differ from other wireless technologies. Devices are reasonably cheap, and still maintain accurateness/preciseness and reliability. Although, energy efficiency is considered to be the major requirement for these type of devices [6, 7].

3.2   Methods for Multi-Criteria Decision-Making

Multi-Criteria Decision-Making (MCDM) is a process that takes a decision upon a set of available solutions, assessing which one is more suitable across diverse, contradicting, qualitative and/or quantitative criteria. MCDM methods have been applied to engineering problems, providing useful insights to decision makers, making their decisions more qualified to overcome complex problems [8]. Based in [8-10], as others could be point out, three of the most widely used decision making methods in literature are the: AHP (Analytic Hierarchy Process); PROMETHEE (Preference Ranking Organization Method for Enrichment Evaluation); and ELECTRE (Elimination Et Choix Traduisant la Realite).

Analytic Hierarchy Process (AHP). The Analytic Hierarchy Process (AHP) was proposed by Thomas L. Saaty [11], and it is aimed to solve problems with multiple and conflicting criteria. AHP is a powerful decision making methodology capable of decompose a complex problem into a systematic hierarchy procedure [12], as depicted in Figure 2.
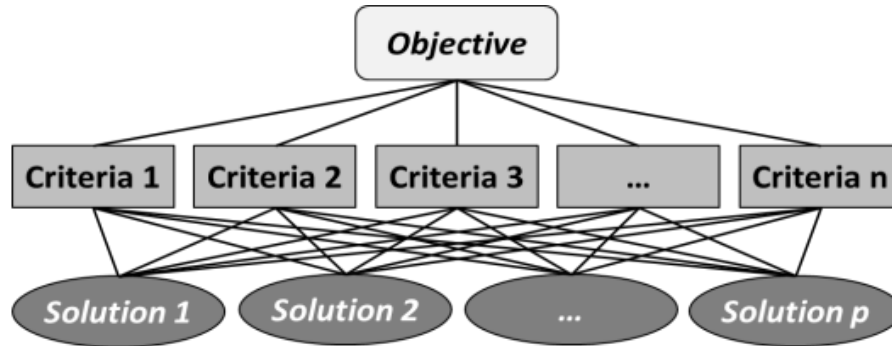
Fig. 2. Example of a criteria and solutions hierarchy.

AHP applies a criteria pairwise comparison based on decision maker judgement, creating the Saaty 1-9 scale to define the criteria priority. In which qualifying a criterion with 1 indicates equal importance, and 9 extremely important. With an n criteria and p solutions it is achieved a decision matrix of $p*n$ size, in which the $\{d_{pn}\}$ element indicates the value for pth solution in respect to the nth criteria. The pairwise comparison matrix is a $m*n$ size, in which the $\{v_{mn}\}$ element indicates the value for mth criteria in respect to the nth criteria. Normalized comparison matrix has its elements calculated as follows: $Nv_{pn}=v_{pn}/w_n$; $w_n=\sum v_{pn}$, $p=1, 2 \ldots$. The contribution of each criteria n for the problem objective, i.e. the eigenvector, is then computed as: $CW_n=\sum w_n/n,\ n=1,2 \ldots$ [13]. Identified the eigenvector, is then possible to rank the solutions through a simple matrix multiplication, i.e.: the decision matrix multiplied by the eigenvector.

PROMOTHEE. Preference Ranking Organization Method for Enrichment Evaluation (PROMETHEE) [14] is a MCDM method, with six different versions based on ranking. The first version, PROMETHEE I, consists on partial ranking; PROMETHEE II on complete ranking; PROMETHEE III performs ranking based on intervals and PROMETHEE IV is a method for the continuous case; PROMETHEE V method has integer linear programming and net flows; finally the PROMETHEE VI method includes a representation of the human brain [13, 15]. The PROMETHEE methodology can be described in five steps, as Figure 3 shows. The decision maker chooses a preference function defined into two independently actions (e.g. Boolean function – value 0 or 1), and then it is applied a solutions comparison. The comparison results and criteria values are used to form a matrix, on which methods can then be applied. For example, PROMETHEE I to create a partial solutions rank and PROMETHEE II to obtain a final solutions ranking [16]. PROMETHEE methods I and IV use preference functions which are more suitable for qualitative criteria. Methods type III and V are normally chosen for quantitative criteria problems. Type II and VI preference functions are used with less frequency.

ELECTRE. Elimination Et Choix Traduisant la Realite (ELECTRE – Elimination and Choice Expressing the Reality) method was created by Bernard Roy [17]. Similar to the previous decision method, ELECTRE has also six variations: type I, II, III, IV, IS and TRI. All ELECTRE methods are based on the same background, but have different process. In ELECTRE I and IS a single solution or a group of solutions are

selected and assigned as a possible solution. ELECTRE II was developed to deal with problems that need to rank solutions from the best to the worst solution. ELECTRE III introduced pseudo-criteria and fuzzy binary outranking relations. ELECTRE IV method was possible to rank solutions without the use of relative criteria coefficients (the only ELECTRE method that did not use it). Finally, ELECTRE TRI assigns categories to solutions [8, 14]. ELECTRE is a preference-based model, performing a pair-wise comparison between solutions. The model assumes three types of relations between solutions: is preferred to; is indifferent to; and incomparable to. ELECTRE brings the notion of thresholds values, used to set concordance and discordance indexes. These indexes are used to classify a solution in relation to another.
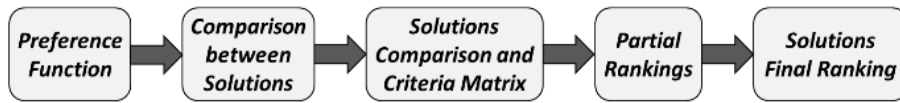

Fig. 3. PROMETHEE methodology.

Summary. The Analytic Hierarchy Process (AHP) follows a prioritization theory, dealing with complex problems that need multi criteria consideration simultaneously. Although, AHP presents some disadvantages. The pairwise comparison based on decision makers' subjective judgment, and the criterion weight has a direct impact on the final score. The PROMETHEE method is consistent, easy to use and does not need great interaction with decision makers. Although, PROMETHEE presents as downside the incapability to react when new alternatives are introduced. The method II is unable to present a preferred solution in a bi-criteria problem, after the method I have been applied. Generally, ELECTRE method does not frequently leads to the case in which one solution stands out from the others. For this reason the method is considered to be more suitable for problems with several solutions and not so many criteria. Also, it is considered to be difficult to understand, and thresholds definition is a problem since there is no "correct" value.

4   Proposed Methodology

This section will describe the proposed methodology which enables users, developers and researchers to assess which hardware platform is more suitable for their own application. The methodology provides means to indicate restrictions if any and/or criteria optimization methods, combined with a criteria prioritization based on decision maker judgement.

Possible solutions, i.e. hardware platforms, are represented by set S in (1):

$$S = \{s_1, s_2, \ldots, s_n\}; n \in N .$$

(1)

The amount of alternatives is a finite number $n \in N$ and is defined by the user, i.e. users select the solutions that will be analyzed. Platforms have a vast number of

features, $m \in N$, although user may or may not consider all of them as important. The set of features/criteria, called F, is defined as in (2).

$$F = \{f_1, f_2, ..., f_m\}; m \in N.$$ (2)

With the sets of possible solutions and criteria defined is then possible to create an Evaluation Table. The Evaluation Table, Ev_{table}, given by (3), is an n-by-m matrix. Rows represent the contemplated solutions and the columns the assessment features. The element $v_{i,j}$ represents the value for feature j of the alternative i.

$$Ev_{table} = \begin{bmatrix} v_{1,1} & v_{1,2} & \cdots & v_{1,m} \\ v_{2,1} & v_{2,2} & \cdots & v_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ v_{n,1} & v_{n,2} & \cdots & v_{n,m} \end{bmatrix} = (v_{i,j}) \in \mathbb{R}^{n \times m}.$$ (3)

Hardware platforms, in our case the possible alternatives, present a vast number of features, and each feature has its own units. Features values are treated as dimensionless. Hardware platforms, in our case the possible alternatives, present a vast number of features, and each feature has its own units. Features values are treated as dimensionless. To perform an analysis on a set of hardware alternatives, considering user's multi-criteria requisites, to obtain an advise on which is the more suitable hardware solution for a specific situation, users must indicate restrictions if any and/or criteria optimization method (minimize or maximize). Currently the proposed methodology presents seven types of restrictions/optimization, from now on call Assessment Constraints, which can be applied by users to assess the more suitable solution(s). The Assessment Constraints set, *AC*, is defined in (4). Each constraint can be applied to one or more criteria. Two or more constraints can be applied to single feature/criteria.

$$AC = \{min, max, mustHave, cannotHave, lessThan, equal, greaterThan\}.$$ (3)

Assessment Constraint *min* and *max* (optimization methods) are used respectively when user's purpose is to minimize or maximize values of a feature. Constraint *mustHave* is used for cases in which a feature must be present. Constraint *cannotHave* is used in the opposite case, i.e. criteria/feature cannot be available in the final proposed solution(s). Constraints *lessThan*, *equal* and *greaterThan*, enables users to set threshold values or look for a specific feature value. When users apply the Assessment Constraints, points are assigned. In the case of *min* and *max*, points are assigned from 1 to n points. For instance, in a minimization applied to a feature f_j, a solution with the smallest value for f_j get n points, and the solution with the higher value in f_j get 1 point. The same principle is used for maximization, 1 point for the lowest f_j value and n points for the higher. For the other Assessment Constraints, it is assigned 0 or n points. For example, the *mustHave* case, a solution with f_j value different from zero gets n points, if the f_j value is zero (which indicates that the

feature is not present in that solution) it get 0 points. When more than one Assessment Criteria is applied to a single feature, assigned points are multiplied. Applying the selected Assessment Criteria to the Evaluation Table, Ev_{table}, is formed the Score Table, ST, formalized next in (5):

$$ST = \begin{bmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,m} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n,1} & p_{n,2} & \cdots & p_{n,m} \end{bmatrix} = (p_{i,j}) \in \mathbb{R}^{n \times m}. \tag{4}$$

To this point, users are able to optimize features using minimization or maximization methods. Are able to set binding rules between solutions and its features, as well as define thresholds for feature values. However, it is not yet possible to assign preference between features, i.e. features priority. To achieve this goal, the proposed methodology takes use of Analytic Hierarchy Process (AHP) presented in Section 3.2. AHP applies to features a pairwise comparison based on decision maker (users) judgement, using the Saaty 1-9 scale to establish criteria priority. Qualifying a feature with 1 indicates equal importance, and 9 extremely important. The AHP comparison matrix, AHP_{table}, is given in (6). The element $\delta_{k,j}$ represents the comparison value for feature j and feature k. It is also important to note that the comparison value, $\delta_{k,j}$, has the inverse value of $\delta_{j,k}$. Specifically, the impact of feature j in feature k has the inverse importance of feature k in feature j. The AHP comparison matrix, AHP_{table}, is then normalized as presented in (7). The contribution of each feature/criteria on the selection of a proper solution is given by the eigenvector and it is computed as described in (8). The final solution ranking, FSR, is then given by (9).

$$AHP_{table} = \begin{bmatrix} \delta_{1,1} & \delta_{1,2} & \cdots & \delta_{1,m} \\ \delta_{2,1} & \delta_{2,2} & \cdots & \delta_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ \delta_{m,1} & \delta_{m,2} & \cdots & \delta_{m,m} \end{bmatrix} = (\delta_{k,j}) \in \mathbb{R}^{m \times m}. \tag{5}$$

$$NAHP_{table} = \frac{(\delta_{k,j})}{\sum_{k=1}^{m}(\delta_{k,i})} \; ; \; k = 1, \ldots, m; \; j = 1, \ldots, m. \tag{6}$$

$$E_{vector} = \frac{\sum_{k=1}^{m}(N\delta_{k,j})}{m} \; ; \; k = 1, \ldots, m; \; j = 1, \ldots, m. \tag{7}$$

$$FSR = (E_{vector} * ST^T)^T. \tag{8}$$

5   Case Study on Hardware Platforms

The selected hardware platforms chosen to validate the proposed methodology are based on Arduino platform. Five different devices were selected, represented by set S, as described in (10). Arduino devices present several features and characteristics suitable for the development of sensor networks. Among the vast number of features, were selected eight to serve as assessment criteria, as described in (11).

Table **1** summaries the selected hardware platforms and values for each feature. Possible solutions and features are placed in the same order as given in (10, 11).

$$S = \{Uno\ Rev3, Mega\ 2560\ Rev3, DUE, Mini\ 05, Micro\}. \tag{9}$$

$$F = \left\{ \begin{array}{c} Energy\ [mA], Cost\ [\text{€}], Clock\ [MHz], \\ Memory\ [KB](Data\ RAM), Memory\ [KB](Program\ Flash), \\ Used\ Memory\ [bytes](RAM), \\ Used\ Memory\ [bytes](Program\ Flash), Weight[g] \end{array} \right\}. \tag{10}$$

Table 1. Hardware Solutions versus Assessment Criteria.

Solutions:	Features/Criteria:							
	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
s_1	25.5	20	16	2	32	9	450	25
s_2	34.2	35	16	8	256	9	642	37
s_3	113	36	84	96	512	0	10492	36
s_4	31.5	14	16	2	32	9	450	6
s_5	41	18	16	2.5	32	157	4264	13

The values for features *Cost, Clock speed, Data Ram Memory, Program Flash Memory and Weight*, respectively $(f_2, f_3, f_4, f_5\ and\ f_8)$, were retrieved from Arduino page[1]. Values for features *Used Memory Ram and Program Flash* when an *Empty application* is installed on the selected Arduino devices, respectively $(f_6\ and\ f_7)$, were retrieved using Arduino IDE version 1.5.7. Arduino IDE was used to program the *Empty application* into devices. *Empty application* is the Arduino operating system without any kind of operation, mainly the operating system footprint. Values of feature *Energy* (f_1), were obtained by placing a multimeter between the power source and the Arduino and then was read the current consumed (milliamps). The Evaluation Table, Ev_{table}, can be obtained directly from Table **1**.

To define the Analytic Hierarchy Process (AHP) table, AHP_{table}, reflecting user criteria importance/preference, was performed a research using popular search tools (IEEE eXplore, Harzing's Publish and BASE). The research was based on titles that

---

[1] https://www.arduino.cc/

contain some key words. The key words "Internet" and "Things" were stuck and varying worlds such "energy", "cost", "weight", etc. The results are summarized in Table 2. The AHP table, AHP_{table}, presented in (12) is formed basically from the preformed research. Notice that this AHP table will be used in the application scenario in Section 6. It also reflects a preference example, since criteria importance can differ widely from one user to others, and it also depends on application purposes.

Table 2. Research Results Summary: A criteria preference/importance.

	Energy	Cost	Processor	Memory	Weight
Percentage [%]	83.9	9.6	3.5	3	0

$$AHP_{table} = \begin{bmatrix} 1 & 2 & 3 & 4 & 4 & 4 & 4 & 9 \\ 1/2 & 1 & 2 & 2 & 2 & 2 & 2 & 9 \\ 1/3 & 1/2 & 1 & 1 & 1 & 1 & 1 & 9 \\ 1/4 & 1/2 & 1 & 1 & 1 & 1 & 1 & 9 \\ 1/4 & 1/2 & 1 & 1 & 1 & 1 & 1 & 9 \\ 1/4 & 1/2 & 1 & 1 & 1 & 1 & 1 & 9 \\ 1/4 & 1/2 & 1 & 1 & 1 & 1 & 1 & 9 \\ 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1 \end{bmatrix}. \tag{11}$$

From AHP_{table}, is possible to compute the normalized AHP comparison matrix, $NAHP_{table}$, from which the eigenvector, i.e. importance of each feature on selecting the more suitable solution, is obtain. The eigenvector is presented in (13).

$$E_{vector} = \{0.32 \quad 0.18 \quad 0.10 \quad 0.10 \quad 0.10 \quad 0.10 \quad 0.10 \quad 0.01\} \tag{12}$$

6 Application Example: Small Storage Unit

For application example was selected one scenario, a Small Storage Unit, where it is needed a hardware platform which presents a small energy consumption, it is cheap and has a large amount of memory available. Since there is no available data regarding energy consumption for the specific application, let's consider the energy consumption of the operating system footprint (in this case Arduino operating system). Accordingly, is then possible to build the Assessment Constraints set, $AC_{scenario}$, presented in (14), and the Score Table, $ST_{scenario}$, as described in (15):

$$AC_{scenario} = \left\{ min_{f_1}, min_{f_2}, max_{f_4}, max_{f_5} \right\} \tag{13}$$

The $AC_{scenario}$, presented in (14) minimizes the first criteria (energy consumption), min_{f_1}, to classify the available hardware platforms in energy

consumption terms. Constraint, min_{f_2}, to minimize the cost, and constraints max_{f_4} and max_{f_5}, to maximize the available memory.

$$ST_{scenario} = \begin{bmatrix} 5 & 3 & 1 & 2 & 3 & 1 & 1 & 1 \\ 3 & 2 & 1 & 4 & 4 & 1 & 1 & 1 \\ 1 & 1 & 1 & 5 & 5 & 1 & 1 & 1 \\ 4 & 5 & 1 & 2 & 3 & 1 & 1 & 1 \\ 2 & 4 & 1 & 3 & 3 & 1 & 1 & 1 \end{bmatrix} \tag{14}$$

Results Comparison. Table 3 presents the final classification results for each available hardware solution. Three different evaluations are available to perform a methods comparison. The first column shows the results when AHP method, alone, is applied to features values. As discussed in Section 3.2, this method presents a direct influence of feature values on the final score. Also, it is not able to reflect user assessment constraints (e.g. min, max). Considering, the objective to minimize all selected criteria, this method advice **Uno Rev3**. But the scenario also includes memory maximization, leading to **DUE** solution. Normally, using AHP method the solution with the higher score is considered the best one. In this scenario that is wrong, excessive energy consumption and cost. Consequently, the result is inconclusive for the presented case study. Since the application objective is more than that, the midway result, $ST_{scenario}$, which consists only on assessment constraints enforcement, indicates that the proper solution is **Mini 05**. The complete proposed methodology execution indicates the same solution, **Mini 05**, but reflects criteria preference and removes all solutions draws. This methodology brings to developers the capability to select the proper solution for a certain task from a set of available solutions. Provides methods to select criteria importance and it is also able to apply constraints functions to features, improving solutions assessment.

Table 3. Application Example Results.

	AHP	$ST_{scenario}$	FSR
Uno Rev3	61.38	17	2.93
Mega 2560 Rev3	107.81	17	2.40
DUE	1124.9	16	1.77
Mini 05	61.96	18	2.96
Micro	448.97	16	2.24

7 Conclusion and Future Work

There are many features/criteria which can influence the overall performance of an application running in a hardware platform. Also, developers have different perspectives of which feature is more important. The proposed methodology enables users, developers and researchers to assess which hardware platform is more suitable

for their own application, depending on the application scenario. It is possible to indicate restrictions (e.g.: must have a certain feature, thresholds for feature values), and apply criteria optimization (minimize or maximize). The solutions ranking is attained using a well-known decision making method, Analytic Hierarchy Process (AHP), aimed to solve problems with multiple and conflicting criteria. Methodology results were able to point out the better/proper solution from the available ones. It also presents a more clear solution ranking compared to the AHP method, which presents an ambiguous result. Future work involves the extension of the analysis to different hardware elements/features. A further step will consider the inclusion of hardware platforms formal descriptions, to facilitate the inclusion process of new criteria.

References

1. Edgar M. Silva and Pedro Maló. Iot testbed business model. In Advances in Internet of Things, volume 4, 37–45, 2014.
2. Boucher TO. and Mcstravic EL. Multi-attribute evaluation within a present framework and its relation to the analytic hierarchy process. The Engineering Economist, 55–71, 1991.
3. Xiaoting Wang and Evangelos Triantaphyllou. Ranking irregularities when evaluating alternatives by using some electre methods. Omega, 36(1):45–63, 2008.
4. Multi-criteria analysis: a manual, 2009. Access in October 20, 2016. http://eprints.lse.ac.uk/12761/1/Multi-criteria_Analysis.pdf.
5. A. Jones, E. Subrahmanian, A. Hamins and C. Grant, "Humans' Critical Role in Smart Systems: A Smart Firefighting Example," in IEEE Internet Computing, vol. 19, no. 3, pp. 28-31, May-June 2015. doi: 10.1109/MIC.2015.54
6. Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. Computer Networks, 52(12):2292–2330, August 2008.
7. Antonio Moschitta and Igor Neri. Power consumption assessment in wireless sensor networks. ICT-Energy-Concepts Towards Zero–Power Inform. and Comm. Techn. 2014.
8. Kolios, A.; Mytilinou, V.; Lozano-Minguez, E.; Salonitis, K. A Comparative Study of Multiple-Criteria Decision-Making Methods under Stochastic Inputs. Energies 2016, 9, 566.
9. Sabaei, D., Erkoyuncu, J., Roy, R.: A Review of Multi-criteria Decision Making Methods for Enhanced Maintenance Delivery, Procedia CIRP, vol 37, 2015, Pages 30-35, ISSN 2212-8271, http://dx.doi.org/10.1016/j.procir.2015.08.086.
10. Aruldoss, M. , Lakshmi, T. M. , & Venkatesan, V. P. A Survey on Multi Criteria Decision Making Methods and Its Applications. American J. of Information Syst., 1(1), 31-43 2013.
11. T. L. Saaty, "A scaling method for priorities in hierarchical structures," J. Math. Psychol., vol. 15, no. 3, pp. 234–281. Jun, 1977.
12. Aldlaigan, A.,Buttle, F.A. "A new measure of bank service quality". International Journal of Service Industry Management, Vol. 13, pp. 38-362. 2002.
13. José Figueira, Salvatore Greco, Matthias Ehrogott. Multiple Criteria Decision Analysis: State of the Art Surveys, volume 78. Springer-Verlag New York, international series in operations research & management science edition, 2005.
14. Brans, J.; Vincke, P. A preference ranking organization method: The PROMETHEE method for multiple criteria decision-making. Manag. Sci. 31, 647–656. 1985.
15. Kristo Mela, Teemu Tiainen, Markku Heinisuo, Comparative study of multiple criteria decision making methods for building design, Advanced Engineering Informatics, vol. 26, Issue 4, pp. 716-726, October 2012.

16. Mateo, J.R.S.C. Multi-Criteria Analysis in the Renewable Energy Industry; Springer-Verlag: London, UK, 2012.
17. Roy, B. The outranking approach and the foundations of electre methods in Theory and Decision vol. pp.31-49. 1991.