Towards Understanding the Carbon Impact in End-to-end Sensing Pipelines

Harsh Desai*
harshd@andrew.cmu.edu
Carnegie Mellon University
Pittsburgh, USA

Sara McAllister*
sjmcalli@cs.cmu.edu
Carnegie Mellon University
Pittsburgh, USA

Nathan Beckmann
beckmann@cs.cmu.edu
Carnegie Mellon University
Pittsburgh, USA

Brandon Lucia
blucia@andrew.cmu.edu
Carnegie Mellon University
Pittsburgh, USA

Abstract

The growth of sensing devices enables a wide range of previously untenable applications from sustainable agriculture to wildlife monitoring. At the same time, this growth necessitates considering the sustainability impact of these devices. Such devices capture sensor data, process it locally and radio-transmit it to the cloud via internet-enabled basestations. While some prior work has begun inspecting emissions at a device level, we need to understand the carbon impact of the entire sensing pipeline — from the sensing device to the cloud. Simply focusing on the device leaves several end-to-end impacts unexplored, both negative and positive. In this paper, we describe this end-to-end view of the sensing pipeline and show how one design decision in the sensing device affects the *entire* pipeline's carbon impact.

1 Introduction

Wireless embedded sensor devices enable a wide range of important applications, from sustainable agriculture [5] to wildlife monitoring [20] and urban sensing [7]. These devices collect information from their environments, using on-device sensors and microcontrollers, while drawing power from a battery or harvested energy. The past decade has seen such devices becoming cheaper and more compact, with increasing hardware-software support from academia [7, 10, 19] and industry alike [4]. A recent report [15] predicts that there will be a trillion sensor devices deployed by 2035.

Unfortunately, as sensor deployments grow in size, their environmental impact becomes an important factor to consider. Given the pressing need for minimizing carbon emissions across computing domains [13, 17, 18], *we must study the carbon footprint in deploying and maintaining sensing devices for an ethical and sustainable future.* Few emerging efforts have begun exploring this issue, providing carbon models for sensor maintainence [23], desiging devices to 'transient' [6] and considering recycled components when designing circuits [21]. Such efforts mainly focus on individual sensor devices.

However, optimizing just sensing devices' carbon leaves out large pieces of the *sensing pipeline* needed to understand the end-to-end carbon impact of sensing. More specifically, once sensing devices collect data they need to transmit it using an on-device radio through a communication layer to a cloud-based datacenter. The data movement and subsequent cloud operations are an important part of sensing's carbon overhead, particularly since datacenter sustainability is simultaneously garnering increasing attention [13, 22]. However, these datacenter optimizations are also happening without knowledge of the rest of the pipeline [1, 24, 26].

In this work, we draw attention to the end-to-end sensing pipeline, and how policy decisions made at any point in the pipeline can have an impact on sustainability across the pipeline. We assert that *sustainability research in sensing deployments will be effective and ethical when it accounts for such end-to-end impact.*

This end-to-end impact can manifest either negatively — losses we need to avoid — or positively — opportunities we can exploit. As an example of negative impact, a sensing device might save its battery by offloading data processing onto the cloud, a strategy that elongates battery life and reduces carbon costs of battery replacements. However, the offloaded processing engages computing and storage on the datacenter, incurring additional carbon emissions. This might lead to an overall loss in terms of carbon impact; avoiding such negative impact requires careful characterization and modeling of cross-pipeline effects and dynamic systems that can globally optimize for carbon impact.

On the positive side, consider a sensing device which uses on-device machine learning (ML) to identify and transmit only the inputs that are interesting to the application. Such a device may also transmit the prediction scores along with 'interesting' images, so that a datacenter scheduler can use them to perform carbon-aware scheduling. An example of carbon-aware scheduling could be where high-confidence 'interesting' images are processed with low-latency, while low-confidence 'interesting' images are processed in lower-priority batches.

* Both authors contributed equally to this work.

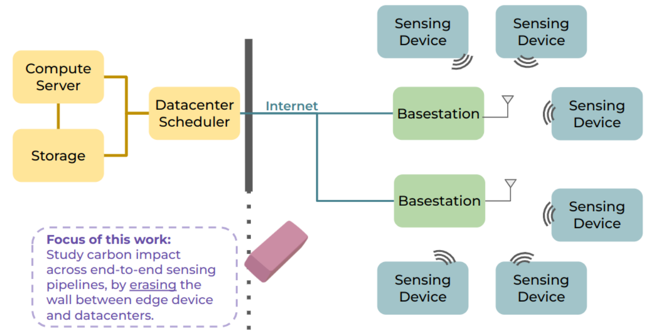

Figure 1. Figure shows a sensing pipeline where sensor devices collect and transmit data to internet-enabled basestations, where the data is exposed to the end user via the cloud-based datacenter. We assert that sustainability efforts targeting sensing pipelines should account for end-to-end impact, instead of focusing on separate segments of the pipeline.

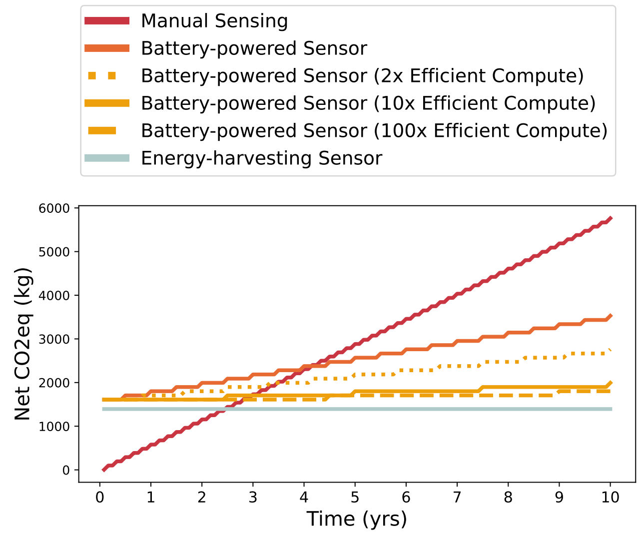

Figure 2. Figure shows the net carbon impact for a smart agriculture application, comparing different methods of capturing sensor data (temperature, humidity). We observe that the using efficient computing methods, and/or replacing the battery with harvested energy reduces the maintainence costs (in terms of carbon) associated with battery replacements.

Through this paper, we present a call-to-arms for sustainability efforts to focus on end-to-end impact, as opposed to focusing on separate segments of the sensing pipeline. We first present the sensing pipeline (section 2) and then discuss a case study: adding on-device ML to sensing devices (section 3).

2 End-to-end Sensing Pipelines

We believe that the carbon footprint of sensing devices needs to be taken holistically: from the sensing device deployment to the communication layer between the sensing devices and data center to the offloaded compute and storage in the datacenter. Changes to deployments and sensing algorithms can cause effects back to the datacenter, both increasing or decreasing the sustainability of this entire *sensing-pipeline*. As seen in Figure 1, there are three main components of our sensing pipeline: the sensing devices, the communication layer, and the datacenter. Each component causes carbon emissions.

Sensing devices. The sensing pipeline starts with the sensor device deployments. Once deployed, these devices then operate their sensors to collect data, often do preliminary data analysis, and eventually send that data back to the cloud. Sensing, processing data, and sending data all require energy — a limited resource on sensing devices that are reliant on battery capacity or harvesting energy from the surrounding [7, 9, 16]. These devices have inherent carbon emissions from manufacturing and deploying the devices, but also embody additional carbon each time the battery has to be replaced.

Communication Layer. When the sensor collects data, it has to forward that data to be processed and permanently storage through a communication layer. A common architecture of this layer is for individual sensors to wirelessly send their updates to a base station that is physically close to each of the servers [8]. The basestation is connected to the internet and forwards the data to the desired endpoint — typically the datacenter. In this part of the pipeline, emissions comes from creating and running the basestation and the internet infrastructure — which needs to be appropriately accounted for between different internet users.

Datacenter. Once the data arrives at a datacenter, the data is routed to server to injest that data into a database or storage system. At the same time, for time-sensitive applications, this data will be forwarded to a stream-analytics framework and be added to batch analytics [3, 25]. Ultimately, these tasks require time on a compute server, whether for real-time processing or batch processing, and space in the distributed storage system. Importantly, the carbon emissions overheads here are dependent on the amount of data coming in.

3 Case Study: Sensor nodes with ML

We now present a case study to demonstrate the importance of holistic, end-to-end sustainability analysis when it comes to sensing pipelines. We consider a smart agriculture application [5] as the reason to deploy sensor devices. The application requires periodic temperature and humidity information from several locations along a 100 sq. mile farm.

Choice of sensing technology. The first important choice for this application is how we are deploying sensors, showing both the importance of deploying sensors and optimizing the carbon emissions of these sensors. Figure 2 shows the net carbon emissions from the smart agriculture application, for several methods of sensing.

We compare three systems: manual sensing where a monitoring person drives to each location and logs the required information, a battery-powered sensor deployment, and an energy-harvesting sensor. Manual sensing accrues carbon emissions as the user has to drive around to each location periodically (once in two months in our example). We compare this with a battery-powered sensor, similar to the one proposed in [23], deployed at each location. Each device is equipped with a Ambiq Apollo 4 microcontroller [2], a temperature+humidity sensor and a LoRa radio [14], all powered using three AA batteries (2800mAh). The device captures one-minute-long data once every five minutes, sending an averaged digest once every hour over the radio. Deploying multiple battery-powered devices incurs embodied carbon at deployment time, and incurs periodic carbon emissions when the user has to drive to each location for replacing batteries.

We see that even with these carbon overheads, deploying sensors allows capturing significantly finer-grained information, while reducing carbon emissions in the longer run compared with manual sensing.To further optimize the carbon emissions for sensing devices, one can consider improving the efficiency of on-device compute (using newer architectures like [10]) and/or entirely replacing the battery with energy harvester (e.g. solar cells like in [7]). Using more efficient compute allows the battery to last longer, resulting in fewer battery replacements and associated carbon emissions.

Even these optimizations do not approach the carbon emissions from harvested energy, which eliminate batteries completely including the embodied and periodic carbon emissions associated with batteries. Operating sensing devices on harvested energy is an emerging field of on-going research, and needs further developments to be reliable and commercially viable.

On-device ML. Prior work [7, 11, 16] has proposed using on-device machine learning (ML) techniques to identify and transmit only the data that is ‘interesting‘ to the application. Uninteresting data is to be discarded. On-device ML significantly reduces the data transmitted from the device. If the device is bottlenecked by radio energy, on-device ML improves overall energy-efficiency of the device and causes the battery to last longer – a sustainability win. If the device is bottlenecked by compute energy, on-device ML might worsen the energy-efficiency of the device and cause the battery to drain faster – an apparent sustainability loss.

If a sustainability-aware system designer were to focus only on the device, they might determine that on-device ML is bad for the environment in some cases. However, this argument fails to account for the carbon emissions caused on the datacenter side for uninteresting data. Figure 3 shows the carbon emissions [12] associated with storing sensor data (images in our example). We study the emissions for storing data on Flash drives and Hard Disk Drives (HDDs), where the sensor device does not perform on-device ML. We also study a case where the sensor device only sends ‘interesting‘ images, which are then stored on HDDs in the datacenter. We observe on-device ML can provide large wins on the datacenter side, by using significantly less storage and leading to longer storage lifetimes. *These opportunities can only be found by studying the end-to-end carbon impact in sensing pipelines to determine if a local policy benefits sustainability or not.*

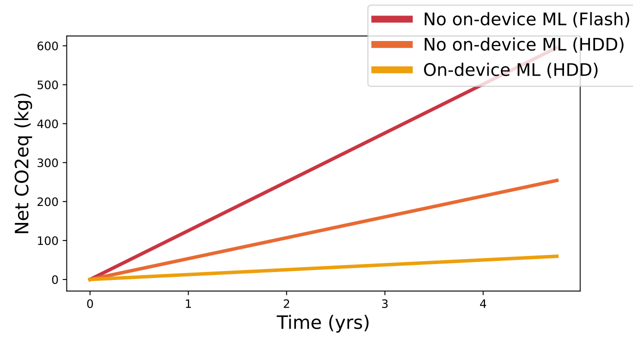

Figure 3. Figure shows the embodied carbon impact for storing sensor data (example shows images collected once per second). We observe that using ML on the sensing devices makes has many downline impacts on the carbon emissions of the pipeline — including cutting emissions and allowing better storage choices.

Holistic-impact studies inspiring research directions. Along with determining whether a sustainability policy is beneficial or not, studying the carbon impact across end-to-end sensing pipelines also can open new avenues for research. As an example, if the sensor device transmits ‘interesting‘ data along with its associated prediction scores, the datacenter scheduler could process the received data in different ways based on the received scores. Data indicated as ‘interesting‘ with low-confidence could be relegated to latency-insensitive batch processing for generating larger carbon wins. Such data annotations could also inform intelligent caching on the storage side, where low-probability ‘interesting‘ data could be stored on carbon-cheaper storage leading to sustainability wins.

Another line of research investigation could look at distributing applications across sensor nodes in a carbon-efficient manner. The activity on each individual sensor node might be dynamically variable, depending on its deployment environment. For example, sensors tracking cattle might see different amounts of activity depending on where they are deployed. Further, each device might have a different amount of resources at any given time (compute capability, battery charge, etc). The datacenter could use such activity statistics and per-device hetergeneity to inform device-level policies, optimizing sustainability across the entire sensing pipeline.

4 Conclusion

In this paper, we presented a call-to-arms for researchers to focus on the end-to-end sustainability impact in sensing pipelines. We presented examples motivating the deployment of sensor devices, along with unintuitive conclusions that can be drawn when studying the entire sensing pipeline. We presented some directions for future research that can exploit this end-to-end impact. By making sustainability research efforts more holistic in sensing pipelines, researchers can avoid unintended worsening in carbon emissions. Studying end-to-end impact will also unlock new research and smarter carbon-aware policies, resulting in a more ethical and sustainable approach to sensing.

References

[1] Bilge Acun, Benjamin Lee, Fiodar Kazhamiaka, Kiwan Maeng, Udit Gupta, Manoj Chakkaravarthy, David Brooks, and Carole-Jean Wu. Carbon explorer: A holistic framework for designing carbon aware datacenters. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, ASPLOS 2023, page 118–132, New York, NY, USA, 2023. Association for Computing Machinery.

[2] Ambiq. Apollo 4. https://ambiq.com/apollo4/.

[3] AWS. How AWS IoT works. https://docs.aws.amazon.com/iot/latest/developerguide/aws-iot-how-it-works.html.

[4] Thomas Ball, Abhijith Chatra, Peli de Halleux, Steve Hodges, Michal Moskal, and Jacqueline Russell. Microsoft makecode: Embedded programming for education, in blocks and typescript. In *SPLASH-E 2019: Proceedings of the 2019 ACM SIGPLAN Symposium on SPLASH-E*, pages 7–12. ACM, ACM, October 2019.

[5] Ranveer Chandra, Swami Manohar, Tusher Chakraborty, Jian Ding, Zerina Kapetanovic, Peeyush Kumar, and Deepak Vasisht. Democratizing data-driven agriculture using affordable hardware. *IEEE Micro*, 42(1):69–77, January 2022.

[6] Tingyu Cheng, Gregory D. Abowd, Hyunjoo Oh, and Josiah Hester. Transient internet of things: Redesigning the lifetime of electronics for a more sustainable networked environment. In *Proceedings of the 2nd Workshop on Sustainable Computer Systems*, HotCarbon '23, New York, NY, USA, 2023. Association for Computing Machinery.

[7] Harsh Desai, Matteo Nardello, Davide Brunelli, and Brandon Lucia. Camaroptera: A long-range image sensor with local inference for remote sensing applications. *ACM Trans. Embed. Comput. Syst.*, 21(3), may 2022.

[8] Adwait Dongare, Craig Hesling, Khushboo Bhatia, Artur Balanuta, Ricardo Lopes Pereira, Bob Iannucci, and Anthony Rowe. Openchirp: A low-power wide-area networking architecture. In *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pages 569–574, 2017.

[9] Francesco Fraternali, Bharathan Balaji, Yuvraj Agarwal, Luca Benini, and Rajesh Gupta. Pible: battery-free mote for perpetual indoor ble applications. In *Proceedings of the 5th Conference on Systems for Built Environments*, BuildSys '18, page 168–171, New York, NY, USA, 2018. Association for Computing Machinery.

[10] Graham Gobieski, Souradip Ghosh, Marijn Heule, Todd Mowry, Tony Nowatzki, Nathan Beckmann, and Brandon Lucia. Riptide: A programmable, energy-minimal dataflow compiler and architecture. In *2022 55th IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pages 546–564, 2022.

[11] Graham Gobieski, Brandon Lucia, and Nathan Beckmann. Intelligence beyond the edge: Inference on intermittent embedded systems. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '19, page 199–213, New York, NY, USA, 2019. Association for Computing Machinery.

[12] Udit Gupta, Mariam Elgamal, Gage Hills, Gu-Yeon Wei, Hsien-Hsin S. Lee, David Brooks, and Carole-Jean Wu. ACT: designing sustainable computer systems with an architectural carbon modeling tool. In *Proceedings of the 49th Annual International Symposium on Computer Architecture*. ACM, 2022.

[13] Udit Gupta, Young Geun Kim, Sylvia Lee, Jordan Tse, Hsien-Hsin S. Lee, Gu-Yeon Wei, David Brooks, and Carole-Jean Wu. Chasing carbon: The elusive environmental footprint of computing. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, pages 854–867, 2021.

[14] HopeRF. RFM95W LoRa Module. https://www.hoperf.com/data/upload/portal/20190801/RFM95W-V2.0.pdf, 2019.

[15] ITU. ARM predicts 1 trillion IoT devices by 2035 with new end-to-end platform. https://www.itu.int/hub/2020/04/arm-predicts-1-trillion-iot-devices-by-2035-with-new-end-to-end-platform/.

[16] Neal Jackson, Joshua Adkins, and Prabal Dutta. Capacity over capacitance for reliable energy harvesting sensors. In *The 18th International Conference on Information Processing in Sensor Networks*, IPSN'19. ACM, April 2019.

[17] Nicola Jones et al. How to stop data centres from gobbling up the world's electricity. *Nature*, 561(7722):163–166, 2018.

[18] Bran Knowles. Acm techbrief: Computing and climate change, 2021.

[19] Christopher Kraemer, Amy Guo, Saad Ahmed, and Josiah Hester. Battery-free makecode: Accessible programming for intermittent computing. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 6(1), mar 2022.

[20] Ting Liu, Christopher M. Sadler, Pei Zhang, and Margaret Martonosi. Implementing software on resource-constrained mobile sensors: Experiences with impala and zebranet. In *Proceedings of the 2Nd International Conference on Mobile Systems, Applications, and Services*, MobiSys '04, pages 256–269, New York, NY, USA, 2004. ACM.

[21] Jasmine Lu, Beza Desta, K. D. Wu, Romain Nith, Joyce E Passananti, and Pedro Lopes. ecoeda: Recycling e-waste during electronics design. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, UIST '23, New York, NY, USA, 2023. Association for Computing Machinery.

[22] Jialun Lyu, Jaylen Wang, Kali Frost, Chaojie Zhang, Celine Irvene, Esha Choukse, Rodrigo Fonseca, Ricardo Bianchini, Fiodar Kazhamiaka, and Daniel S. Berger. Myths and misconceptions around reducing carbon embedded in cloud platforms. In *Proceedings of the 2nd Workshop on Sustainable Computer Systems*, HotCarbon '23, New York, NY, USA, 2023. Association for Computing Machinery.

[23] Pol Maistriaux, Thibault Pirson, Maxime Schramme, Jérôme Louveaux, and David Bol. Modeling the carbon footprint of battery-powered iot sensor nodes for environmental-monitoring applications. In *Proceedings of the 12th International Conference on the Internet of Things*, IoT '22, page 9–16, New York, NY, USA, 2023. Association for Computing Machinery.

[24] Sara McAllister, Benjamin Berg, Julian Tutuncu-Macias, Juncheng Yang, Sathya Gunasekar, Jimmy Lu, Daniel S. Berger, Nathan Beckmann, and Gregory R. Ganger. Kangaroo: Caching billions of tiny objects on flash. In *ACM SOSP*, 2021.

[25] Microsoft Azure. Condition monitoring for industrial IoT. https://learn.microsoft.com/en-us/azure/architecture/solution-ideas/articles/condition-monitoring.
[26] Jaylen Wang, Udit Gupta, and Akshitha Sriraman. Peeling back the carbon curtain: Carbon optimization challenges in cloud computing. In *Proceedings of the 2nd Workshop on Sustainable Computer Systems*, HotCarbon '23, New York, NY, USA, 2023. Association for Computing Machinery.