This article has been accepted for publication in IEEE Pervasive Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MPRV.2024.3386337

Special Issue: Pervasive Sustainability

The Necessary Shift: Towards a Sufficient Edge Computing

Klervie Toczé, *inköping University, Sweden*

Simin Nadjm-Tehrani, *inköping University, Sweden*

Abstract—Edge computing is becoming a reality and attracts an increasing interest both from academia and industry. This is driven by its promises of enabling/improving use cases thanks to e.g. lower latency or alleviated network load. This paves the way for edge computing having a huge impact on our daily lives in the (near) future. However, except works dealing with energy efficiency, studies of the (un)sustainability of edge computing are almost nonexistent, which is worrying. In this paper, we advocate the need to go beyond energy efficiency and face the resource impact of edge computing. At this point when we are still able to influence design choices, it is the responsibility of this community to ensure future systems do not become unsustainable down the line. In particular, we suggest embracing a sufficiency mindset, aiming at reducing absolute resource impact and defining what is a good enough service level. After explaining why we need to move beyond efficiency, we explore the concept of sufficiency and identify related challenges. Then, we propose a first version of an edge sufficiency toolkit as a helper for shifting towards a sufficiency mindset. Finally, we illustrate the use of this toolkit in a case study.

The edge computing paradigm, which is about moving resources (e.g. computational, storage) closer to the end users, at the edge of the network [1] is becoming a reality. Large companies such as Microsoft or Amazon have added edge computing-related services to their offerings and it is estimated that the edge will be a major player in the future, both in terms of investments in the infrastructure [2] and of how much data it handles.

At the same time, the current state of the world is very concerning. From an environmental perspective, numerous natural disasters due to climate change are a very tangible reminder of the impact humans have on the planet, an impact that is actually threatening our future. This is documented in e.g. the latest Intergovernmental Panel on Climate Change (IPCC) report [4]. One pinpointed issue is the overall unsustainable consumption of natural resources. Because digital technology is today pervasive in our way of living, it contributes to a large part of our impact: digital technologies emitted 3.5% of the greenhouse gases in 2019. This share grows rapidly and could exceed 7% by 2025 [5]. While digital technologies can help tackling sustainability challenges, it is necessary to also look into how digital technologies themselves can be made more sustainable.

The deployment of edge computing will irretrievably contribute to an even larger resource use, especially if it is done in the usual way, i.e. without thinking about sustainability first. As such a large resource use is unsustainable, there is a need to think differently this time. This is a task for every edge researcher and practitioner. Indeed, technology is not value neutral [6] and we are responsible for what we create.

We thus argue that there is a need for more sustainability-oriented research and practices in edge computing. So far, the works including sustainability thinking have been mainly focusing on one of the three strategies for sustainability [7], namely efficiency. However, efficiency strategies are known to be subject to rebound effects that are hard to anticipate and often contribute to worsening the initial issue [6]. Therefore, we lift the need to diversify the strategies considered and advocate consideration of the sufficiency strategy. Our goal with this work is to raise awareness about sufficiency and the associated challenges, as well as

---

XXXX-XXX © 2023 IEEE
Digital Object Identifier 10.1109/XXX.0000.0000000

This article has been accepted for publication in IEEE Pervasive Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MPRV.2024.3386337

to ease shifting to the sufficiency mindset with a few tools.

Resource use within edge computing happens in all life-cycle phases, from manufacturing to handling the resulting e-waste. Likewise, the edge technology landscape is wide, including for example different types of systems (e.g. multi-access edge, enterprise edge) and devices (more or less resource-constrained). Researchers and practitioners can be involved in edge development and/or operation. While the concepts and challenges introduced in this work are applicable to all the above, *how* to tackle them may vary (including adapting the tools). Our starting point for proposing tools and examples is within the development of algorithms for the use phase of multi-access edge infrastructures.

THE CURRENT SITUATION

Alarming forecasts about global rise of temperatures in connection with extreme weather events (e.g. heat waves or record floods) have multiplied in the past years, calling for urgent actions towards *sustainability*.

Sustainability

Several frameworks have been designed to describe sustainability such as the UN Sustainable Development Goals[1]. These include environmental, social and economic dimensions. Another concept is the one of planetary boundaries [8], delimiting a safe operating space within nine processes regulating the stability and resilience of the Earth.

Sustainability within edge computing

Sustainability is unfortunately often absent from the discussions in the edge community, or reduced to energy efficiency. It may be because of the “neutral” technology assumption, i.e. that it is only how it is used which is harmful or beneficial [6]. Another reason may be the belief that edge computing can contribute to “greening” our practices, e.g. through energy efficiency or the development of “smart” technologies, and that no further effort is required.

This absence is actually not specific to edge computing but happens in the IT field in general. Notable exceptions of works considering how to think about IT and sustainability are, however, appearing in recent years [9], [10]. In the specific field of edge computing, sustainability is one of the arguments motivating edge computing research [11] and works tackle e.g. modeling the CO_2 footprint of applications [12] or optimizing renewable energy use [13].

In our previous work [14], we identified seventeen edge computing sustainability issues and formalize them as unsustainable patterns. They cover a diverse range of issues, ranging from digital exclusion to one-sided infrastructure control. Among them, several patterns highlight the need to produce and consume less, differently and better. For example, edge computing leads to hardware multiplication and falls into the efficiency trap. One countermeasure presented is considering the notion of sufficiency.

The concept of digital sufficiency has been recently explored by Santarius et al. [7], considering IT as a whole. Moreover, Madon and Lago [15] explored digital sufficiency in the context of flexible work enabled by cloud computing. In this paper, we explore sufficiency in the context of edge computing.

MOVING BEYOND EFFICIENCY

Before diving into sufficiency, we present a short illustrative example to show why we shouldn’t only focus on efficiency for reducing our footprint.

Efficiency is typically considered to be about using the least amount of inputs to achieve the highest amount of output. Now, imagine a fictional device A: it can provide 1000 Million Instructions Per Second (MIPS), has a power draw of 25W and cost 1000$. Research and development are conducted to make this device more efficient, with regards to cost and energy. This leads to the creation of device B: it can provide 500 MIPS, has a power draw of 10W and cost 200$. It is more cost-efficient as you get more MIPS for one dollar spent (2.5 instead of 1) and more energy-efficient as you also get more MIPS for one Watt spent (50 instead of 40), i.e. you have higher output (MIPS) with lower input (dollar or Watt).

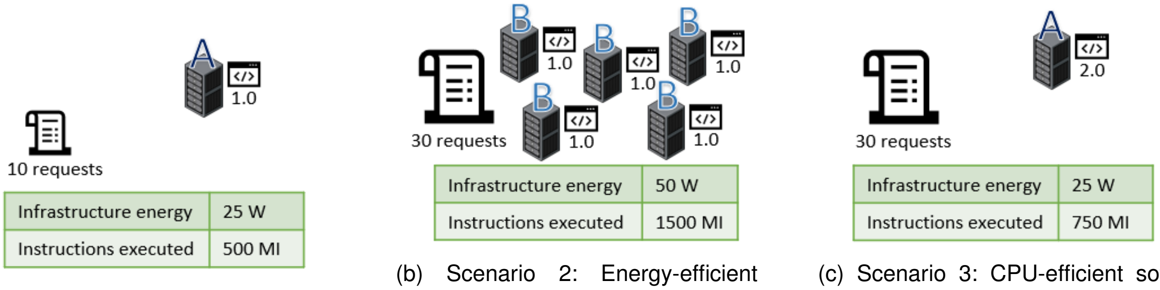
Next consider the edge infrastructure of Scenario 2 (Figure 1b). It uses the cost- and energy-efficient device B instead of device A (Figure 1a). Both devices run version 1.0 of a given edge software, where one request requires 50 MI. Leveraging the lower cost of device B, Scenario 2 enables increasing the computing capacity of the edge infrastructure at the same cost as Scenario 1, for example by covering new geographical areas. More users can be served and the demand increases. However, despite device B being more energy-efficient, the total energy for powering the infrastructure is higher in Scenario 2.

Another example: a new version (2.0) of the software is developed. This version is more CPU-efficient,

---

[1] https://sdgs.un.org/goals

This article has been accepted for publication in IEEE Pervasive Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MPRV.2024.3386337



(a) Scenario 1: Baseline.

(b) Scenario 2: Energy-efficient hardware.

(c) Scenario 3: CPU-efficient software.

FIGURE 1: Illustrating why we should think beyond efficiency.

one request requires 25 MI instead of 50. The software is faster, which leads to more users using it, so the demand increases. Therefore, Scenario 3 (Figure 1c), which is using the CPU-efficient software, is executing more CPU instructions than Scenario 1.

In both cases above, using the most efficient hardware or software actually leads to an increase of resource usage, for the very same resource that the efficiency improvements target (energy or CPU). Such a situation is commonly happening and is known as a rebound effect or the Jevons paradox[2].

This happens because an efficiency mindset is agnostic to how the demand and supply sides look like from a resource perspective. Hence, it is often the case that efficiency improvements lead to a demand increase. Therefore, looking closely at the demand and supply sides is necessary to increase the chances of an actual lower resource use. This is why we advocate for moving away from efficiency as the main strategy.

A final note: of course one can create examples where using more efficient hardware or software will lead to a decrease of resource use. This is not the point here. We want to illustrate why we should not mainly focus on efficiency, not that efficiency should not be pursued. Efficiency *is* useful, but relying solely on it will not solve the current issues. Similarly, these examples are simplified and consider an isolated edge system. In practice, one should also take into consideration the effects of different systems interacting with each other.

SUFFICIENCY AND ASSOCIATED CHALLENGES

We therefore argue for shifting towards the sufficiency strategy as the primary focus.

Defining sufficiency

At a high level, sufficiency policies are defined by the IPCC as “a set of measures and daily practices that avoid demand for energy, materials, land and water while delivering human well-being for all within planetary boundaries” [4]. Therefore, sufficiency is focusing on reducing the absolute level of resource consumption while still maintaining or improving living conditions for all [7].

Hence, sufficiency can be seen as a two-step process. First, one has to define which goods and services are necessary for human well-being and to which extent. Then, given this defined level of production and consumption, one can work on reducing the absolute amount of resource needed to achieve this level. The aim is to use the minimum amount of resources to achieve well-being for all.

The second step of the process may be reduced to an optimization problem, which is minimizing resource consumption subject to quality of service (QoS) constraints[3]. The edge community being used to deal with optimization problems, it may be tempting to stop here. However, the sufficiency concept is more than that. The first step is at least, if not more, as important as the second.

Hence, there are two challenging dimensions: 1) reducing resource use in absolute numbers and 2) defining the actual needed level of production and consumption. Moreover, ensuring human well-being for all means that the needed level will imply a reduction for some, and an increase for others, based on their current level. Hence, sufficiency is **not** about denying the right to a decent life to those in need.

---

[2] It was first described by William Stanley Jevons with regards to the increased use of coal when coal-efficient steam machines were created.

[3] Note that it is different from maximizing QoS subject to an energy constraint.

This article has been accepted for publication in IEEE Pervasive Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MPRV.2024.3386337

Challenges

Several challenges arise when shifting to sufficiency. This section presents relevant ones for edge computing. Note that they are not necessarily specific to edge computing and can be relevant for other areas as well.

Challenge 1: Getting a comprehensive picture of resource use Edge computing evolves by interactions among several actors: end users, application developers, researchers, infrastructure owners, etc.. It is important that all of them can get a complete picture of resource use, both to be aware of own use level (e.g. in comparison to what is defined as sufficient) and to compare one's level over time when taking reduction-promoting actions. This is challenging as the resources used are diverse (hence likely to require different measuring tools), and for some actors will require insights into other actors to get a full picture [16]. For example, an end user may use different applications running on infrastructures owned by several companies. Finally, the tools for gauging the resource use should themselves not be too resource-intensive.

Moreover, the resource use picture should be accurate, otherwise there is a risk that sufficiency efforts are wrongly targeted. This may prevent an actual resource use decrease, e.g. if the use of a given type of resource is under-estimated no efforts will target it. The most accurate way is to directly measure the use, however it is not possible nor recommended to do this for every edge device, all the time. This is too costly, both from an environmental and an economic perspective. There is thus a need for good estimations to complement measurements. Providing such estimations is a challenging task. For example, reporting the energy consumption of devices is complex [16] and too often done in a naive way. Works such as the one by Ahvar et al. [17], who propose an energy model for some edge-related architectures need to be expanded. Moreover, an accurate resource use picture should encompass all the device phases, i.e. the use phase, but also the production and the end-of-life phases. How to get data for the whole lifecycle (through e.g. LCA studies) and present it in an understandable way to trigger actions is a challenging task.

Once the resource use data is gathered, the next step is to present it. Here, it is important to reflect on which metrics are used and how they help (or not) moving towards sufficiency. For example, showing the average energy consumption may have a different impact on the actions taken than showing the total energy consumption. The challenge lies in identifying the set of most impactful metrics for sufficiency for every work.

Finally, the resource use picture should handle the potential existence of rebound effects affecting resources that are not the ones usually considered. This is a challenging task as some impact may be concealed because edge computing is not an isolated system but interacts with many others.

Challenge 2: Taking responsibility The diversity of actors at the edge poses a challenge regarding responsibility. Who is actually responsible for the increasing resource use due to edge computing? And who should do something about it?

As the resource use of edge computing emerges due to a range of edge services utilized, one “obvious” answer is that the one creating the demand (i.e. end users and to some extent application developers) is responsible and should act. Indeed, it is urgent to reflect and act upon what we use edge computing for and the extent of this use. However, we argue that resource use is not the sole responsibility of the demand side and that the supply side also has a role to play. For example, by initiating the good-enough discussion with their users. The whole paradigm has to be re-thought with a sufficiency lens to achieve a significant change.

Challenge 3: Reducing the absolute resource use Building upon the comprehensive picture of resource use (see Challenge 1) for a given edge solution (e.g. an algorithm, an architecture, a device, an infrastructure, etc.) there is a need to be able to fairly compare it to alternatives to promote the one with the lowest footprint. For that, more tools and methods enabling fair comparisons have to be developed. For example, edge benchmarking is still an open-issue, although some efforts are ongoing [18]. Moreover, relevant metrics have to be identified.

Also, representative data sets have to be available. First, data should be representative in order to help focus on real issues and not miss or invent some. This is challenging, as the current lack of representative edge data sets indicates. Secondly, data should be available so that comparing different techniques is as simple as possible in order not to waste effort on reproduction. Similarly, approaches that ease reproduction, such as providing code and experimental details have to be encouraged (or enforced). Given the current low occurrence of such practices, this is also a challenging point.

Another challenge is that edge computing systems are not isolated but interact with other systems such as the cloud or the telecommunication networks. How these interactions influence the resource use and as

This article has been accepted for publication in IEEE Pervasive Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MPRV.2024.3386337

such the reduction possibilities is an open question. Especially, one should be careful that an absolute reduction of resource use at the edge does not lead to an increased resource use in an interconnected system.

Challenge 4: Defining appropriate levels of production and consumption First, it is challenging to answer honestly the question: how much edge do we *actually* need and for what purposes? As an active part of the edge community, either as researcher or practitioner, we are biased. Other external stakeholders should be involved but how to accomplish this is to be determined.

When setting the appropriate levels, the basic needs of the user should be met. While this may be rather easy to define for resources like water or food, it is not that easy for edge computing. Many edge applications have an entertainment purpose, hence what level of entertaining should be considered as the basic need level? Who should decide? Moreover, how to enforce that only basic needs are met and no more? Should this be enforced? By whom then? On the contrary, other edge applications are safety-critical and reducing their resources too much may lead to catastrophic outcomes. These are challenging topics that are typically outside the usual expertise area of people involved in edge computing.

As a part of defining these levels, performance should be redefined, when acceptable with regards to the application characteristics. Indeed, the current pursuit of *always best and better* is resource-hungry. The necessary shift from this to *good-enough* (i.e. to considering performance as a constraint and no more as a goal) will likely be challenging. Note that this does not prevent having (very) high performance constraints when required (e.g. in safety-critical use cases).

The edge computing paradigm is home to a wide range of use cases [3]. Some of them can bring us closer to a sufficient edge computing while other won't. Complex use cases such as smart waste management (where garbage containers are equipped with sensors) can exhibit both sufficient and non-sufficient dimensions. It contributes to sufficient fuel use by only targeting the place where collection is needed but at the (likely) cost of many resources used for the sensors since all containers have to be equipped. The challenge is to analyze the use cases with sufficiency lenses to focus our efforts and spend resources wisely.

Defining the appropriate levels will also be challenging due to the edge workload characteristics. Indeed, this workload is created by stationary and mobile users, and thus can vary quite drastically in space and time. The aggregate load may exhibit sudden spikes. Therefore, the defined levels have to take this into consideration. How to deal with this need for flexibility in a sufficient way is complex.

Challenge 5: Daring to face the issue and start acting The above challenges are drastically questioning the current way of doing things. Working towards sufficiency is thus likely to be facing resistance and be emotionally straining.

A resistance that likely will be encountered is about the economical dimension of sufficiency. This is also discussed in related domains [19]. Some may posit that sufficiency is never going to happen because it goes against business as usual: no company will cut its resource use if it fears this gives the competitors an advantage and/or lead to profit reductions. This highlights the need for global policies and a general change of mindset. If the planet goes under, profits will anyway be irrelevant. Moreover, since some issues have been created because of the current economic model, why should it be considered central? The necessary changes will indeed be tough but that is not a reason for looking away.

Another sort of resistance is what one may call an inner resistance: once facing that the problems are huge, it is quite natural to feel discouraged and think "Why bother? What I can do is never going to make a difference". One may also struggle with the fact that one has been contributing to the problem.

A third challenging aspect is that sufficiency requires us to reflect on more than the technological aspects, which may be outside our comfort zone. All of this may prevent us from acting, paralyzed by fear or anxiety.

In a nutshell, making the necessary shift to sufficiency is not going to be an easy path. However, recall its aim: a planet where everyone can live a decent life. Hence, we argue that sufficiency should be pursued despite any initial difficulties. A more frequent consideration of sufficiency in works will contribute to alleviating both external and inner resistances.

EDGE SUFFICIENCY TOOLKIT

How to act then? In this section, we gather a first set of tools that can be used for adding a sufficiency perspective to an edge computing work.

Checklist

The first tool is the Edge Sufficiency Checklist. It is relevant for all challenges. Inspired by the ACM

This article has been accepted for publication in IEEE Pervasive Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MPRV.2024.3386337

SIGPLAN Empirical Evaluation Checklist[4], its aim is to help shifting to a sufficiency mindset and trigger a reflection around it in the context of a specific work.

Figure 2 shows a first version of an Edge Sufficiency Checklist. It contains five main statements acting as to-be-thought-of items. Each statement is accompanied by comments, questions, and examples to help understanding what to include when considering each statement. Note that using this checklist is in itself not a guarantee that the work presents a sufficient solution. Also, not all boxes may be relevant to a given work and a single work does not have to address all the potential issues lifted by using the checklist. Identifying the issues and raising awareness about the resource impact of techniques and methods is also valuable. In brief, the checklist should not be followed blindly.

- ☐ **A-Identify all the resources involved.** Think large and beyond the ones usually studied. For example: Human resources/software/hardware needed to design and/or maintain the system; Raw materials used to create the system.
- ☐ **B-Aim to get a comprehensive view of the resource use.** Focus on absolute numbers. For each identified resource, have a "total use" metric. E.g. total weight of e-waste generated over the system lifetime, total water use, total electricity consumption and not only electricity intensity.
- ☐ **C-Analyze the impact on resource use for any studied method or technique.** Is the method or technique leading to increased absolute resource use? Compile and compare the metrics (from B) for all alternatives when having to select one. E.g. when proposing a new edge scheduler include resource consumption comparison to existing ones in addition to the performance comparison.
- ☐ **D-Question the demand level.** Can the demand be reduced? Is the work creating new demand? For example, what is the lowest video quality acceptable for the target application? Can the data be pre-processed? Does the user have to get new devices to use this new service and how many?
- ☐ **E-Challenge the definition of a satisfying service level.** What is a good enough service level in this context? What is the resource use impact of increased performance? Can the user compromise on some service level metric? Which one(s)? Are differentiated service levels available and is there a tangible gain for the user that does not subscribe to extra (not strictly necessary) features?

FIGURE 2: Edge Sufficiency Checklist 1.0.

The checklist should not be seen as comprehensive and could be expanded with e.g. examples of common pitfalls and good practices. We hope that this work triggers interest into developing an improved version of this checklist.

For each resource used	
Over the given period	Total amount used
	Maximum amount used
	Share of the available amount used

TABLE 1: A metric set for sufficiency.

Metric set

The second tool is a metric set. It is needed for comparing solutions from a sufficiency perspective and is relevant for challenges 1,3,4 and 5. . Although sustainability as a whole is a very broad topic, it is common in the literature to reduce it to a few metrics. For example, when Varghese et al. [11] include sustainability in the arguments for edge computing, the only metrics mentioned are electricity consumption and the associated carbon footprint. We argue that other metrics should be included.

Table 1 presents the proposed metric set. It comprises three metrics per resource used. These three metrics are measured over a given period of time, e.g. the length of an experiment, the lifecycle of a product.

The first metric is the total amount of the resource that is used in the given period. This is the central metric as the aim of sufficiency is to be able to lower this amount. The second metric is the maximum amount used at a single point during the given period. This is useful to identify an overprovisioned system. Finally, the third metric is the share of the available resource that is used over the given period[5]. Together with the previous metric, this is of interest to reflect on how the resource is provisioned and whether it is possible to reduce this provisioning if resources are not utilized.

Two important things should be noted about the metric set. First, it is a metric **set**, not a single metric. Indeed, as sufficiency is about reducing our resource use, and since edge computing involves several type of resources (e.g. network and computation), the use of several metrics is necessary. Secondly, none of these metrics is new or complex. Measuring for sufficiency is easy!

Resource use acknowledgment

The third tool is what we call the resource use acknowledgment. It is a statement to be included in e.g. research articles about the footprint of the work

---

[4] https://www.sigplan.org/Resources/EmpiricalEvaluation/

[5] Note that this metric is only relevant for provisioned resources.

This article has been accepted for publication in IEEE Pervasive Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MPRV.2024.3386337

conducted and is relevant for challenges 1,2,3 and 5. . This is inspired by the Machine Learning (ML) CO2 Impact Calculator[6], which advocates emission reporting for ML experiments.

Similarly, we advocate a systematic reporting of the resource use in edge computing. This reporting should ideally be easily performed, to minimize the amount of resource (in the very broad sense) spent on anything else than the actual work. As far as we know, no calculator is available for non-ML experiments. A first step can instead be reporting hardware used and experimental phase duration. The main aim being to make both authors and readers aware of the extent of resource use. The focus is *not* on getting the most exact number, as long as the reporting is honest. To avoid concealing strategies, any type of ranking or reward based on what is self-reported in the resource use acknowledgment should be discouraged.

Travel guide

The last tool is a “travel” guide for researchers and practitioners wanting to shift towards sufficiency. It is relevant for challenges 4 and 5. We are well-aware that going along this way is unfortunately deviating from the current mainstream path, and it is easy to feel lost. These recommendations are not edge-specific and can be complemented e.g. by Silberman [20] or Pargman and Eriksson [21].

Start where you are. Even if one can feel powerless given the scale of the issues, we believe everyone can contribute at their level. It is not needed to become an expert in sustainability before acting. Start putting on “sufficiency glasses” in your everyday work. You are the most relevant person to incorporate sufficiency in what you do.

Target baby steps. All sufficiency work is valuable, regardless of its span. It is ok to first focus on a smaller part of the problem and not be sufficient in every sense. Small imperfect actions in the right direction are better than no action.

Help others to build on your work. Provide all the data and tools necessary for reproducing your work to avoid wasting resources. Best practices for reproducibility can be adapted from nearby fields such as cloud computing [22].

Embrace interdisciplinary work. Edge computing systems are complex systems with environmental, social and economical dimensions. As such, many disciplines can be involved. Take any opportunity to give an interdisciplinary dimension to your work. It is especially useful to identify rebound effects.

Dare to question the rules. Sufficiency is on the opposite side of what is traditionally valued such as best performance and growth. Hence, considering sufficiency implies to question the relevance of things considered “obvious” or “standard”.

Take care of yourself. Facing the current issues and acknowledging your own impact is straining. Don’t forget to refill your energy. For this purpose, joining and engaging in communities of researchers and practitioners sharing the same objective (and likely the same struggles) is very valuable.

Concrete examples of actions relevant for the edge computing community (where increased awareness is needed) are to raise the question of sustainability/sufficiency in the meetings/projects/conferences you are active in or include a discussion or study about it in your next report/article.

CASE STUDY

This case study exemplifies how the toolkit can be used.

Setup

We selected a popular edge computing simulator, iFogSim2,which is publicly available and contains example test files. We pick two (named Clustered[7] and Edgeward[8]) as examples of algorithms to be evaluated with sufficiency in mind. The two algorithms have the same goals. Both are run with 5 mobile users to have the same workload. The checklist reflections, code, result files, additional graphs, and more are available on Gitlab[9].

Metric set

To analyze how the two placement algorithms perform from a sufficiency perspective, different metrics (from the metric set presented previously) are considered. Only a selection is shown for space reasons.

Figure 3a shows that the total energy consumed over the experiment time is slightly lower for Edgeward in the case of cloud and edge devices, and the same for user devices. The reduction is 725 MJ for the edge devices (3.2%) and 134 MJ for the cloud device (4.7%). Note the importance of showing absolute numbers and

---

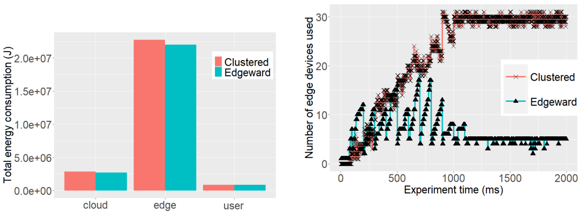
[6] https://mlco2.github.io/impact/

[7] CardiovascularHealthMonitoringApplication.java

[8] CardiovascularHealthMonitoringApplicationEdgeward.java

[9] The material will be made available upon acceptance.

This article has been accepted for publication in IEEE Pervasive Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MPRV.2024.3386337



(a) Resource: energy.

(b) Resource: edge devices.

FIGURE 3: Case study results.

not percentages only as the highest reduction in MJ corresponds here to the lowest percentage.

Figure 3b shows that the maximum number of edge devices used within a given millisecond is higher for Clustered (31 versus 19). The maximum number is reached during a resource use spike for Edgeward, whereas the resource use seems to stabilize around this maximum for Clustered. This type of insight is useful for designing a sufficient infrastructure.

Resource use acknowledgment

To create and run the experimental part of this case study, a Dell Precision 5520 with a Samsung S24D340h external screen were used during around 7 hours. The computer is assumed to be under load for this whole duration[10], with a power consumption of 100W. The external screen has an average power consumption of 18W. The approximated total energy consumption is therefore 3 MJ.

Discussion

This case study shows how the edge sufficiency toolkit can be used for the evaluation of two placement algorithms. The idea is that such evaluation can help choosing algorithms that will enable coming closer to a sufficient edge.

One first remark is that the toolkit being generic, it has to be adapted when used for a specific case. For example, given that this case study does not include experiment design, but reproduces an existing one, the last two checklist items ("question the demand" and "challenge the definition of satisfying service") are not in scope.

Secondly, there is an obvious lack of methods and tools for conducting such sufficiency studies, thus some parts of the case study were not conducted according to a standard (as there is none). However, following the travel guide presented earlier, we argue that we need to start somewhere in order to move forwards. It is easier to step-wise improve than to create something perfect from the start. We also make available all the material to ease reproduction and extension.

Thirdly, calculating or even estimating the impact of your own research is hard. For example, we could report the number of hours spent on the experiment only because we planned to report it from the start and accordingly logged it. Here as well, there is a need to create user-friendly methods and tools. Inspiration may be taken from the upcoming EU sustainability reporting standards for companies[11].

We finally acknowledge that this case study has limitations, e.g. the metric set focused on two resources but there are more involved, and the metric set overhead was not studied. Moreover, the case study focuses on the resource consumption analysis of two algorithms, but similar studies can be conducted for other aspects of edge computing. For example, comparing two physical edge architectures or different edge design approaches. Each study should tailor the sufficiency requirements to its specific use case.

CONCLUSION

In this work, we advocate the need for edge computing to shift towards sufficiency to reduce its footprint. We first discuss sustainability and why efficiency cannot alone achieve the wanted results. We then present sufficiency and associated challenges, both technical and non-technical ones. To make the shift, we propose a toolkit with a first version of four tools and exemplify one possible use of it through a case study.

We believe there is a need for more work focusing on sufficiency, both for using it when researching or deploying edge computing (and other technologies) but also as a topic of its own. We plan on further development of the toolkit and associated tools and hope for a collaborative effort both within the edge computing community and beyond.

ACKNOWLEDGMENTS

This work has been supported by the Swedish national graduate school in computer science (CUGS). We

---

[10] This is a pessimistic assumption as the experiments are fast to run (less than a minute each) and most of the time was spent programming.

[11] https://finance.ec.europa.eu/capital-markets-union-and-financial-markets/company-reporting-and-auditing/company-reporting/corporate-sustainability-reporting_en

This article has been accepted for publication in IEEE Pervasive Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MPRV.2024.3386337

thank Rodrigo Saar de Moraes for our discussions about parts of this work.

REFERENCES

1. M. Satyanarayanan, “The emergence of edge computing,” *Computer*, vol. 50, no. 1, pp. 30–39, 2017.
2. State of the Edge, State of the Edge 2020: A Market and Ecosystem Report for Edge Computing. [Online]. Available: https://www.lfedge.org/wp-content/uploads/2020/04/SOTE2020.pdf
3. C. Perera, Y. Qin, J. C. Estrella, S. Reiff-Marganiec, and A. V. Vasilakos, “Fog Computing for Sustainable Smart Cities: A Survey,” *ACM Comput. Surv.*., vol. 50, no. 3, 2017.
4. IPCC, *Climate Change 2022: Impacts, Adaptation and Vulnerability*. Cambridge, UK and New York, NY, USA: Cambridge University Press, 2022.
5. H. Ferreboeuf, M. EfouiHess, and X. Verne, Environmental impacts of digital technology: 5year trends and 5G governance. [Online]. Available: https://theshiftproject.org/wp-content/uploads/2023/04/Environmental-impacts-of-digital-technology-5-year-trends-and-5G-governance_March2021.pdf
6. M. Huesemann and J. Huesemann, *Techno-fix : why technology won’t save us or the environment*. New Society Publishers, 2011.
7. T. Santarius, J. C. T. Bieser, V. Frick, M. Höjer, M. Gossen, L. M. Hilty *et al.*, “Digital sufficiency: conceptual considerations for ICTs on a finite planet,” *Ann. Telecommun.*., vol. 78, pp. 277–295, 2023.
8. J. Rockström, W. Steffen, K. Noone, Å. Persson, F. S. Chapin, E. Lambin, *et al.*, “Planetary Boundaries: Exploring the Safe Operating Space for Humanity,” *Ecology and Society*., vol. 14, no. 2, 2009.
9. B. Nardi, B. Tomlinson, D. J. Patterson, J. Chen, D. Pargman, B. Raghavan, *et al.*, “Computing within Limits,” *Commun. ACM.*, vol. 61, no. 10, pp. 86–93, 2018.
10. C. Becker, *Insolvent: How to Reorient Computing for Just Sustainability*. MIT Press, 2023.
11. B. Varghese, E. de Lara, A. Y. Ding, C.-H. Hong, F. Bonomi, S. Dustdar, *et al.*, “Revisiting the Arguments for Edge Computing Research,” *IEEE Internet Computing.*, vol. 25, no. 5, pp. 36–42, 2021.
12. B. Ramprasad, A. da Silva Veith, M. Gabel, and E. de Lara, “Sustainable Computing on the Edge: A System Dynamics Perspective.” in *Proceedings of the 22nd International Workshop on Mobile Computing Systems and Applications (HotMobile ’21)*, 2021.
13. G. Perin, M. Berno, T. Erseghe and M. Rossi, “Towards Sustainable Edge Computing Through Renewable Energy Resources and Online, Distributed and Predictive Scheduling,” *IEEE Transactions on Network and Service Management*, vol. 19, no. 1, pp. 306–321, 2022.
14. K. Toczé, M. Madon, M. Garcia, and P. Lago, “The Dark Side of Cloud and Edge Computing: An Exploratory Study,” in *LIMITS ’22: Workshop on Computing within Limits*, 2022.
15. M. Madon and P. Lago, ““We are always on, is that really necessary?” Exploring the path to digital sufficiency in flexible work,” in *International Conference on ICT for Sustainability (ICT4S 2023)*, 2023.
16. E. J. Vergara, S. Nadjm-Tehrani, and M. Asplund, “Fairness and Incentive Considerations in Energy Apportionment Policies,” *ACM Trans. Model. Perform. Eval. Comput. Syst.*, vol. 2, no. 1, 2016.
17. E. Ahvar, A. -C. Orgerie and A. Lebre, “Estimating Energy Consumption of Cloud, Fog, and Edge Computing Infrastructures,” *IEEE Transactions on Sustainable Computing*, vol. 7, no. 2, pp. 277-288, 2022.
18. B. Varghese, N. Wang, D. Bermbach, C.-H. Hong, E. de Lara, W. Shi *et al.*, “A Survey on Edge Performance Benchmarking,” *ACM Computing Surveys*., vol. 54, no. 3, pp. 1–33, 2021.
19. J. C. de Man, J. O. Strandhagen, “An Industry 4.0 Research Agenda for Sustainable Business Models,” *Procedia CIRP*., vol. 63, pp. 721–726, 2017.
20. M. S. Silberman, “Information systems for the age of consequences,” in *First Monday*., vol. 20, no. 8, 2015.
21. D. Pargman and E. Eriksson, “Exploring Inner Transition: Expanding Computing for Sustainability,” in *LIMITS ’23: Workshop on Computing within Limits*, 2023.
22. A. V. Papadopoulos, L. Versluis, A. Bauer, N. Herbst, J. von Kistowski, A. Ali-Eldin, *et al.*, “Methodological Principles for Reproducible Performance Evaluation in Cloud Computing,” *IEEE Transactions on Software Engineering*., vol. 47, no. 8, pp. 1528–1543, 2021.

Klervie Toczé is a PhD student at Linköping university. Her current research interests include resource management, edge computing, and sustainability. Klervie received the Licentiate degree in computer science from Linköping University. Contact her at klervie.tocze@liu.se.

Simin Nadjm-Tehrani is a full professor at Linköping University. Her current research interests include dependability and cybersecurity in critical systems and infrastructures. Simin received the Ph.D. degree in computer science from Linköping University. Contact her at simin.nadjm-tehrani@liu.se.