A First-Order Model to Assess Computer Architecture Sustainability

Lieven Eeckhout (Ghent University)

Abstract—Sustainability in general and global warming in particular are grand societal challenges. Computer systems demand substantial materials and energy resources during production and usage. A key question is how computer architects can design sustainable computer systems. Given the inherent data uncertainty, this paper proposes a deliberately simple first-order model to assess computer architecture sustainability based on first principles. Several case studies illustrate the insight that the model provides as well as its broad applicability.

1 Introduction

As the world population and average affluence continues to increase, the world-wide demand for resources, both materials and energy, continues to grow. The extraction of raw materials, the manufacturing of products, transportation, usage, and finally depletion and recycling requires huge amounts of energy, most often provided by fossil fuels. This in turn leads to global warming and climate change as a result of increased greenhouse gas (GHG) emissions.

Information and communication technology (ICT), due to its demand for raw materials and as a result of carbon emissions during the manufacturing and use of electronic devices, has a non-negligible impact on sustainability. A recent study reports that ICT contributes for about 2.1 to 3.9% of the world's global greenhouse gas (GHG) emissions, which is likely to increase (substantially) in the near future [4].

A key question that arises is how and to what extent computer architects can take sustainability into account when designing computer systems. Our community just recently embarked on this endeavor. Gupta et al. [6] provide a comprehensive characterization of the environmental footprint of computing for both mobile devices, always-connected computers, and hyperscale datacenters. As a follow-up, the ACT model [7] was proposed for computer architects to analyze a computer system's sustainability at design time. The ACT model relies on detailed numbers from industry processes regarding various steps during production. While this is an important step for our community, the ACT model's overall accuracy is limited by the "*lack of up-to-date carbon emission data for the latest compute [...] technologies*". Furthermore, the ACT model hopes to "*encourage industry to publish more detailed carbon characterizations to standardize carbon footprint accounting*" [7]. Given the lack of detailed up-to-date numbers, this work takes a different approach by proposing a first-order model to provide insight, intuition and guidance for computer architects in research and early-stage development of individual chips. This model is envisioned as a useful complement to a more detailed model such as ACT.

The first-order model proposed in this work embraces the inherent uncertainty to model the environmental footprint by being deliberately simple and flexible. The model is based on first principles, using proxies for the embodied footprint due to manufacturing, and the operational footprint during device usage. The proxies relate to what computer architects have control over at design time, i.e., the proxy for embodied emissions is chip area, while the proxy for operational emissions is energy and power consumption assuming a fixed-work and fixed-time scenario, respectively. The model further includes a parameter to weigh the relative importance of the embodied versus operational footprint to account for variation in product use and lifetime, and to anticipate the effect of the infamous rebound effect. The first-order model provides insight into how early-stage design decisions affect sustainability, as we illustrate through several case studies to demonstrate the value and broad applicability of the model.

2 Motivation

Developing a detailed sustainability model for computer architects to steer the design process is extremely complex and involved. There is inherent uncertainty in modeling the environmental footprint due to data limitations and various unknowns. While some contributing factors are known and can be accounted for, such as the use of materials and energy, as well as the amount of chemicals and gases emitted during manufacturing, others are unknown, or at least, there is substantial uncertainty about the specific values for each of the contributing factors. This is the case for both the embodied as well as the operational footprint. In particular, not all stages of the manufacturing process are documented at the same desired level of detail, if at all. As an example, a recent study by Imec [5] uses approximations to quantify the production footprint. Further, Life Cycle Assessment (LCA) reports use industry averages when parameters are unknown for the specific production process, see for example Apple's iPhone 12 [2].

There is even more uncertainty when it comes to the operational footprint. The degree and intensity of use of a device is simply impossible to assess at design time. The operational footprint depends on the user, the intensity of use, product lifetime, and geographic location of the user which determines the power grid mix. Operational footprint hence needs to be estimated using historical data for similar products. To make things worse, improving the efficiency of a device at design time often has the unintended side-effect of a rebound effect, also known as Jevons' paradox [1], which essentially means that an improvement in efficiency leads to an increase in demand and/or usage, which ultimately leads to a net increase in the environmental impact which the designer originally intended to reduce. For example, a device that is cheaper to manufacture because it uses less materials and energy during production, may be sold at a lower price, which may stimulate its sales, ultimately leading to a larger overall embodied footprint. Likewise, a computer system that is more efficient in terms of performance, power or energy could stimulate its use, up to the point that the overall operational footprint increases.

In conclusion, there is inherent data uncertainty. We hence opt for a first-order model that is built upon first principles so that computer architects can gain insight and reason about sustainability implications at early stages of the design cycle without being tampered with inaccurate and missing data, and unknown and unintended side-effects and parameters.

3 First-Order Model

The first-order model uses proxies for the embodied and operational footprints, as well as a parameter to denote the relative importance of the embodied versus operational footprint.

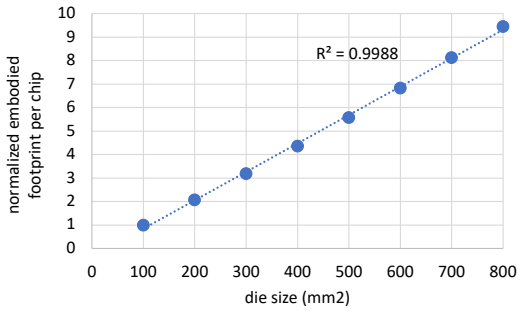


Fig. 1: Embodied footprint per chip as function of die size.

3.1 Embodied Footprint

The proxy for the embodied footprint is chip area. The unit of production in a semiconductor fabrication plant is a wafer, which is then sliced up to obtain individual chips. Producing a single wafer involves the usage of a plethora of materials, some of which are rare earth materials, come from politically instable regions in the world and/or require huge amounts of energy during the extraction process. Producing a wafer also incurs huge amounts of energy. Imec [5] recently published the amount of energy needed for a range of CMOS technology nodes from 28 nm to 3 nm. The annual growth rate in energy needs is estimated to be around 11.9%, which is a result for increased complexity with increasing number of process steps, increasing number of metal layers, new extreme ultraviolet lithography (EUV) equipment, etc. In addition, the production process also leads to emissions of chemicals and gases including fluorinated compounds (e.g., SF_6, NF_3 and CF_4, among others), which is estimated to increase by 9.3% per year. Finally, the semiconductor manufacturing process requires increasing amounts of ultra-pure water consumption, which is estimated to increase by 8.0% per annum.

What we, computer architects, have control over when it comes to embodied footprint is chip size. A big chip (large die size) means fewer chips per wafer, and thus a larger embodied footprint per chip. In contrast, a small chip means more chips per wafer, which implies a smaller per-chip embodied footprint. The number of chips one can obtain per wafer, and the embodied footprint per chip, thus depends on the chip's die size. de Vries [3] provides a formula that empirically derives the number of chips per wafer CPW as a function of die size A:

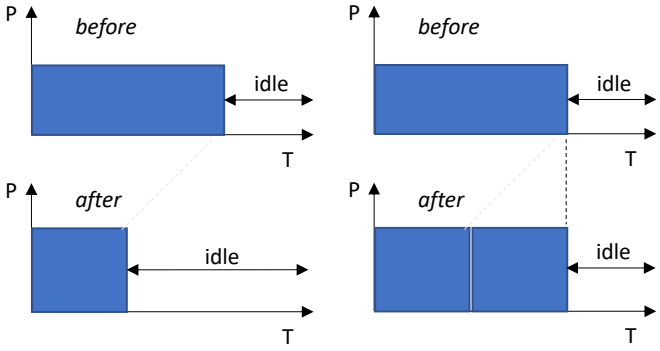
$$CPW = \frac{\pi d^2}{4A} - 0.58\frac{\pi d}{\sqrt{A}},$$

with d the wafer's diameter (e.g., 300 mm^2). Since the unit of fabrication is a wafer, the embodied footprint per chip is hence inversely proportional to *CPW*. Figure 1 shows the embodied footprint per chip as a function of die size in the region of practical concern, up to 800 mm^2 (approximately the reticle limit) and normalized to 100 mm^2. We conclude that the embodied footprint per chip is approximately linearly proportional to a chip's die size. The fact that bigger chips lead to lower yield and thus a higher embodied footprint, further enforces the observation that the embodied footprint of a chip strongly correlates with its size. We hence use chip size A as a proxy for the embodied footprint per chip.

3.2 Operational Footprint

The operational footprint relates to the total energy consumed by a device over its entire lifetime. We consider two scenarios: a fixed-work scenario and a fixed-time scenario.

The *fixed-work* scenario assumes that a device performs a fixed amount of work during its entire lifetime, see Figure 2(a). In this scenario, the total operational footprint equals the total energy consumed to get the work done. (We assume that the device is turned off when not in use, or its idle power is negligible.) It is important to note here that total energy consumption is critical, which is the product of (average) power consumption and execution time. This implies that a high-performance computer system may have a smaller operational footprint than a lower-performance system, even if it consumes more power: energy consumption is reduced if the performance improvement outweighs the increase in power consumption, as in the example given in Figure 2(a).



(a) fixed-work (b) fixed-time

Fig. 2: Operational footprint relates to (a) energy under a fixed-work scenario and (b) power under a fixed-time scenario.

The fixed-work scenario makes the (simplifying) assumption that the amount of work done by a device is fixed for its entire lifetime, e.g., a smartphone or server performs the same of work as a similar product from, say, four years ago. This is not always what happens in practice. As aforementioned, as devices become more efficient, demand increases — i.e., the infamous rebound effect. Hence, it is not unexpected that an optimization of some sort (be it a performance, energy or power optimization) leads to additional usage of the device. In particular, a performance optimization will likely incentivize the user to issue even more work, simply because it is possible, and because it has become cheaper to do so. The result is that the device is used more intensively. The *fixed-time* scenario assumes that a more efficient device is used for the same duration of time as a less efficient device, see Figure 2(b). This implies that the more efficient device performs more work. Because time is constant under the fixed-time scenario, the device's total lifetime energy consumption is proportional to its power consumption.

Based on the above reasoning, we consider two proxies for operational footprint, namely energy and power. Energy is the proxy when assuming that the amount of work is fixed, while power is the proxy when assuming that time is fixed. Which scenario is most representative depends on the use case of the computer (sub)system under consideration. An example of a fixed-work scenario is a video decoder that decodes a fixed number of frames per unit of time. An example of a fixed-time scenario is an always-on network interface. In such cases, operational footprint can and should be approximated with its respective proxy. However, many other practical examples do not strictly fall under these two scenarios. In particular, a more efficient computer system may incentivize more frequent use. It is hence important to consider operational footprint under both scenarios as the typical use case might not be known ahead of time. In other words, depending on the degree of the rebound effect, the fixed-time scenario might be more appropriate. Note that the rebound effect could lead to an even larger increase in the amount of work being done, reaching well beyond the fixed-time scenario, i.e., the more efficient device is used for an even longer duration than the original device. If so, the relative weighting of the operational footprint relative to the embodied

footprint should be increased, as we discuss next.

3.3 Embodied versus Operational Footprint

To assess a computer system's total footprint one has to weigh the embodied and operational footprint. The relative importance of the embodied versus operational footprint is hard to assess though at design time as it depends on a number of factors. For one, the ratio varies across devices. Gupta et al. [6] conclude that embodied emissions dominate for battery-powered devices, e.g., smartphone, smart watches, tablets. On the contrary, operational emissions dominate for always-connected personal devices, e.g., desktop computers, game consoles, speakers. In the datacenter, due to the transition towards green energy sources for empowering the IT and cooling equipment, embodied emissions tend to dominate.

Further, the ratio also depends on the lifetime of the device. The longer the lifetime of the device, the higher weight the operational footprint carries in the total footprint, and the less significant the embodied footprint is. Moreover and as alluded to before, the rebound effect may increase the usage of more efficient devices, possibly increasing the relative importance of operational emissions. The fixed-time scenario discussed in the previous section assumes that a more efficient device is used for the same amount of time as a less efficient device. If the time spent on the device increases even further, this implies that even more work is done during the lifetime of the device compared to a fixed-time scenario.

Finally, whether green energy sources are used during product manufacturing and/or product lifetime also affects the relative ratio of the embodied versus operational footprint. Note that even if manufacturing would be done using only green energy, the embodied footprint still incurs a substantial environmental impact as a result of the materials used and the chemicals and gases emitted during manufacturing. As is clear from the above discussion, it is non-trivial, if at all possible, to assess the relative importance of embodied versus operational emissions at early stages of the design cycle. We hence capture the relative importance of the embodied versus operational footprint as a parameter in the model.

3.4 Putting It Together: Overall Model

Based on the above discussion, the first-order model for a computer system's total environmental footprint F proposed in this work considers two scenarios. Under the fixed-work scenario, the environmental footprint F equals:

$$F_{fixed-work} = \alpha \cdot A + (1-\alpha)E,$$

with A chip area, E energy consumption, and α the embodied-to-operational weight. Under the fixed-time scenario, the environmental footprint F equals:

$$F_{fixed-time} = \alpha \cdot A + (1-\alpha)P,$$

with P power consumption. Again, which of the two scenarios to consider and how to set the α parameter depends on the anticipated use case. In many practical situations, the exact use case is unknown, and yet we need to make a holistic assessment about the sustainability of a computer system under design in terms of both embodied and operational footprint, while being considerate of potential rebound effects. It is hence advised to consider multiple scenarios and ranges of α's to understand the sustainability impact of a particular design in light of the inherent data uncertainty, as demonstrated next.

It is worth noting that how the first-order model accounts for the embodied and operational footprint is similar to ACT [7]. However, there are important differences. Besides the fact that ACT relies on detailed carbon emission numbers, as mentioned before, another key difference is that the first-order model includes a fixed-time scenario to account for the rebound effect (Jevons' paradox) unlike ACT which implicitly assumes a fixed-work scenario. In addition, the first-order model allows for easily exploring the impact of inherent data uncertainty on sustainability by varying the α parameter.

4 Case Studies

We now consider a number of case studies to illustrate the applicability and usefulness of the model.

4.1 Die Shrink

In the first case study we consider a die shrink, i.e., we implement an existing design in a new next-generation chip technology, while considering both classical and post-Dennard scaling [12]. A die shrink reduces chip area by 50% from one technology node to the next. The reduction in chip area is offset in part by the increase in energy consumption due to manufacturing in a new tech node — a recent study reports that energy consumption increases by on average 25.2%, and the amount of chemicals and gases emitted during manufacturing increases by on average 19.5% [5]. Overall, a die shrink leads to a net reduction in embodied emissions.

When it comes to operational emissions, we make a distinction between classical scaling versus post-Dennard scaling [12]. Assuming classical scaling first, power consumption reduces by a factor 2×, and because the circuit can be clocked at 1.41× higher frequency, energy consumption is reduced by a factor 2.82×. In other words, operational emissions reduce under classical scaling, under both the fixed-work and the fixed-time scenarios. In contrast, under post-Dennard scaling, power consumption remains the same, while energy reduces by a factor 1.41×. This implies that operational emissions reduce under the fixed-work scenario while remaining unchanged under the fixed-time scenario.

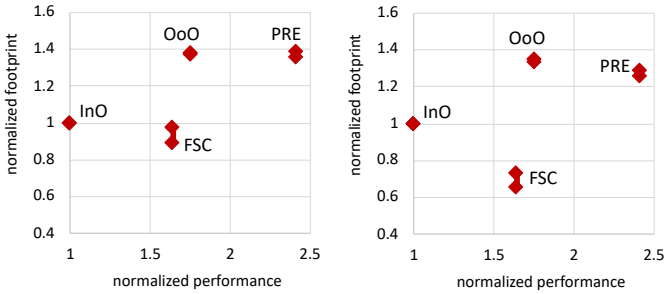
Overall, it is safe to conclude that a die shrink leads to a reduction in environmental footprint. In other words, computer systems would have become more sustainable over time if we would have leveraged Moore's Law to make our chips smaller. This is not what we have seen though in practice. Architects have used the additional transistors when moving from one technology node to the next to add functionality (e.g., more cores, more cache, more accelerators, etc.), which has led to an overall increase in environmental footprint: the increase in embodied emissions offsets the reduction in operational emissions observed as a result of energy and power optimizations.

4.2 Core Microarchitecture

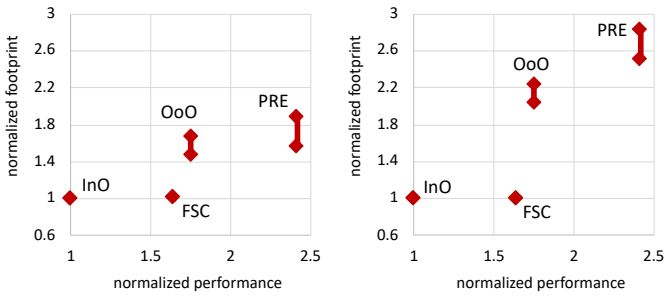
The second case study considers four microarchitectures: (1) an in-order (InO) core, (2) a Forward Slice Core (FSC) [10], (3) an out-of-order (OoO) core, and (4) an OoO core enhanced with Precise Runahead Execution (PRE) [11]. We take the chip area, power, energy and performance numbers from [10] for comparing InO, FSC and OoO; and we consider scaled numbers for PRE [11].[1] Essentially, FSC achieves a level of performance that is comparable to OoO at a small area and power overhead over InO. PRE improves performance over OoO at a small area cost but a large power cost compared to OoO.

Figure 3 reports the total footprint of these microarchitectures as a function of performance assuming a fixed-work scenario, when (a) the embodied emissions dominate (α varies from 0.7 to 0.9), and (b) the operational emissions dominate (α varies from 0.1 to 0.3). Subfigures (c) and (d) report similar results under a fixed-time scenario. Design points in the bottom-right are optimal, i.e., highest performance and lowest environmental footprint. These results suggest that FSC has a lower total footprint than InO, especially under a fixed-work scenario when the operational footprint dominates. The reason is reduced energy consumption over InO. Under a fixed-time

1. The baseline OoO core assumed in [11] is substantially more aggressive than the OoO core assumed in [10], hence we cannot directly compare these numbers. We use the relative area, power, energy and performance numbers here for illustrative purposes only.

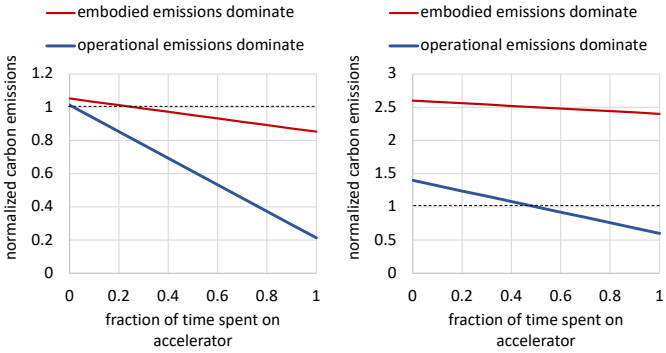


(a) fixed-work, α in $\{0.7, 0.9\}$ (b) fixed-work, α in $\{0.1, 0.3\}$



(c) fixed-time, α in $\{0.7, 0.9\}$ (d) fixed-time, α in $\{0.1, 0.3\}$

Fig. 3: Comparing the total footprint of the InO, FSC, OoO and PRE microarchitectures assuming fixed-work and fixed-time scenarios, and different embodied versus operational weights.



(a) 6.5% extra chip area (b) 200% extra chip area

Fig. 4: Total footprint of hardware specialization (normalized to OoO core) for an accelerator that incurs (a) 6.5% extra chip area and (b) twice as much chip area.

scenario, FSC's footprint is similar to InO while achieving higher performance, hence it is still the preferred architecture.

An interesting observation is to be made when comparing PRE versus OoO. Assuming a fixed-work scenario, PRE incurs a (slightly) lower footprint than OoO, especially when operational emissions dominate, and hence it is the preferred microarchitecture. However, under a fixed-time scenario, PRE incurs a significantly worse footprint than OoO. In other words, increased work as a result of the system being more efficient (i.e., Jevons' paradox), leads to an overall increase in operational emissions due to its higher power consumption. This leads to the conclusion that PRE is not strictly better than OoO, but simply a different trade-off point, i.e., higher performance at the cost of a higher environmental impact.

4.3 Hardware Specialization and Dark Silicon

Hardware specialization and accelerators are widely seen as a way to continue to improve performance in a power- and energy-efficient way in the post-Dennard era. As an example, Hameed et al. [8] propose an H.264 accelerator that incurs 6.5% extra chip area when delivering similar performance and consuming 500× less energy compared to an out-of-order core. Figure 4(a) reports the total footprint of the OoO core with the accelerator, normalized to the OoO core without the accelerator, as a function of the fraction of time spent on the accelerator when embodied emissions dominate ($\alpha = 0.8$) and operational emissions dominate ($\alpha = 0.2$). The total footprint goes down as the accelerator is used more intensively, i.e., the reduced operational footprint amortizes the increased embodied footprint of the accelerator. It is interesting to note that when the embodied emissions dominate (which appears to be the case in mobile devices [6]), it is particularly important that the accelerator be used intensively to amortize the increased embodied footprint.

A modern-day processor is a system-on-chip (SoC) featuring a number (tens) of accelerators [9]. Not all accelerators can be powered on all the time due to power constraints — a phenomenon called dark silicon [12]. Figure 4(b) reports the same data as Figure 4(a) with one important difference, namely that the accelerator occupies two thirds of the entire chip — we still assume that an accelerator consumes 500× less energy for the same level of performance. If the embodied footprint dominates (again, the likely case today [6]), it is clear that hardware specialization leads to a substantial increase in total environmental footprint. If the operational footprint dominates, the accelerators should be used intensively to amortize the embodied footprint to reduce the overall footprint.

5 Conclusion

Given its major impact on society, it is our responsibility as computer architects to take the environmental footprint into account when designing computer systems. This paper proposed a simple model based on first principles to drive computer architecture design decisions while considering both embodied and operational emissions. The model is deliberately simple to enable and encourage insight and intuition while facing significant degree of uncertainty at early stages of the design in terms of the materials used, the production process, as well as the energy usage during a computer system's lifetime. The usefulness and broad applicability of the model was illustrated through several case studies.

References

[1] B. Alcott. Jevons' paradox. *Ecological Economics*, 54(1), 2005.
[2] Apple. iPhone 12 product environmental report, 2020. URL https://www.apple.com/environment/pdf/products/iphone/iPhone_12_PER_Oct2020.pdf.
[3] D. K. de Vries. Investigation of gross die per wafer formulas. *IEEE Transactions on Semiconductor Manufacturing*, 18(1):136–139, 2005.
[4] C. Freitag, M. Berbers-Lee, K. Widdicks, B. Knowles, G. S. Blair, and A. Friday. The real climate and transformative impact of ICT: A critique of estimates, trends, and regulations. *Patterns*, 2(9), 2021.
[5] M. Garcia Bardon, P. Wuytens, L.-A. Ragnarsson, G. Mirabelli, D. Jang, G. Willems, A. Mallik, A. Spessot, J. Ryckaert, and B. Parvais. DTCO including sustainability: Power-performance-area-cost-environmental score (PPACE) analysis for logic technologies. In *IEEE International Electron Devices Meeting (IEDM)*, 2020.
[6] U. Gupta, Y. G. Kim, S. Lee, J. Tse, H.-H. S. Lee, G.-Y. Wei, D. Brooks, and C.-J. Wu. Chasing carbon: The elusive environmental footprint of computing. In *HPCA*, 2021.
[7] U. Gupta, M. Elgamal, G.-Y. W. G. Hills, H.-H. S. Lee, D. Brooks, and C.-J. Wu. ACT: Designing sustainable computer systems with an architectural carbon modeling tool. In *ISCA*, 2022.
[8] R. Hameed, W. Qadeer, M. Wachs, O. Azizi, A. Solomatnikov, B. C. Lee, S. Richardson, C. Kozyrakis, and M. Horowitz. Understanding sources of inefficiency in general-purpose chips. In *ISCA*, 2010.
[9] M. D. Hill and V. J. Reddi. Accelerator-level parallelism. *Communications of the ACM*, 64(12), 2021.
[10] K. Lakshminarasimhan, A. Naithani, J. Feliu, and L. Eeckhout. The forward slice core microarchitecture. In *PACT*, 2020.
[11] A. Naithani, J. Feliu, A. Adileh, and L. Eeckhout. Precise runahead execution. In *HPCA*, 2020.
[12] G. Venkatesh, J. Sampson, N. Goulding, S. Garcia, V. Bryksin, J. Lugo-Martinez, S. Swanson, and M. B. Taylor. Conservation cores: Reducing the energy of mature computations. In *ASPLOS*, 2010.