Gaining confidence on dependability benchmarks' conclusions through "back-to-back" testing

Miquel Martínez, David de Andrés and Juan-Carlos Ruiz
Instituto de Aplicaciones de las TIC Avanzadas (ITACA)
Universitat Politècnica de València, Campus de Vera s/n, 46022, Spain
Email: {mimarra2, ddandres, jcruizg}@disca.upv.es

Submission Type: Practical Experience Report
Keywords: dependability benchmarking, result analysis, back-to-back testing

***Abstract*—The main goal of any benchmark is to guide decisions through system ranking, but surprisingly little research has been focused so far on providing means to gain confidence on the analysis carried out with benchmark results. The inclusion of a back-to-back testing approach in the benchmark analysis process to compare conclusions and gain confidence on the final adopted choices seems convenient to cope with this challenge. The proposal is to look for the coherence of rankings issued from the application of independent multiple-criteria decision making (MCDM) techniques on results. Although any MCDM method can be potentially used, this paper reports our experience using the Logic Score of Preferences (LSP) and the Analytic Hierarchy Process (AHP). Discrepancies in provided rankings invalidate conclusions and must be tracked to discover incoherences and correct the related analysis errors. Once rankings are coherent, the underlying analysis also does, thus increasing our confidence on supplied conclusions.**

I. Introduction

Since the seminal research carried out during the DBench European project more than 10 years ago [1], lots of efforts have been done in dependability benchmarking resulting in the current availability of a wide variety of dependability, security and resilience benchmarks. The similarities among existing proposals is not surprising, since most of them rely on the DBench experimental framework, which is adapted and extended in each proposal attending to the variety of constraints imposed by each particular system, application domain and/or context of use.

Despite the interest for comparing different component and system implementations, configurations and parametrisations, dependability benchmarking has attained so far a limited industrial adoption. Discussing the root causes of this situation falls beyond the scope of this paper but what seems quite clear is that, in some cases, the requirements imposed to dependability benchmarks by the academia are different from those expected by the industry. Approaches, like the SPEC Research IDS Benchmarking Working Group [2], aims at mitigating that problem by fostering innovative research through exchange of ideas and experiences between academia and industry, although there exists a long way to go.

The vision of industrials of what is a dependability benchmark is usually quite pragmatic; they consider such type of benchmarks as tools to support, or automate to some extent, the process of selecting the most suitable components for the particular type of systems they produce. As a result, they ask for the provision of a limited number of results (if possible one) in order to accelerate and simplify the final selection/decision process underlying any benchmarking effort. On the other hand, researchers prefer to provide the so-called *necessary* (sometimes large) number of measures to establish a precise and well-reasoned ranking among all benchmarked targets. It must be noted that this approach is not a problem by itself. The problem is that different rankings and conclusions can be issued from the analysis of the very same set of benchmarking measures. One of the aspects leading to that situation is the lack of any explicit representation of the analysis procedure followed to issue conclusions, which limits in practice the repeatability of such procedure. This situation should not be a surprise for the reader since analysing benchmarks results refers to a well-known and subjective multi-attribute analysis process [3]. As a result, and despite the pertinence and correctness of conclusions, the analysis performed must be always studied attending to the particular subjective (judgmental) analysis criteria used by the decision maker.

The use of multiple-criteria decision-making (MCDM) techniques provides means to explicitly represent the analysis process followed when interpreting (benchmarking) results under the form of a multi-attribute decision model, also called quality model. Multicriteria decision problems may have different goals that are very close to those pursuit when analysing dependability benchmarks results [3]: i) to eliminate a number of worst alternatives, or ii) to choose a number of best alternatives, or ii) to rank the alternatives. In the problem of elimination or choice, the order between the eliminated or chosen alternatives could be also important. In this case we have a mixed problem of iv) choice and ranking. It must be noted that the consideration of MCDM techniques in the definition of dependability benchmarks is not something new [4] [5]. However, existing proposals limit their purpose to the use of MCDM techniques to make explicit the quality model followed to analyse benchmarking measures. This eliminates uncertainties in the process followed to analyse measures, thus improving its repeatability.

This paper makes an step forward in that direction and exploits the differences existing among various MCDM techniques in order to diversify the analysis process and gain confidence in conclusions. It must be underlined that the approach is not useful for checking the correctness of the analysis

process itself. The proposal limits its scope to the comparison of the conclusions issued from applying two different MCDM techniques, attending to the same analysis criteria, despite its correctness, to an existing set of benchmarking measures. By checking the existence of discrepancies in the conclusions, one can detect misuses of MCDM techniques, thus being able to fix existing interpretation errors. Once conclusions issued from the application of MCDM techniques are coherent, one can gain confidence on the consistency of reported conclusions, even if such conclusions are not correct because of a problem in the interpretation of input requirements.

This paper is structured as follows. First, section II introduces the case study that will illustrate the proposal all through the rest of the paper. It will also exemplify the application of an MCDM technique named Logic Score of Preferences (LSP) to the considered case study. Then, section III provides a high level view of the approach, details how to apply an alternative MCDM technique, called Analytic Hierarchy Process (AHP) to the same case study, and describes the process followed to detect inconsistencies between rankings promoted by LSP and AHP techniques. Finally, section IV shows the usefulness of the approach, section V discusses benefits and drawbacks of the proposal, and section VI closes the paper.

II. Case study

Wireless Mesh Networks (WMNs) are a particular type of ad hoc networks which is currently being used, among other things, to provide cheaper and more flexible access to Internet than their wired counterparts to isolated or remote areas. As these networks may be deployed in very different scenarios, they may be subjected to a wide range of perturbations (both accidental faults and malicious attacks). Accordingly, and taking into account that a single perturbation has been considered as the most important for each scenario, the aim of this case study is to determine in which of the five proposed scenarios it could be more interesting to deploy that network. Results will be analysed by means of a multiple-criteria decision-making (MCDM) method, the Logic Score of Preferences (LSP), to score and rank the different considered scenarios.

A. Experimental set up and results

The considered WMN consists of 16 static nodes deployed as shown in Fig. 1. REFRAHN, the *Resilience Evaluation FRamework for Ad Hoc Networks* supporting this experimentation, makes use of real devices as network nodes, but emulates their visibility by packets filtering. So, the experimental platform for this case study comprises 10 Linksys WRT54GL routers (200 MHz MIPS processor, 16 MB of RAM, IEEE 802.11b/g Broadcom BCM5352 antenna) running a WRT distribution (White Russian), and 6 HP 530 laptops (1.46 GHz Intel Celeron M410 processor, 512 MB of RAM, internal IEEE 802.11b/g Broadcom WMIB184G wireless card, 4 Li-Ion cells battery (2000 mAh)) running an Ubuntu 7.10 distribution.

Communications are managed by *olsrd* (www.olsr.org), the most extended implementation of the popular Optimized Link State Routing (OLSR) protocol, in its version 0.4.10. The applicative traffic addressed to exercise the network is defined in terms of synthetic UDP Constant Bit Rate (CBR) data flows

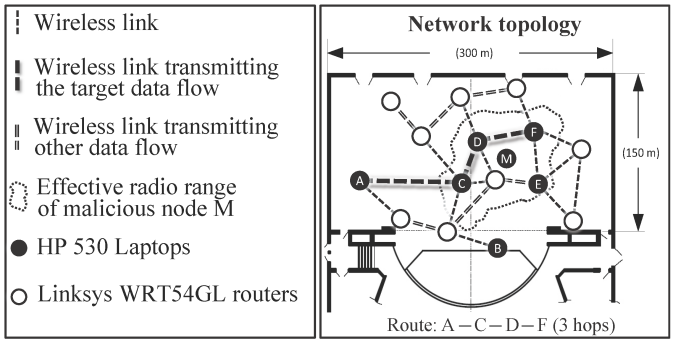

Fig. 1. Wireless mesh network topology

of 200 Kbps, similar to those observed in daily scenarios [6]. The workload then consists in three of these data flows being exchanged among network nodes.

The perturbations considered in this study (fault- and attack-load) is a subset of the most harmful faults in the domain of WMNs [7]. These perturbations define the five different scenarios considered for this case study: one in which accidental faults, like ambient noise (A), are the most predominant, and the rest where the routing protocol faces various malicious attacks, such as selective forward (S), jellyfish (J), tampering (T), and flooding (F) attacks. Only one of this perturbations will be injected in each of the five considered scenarios, so it could be possible to determine the impact of such particular perturbation on the network. The target data flow for all the considered perturbation is the 3-hop communication between nodes A and F in Fig. 1. Whenever a perturbation requires the participation of a malicious node to perpetrate the attack, node M in Fig. 1 will play that role.

The set of measures that will be used to characterise the behaviour of the network in presence of perturbations consists of 5 different measures: i) the average amount of traffic effectively received during experimentation (*throughput*), ii) the average packets delay in milliseconds (*delay*), iii) the percentage of the time the routes are available for inter nodes communication (*availability*), iv) the percentage of packets whose data remain unaltered (*integrity*), and v) the average energy consumed by nodes (*energy*).

For each of the considered scenarios, including a perturbation free one, a total of 15 experiments were executed with a duration of 9 minutes each. The average results obtained from all the experiments performed in each scenario are presented in Table I.

As can be seen, the interpretation of the whole set of results listed in Table I is not straightforward, and multiple-criteria decision-making (MCDM) methods are really helpful to guide the comparison among different scenarios [8]. Our prior research focused on integrating one of these methods,

TABLE I. Experimental results for each scenario

Scenario	Throughput (Kbps)	Delay (ms)	Availability) (%)	Integrity (%)	Energy (J)
(A)mbient noise	145.2	48.2	73.6	92.12	8.2
(S)elective forwarding	121	42	91.2	97.53	8
(J)ellyfish	184.8	1086.5	88.7	98.54	10.3
(T)ampering	183.6	39.7	93.1	5.2	10.6
(F)looding	149	62.9	72.1	97.56	15.4

the Logic Score of Preferences (LSP) in particular, into the common dependability benchmarking flow [9] [4].

B. Multiple-criteria decision-making: LSP as an example

LSP aims at characterising each system through a single 0-to-100 score which could be used to easily compare and rank eligible alternatives. The final score of the system is obtained by the successive aggregation of intermediate scores according to a defined *criteria tree hierarchy*. Each aggregation takes into account the particular contribution (*weight*) of each subcriterion to the upper level criterion and the intensity of their relation (*operator*). The scores for the base level criteria are obtained by normalising the obtained results according to a given minimum and maximum values (*thresholds*). All these elements, hierarchy tree, weights, operators, and thresholds, constitute the so called *quality model*. The scores for the rest of upper level criteria are computed by using the generalised power mean (see Equation 1).

$$score = \sum_{i=1}^{\text{number of subcriteria}} (\text{weight}_i \text{score}_i^{\text{operator}})^{\frac{1}{\text{operator}}} / \sum_{i=1}^{\text{number of subcriteria}} \text{weight}_i = 1 \quad (1)$$

The quality model for a given system should be specified prior to experimentation, so its definition is not influenced by experimental results. The constituent elements of the quality model should faithfully reflect the requirements the system must meet. This is the available information for the considered case study: *"The main concern of the deployed WMN focuses on the dependability of supported communications, as sensitive information that should not be altered will be exchanged among network nodes. Thus, preserving the integrity of exchanged packets is of primary importance, whereas the availability of the routes although still of interest is a secondary matter. The network performance also contributes to provide a good quality service, but is not as much important as its dependability. Increasing the network throughput is the main priority to increase the network performance, whereas the delay of the packets is not so important as long as they finally reach their destination. As the nodes of the network will be continuously powered, reducing their energy consumption can be considered as a nice bonus, but not a strong requirement."* Taking all this into account, the quality model reflecting our criteria and optimisation goals for the considered WMN is depicted in Fig. 2. It must be noted that, although not included for space constraints, the thresholds used to normalise the measures, and the required normalisation functions, should also be extracted from the requirements, existing literature, or practical experience.

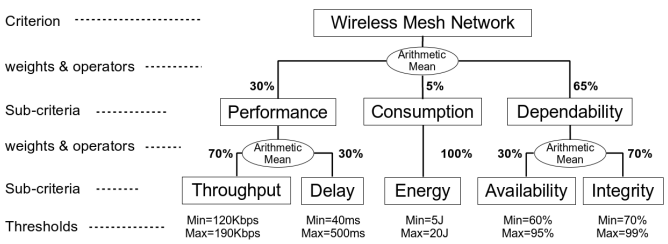

Fig. 2. LSP quality model for the considered case study

After applying the proposed quality model to the results listed in Table I, the scores obtained for each of these scenarios lead to the following ranking (from best to worst): J (83.44), S (73.83), T (68.76), A (62.61), F (49.65). Accordingly, the considered WMN is best suited to be deployed in scenarios facing jellyfish and selective forwarding attacks, whereas it is not really usable in scenarios facing flooding attacks.

C. Limitations of the approach

As shown, MCDM methods, like LSP used in this case study, are powerful tools to ease the comparison among different alternatives to select that optimising the defined criteria. However, some values of the multiattribute decision models, like the weights, operators, and thresholds in the quality model presented in Fig. 2, are often subjective (judgemental). Lack of precision and accuracy when specifying the requirements of the system (criteria and goals), like the vague natural language description presented in this case study, the misinterpretation of these requirements, or their mapping into quality model attributes, are very important sources of uncertainty. As final rankings provided by MCDM methods are sensitive to changes in their input parameters [10], those uncertainties may lead to very different decisions.

Accordingly, the question of which is the level of confidence that can be placed on the ranking provided by MCDM methods when applied to dependability benchmarks arises. Any variation in the quality model attributes, either due to misinterpretations or vague specifications, may result in wrong decisions which may greatly comprise the dependability of target systems. Hence, the provision of mechanisms to detect and even diagnose any potential inconsistency in the ranking obtained via MCDM methods is indispensable to increase our confidence on provided conclusions.

III. Proposal

The main problem once conclusions are provided by the selected MCDM method is that there is no way of determining whether they are right, or at least they seem coherent, taking into account the existing sources of uncertainty in the definition of the required quality model. However, in this context, techniques like *back-to-back testing* may prove useful to detect and possibly diagnose potential flaws in the conclusions obtained.

Back-to-back testing involves cross-comparison of all responses obtained from functionally equivalent components [11]. If any of the comparisons signals a difference the problem is further investigated and, if necessary, a correction is applied. Translating this approach into the considered dependability benchmarking context involves i) applying different MCDM methods to analyse the results issued from experimentation, and ii) comparing the provided rankings to detect existing inconsistencies. If those rankings are coherent, although their correctness cannot be completely guaranteed, the confidence that can be placed on them highly increases. In concrete, this case study promotes the use of a MCDM method called *Analytic Hierarchy Process* (AHP) [12], in parallel with LSP, to achieve this goal.

Next sections describe in detail the AHP technique and the process defined to find out any meaningful dissimilarities between LSP and AHP conclusions.

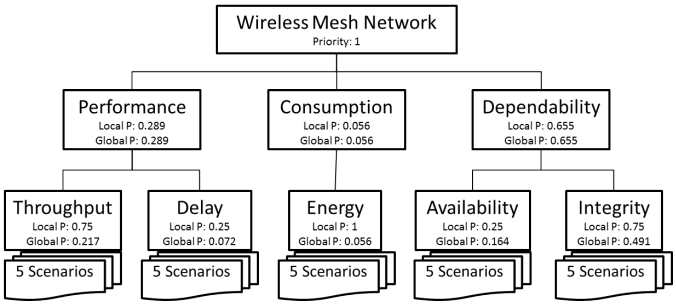

Fig. 3. AHP quality model for the considered case study

A. AHP as an alternative MCDM

AHP [12] is a MCDM method that, instead of a final score, provides a *priority* for each considered alternative reflecting its contribution to the goals optimisation. It shares procedural similarities with LSP, as it also makes use of a hierarchical quality model to aggregate subcriteria into higher level criteria. Accordingly, the very same criteria tree hierarchy may be used for both LSP (see Fig. 2) and AHP (see Fig. 3) techniques, although the parameters used to characterise the model are not exactly the same and are determined in a different way. This similarity will later ease the comparison between final rankings.

As Fig. 3 depicts, the AHP quality model just considers the contribution of each subcriterion to the upper level criterion through their relative *priorities*. These priorities are obtained by means of the pairwise comparison of all the subcriteria contributing to a given criterion. Those comparisons are assigned a number (*intensity*) stating how many times more important or dominant one criterion is over another regarding the criterion with respect to which they are compared. Table II [13] lists the different values (from 1 to 9) denoting the intensity of the importance of criterion A with respect to criterion B.

The pairwise comparison of all the criteria contributing to a given criterion is represented in a matrix form, in such a way that if the intensity of criterion A with respect to criterion B is X, then the intensity of criterion B with respect to criterion A is $1/X$. Table III shows the resultant matrix for the pairwise comparison of *Performance*, *Consumption*, and *Dependability*, with respect to the *Wireless Mesh Network* according to the requirements expressed in Section II-B. In this case, *Dependability* is considered more important than *Performance*, and absolutely more important than *Consumption*, whereas *Performance* is considered just much more important than *Consumption*. Resulting priorities can be derived from the principal right eigenvector of the matrix. However, a fair estimation can be obtained through a more straightforward procedure that will be used in this case study: i) compute the geometric mean for each row of the matrix, ii) sum up the geometric mean obtained for each row, and iii) divide each geometric mean by the total sum. After applying this procedure, shown in Table III, the contribution of each criterion to the final goal is of 0.29 for *Performance*, 0.655 for *Dependability*, and 0.055 for *Consumption*.

TABLE II. THE FUNDAMENTAL SCALE OF ABSOLUTE NUMBERS FOR PAIRWISE COMPARISON

Definition	Description	Intensity[a]
Equal	A and B are equally important	1
Moderate	A is somewhat more important than B	3
Strong	A is much more important than B	5
Very strong	A is very much more important than B	7
Extreme	A is absolutely more important than B	9

[a] Intensities of 2, 4, 6 and 8 can be used to express intermediate values. Very close importance values can be represented with 1.1–1.9.

TABLE III. PAIRWISE COMPARISON MATRIX OF THE MAIN CRITERIA WITH RESPECT TO THE GOAL

Wireless Mesh Network					
	Performance	*Dependability*	*Consumption*	Row's GeoMean	Priority
Performance	1	1/3	7	1.326	0.29
Dependability	3	1	9	3	0.655
Consumption	1/7	1/9	1	0.251	0.055
			SUM	4.577	

This procedure is recursively applied to compute the priorities for subcriteria with respect to the upper level criterion. Table IV lists the resulting matrices for *Performance* and *Dependability*.

Finally, the pairwise comparison is performed among the different alternatives to determine their contribution to the base level criteria defined in the tree hierarchy. Table V lists the resulting matrices for *Throughput*, *Delay*, *Energy*, *Availability*, and *Integrity*.

When defining these matrices, it is important keeping the consistency of the pairwise comparisons. For example, if the intensity of criterion A with respect to criterion B is 3, and the intensity of criterion B with respect to criterion C is 3, then to keep the consistency, the intensity of criterion A with respect to criterion C should be more than 3. The consistency of the pairwise comparison matrices is computed by the so called *Consistency Index* (CI) [12] and, although it will not be described here due to space constraints, all the matrices defined in this case study proved to be consistent.

Priorities must be understood at two levels: *local* priorities to their upper criterion, directly obtained from the defined matrices, and *global* priorities with respect to the goal, computed as the local priority multiplied by the global priority of its upper level criterion. For example, *Throughput* and *Delay* have a local priority of 0.75 and 0.25 respectively, computed from the matrix defined in Table IV. This is their priority with respect to *Throughput*. However, their global priority with respect to the final goal is $0.75 \times 0.289 = 0.217$ and $0.25 \times 0.289 = 0.072$, respectively. Fig. 3 depicts all the local and global priorities for the defined criteria.

This very same procedure is then applied for the priorities obtained for each alternative with respect to the base level criteria. For instance, Scenario A has a local priority of 0.120 with respect to *Throughput* and a global priority of $0.120 \times 0.217 = 0.026$ with respect to the global goal

TABLE IV. PAIRWISE COMPARISON MATRICES FOR THE SUBCRITERIA WITH RESPECT TO PERFORMANCE AND DEPENDABILITY

Performance		
	Throughput	*Delay*
Throughput	1	3
Delay	1/3	1

Dependability		
	Availability	*Integrity*
Availability	1	1/3
Integrity	3	1

TABLE V. PAIRWISE COMPARISON MATRICES FOR ALTERNATIVES WITH RESPECT TO THE BASE LEVEL CRITERIA

Throughput

	A	S	J	T	F
A	1	2	1/3	1/3	1
S	1/2	1	1/5	1/5	1/2
J	3	5	1	1	3
T	3	5	1	1	3
F	1	2	1/3	1/3	1

Delay

	A	S	J	T	F
A	1	1	9	1	3/2
S	1	1	9	1	3/2
J	1/9	1/9	1	1/9	1/9
T	1	1	9	1	3/2
F	2/3	2/3	9	2/3	1

Availability

	A	S	J	T	F
A	1	1/5	1/5	1/5	1
S	5	1	1	1	5
J	5	1	1	1	5
T	5	1	1	1	5
F	1	1/5	1/5	1/5	1

Integrity

	A	S	J	T	F
A	1	2/3	2/3	9	2/3
S	3/2	1	1	9	1
J	3/2	1	1	9	1
T	1/9	1/9	1/9	1	1/9
F	3/2	1	1	9	1

Energy

	A	S	J	T	F
A	1	1	2	2	4
S	1	1	2	2	4
J	1/2	1/2	1	1	2
T	1/2	1/2	1	1	2
F	1/4	1/4	1/2	1/2	1

according to its contribution to *Throughput*. Resulting priorities for each alternative are then added up to obtained their final priority. These priorities are then used to rank the alternatives according to their contribution to the optimisation of the goal. In this case study, the final ranking from best to worst is: J (0.2626), S (0.2252), F (0.1861), A (0.1632), and T (0.1629). So, the target WMN is best suited to be deployed in scenarios facing jellyfish and selective forwarding attacks, whereas it should not be considered for scenarios suffering ambient noise or tampering attacks.

B. Detecting inconsistencies in provided rankings

The use of two different MCDM methods enables the comparison of the provided rankings to increase de confidence that can be placed on the provided conclusions. Basically, the rankings obtained by applying the LSP and AHP quality models to the results of the dependability benchmark are compared to check whether they are coherent or not. As both techniques follow a different procedure to compute final rankings any misinterpretation of the requirements, procedural errors, or simple transcription mistakes may probably reflect on the provided output (ranking). But, as both techniques are based on the same criteria hierarchy tree, this enables the possibility of tracking inconsistencies down the tree to look for their origin. So not only potential problems may be detected but, in some cases, also diagnosed. The flow diagram representing the procedure to be followed for the back-to-back testing of LSP and AHP rankings is depicted in Fig. 4.

The very first step consists in comparing the rankings for the root of the criteria hierarchy tree (goal). In case that no meaningful inconsistencies are found, then the process ends and the rankings are considered coherent. This is what happens

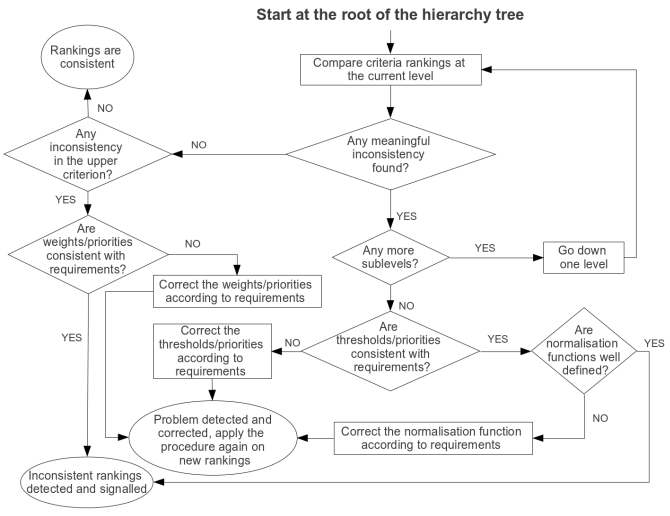

Fig. 4. Flow diagram for back-to-back testing LSP and AHP rankings

in this case study, as alternative scenarios are sorted as J-S-F-A-T from best to worst by both techniques. It must be noted that very small inconsistencies may appear due to the different nature of the considered MCDM methods. For instance, two alternatives may present very close scores/priorities but take reversed positions in both rankings. This probably does not invalidate the provided rankings, but points out that these alternatives are really so close to optimise the goal that they could be considered as interchangeable. In case that more meaningful inconsistencies are found it is necessary to go down the hierarchy tree to look for their origin.

The rankings for the next level subcriteria are also check for inconsistencies. If no meaningful inconsistencies are found this means that, probably, the problem is related to the weights (LSP)/priorities (AHP) computed for the upper level criterion, which should be checked against the requirements. Otherwise, it is necessary to go further down the hierarchy tree in a recursive way.

Finally, in case that the lowest level criteria are reached, and no discrepancies are found, this probably means that thresholds (LSP)/weights (AHP) are not correctly defined at this level, which should be checked against the requirements. If this check is inconclusive, then the problem is likely related to the function used to normalise the measures (LSP).

If all these checks are fruitless, then it is not possible to diagnose the origin of the inconsistency to correct it but, al least, a potential problem in the provided conclusions is detected and signalled. Likewise, it is not possible to ensure the correctness of the provided rankings but the confidence that can be placed on its correctness is largely increased.

IV. VALIDATION OF THE PROPOSED APPROACH

The proposed back-to-back testing approach to check the consistency of rankings computed from dependability benchmarks results offers a promising procedure to increase the confidence that can be placed on such conclusions. However, it is necessary to determine whether that procedure is robust enough to detect and even diagnose inconsistencies in those rankings derived from different sources of uncertainty when determining the parameters of the defined quality models.

To show the feasibility of this approach, three different LSP quality models have been proposed (M1, M2, M3), in addition to the two original LSP (M0) and AHP models previously defined in this case study. The first new LSP quality model (M1) includes a misinterpretation (or different interpretation) of the requirements specified in Section II-B in a vague natural language, in such a way that the contribution of *Performance*, *Dependability*, and *Consumption* to the goal is now 0.3, 0.5, and 0.2, respectively. The second quality model (M2) presents a simple transcription error, as the contribution of *Availability* and *Integrity* to *Dependability* has been reversed (0.7 and 0.3, respectively). Finally, another source of uncertainty, related to

TABLE VI. SCORES/PRIORITIES OBTAINED FOR ALL CRITERIA AFTER APPLYING THE DEFINED LSP/AHP QUALITY MODELS. SCORING DIFFERENCES WITH RESPECT TO THE ORIGINAL LSP MODEL (M0) ARE HIGHLIGHTED IN LIGHT GREY.

Scenario/Subcriterion		Measure	LSP score		AHP priority	Subcriterion	LSP score			AHP priority	LSP goal score				AHP goal priority
			M0, M1 & M2	M3			M0 & M1	M2	M3		M0	M1	M2	M3	
A	Throughput	145.2	36	36	0.0261	Performance	54.66	54.66	55.2	0.1509	62.61	64.66	52.89	62.77	0.1632
	Delay	48.2	98.22	100	0.0176										
	Availability	73.6	38.857	38.857	0.0096	Dependability	65.05	50.08	65.05	0.1565					
	Integrity	92.12	76.28	76.28	0.0929										
	Energy	8.2	78.67	78.67	0.0169	Consumption	78.67	78.67	78.67	0.0169					
S	Throughput	121	1.428	1.428	0.0140	Performance	30.87	30.87	31	0.1093	73.83	71.86	72.33	73.87	0.2252
	Delay	42	99.56	100	0.0176										
	Availability	91.2	89.143	89.143	0.0482	Dependability	93.194	90.879	93.194	0.2696					
	Integrity	97.53	94.93	94.93	0.1285										
	Energy	8	80	80	0.0169	Consumption	80	80	80	0.0169					
J	Throughput	184.8	92.571	92.571	0.0755	Performance	64.8	64.8	64.8	0.2674	83.44	79.12	79.17	83.44	0.2626
	Delay	1086.5	0	0	0.0020										
	Availability	88.7	82	82	0.0482	Dependability	93.489	86.92	93.489	0.2696					
	Integrity	98.54	98.413	98.413	0.1285										
	Energy	10.3	64.67	64.67	0.0084	Consumption	64.67	64.67	64.67	0.0084					
T	Throughput	183.6	90.571	90.571	0.0755	Performance	93.6	93.6	93.6	0.3215	49.65	54.8	74.24	49.65	0.1629
	Delay	39.7	100	100	0.0176										
	Availability	93.1	94.571	94.571	0.0482	Dependability	28.371	66.2	28.371	0.0936					
	Integrity	5.2	0	0	0.0132										
	Energy	10.6	62.67	62.67	0.0084	Consumption	62.67	62.67	62.67	0.0084					
F	Throughput	149	41.428	41.428	0.0261	Performance	57.51	56.51	51.26	0.1509	68.76	61.83	53.05	66.89	0.1861
	Delay	62.9	95.021	74.2	0.0176										
	Availability	72.1	34.571	34.571	0.0096	Dependability	76.895	52.710	76.895	0.2108					
	Integrity	97.56	95.03	95.03	0.1285										
	Energy	15.4	30.67	30.67	0.0042	Consumption	30.667	30.667	30.667	0.0042					

the definition of the thresholds used to normalise the obtained measures is considered in the third quality model (M3). In this case, the thresholds for the *Delay* have been tighten in excess ([50, 100] instead of [40, 500]). The different scores and priorities obtained by means of all these quality models are listed in Table VI. According to these figures, Table VII lists the final ranking provided by these quality models for the different considered criteria.

As the rankings for AHP and M0 have been already proved to be consistent in Section III-B, let us move to comparing rankings for AHP and M1. As Table VII shows Scenarios F and A swap positions in the provided rankings, thus pointing out a potential inconsistency in the defined quality models. Following the proposed diagram flow (see Fig. 4), the rankings for the criteria at the next level are also checked. In this case no further discrepancies are found, so the problem should be related to the weights/priorities (pairwise comparison matrices) defined for the highest level of the hierarchy. Whether the parametrisation of one or the other model, or neither of them, faithfully represents the requirements of the system is for the benchmark analyser to decide. Corrective actions at this level are required and new rankings should be compared again.

Great inconsistencies are also found when comparing rankings for AHP and M2, as the worst scenario for AHP is considered the second best for M2. As in the previous example, the rankings for the criteria at the next level are also examined to search for further discrepancies. In this case, the ranking for the *Dependability* criterion also presents inconsistencies. According to the proposed diagram flow, now it is time to check the next (lowest in this case) level of the hierarchy.

TABLE VII. BEST TO WORST RANKING OF CONSIDERED SCENARIOS. DIFFERENCES WITH RESPECT TO AHP RANKING ARE IN BOLDFACE

Quality model		Performance	Dependability	Consumption	WMN
AHP		T-J-F/A-S	J/S-F-A-T	S/A-J/T-F	J-S-F-A-T
LSP	M0	T-J-F-A-S	J-S-F-A-T	S-A-J-T-F	J-S-F-A-T
	M1	T-J-F-A-S	J-S-F-A-T	S-A-J-T-F	J-S-**A**-**F**-T
	M2	T-J-F-A-S	**S**-**J**-**T**-**F**-**A**	S-A-J-T-F	J-**T**-**S**-**F**-**A**
	M3	T-J-**A**-**F**-S	J-S-F-A-T	S-A-J-T-F	J-S-F-A-T

No discrepancies are found for *Availability* and *Integrity*, so the problem should be related to the weights/priorities assigned at the *Dependability* level. The requirements specified in Section II-B clearly state that *"preserving the integrity [...] is of primary importance, whereas the availability [...] is a secondary matter,"* so it is easy to determine that the weights for M2 are wrong. After correcting the error, new rankings must also be compared again.

Finally, when comparing rankings for AHP and M3, no inconsistencies can be found according to the proposed diagram flow (see Fig. 4). However, the rankings at *Performance* and *Delay* levels present some discrepancies, and the question of whether this approach is really sound arises. It must be noted that, as stated in Section II-C, MCDM methods are sensitive to input parameters. This means that variations in the input parameters may vary the final ranking. However, this also means that there exist different value ranges for these parameters that do not affect the provided ranking. This is clearly the case of the variation induced in the thresholds for *Delay*. The contribution of the *Delay* to the final goal is not so important, and the dispersion of the measures for each scenario is so small, that the inconsistency is just filtered or masked by the quality model. Obviously, this issue could also be signalled to benchmark analysers, but it seems fairly simpler to make it transparent to them as it really does not affect the final conclusion. The sensitivity of MCDM methods will be further discussed on Section V.

As the considered examples have shown, the proposed approach is able to properly track ranking inconsistencies down the criteria hierarchy tree to find the source of these discrepancies. Hence, back-to-back testing the final rankings provided by MCDM methods prove to be a feasible solution to guide the analysis of dependability benchmarking results and increase the confidence that can be placed on drawn conclusions.

V. Discussion

MCDM, as a subdiscipline of operational research, has been supporting decision-making processes for many years in very different application domains. Despite its long tradition, there still exists a recognised fundamental paradox in MCDM. Every single MCDM method claims to offer the best decision but, when different methods are taken into account, not all of them select the same alternative [14]. Accordingly, determining which is the most suitable MCDM method to analyse dependability benchmarking results in a given context could also required the use of another MCDM method, leading to another paradox. One possible option is considering *rank reversals*.

Rank reversals [15] are a particular problem of some MCDM methods which, when subjected to small variations in their inputs or quality model parameters, may produce contradictory rankings. Special tests are usually defined to detect whether this problem affects the solution provided by a given MCDM method, and alternative methods should be then considered. For example, let us assume that two different routing protocols are being benchmark to select the most suitable to be deployed in a given WMN. Protocol A exhibits more *Throughput* than protocol B, but its overall quality is lower. So, a decision maker could sacrifice the network quality if he considers that is utterly important to obtain the highest possible *Throughput*. However, if a third protocol C is benchmarked, which presents much lower *Throughput* than B but with a very similar overall quality, then the perception of the decision maker may be biased and see protocol B as a more attractive option. As can be seen from the example, rank reversals may also be caused by rational decisions, so they are not always indicative of a faulty decision-making process. Distinguishing whether rank reversals are due to one or the other cause is still a hot topic in the operational research community.

That is why, the back-to-back testing of different MCDM methods to detect any potential inconsistencies in the conclusions provided appears as a sensible option to increase our confidence on final rankings. It is not necessary to determine which is the best MCDM method in absolute terms, but just that provided rankings are consistent with the requirements used to interpret dependability benchmark results. In this proposal, LSP and AHP methods have been selected, as they both can share the same criteria hierarchy tree for their respective quality models. This feature enables the navigation through the different levels of the hierarchy to diagnose the possible source of detected inconsistencies. Obviously, not all MCDM share this feature, but some of them are likely to share other features that could make them compatible to be also used for back-to-back testing. Classifying existing MCDM methods according to shared features, thus enabling the application of different sets of MCDM methods according to, for instance, target application domains, number of considered criteria, or sensitivity to obtained experimental results, is a very interesting topic for further research [16].

As previously mentioned, MCDM methods present different degrees of sensitivity to variations in incoming data or quality model parameters [17]. Those methods with a high sensitivity are more likely to exhibit rank reversal behaviours due to intrinsic sources of uncertainty when defining the quality model and, thus, should not be considered for back-to-back testing when more reliable methods are available. Not so sensitive methods are of great interest, as the uncertainty induced in the quality model (like thresholds, weights, and operators), can be reduced or even masked by the model itself. Accordingly, by estimating the sensitivity of proposed quality models in advance it could be possible to determine and select the least sensitive models, or which parameters should be carefully tuned so as to prevent later inconsistencies. This is also a hot topic requiring further research.

VI. Conclusions

This practical experience report proposes the exploitation of the diversity existing between different multiple-criteria decision making (MCDM) techniques in order to gain confidence on conclusions issued from the analysis of benchmark measures.

The proper comparison of different techniques' results is quite challenging since the diversity existing in the techniques is translated to their related quality models. A concrete approach is proposed to compare the quality model defined by the Logic Score of Preferences (LSP) technique with the one induced, applying the same criteria hierarchy, by the Analytic Hierarchy Process (AHP) method. When rankings issued from both techniques are incoherent, one can detect potential sources of errors in the analysis process and, sometimes, fix them. When they are coherent, one gains confidence on the consistency of drawned conclusions. However, it must be underlined that, despite the coherence of the rankings provided by considered MCDM techniques, conclusions may be incorrect in cases, like where the functional or non-functional requirements of the target systems have been incorrectly captured.

We cannot currently state that this approach applies regardless the couple of MCDM techniques selected for analysis, since their related quality models may exhibit different levels of sensitivity to parameters and input data, which can result in rank reversals. Classifying existing MCDM methods according to their features in order to enable their combined or complementary use attending to aspects relating to the considered application domain, number of criteria, or sensitivity to existing benchmarking measures, remains today an open topic requiring further research. Since the final goal of any benchmark is to drive decisions based on scores and rankings, the final goal of this research is to integrate the use of decision making techniques in the analysis process of dependability benchmarks, something that is today neglected and left in the hands of decision makers acting as benchmark users.

Acknowledgment

Work partially supported by the Spanish project ARENES (TIN2012-38308-C02-01).

References

[1] K. Kanoun and L. Spainhower, Eds., *Dependability Benchmarking for Computer Systems*. Wiley and IEEE Computer Society Press, 2008.

[2] "The SPEC Research IDS Benchmarking Working Group, officially called Benchmarking Architectures for Intrusion Detection in Virtualized Environments," [Online]. Available: http://research.spec.org/en/working-groups/ids-benchmarking-working-group.html, 2013.

[3] M. Koksalan, J. Wallenius, and S. Zionts, *Multiple Criteria Decision Making: From Early History to the 21st Century*. World Scientific Publishing Company; 1 edition (June 6, 2011), 2012.

[4] J. Friginal, D. de Andrés, J.-C. Ruiz, and P. Gil, "Coarse-grained resilience benchmarking using logic score of preferences: ad hoc networks as a case study," in *Proceedings of the 13th European Workshop on Dependable Computing*, ser. EWDC '11. New York, NY, USA: ACM, 2011, pp. 23–28.

[5] M. Martínez, D. de Andrés, J.-C. Ruiz, and J. Friginal, "Analysis of results in dependability benchmarking: Can we do better?" *M&N 2013, International Workshop on Measurements and Networking*, pp. 127–131, 2013.

[6] "Hillsdale WMN," Online: http://dashboard.openmesh.com/overview2.php?id=Hillsdale, 2012.

[7] I. F. Akyildiz and X. Wang, "A survey on wireless mesh networks," *IEEE Communications Magazine*, vol. 43, no. 9, pp. S23–S30, 2005.

[8] E. Triantaphyllou, *Multi-Criteria Decision Making Methods: A Comparative Study*. Springer, 2000.

[9] J. Friginal, D. de Andrés, J. C. Ruiz, and R. Moraes, "Using Dependability Benchmarks to Support ISO/IEC SQuaRE," in *IEEE 17th Pacific Rim International Symposium on Dependable Computing*, 2011, pp. 28–37.

[10] A. Ekárt and S. Z. Németh, "Stability analysis of tree structured decision functions," *European Journal of Operational Research*, vol. 160, pp. 676–695, 2005.

[11] M. A. Vouk, "Back-to-back testing," *Information and Software Technology*, vol. 32, no. 1, pp. 34–45, jan 1990.

[12] T. Saaty, "What is the analytic hierarchy process?" in *Mathematical Models for Decision Support*, ser. NATO ASI Series, G. Mitra, H. Greenberg, F. Lootsma, M. Rijkaert, and H. Zimmermann, Eds. Springer Berlin Heidelberg, 1988, vol. 48, pp. 109–121.

[13] ——, "Decision making with the analytic hierarchy process," *International Journal of Services Sciences*, vol. 1, no. 1, pp. 83–98, 2008.

[14] E. Triantaphyllou and S. H. Mann, "An examination of the effectiveness of multi-dimensional decision-making methods: a decision-making paradox," *Decision Support Systems*, vol. 5, no. 3, pp. 303–312, Sep. 1989.

[15] V. Belton and T. Gear, "On a short-coming of saaty's method of analytic hierarchies," *Omega*, vol. 11, no. 3, pp. 228–230, 1983.

[16] S. H. Zanakis, A. Solomon, N. Wishart, and S. Dublish, "Multi-attribute decision making: A simulation comparison of select methods," *European Journal of Operational Research*, vol. 107, no. 3, pp. 507–529, 1998.

[17] W. Wolters and B. Mareschal, "Novel types of sensitivity analysis for additive MCDM methods," *European Journal of Operational Research*, vol. 81, no. 2, pp. 281–290, 1995.