Analysis of results in Dependability Benchmarking: Can we do better?

Miquel Martínez, David de Andrés, Juan-Carlos Ruiz
STF-ITACA
Universitat Politècnica de València
Campus de Vera s/n, 46022, Spain
Email: {mimarra2, ddandres, jcruizg}@disca.upv.es

Jesús Friginal
LAAS-CNRS
7 avenue du Colonel Roche, F-31077 Toulouse, France
Email: jesus.friginal@laas.fr

***Abstract*—Dependability benchmarking has become through the years more and more important in the process of systems evaluation. The increasing need for making systems more dependable in presence of perturbations has contributed to this fact. Nevertheless, even though many studies have focused on different areas related to dependability benchmarking, and some others have focused on the need of providing these benchmarks with good quality measures, there is still a gap in the process of the analysis of results. This paper focuses on providing a first glance at different approaches that may help filling this gap by making explicit the criteria followed in the decision making process.**

I. Introduction

For many years the evaluation of a system's features made reference to the evaluation of those related to its performance. Nevertheless, the need for providing dependable systems in presence of perturbations, has lead to a current state of affairs in which many people from both academia and industry evaluate the dependability of systems, in addition to their performance, with comparison and selection purposes. This process, usually known as dependability benchmarking in the research community, has been tackled in many works in the literature where it is applied to different application domains, such as *web servers* [1], *on-line database transactional systems*[2], or *automotive systems*[3], among others.

Most of these works base their benchmarking process on the guidelines established in [4], so as to ensure portable, scalable, and non-intrusive procedures that may lead to repeatable and reproducible experiments. Other works, like [5], focus on dependability measurement to integrate into existing dependability benchmarking processes the common practice followed in metrology. But even though remarkable studies can be found on how to evaluate dependability features in many different systems [6], when it comes to analyse the results obtained in the experiments in order to provide meaningful conclusions, it can be found that evaluators base their conclusions in their own criteria. This presents a problem when different evaluators want to compare their results with the ones presented in another work. This fact has been pointed out in studies like [7] where, among other things, raw data from different experiments and evaluators can be shared, analysed and correlated to obtain good quality measures. However, the purpose of this paper is not focused on data sharing or obtaining quality measures from experimentation, but in pointing out a fact that is present in most dependability benchmarking related works performed so far, and that in our knowledge has not been properly considered yet, the **conclusions reproducibility**.

After analysing many works from the literature, like those presented in [6], it can be observed that the most commonly followed approach consists in presenting the raw measures (computed from the raw measurements/data obtained for each experiment) characterising different system's features, and drawing some conclusions from them. The process of how to compute measures from measurements is usually detailed in depth to show the correctness of such process and enabling other researchers to obtain the same measures. However, as mentioned before, conclusions are usually based on the evaluator's criteria (which is not a bad thing), but the process on how the measures are analysed to provide such conclusions is usually missing, making sometimes hard to understand how the evaluator has come up with them. It is known that in order to compare the results obtained from different experiments, all results must have been obtained following the same process, otherwise comparing them would not provide meaningful conclusions. Thus, a question raises: "*starting with the same results, can we consider useful two different conclusions obtained through different criteria?*" Well, this is not a yes/no answer, it depends. All conclusions extracted from results may be right according to a certain criteria, or wrong according to another, and here is where lies the importance of making explicit the considered criteria in the analysis process.

When reviewing dependability benchmark analyses where the criteria used to obtained the conclusions are missing, external evaluators may disagree with these conclusions, and thus state that the work is not correct. But if the criteria were explicitly defined, external evaluators could understand the reasoning behind those conclusions and thus argue about the analysis process, but not about the work done.

Section II shows a brief analysis of i) different possible profiles for evaluators, who are the consumers of those conclusions drawn from dependability benchmarking studies, and ii) the different techniques applied that lead to those conclusions. An example that illustrates the benefits of using decision support techniques and the lacks covered by them is presented in Section III, followed in Section IV by a discussion about the feasibility of introducing these methodologies into the common dependability benchmarking process. Finally, the main challenges to be faced are summarised in Section V.

II. Background

The number of measures obtained when evaluating a system is usually related to the difficulties found to present the results to end users. For that reason, many benchmarks provide a single score for each system. For instance, when observing the different set of benchmarks provided by the Embedded Microprocessor Benchmark Consortium (EEMBC) [8], all of them get a whole bunch of measures (16 in the case of EEMBC's AutoBench 1.1) but provide a single global measure (Automark for EEMBC's AutoBench 1.1) for a system by calculating a geometric mean with all the given measures. But, when providing a set of measures, it should be taken into account that there are different evaluator profiles that may need to consume these measures. For example, while people from academia may want as many individual measures as possible to exactly determine the effect of certain improvements or new configurations in a system, people from industry, in the other hand, could prefer a single global measure for a straight comparison among competing Commercial Off-The-Shelf (COTS) to be integrated into the system. Likewise, there will probably be some other users requiring more than a single measure, to be able to analyse a system according to different perspectives, but not tens of measures, which make the analysis really hard and often meaningless.

There are different approaches to represent and analyse the multiple measures obtained from evaluation. Although each approach has its own particularities, all of them have to face a common problem: *how to characterise decision criteria within a friendly and usable model*. Choosing a certain kind of representation for measures has important consequences in terms of expressiveness. Simplistic approaches may skew in excess the representation of the model, whereas representations with a high expressiveness can add unnecessary complexity to the model or can be cumbersome in its use for decision making. Therefore it is important to find an equilibrium between representing as much information as possible and maintaining a good degree of usability.

Measures aggregation is a common approach usually applied in the community of dependability benchmarking to ease the comparison among systems. However, it is surprising that so far there is still a lack of unified criteria when addressing the aggregation of measures and their subsequent analysis. Common methods applied by users for aggregation range from simple mathematical operations (e.g., addition, arithmetic mean or geometric mean) to more serious and systematic distribution fitting [9] and custom formulae [10] approaches.

Kiviat or radar diagrams [11] are graphical tools that represent the results of benchmarks in an easy-to-interpret footprint. They can show different measures using only one diagram and, although some training is required, the comparison of different diagrams is fairly simple. The scalability of Kiviat diagrams enables the representation of up to tens of measures. However, managing such a huge amount of information may difficult the interpretation and analysis of results. The problem previously stated is solved in [12] throughout the use of an analytical technique named the *figure of merit* which, imposing certain restrictions to the graph axes, synthesises all the measures into a unique numerical value associated to the footprint shape. However, the problem of this solution, as it happens with most techniques using the mean or the median, is that valuable information could be hidden behind a unique number, and consequently, the comparison between systems could result quite vague [13].

Generally, these techniques focus just on aggregating results and do not provide any insights on how to cope with the interpretation of issuing scores. Nevertheless, there are other techniques that can be used to aggregate the measures while making explicit the decision criteria followed. One of these techniques is the Logic Scoring of Preferences (LSP) [14], a method for combining a large number of criteria into one score. In order to achieve this, an aggregation tree has to be built, where the leaves of this tree are the raw measures. An elementary criterion is defined for each measure, where each criterion has a minimum and a maximum value that define the interval containing the accepted values for each specific measure. The values of the obtained measures are then normalized according to these (minimum and maximum) thresholds. All the measures are aggregated into higher-level features using operators and weights that determine the contribution of each low-level measure to the higher-level one. The final result is a global score that can be used to compare the evaluated system.

Yet another technique that makes explicit the decision criteria is the Analytic Hierarchy Process (AHP) [15]. This technique is widely used in other contexts as a decision making process. As happens in the LSP technique, measures are aggregated using a decision tree, where the leaves are the raw measures and the root is a global score for the system. All the measures are compared two-by-two to determine their contribution to the higher-level criterion. This contribution is obtained by computing the principal right eigenvector of the matrix containing the result of the two-by-two comparison. This process is recursively applied to all levels of the decision tree, thus ending with a global score (priority) for each system that allows their comparison.

As can be seen, exiting aggregation techniques can be classified in those just providing a single score, thus enabling a straightforward comparison of systems, and those based on a hierarchical aggregation of measures, usually in a tree-like form, which enables the navigation from raw measures to a single scores through different levels. Although simple aggregation approaches have been used along the years in the field of dependability benchmarking, it is surprising to note that more complex schemes have not been considered yet. Accordingly, it is necessary to study to what extent they could fulfill the requirements of benchmark evaluators and thus prove their suitability for this domain.

III. Proof of concept

In order to show the difference between making explicit or not the decision criteria when analysing dependability benchmark results, this case study makes use of the results obtained in [16], where authors evaluate the behaviour of an ad hoc network in presence of perturbations. For this example, a small subset of the measures obtained in the work are used to ease the understanding of the whole process. Nevertheless, studies applying these techniques to a large set of measures can be found in the literature like in [17].

The results in Table I represent the measures obtained from an ad hoc network in presence of one of the following

attacks: *Replay attack*, *Flooding attack* and *Tampering attack*. Due to the adaptation capabilities of ad networks, the system kept working in presence of the injected perturbations, but their impact could be observed on the system's performance and dependability degradation. The selected measures for the example are described next:

Availability
: Percentage of time the communication route established between sender and receiver is ready to be used.

Integrity
: Percentage of packets whose content has not been unexpectedly modified.

Throughput
: Average throughput of the network in kilobits per second.

TABLE I: Measures obtained from the study done in [16]

Measure	Replay attack	Flooding attack	Tampering attack
Availability (%)	75.20	65.00	90.33
Integrity (%)	99.44	98.23	62.90
Throughput (Kbps)	70.90	80.18	96.45

Obtained measures will be analysed by two different evaluators using two different techniques to aggregate the results, and determine which attack impacts the network the most. The first evaluator (**Ev1**) will aggregate the results obtained using a geometric mean (like it is done in the EEMBC), while the second evaluator (**Ev2**) will use the LSP technique described before.

The main purpose of Ev1 is to compare the system's behaviour in presence of perturbations in an easy way, so the geometric mean suits perfectly for this purpose. The scores obtained by Ev1 after aggregating the measures through Equation 1 are shown in Table II.

$$\sqrt[3]{Availability * Integrity * Throughput} \qquad (1)$$

Ev2 is using the LSP technique, which explicitly defines the reasoning behind the decision process through a mathematical model. The decision criteria followed by Ev2 to aggregate the measures is depicted in Figure 1 as an aggregation tree. *Availability* and *Integrity* measures are aggregated into a higher-level feature of the system called *Dependability*, and *Integrity* has been considered of more importance than *Availability* to determine the Dependability of the system. It is to note that this is just taken as an example of aggregation, and it does not mean that these two measures represent the dependability of the system as defined in [18]. Ev2 also considers that to determine a global score for the system, *Dependability* is slightly less important than *Performance*.

In the aggregation tree, *Min* and *Max* values represent the threshold values that define the interval of accepted values for each measure. The **M** inside a circle, represents the mean operator, but many different kind of operators can be used depending on the evaluator's requirements. A deeper analysis

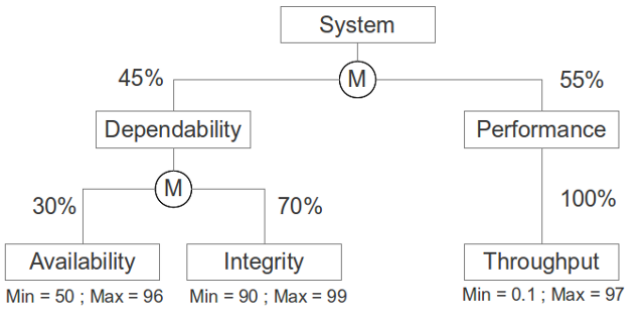

Fig. 1: Aggregation tree defined by the second evaluator (Ev2) to determine the system score

of these operators is performed in [17], where up to 20 different operators are defined. Table II shows the scores obtained by Ev2.

TABLE II: Scores obtained by the first (Ev1) and second evaluators (Ev2)

Evaluator	Replay attack	Flooding attack	Tampering attack
Ev1	80.9359	79.9971	81.8329
Ev2	80.9859	77.2679	55.5521

The different analyses performed by both evaluators lead them to different conclusions. From the results obtained by Ev1, the conclusion is that all the attacks have a similar impact on the system, with the "tampering attack" being slightly more benign, whereas the results obtained by Ev2 show that the "replay attack" has the lowest impact of the three attacks, being the "tampering attack" the most dangerous. As can be seen from this simple analysis, contradictory results can be obtained from the same set of results just because the interpretation process has not been accurately predefined. Nevertheless, this does not mean that one conclusion is right and the other is wrong. The purpose of the example was to prove that there are other methodologies that can be applied for the analysis of results that provide much more information about the criteria followed by the evaluator when presenting experiment's conclusions. Indeed, Ev2 can also take decisions based on intermediate results issued from the hierarchical aggregation of measures. For instance, the "replay attack" still has the lowest impact on the system from a *Dependability* viewpoint (according to the defined high-level feature). However, when just considering the *Performance* of the system (the other high-level feature), the "tampering attack" is the one providing the best scoring but, as previously described, is the worst case when considering the system as a whole.

It is easy to perceive that different aggregation techniques may lead to different conclusions, but there are other problems that may arise from the absence of information about the criteria used. For example, **Ev3** represents another evaluator willing to analyse the data shown in Table I using the LSP methodology. Ev3 presents the same aggregation tree that Ev2, and also the same thresholds for the different measures but, in

this case, Ev3 is considering *Dependability* far more important than *Performance*. Figure 2 depicts the aggregation tree with the weights established by Ev3, and Table III lists the scores obtained after measures aggregation.

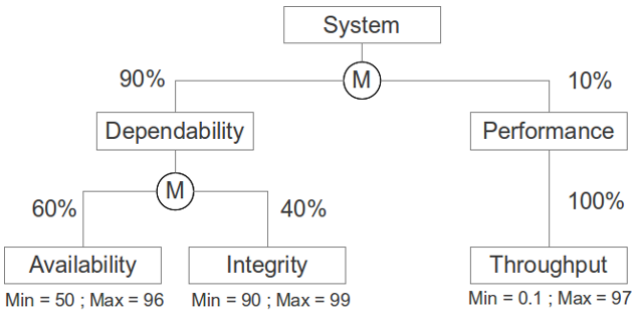

Fig. 2: Aggregation tree defined by the third evaluator (Ev3)

TABLE III: Scores obtained by the third evaluator (Ev3)

Evaluator	**Replay attack**	**Flooding attack**	**Tampering attack**
Ev3	72.8638	58.7766	57.2866

As can be appreciated from the results shown in Table II and Table III, both evaluators (Ev2 and Ev3) provide the same classification when ranking the perturbations from low to high impact on the system: i) "Replay attack", ii) "Flooding attack", and iii) "Tampering attack". This example points out the need of making explicit the criteria followed by the evaluator when analysing the results, because whereas Ev3 is considering that *Dependability* features are more relevant to determine the quality of the system in presence of perturbations, Ev2 considers *Performance* metrics slightly more relevant and, in both case, the same ranking is obtained. Thus, not providing an explicit definition of the decision process may lead readers to misunderstand the reasoning followed to obtain the conclusions, resulting in misleading results when the wrong decision making process will be applied to future experiments performed by that people.

IV. Discussion

Usually, the criteria used by evaluators is subjective and is determined by the application context of target system. This means that, when evaluating a web server that accesses a database in presence of attacks, for example, the criteria used to extract conclusions from results obtained should not be the same if that server that manages an industry's private information than if it manages posts in a cooking blog. So, this context is very important and must be taken into consideration when specifying the decision making process to be followed. However, while some of the presented methodologies lack the means to support approach (like Geometric mean or Kiviat diagrams), methodologies like LSP or AHP not only make explicit the criteria used for measures aggregation, but also remove any possible uncertainty in the process, as mathematical models would present less ambiguities than natural language.

A hierarchical representation of the analysis, which enables the navigation from coarse-grain (global score) to fine-grain (raw measures) through medium-grain (intermediate features) viewpoints, opens the doors to evaluators with many different profiles. For example, i) developers may get as many raw measures as desired to have a complete and detailed picture of the system under development, ii) administrators may prefer having a reduced number of aggregated scores characterising different features of the system while tuning its configuration, whereas iii) end users with low expertise may obtain just a single score characterising the quality of the deployed system.

Although the benefits of these approaches seem indubitable, there are a lot of questions still to be solved, like i) how to integrate decision making processes in the common dependability benchmarking process, ii) in case of methodologies being complementary, how can they be combined to make the most of them and ease the decision making process, or iii) in case of methodologies being exclusive, in which scenarios should each of them be applied. Accordingly, there is still a long way to go before the dependability benchmarking community embraces these practices.

V. Conclusions

Along the years, dependability benchmarking has evolved into a mature discipline with applicability in many different areas. Most related works focused on the specification of clear guidelines for the definition and execution of dependability benchmarks, whereas others introduced well formed processes to define good quality measures for the evaluation processes. Nevertheless and although the main goal of these benchmarks is to compare and select among different products or systems those providing the best trade-off between performance and dependability, paradoxically no effort has been devoted yet to provide an accurate and unambiguous decision making process. Common aggregation processes followed to evaluate systems in dependability benchmarking lack rigorousness and vary continuously from one work and evaluator to another. In many cases, the decision criteria applied to analyse the resulting measures is not made explicit, thus making more difficult the fair comparison of results obtained in different experiments and/or by different evaluators.

This work can be considered as a first step forward to pave the way for integrating decision making methodologies into the dependability benchmarking process to enable the **conclusions reproducibility**.

Acknowledgements

This work is partially supported by the Spanish project ARENES (TIN2012-38308-C02-01), the ANR French project AMORES (ANR-11-INSE-010), and the Intel Doctoral Student Honour Programme 2012.

References

[1] J. Dures, M. Vieira, and H. Madeira, "Dependability benchmarking of web-servers," in *Computer Safety, Reliability, and Security*, ser. Lecture Notes in Computer Science, M. Heisel, P. Liggesmeyer, and S. Wittmann, Eds. Springer Berlin Heidelberg, 2004, vol. 3219, pp. 297–310.

[2] M. Vieira and H. Madeira, "A dependability benchmark for oltp application environments," in *Proceedings of the 29th international conference on Very large data bases - Volume 29*, ser. VLDB '03. VLDB Endowment, 2003, pp. 742–753.

[3] J.-C. Ruiz, P. Yuste, P. Gil, and L. Lemus, "On benchmarking the dependability of automotive engine control applications," in *IEEE/IFIP International Conference on Dependable Systems and Networks*, 2004, pp. 857–866.

[4] DBench, "Dependability Benchmarking," IST Programme, European Commission, IST 2000-25425, [Online]. Available: http://www.laas.fr/DBench, 2013.

[5] A. Bondavalli, A. Ceccarelli, L. Falai, and M. Vadursi, "A new approach and a related tool for dependability measurements on distributed systems," *IEEE Transactions onInstrumentation and Measurement*, vol. 59, no. 2, pp. 820–831, 2010.

[6] K. Kanoun and L. Spainhower, *Dependability benchmarking for computer systems*. John Wiley & Sons, 2008, vol. 72.

[7] A. Ceccarelli, "Analysis of critical systems through rigorous, reproducible and comparable experimental assessment," Ph.D. dissertation, 2012.

[8] EEMBC, "Embedded microprocessor benchmark consortium." [Online]. Available: http://www.eembc.org

[9] G. Concas, M. Marchesi, S. Pinna, and N. Serra, "Power-laws in a large object-oriented software system," *IEEE Transactions on Software Engineering*, vol. 33, pp. 687–708, October 2007.

[10] Y. A. Al-Sbou, R. Saatchi, S. Al-Khayatt, R. Strachan, M. Ayyash, and M. Saraireh, "A novel quality of service assessment of multimedia traffic over wireless ad hoc networks," in *Proceedings of the 2008 The Second International Conference on Next Generation Mobile Applications, Services, and Technologies*, 2008, pp. 479–484.

[11] K. W. Kolence and P. J. Kiviat, "Software unit profiles and Kiviat figures," *ACM/Sigmetrics Performance Evaluation Review*, vol. 2, no. 3, pp. 2–12, 1973.

[12] M. F. Morris, "Kiviat graphs: conventions and figures of merit," *ACM/Sigmetrics Performance Evaluation Review*, vol. 3, no. 3, pp. 2–8, 1974.

[13] D. de Andres, J. C. Ruiz, and P. Gil, "Using dependability, performance, area and energy consumption experimental measures to benchmark ip cores," in *Forth Latin American Symposium on Dependable Computing (LADC)*, 2009, pp. 49–56.

[14] J. Dujmovic and R. Elnicki, *A DMS Cost/Benefit Decision Model: Mathematical Models for Data Management System Evaluation, Comparison, and Selection*. National Bureau of Standards, Washington D.C., No. GCR 82-374. NTIS No. PB 82-170150, 1982.

[15] 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. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-83555-1_5

[16] J. Friginal, D. de Andres, J.-C. Ruiz, and P. Gil, "On selecting representative faultloads to guide the evaluation of ad hoc networks," in *Dependable Computing (LADC), 2011 5th Latin-American Symposium on*, april 2011, pp. 94 –99.

[17] J. J. Dujmovi and H. Nagashima, "LSP method and its use for evaluation of java IDEs," *International Journal of Approximate Reasoning*, vol. 41, no. 1, pp. 3 – 22, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0888613X05000423

[18] A. Avizienis *et al.*, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no. 1, pp. 11–33, Jan–March 2004.