Conference Paper

Towards a Methodology for Experimental Evaluation in Low-Power Wireless Networking

Author(s):
Jacob, Romain; Boano, Carlo A.; Raza, Usman; Zimmerling, Marco; Thiele, Lothar

Publication Date:
2019-04

Permanent Link:

https://doi.org/10.3929/ethz-b-000325096 →

Towards a Methodology for Experimental Evaluation in Low-Power Wireless Networking

Romain Jacob
ETH Zurich
jacobr@ethz.ch

Carlo Alberto Boano
Graz University of Technology
cboano@tugraz.at

Usman Raza
Toshiba Research Europe Limited
usman.raza@toshiba-trel.com

Marco Zimmerling
TU Dresden
marco.zimmerling@tu-dresden.de

Lothar Thiele
ETH Zurich
thiele@ethz.ch

ABSTRACT

Making experimental research on low-power wireless networking *repeatable*, *reproducible*, and *comparable* is a long overdue step that hinders a wide acceptance of this technology within the industry. In this paper, we start to fill this gap by proposing and applying a *well-defined methodology* that specifies how to plan and execute experiments, as well as how to report their results. We further discuss potential definitions for repeatability, replicability, and reproducibility in the context of low-power wireless networking.

CCS CONCEPTS

• **Networks** → **Network performance evaluation**.

KEYWORDS

Methodology; Experiment Design and Analysis

ACM Reference Format:
Romain Jacob, Carlo Alberto Boano, Usman Raza, Marco Zimmerling, and Lothar Thiele. 2019. Towards a Methodology for Experimental Evaluation in Low-Power Wireless Networking. In *2nd Workshop on Benchmarking Cyber-Physical Systems and Internet of Things (CPS-IoTBench '19), April 15, 2019, Montreal, QC, Canada.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3312480.3313173

1 INTRODUCTION

A scientific contribution can be considered valid only when the results have been reproduced by others. While this seems obvious, the current practice in low-power wireless research is a far cry from this goal. Even if the source code is available, the description of the evaluation setup and how the results are derived from the raw measurements are often incomplete and invalid from a statistical standpoint. More fundamentally, it remains an open question how can results be considered reproducible in the face of uncontrollable variability in the test environment (*e.g.*, a testbed).

CPS-IoTBench '19, April 15, 2019, Montreal, QC, Canada

ACM ISBN 978-1-4503-6693-9/19/04…$15.00
https://doi.org/10.1145/3312480.3313173

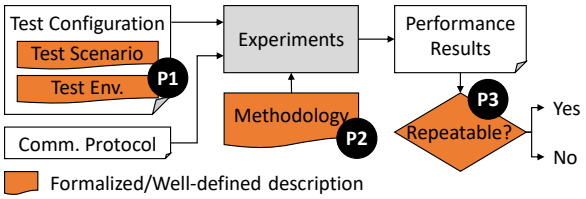


Figure 1. Towards benchmarking of wireless networking. *To improve the reproducibility and comparability of research contributions, a formalized description of the test configuration description (**P1**) and a well-defined methodology to conduct experiments (**P2**) are necessary. Only then, one can start debating about the repeatability and reproducibility of the results (**P3**). This paper focuses on **P2**+**P3**.*

In recent years, the low-power wireless community has started to work on improving the reproducibility and comparability of experimental results [4, 11–13] The goal is to derive a set of *benchmark problems* that can be used as a recognized yardstick to compare the performance of different networking solutions (*e.g.*, routing vs. flooding), different platforms (*e.g.*, single-core vs.multi-core), or even different low-power wireless technologies (*e.g.*, BLE vs. IEEE 802.15.4) in relevant scenarios inspired by real-world applications.

A benchmark problem associates a given *test configuration* with a set of relevant *performance metrics*. Such a well-defined setup is meant to enable a quantitative performance comparison of different low-power wireless communication protocols. However, to improve the reproducibility and comparability of protocol performance, defining benchmark problems is just one piece of the puzzle. We identify six sub-parts in solving this complex problem:

- **P1** A common framework to describe the *test configuration* of wireless networking experiments. Such a framework should include both the test scenario (*i.e.*, the traffic pattern and load), as well as the test environment (*i.e.*, the testbed infrastructure or simulator tool used in the evaluation).
- **P2** A well-defined *experimental methodology* that prescribes how to plan, execute, and report experimental results. For example, such methodology should inform the experimenter about which data should be collected and how, the way the collected data should be processed, and how to synthesize the data into a statistically meaningful performance report.
- **P3** Formal definitions of *repeatability*, *replicability*, and *reproducibility* in the context of low-power wireless networking.
- **P4** A well-defined *comparison methodology* that prescribes, for example, how can one claim that "protocol A is better than protocol B."

P5 A set of *benchmark problems*, formulated using the framework to describe the test configuration (**P1**) and executed according to the well-defined experimental methodology (**P2**).

P6 *Technical solutions* (*i.e.*, tools) that let experimenters apply this methodology (thus improving reproducibility) without cluttering research papers with details about the evaluation.

Initial work has been conducted to address **P1** [4]. In this paper, we focus on **P2** and **P3** (see Fig. 1), specifically,

- We outline a well-defined methodology that specifies how experimental evaluation should be conducted, and how results should be reported (Sec. 3). This methodology relies on a sound approach based on non-parametric statistics.
- We apply the proposed experimental methodology in a case study, reporting the performance of seven low-power wireless protocols evaluated on the same test configuration (Sec. 4).
- We propose definitions for repeatability, replicability, and reproducibility in the context of low-power wireless networking, where (potentially large) performance variation is expected due to uncontrollable environmental factors (Sec. 5).

2 BACKGROUND ON STATISTICS

This section introduces the necessary background on non-parametric statistics as the basis for the methodology proposed in Sec. 3.

The output of an experiment is a sequence of measurements, *e.g.*, the end-to-end latency of packets received by one node. We can interpret the set of measurements as an empirical distribution (*e.g.*, of the end-to-end latency). By performing more repetitions of the same experiment, we increase the confidence that the empirical distribution closely matches the true population distribution one would observe when performing infinitely many repetitions.

Previous studies have shown that performance measurements are often not normally distributed [10]. Hence, it is inappropriate to compare sets of performance measurements based on the sample mean and the sample standard deviation. Rather, more robust methods from non-parametric statistics should be applied, which suggest using instead the median or other percentiles.

Assume we have a set of measurements X. After sorting X, the median is the measurement at index $\lfloor n/2 \rfloor$, where n is the number of measurements in X; the p-th percentile, $0 < p < 100$, is the measurement at index $\lceil np/100 \rceil$. Assuming the X measurements are independent and identically distributed (iid), one can derive the probability α that the true percentile p of the population distribution falls in the interval $I = [x_j, x_k]$. I is called a *confidence interval* (CI) for the percentile p with *confidence level* α. For large n, indices j and k can be approximated as $\lfloor (np - z\sqrt{np(1-p)})/2 \rfloor$ and $\lceil 1 + (np + z\sqrt{np(1-p)})/2 \rceil$ respectively, where $z = 1.96$ for a confidence level $\alpha = 95\,\%$ [8]. Tables are available in the literature for small values of n [8, 10]. This allows deriving the minimal number of samples n necessary to give a CI for any percentile p. Typically, the CIs tend to get narrower with more repetitions, *i.e.*, larger n.

3 METHODOLOGY

To improve on the reproducibility and comparability of research contributions, it is paramount to agree on how the evaluation of these contributions should be performed. In other words, a well-defined *experimental methodology* is required.

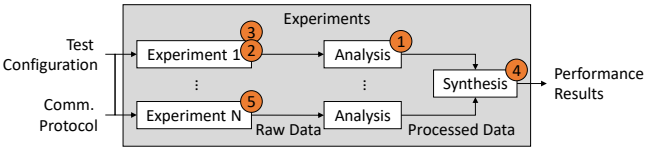


Figure 2. The experimental procedure can be decomposed in three phases: *experiments*, *analysis*, and *synthesis*. *An experimental methodology should answer five questions: (1) Which metrics to compute? (2) Which data to collect? (3) How many samples to collect? (4) How to synthesize results? (5) How many experiments should be performed?*

In order to derive such a methodology, it is important to first dissect the evaluation process itself. Let us assume one protocol is to be evaluated on one test configuration. [1] We argue that the experimental procedure can be decomposed into three main phases:

The experiments which lead to the collection of *raw data*.

The analysis which processes the raw data of each experiment. This usually means computing a set of *metrics*.

The synthesis which aggregates the processed data of multiple experiments in a comprehensive summary. We call this summary a *performance report*.

We illustrate these phases in Fig. 2. A number of strongly interrelated questions must be answered to fully describe this process:

(1) *Which metrics should be computed?* Only the experimenter can decide on the performance dimensions that matter (*e.g.*, reliability). For each dimension, different metrics can be considered and different measures can be useful to compute.

(2) *Which raw data should be collected?* There is a trade-off between the ease of collecting raw data and their richness. The more fine-grained the data being collected are, the more information can be extracted out of them.

(3) *How many samples should be collected?* Given the chosen measures, how many samples are necessary in order to obtain statistically relevant numbers? Answering this question is necessary to define the minimal length of an experiment.

(4) *How to synthesize results into a performance report?* It is a priori not clear how the results of multiple experiments should be aggregated.

(5) *How many experiments should be performed?* Depending on the chosen synthesis approach, how many experiments should be performed in order to obtain statistically relevant performance reports?

In the remainder of this section, we look into each of these questions and make concrete proposals. Altogether, this sketches a well-defined methodology to conduct experimental evaluations.

3.1 Which metrics to compute?

Before deciding which metrics to use in an evaluation, one should reflect on the physical dimensions of interest. Different application domains (*e.g.*, condition monitoring or industrial control) likely focus on different dimensions (*e.g.*, reliability or latency). Each dimension can be investigated using one or several metrics, based on which some *performance measures* are computed.

[1] We do not discuss here how to formally describe a test configuration (**P1**); this is beyond the scope of this paper. We simply assume such formal description exists.

For example, let us consider energy efficiency. This is a physical dimension that can be investigated using, *e.g.*, the radio duty cycle as a metric. If one is interested in the mean depletion time of a node's battery (related to the average energy consumption), a valid measure is, *e.g.*, the median radio duty cycle across all nodes. If one is interested in the expected time before a first node depletes its battery (related to the maximum consumption), a different measure is needed, *e.g.*, the maximal current draw of individual nodes.

As described in Sec. 2, in our context, the physical quantities we study are expected to be non-parametric. Thus, *statistical methods based on mean and variance should not be used*. Instead, the literature recommends using confidence intervals (CI) on sample percentiles [10], which are are good candidates as performance measures. Moreover, performance reports must be based on the same metrics to be comparable. This calls for a consolidation of a core list of "metrics that matter," for which [14] provides a good starting point.

3.2 Which data to collect?

Obviously, the minimal requirement is that the collected data are sufficient to compute the metrics of interest.

Moreover, the more fine-grained or unprocessed the data are, the richer and thus more interesting they are. For example, let us consider the end-to-end latency of individual packets. Raw data containing the transmission and reception timestamps are far more valuable than data containing only their relative difference. The latter only quantifies latency, whilst the former can also provide, *e.g.*, information about the receiving jitter.

3.3 How many samples to collect?

As discussed in Sec. 3.1, a tendency in the data can be described with a median, *i.e.*, the 50th percentile. Instead, if one is interested in extreme performance, a higher percentile (*e.g.*, 95th or even 99th percentile) might be more suited as a measure.

Moreover, these measures are often *predictive*. In other words, we attempt to estimate what the true value of the measure would be if the test were run forever. In such a case, one cannot report on exact values but can only provide CI on the true value, given a certain confidence level α (see Sec. 2).

It follows that the minimal number of samples one must collect depends on the measure of interest. Intuitively, if a few samples can be sufficient to estimate the median, many more are required to estimate the 99th percentile. The required number of samples must be computed given the desired confidence level α and the selected measure. For example, assuming a confidence level $\alpha = 75\%$, [2] the approach described in Sec. 2 yields that 3 samples are enough to report a CI on the median, whereas estimating the 97.7th and 99.8th percentiles requires a minimum of 61 and 1027 samples [10]. [3]

3.4 How to synthesize results?

After analysis, each experiment provides a set of processed data, *i.e.*, the chosen measures (see Sec. 3.1). However, to concisely report on the system performance (and eventually compare different systems – **P4**), it is useful to synthesize these results using what we call *performance indicators*. An indicator is a unique numerical value that synthesizes the system's performance across the whole evaluation along one of the measures. Thus, a performance report synthesizes the whole evaluation into a vector of size M, where M is the number of measures.

The definition of "good" indicators is a priori not trivial. Analog to the discussion in Sec. 3.3, we suggest to define indicators using percentiles on the measures. Again, the percentiles to use depend on the type of performance statements that one is trying to make. The median across all tests can be to used to report on average performance; a higher percentile is needed to investigate extreme performance (*e.g.*, the latency of a real-time protocol).

As the evaluation contains only a finite number of tests, the true percentiles must be estimated using CI at a given confidence level α. This results in two values per measure: the lower and upper bound of the CI. Let us recall the meaning of a CI: The true value of the percentile for the underlying distribution lies somewhere in the interval with probability α. We propose to use as performance indicator the "conservative bound," which depends on the metric. Consider for example the reliability measured as packet reception rate (PRR): the higher the PRR, the better. Thus, to be conservative, one should use the lower bound of the CI as a performance indicator for this metric. It is the opposite for the energy consumption: the lower, the better. Thus, the upper bound of the CI should be used as a performance indicator.

3.5 How many experiments to perform?

The final question to answer in order to complete the experimental evaluation planning is: *How many repetitions should be performed?*

Once again, the answer depends on the type of performance statements one wants to make; in other words, it is subordinate to how the results are synthesized (see Sec. 3.4). A minimum of 3 samples (*i.e.*, 3 repetitions) are required to obtain a 75% CI on the median performance. More repetitions allow one to make stronger statements by increasing the confidence level; *e.g.*, a 95% CI on the median requires 6 samples [10]. Additional repetitions may also help narrowing the CIs by excluding extreme values.

Summary. This section presented a methodology to plan, execute, and report on experimental evaluations. This methodology does not specify *what* should be done in the evaluation (*e.g.*, which metrics to compute), but rather *how* to do it (*e.g.*, how to choose valid measures). In particular, we propose a statistically relevant approach to answer two basic yet difficult questions: (i) *How long should an experiment run?* (ii) *How many repetitions should be performed?*

The core ideas of this methodology can be summarized in the following guidelines:

(1) Use the correct measure for the performance aspects under investigation. To facilitate the comparability between performance results, the same metrics should be used. This calls for a restricted list of metrics that the community should agree to use to investigate common performance aspects of communication protocols such as average energy consumption, worst-case latency, etc.
(2) Fine-grained data allow for a deeper analysis. Whenever possible, the raw data should be collected (and made accessible) with the finest granularity possible.

[2] 75% is a rather low confidence. A higher level, such as 95%, would be preferable.
[3] These percentiles "correspond" to the $\mu + \sigma$ and $\mu + 2\sigma$ for a normal distribution.

(3) Predictive measures should be based on confidence intervals of some distribution percentiles (*e.g.*, the median). The (minimal) duration of a test should be decided based on the chosen measures. The more extreme the percentiles are (*e.g.*, 99th), the more samples are required and hence the longer the test must last in order to reach a given confidence level. The latter should be set to a large value, such as 95%.
(4) The performance report should be synthesized using CIs with a minimal confidence level of 75% (or better: 95%). The *synthesis **should not** be done using the mean or standard deviation.* Unfortunately, although statistically inappropriate in our setting, the mean is commonly used in the community.
(5) Following our proposed methodology, experimenters should often perform many "short" tests (*i.e.*, simply long enough to compute the chosen metrics) rather than a few "long" tests.

4 CASE STUDY

We illustrate our experimental methodology on a practical case study, where we evaluate the performance of seven different protocols on the exact same scenario. [4]

4.1 Evaluation settings

Test scenario. We consider a data collection scenario in a 15-node network with 14 sources (*i.e.*, generating packets) and one sink node. Each source generates 200 application payloads of 2 bytes at a fixed rate of 10 per second. No payload is generated during the first 10 s, after which the generation is periodic, with a pseudo-random offset between the different sources (based on the node IDs). Once the 200 payloads have been generated (*i.e.*, after 20 s), the test runs for 10 s more before it stops. Any application payload not successfully received at the sink node by this time is counted as lost.

It is important to note that this scenario is *terminating*, *i.e.*, it has a definite end. This is different from a test scenario where one aims to estimate steady-state performance. One consequence is that after one run of the scenario, one obtains exact performance measures rather than estimates. For example, one measures the *exact* number of successfully received application packets. The uncertainty lies in the variability of the results across multiple runs, not on the performance that would be obtained if the tests were longer.

Test environment. We use the FlockLab testbed [9] as test environment, using the DPP platform [3]. The latter embeds a TI CC430 SoC featuring a sub- GHz RF core. The list of nodes and the identity of the sink node are fixed and known at design time. Tests are run during night time (between 10pm and 7am) to limit external interference. Further details about the test environment settings are contained in the FlockLab XML test files, which are available together with other additional material of this paper [1].

Performance metrics, measures, and synthesis method. Following our methodology, we first decide on the performance dimensions we aim to investigate with our evaluation before selecting valid measures and a synthesis strategy. As an example, we investigate the following three aspects:

Q1 *How many application payloads can one expect to successfully receive in one execution of the scenario?* This relates to the *average reliability*. A corresponding measure is the overall PRR, which produces one value per test. We synthesize the results using the lower bound of the 95% CI for the median PRR across all tests.

Q2 *How much energy can one expect to be consumed by one node during one execution of the scenario?* This relates to the *average energy consumption* across all nodes. A corresponding measure is the median current draw across all nodes, which produces one value per test. [5] We synthesize the results using the upper bound of the 95% CI for the median across all tests.

Q3 *After how many executions of the scenario will a first source node have depleted its battery?* This relates to the *maximal energy consumption* per source node. A corresponding measure is the maximal current draw of one node. This is computed by considering, for each individual node, the 95% CI of its median current draw across all tests, [6] then taking the maximal upper bound of all the CIs. It produces one value for the whole evaluation.

Ultimately, we synthesize the evaluation results using three normalized performance indicators, one for each metric. [7] By design, the PRR is already normalized. For the energy consumption, we transform the measures $\hat{x}$ into normalized values $x = 1 - \hat{x}/I_{\max}$, where $I_{\max} = 25$ mA is an upper bound for a node's current draw for this configuration. Thus, our three performance indicators range between 0 and 1 where a higher score means better performance.

Length and number of experiments. As the scenario is both terminating and short, each experiment runs the scenario in full. [8]

Our measures and aggregation strategy rely on 95% CI on median values across all tests. This leads to a minimum of 6 repetitions (see, *e.g.*, the tables in [10]). In order to obtain better estimates (*i.e.*, to limit the impact of potential outliers), we perform 20 repetitions. If 20 measurements are available, and sorted like $x_1 \leq x_2 \leq \ldots \leq x_{20}$, the 95% CI for the median is $[x_6, x_{15}]$ [10].

Raw data collection. As discussed in Sec. 3.2, the raw data should provide enough information to compute the metrics of interest, but should also strive to be as detailed as possible, such that further or different processing can be carried out. With this mindset, we collect the following raw data:

- The sink node writes individual received application payloads (2 pseudo-random bytes) into a serial message. The serial dump is provided by FlockLab as part of the test results.
- FlockLab collects current drain measurements of each node, at a rate of 144000 samples per second (1 sample every $\sim$ 7 μs) with a 10 picoampere precision. The test results contain both the complete time series and the average across the whole test, for each node.

A set of processing scripts convert these raw data into our performance indicators. All collected data, scripts, and utilization notes are openly available as complementary materials [1].

[4] All protocols have been designed by Master students for course in fall 2018.

[5] As we have the values for all the nodes, we obtain the *exact median current draw* for each test. This is a *descriptive statistic*.

[6] This is a *predictive statistic*: We try to estimate the true median value for each node by running a limited number of tests. Thus, we only obtain a CI (not an exact value).

[7] The normalization is optional, it simply helps comparing across indicators.

[8] 200 payloads generated at a rate of 10 per second, plus 10 s at the start and the end of the scenario, which hence lasts 40 s in total.

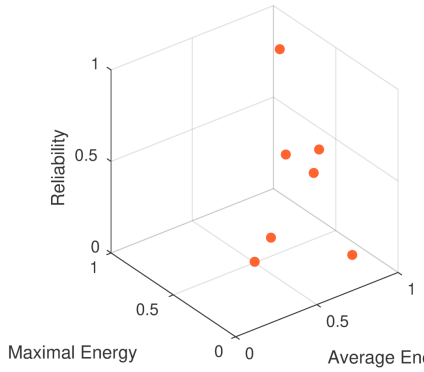


Figure 3. Performance results of all protocols. *A graphical representation of all performance results provides a high-level overview, but may be ill-suited for a static figure in a paper. Ideally, a dynamic version of the graphic should be provided (see [1]). A data table (such as Table 1) is more accurate and more suited to report on final results.*

Table 1. Performance results of all protocols. *While less visually attractive, a table provides more accurate information than a graphic such as Fig. 3.*

Protocol	A	B	C	D	E	F	G
Average Energy	0.82	0.83	0.89	0.86	0.90	0.43	0.25
Worst-case Energy	0.67	0.44	0.82	0.18	0.52	0.27	0.19
Reliability	0.40	0.41	0.89	0.06	0.48	0.27	0.25

4.2 Evaluation results

After the evaluation comes the question of the presentation of the results. Here, the challenge lies in reporting the results in a concise yet informative form.

For this as well, relying on a well-defined methodology helps. The evaluation results can be summarized without ambiguity using the chosen performance indicators, as their definition and derivation are formally specified. For a few indicators, a graphical representation can give a quick overview of the respective performance of different protocols (see Fig. 3). Nevertheless, a data table (such as Table 1) is more precise and should be provided.

Furthermore, whenever relevant, one should provide some representation of all processed data, *i.e.*, the measures. For example, Fig. 4 shows the PRR and median current draw for each test of protocol C, as well as the 95% CI for the median. Such graphic concisely shows the data points, their spread, and their synthesis; hence, it provides more details about the performance of a given protocol than the performance indicators alone.

5 REPRODUCIBILITY OF LOW-POWER WIRELESS NETWORKING

It is commonly recognized that *"an experimental result is not fully established unless it can be independently reproduced"* [2]. Recently, this fundamental statement in experimental research has been publicly made by the Association for Computing Machinery (ACM) [2]. To go further and foster best practices in experimental sciences, the ACM introduced a few years ago a badging system related to artefact reviews for scientific publications, associated with a terminology of reproducibility [2]. This terminology defines three levels

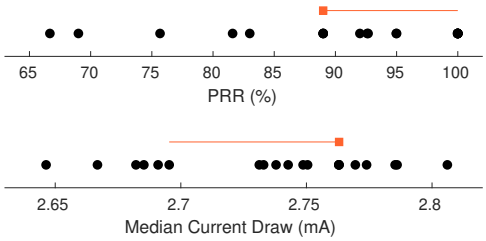


Figure 4. Measurements collected for protocol C. The dots are the measurement points of each individual test. The bar represents the 95% CI on the median, and the square marks the corresponding performance indicator. *This graphic efficiently shows the data points, their spread, and how they will be synthesized. Such representation is encouraged whenever one aims to give more details about the performance of a given protocol.*

of reproducibility which can be summarized as follows:

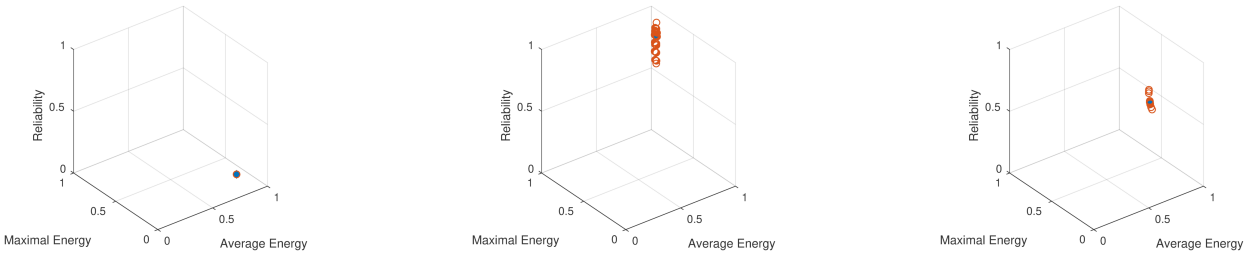
Repeatability	Same team	Same setup
Replicability	Different team	Same setup
Reproducibility	Different team	Different setup

This terminology is intentionally loose, such that it can be adapted to the specifics of different research fields. Whilst the intuition behind these definitions is relatively simple, their formalization is far from trivial. Given the natural variability of wireless experiments, asking performance results to be exactly the same to qualify these results as reproducible does not make much sense. Thus, one should ask the results to be "close." But how to measure this "closeness"? How to assess whether results are sufficiently "close"? Does a hard cut between reproducible or not even make sense; or should we rather aim to *quantify* reproducibility? This paper does answer these (difficult) questions, but does suggest some initial ideas to open the discussion.

Let us first focus on repeatability. Our methodology synthesizes the performance of a given protocol using some indicators (see *e.g.*, the case study in Sec. 4). Now, what does it mean to say: *"these results are repeatable"*? According to the ACM definition, it would mean that, if we were to re-run the complete evaluation, we would find "close" results. In other words, repeatability is concerned with the *confidence in the results obtained in the evaluation.*

One idea to investigate this question is the technique of *bootstrapping* [5]. Bootstrapping is a statistical method based on re-sampling, which allows increasing the accuracy on some population estimates; this can be applied to our problem. Let us assume we perform an evaluation with N repetitions, out of which we compute one vector of performance indicators. A *bootstrap sample* refers to a new synthetic set of N tests, where each test in the bootstrap sample is randomly chosen from the original N tests. For example, if our original test set is $\{1, 2, 3\}$, a bootstrap sample could be randomly created as $\{2, 1, 2\}$. For each bootstrap sample, we can compute a new vector of performance indicators. By creating many bootstrap samples (*e.g.*, 1000), one easily obtains a population of performance vectors out of the original N tests. [9] The idea is that the spread of this population could be used to measure the repeatability of the

[9] While bootstrapping seems to create values out of thin air (hence its name), it has been shown useful *e.g.*, to provide confidence intervals on some population parameters [5]; in our case, this parameter would be the vector of performance indicators.



(a) *The performance indicators computed on the bootstrap population are all closely grouped together, which would tend to indicate very repeatable results.*

(b) *The performance indicators computed on the bootstrap population are very stable with respect to energy, but exhibit a large spread with respect to reliability. Is this "repeatable" or not?*

(c) *The performance indicators computed on the bootstrap population are very stable with respect to average energy, but exhibit a correlation between maximal energy and reliability.*

Figure 5. While it is rather easy to argue that the results for Fig. 5a are "more repeatable" than those in Fig. 5b and 5c, it is yet unclear how to generally assess the repeatability of a single protocol.

result. The intuition is that the "closer" the performance vectors from the bootstrap distribution, the "more repeatable" the result.

As an example, Fig. 5 shows the performance indicators of the bootstrap distributions for three different protocols from our case study. While it is rather easy to argue that the results in Fig. 5a are "more repeatable" than Fig. 5b and 5c, it is yet unclear how to generally assess the repeatability of individual results. Furthermore, it remains to be clarified whether the statistical guarantees of bootstrapping hold for our performance indicators (which are not quite the same as typical estimates, like, *e.g.*, the mean). However, we argue that this avenue deserves further investigations.

We now shortly comment on replicability and reproducibility. These definitions yield that a *different team* is performing the evaluation again. In this context, the "closeness" between the original and the replicated/reproduced studies could be formulated with the following question: *What is the probability that the two sets of results are samples coming from the same underlying distribution?* This question could be answered by using another well-founded statistical method like, *e.g.*, the Kruskal-Wallis test [7]. But here as well, further investigation about its applicability is needed.

6 DISCUSSION AND CONCLUSION

In this paper, we have outlined some necessary steps towards making experimental research on low-power wireless networking *repeatable*, *reproducible*, and *comparable*. We identified the lack of a *well-defined methodology* that specifies how to plan, execute, and report on experimental results as one of the missing ingredients towards this goal. We hence proposed a methodology suitable when experimenting with low-power wireless protocols and applied it on a case study. We have further discussed how complex is to define repeatability, replicability, and reproducibility in the context of low-power wireless networking. Further research is needed to turn our ideas into a full-fledged, validated methodology that serves as an accepted guideline for experimental evaluations in the field. Inspiration can be taken from other disciplines, *e.g.*, clinical studies [6], where reproducibility has been a major concern for a long time.

ACKNOWLEDGMENTS

We thank Balz Maag and Antonios Koskinas for the fruitful discussions and useful pointers provided during this work. This work was supported in part by the German Research Foundation (DFG) within the Emmy Noether project NextIoT (grant ZI 1635/2-1).

REFERENCES

[1] 2019. Complementary Materials. http://romainjacob.net/2019-cps-iotbench.
[2] ACM. 2018. Artifact Review and Badging. https://www.acm.org/publications/policies/artifact-review-badging.
[3] Jan Beutel, Roman Trüb, Reto Da Forno, Markus Wegmann, Tonio Gsell, Romain Jacob, Michael Keller, Felix Sutton, and Lothar Thiele. 2019. Demo Abstract: The Dual Processor Platform Architecture. In *2019 18th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*.
[4] Carlo A. Boano, Simon Duquennoy, Anna Förster, Omprakash Gnawali, Romain Jacob, Hyung-Sin Kim, Olaf Landsiedel, Ramona Marfievici, Luca Mottola, Gian Pietro Picco, Xavier Vilajosana, Thomas Watteyne, and Marco Zimmerling. 2018. IoTBench: Towards a Benchmark for Low-Power Wireless Networking. In *1st Workshop on Benchmarking Cyber-Physical Networks and Systems (CPSBench 2018)*. https://doi.org/10.3929/ethz-b-000256517
[5] Bradley Efron. 1992. Bootstrap Methods: Another Look at the Jackknife. In *Breakthroughs in Statistics: Methodology and Distribution*, Samuel Kotz and Norman L. Johnson (Eds.). Springer New York, New York, NY, 569–593. https://doi.org/10.1007/978-1-4612-4380-9_41
[6] Leonard P. Freedman, Iain M. Cockburn, and Timothy S. Simcoe. 2015. The Economics of Reproducibility in Preclinical Research. *PLOS Biology* 13, 6 (June 2015). https://doi.org/10.1371/journal.pbio.1002165
[7] William H. Kruskal and W. Allen Wallis. 1952. Use of Ranks in One-Criterion Variance Analysis. *J. Amer. Statist. Assoc.* 47, 260 (Dec. 1952), 583–621. https://doi.org/10.1080/01621459.1952.10483441
[8] Jean-Yves Le Boudec. 2011. *Performance Evaluation of Computer and Communication Systems*. Epfl Press.
[9] Roman Lim, Federico Ferrari, Marco Zimmerling, Christoph Walser, Philipp Sommer, and Jan Beutel. 2013. FlockLab: A Testbed for Distributed, Synchronized Tracing and Profiling of Wireless Embedded Systems. In *Proceedings of the 12th International Conference on Information Processing in Sensor Networks (IPSN '13)*. ACM, New York, NY, USA, 153–166. https://doi.org/10.1145/2461381.2461402
[10] H. Schmid and A. Huber. 2014. Measuring a Small Number of Samples, and the 3v Fallacy: Shedding Light on Confidence and Error Intervals. *IEEE Solid-State Circuits Magazine* 6, 2 (2014), 52–58. https://doi.org/10.1109/MSSC.2014.2313714
[11] Markus Schuß, Carlo Alberto Boano, and Kay Römer. 2018. Moving Beyond Competitions: Extending D-Cube to Seamlessly Benchmark Low-Power Wireless Systems. In *CPSBench 2018 - International Workshop on Benchmarking Cyber-Physical Networks and Systems*. 6.
[12] Markus Schuß, Carlo Alberto Boano, Manuel Weber, Matthias Schulz, Matthias Hollick, and Kay Römer. 2019. JamLab-NG: Benchmarking Low-Power Wireless Protocols under Controlable and Repeatable Wi-Fi Interference. In *Proceedings of the 2019 International Conference on Embedded Wireless Systems and Networks*. Junction Publishing, Beijing, China, 12.
[13] Mališa Vučinić, Milica Pejanovic-Djurisic, and Thomas Watteyne. 2018. SODA: 6TiSCH Open Data Action. In *CPSBench 2018 - International Workshop on Benchmarking Cyber-Physical Networks and Systems*.
[14] Dingwen Yuan, Salil S. Kanhere, and Matthias Hollick. 2017. Instrumenting Wireless Sensor Networks — A Survey on the Metrics That Matter. *Pervasive and Mobile Computing* 37 (June 2017), 45–62. https://doi.org/10.1016/j.pmcj.2016.10.001