SuperCode: sustainability PER AI-driven co-design

Broekema, P.C.; Nieuwpoort, R.V. van; Palumbo, F.; Tumeo, A.; Varbanescu, A.; Simmhan, Y.

SuperCode: Sustainability PER AI-driven Co-design

Invited Paper

P. Chris Broekema
broekema@astron.nl
Netherlands institute for radio astronomy (ASTRON)
Dwingeloo, the Netherlands
Leiden Institute for Advanced Computer Science (LIACS)
Leiden, the Netherlands

Rob V. van Nieuwpoort
r.v.van.nieuwpoort@liacs.leidenuniv.nl
Leiden Institute for Advanced Computer Science (LIACS)
Leiden, the Netherlands

ABSTRACT

Currently, data-intensive scientific applications require vast amounts of compute resources to deliver world-leading science. The climate emergency has made it clear that unlimited use of resources (e.g., energy) for scientific discovery is no longer acceptable. To address this challenge, the use of future and emerging computing architectures promises to be much more energy efficient. However, without well optimized code these cannot reach their full potential. Effectively using emerging architectures has proven challenging due to excessive cost and time involved in porting and optimising existing code. We propose a generic AI-driven co-design methodology, using specialized Large Language Models (like ChatGPT), to effectively generate efficient code for emerging computing hardware. Instead of conventional KPI's like computational efficiency or runtime, we propose sustainability as KPI, to emphasize our commitment to do more science with fewer resources. We validate our methodology with two challenging radio astronomy use-cases, terrestrial (LOFAR, SKA) and space-based (OLFAR). The primary transverse goal of SuperCode is to reduce the environmental impact of data-intensive applications by unlocking the use of emerging efficient hardware architectures, through a novel approach of AI-driven co-design. In contrast to normal co-design, where computational performance or efficiency is used as Key Performance Indicator (KPI), we introduce a *sustainability score* instead.

We present the SuperCode project here in this form to introduce the vision behind the project and to disseminate the work in the spirit of Open Science and transparency. An additional aim is to collect feedback and invite potential collaboration partners and use-cases to join the project.

CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence**; • **Applied computing** → *Astronomy*; • **Hardware** → *Emerging technologies*; **Impact on the environment**.

KEYWORDS

AI, co-design, radio astronomy, sustainability, LLMs

CF Companion '25, May 28–30, 2025, Cagliari, Italy

ACM ISBN 979-8-4007-1393-4/2025/05
https://doi.org/10.1145/3706594.3727576

ACM Reference Format:
P. Chris Broekema and Rob V. van Nieuwpoort. 2025. SuperCode: Sustainability PER AI-driven Co-design: Invited Paper. In *22nd ACM International Conference on Computing Frontiers (CF Companion '25), May 28–30, 2025, Cagliari, Italy*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3706594.3727576

1 INTRODUCTION

Data-intensive science, such as radio astronomy and high-energy physics, requires vast amounts of compute resources to deliver world-leading science. The current energy and environmental crises drive a strong desire to do science in a manner that minimizes the environmental impact we make while maximizing the science we can deliver. Modern special-purpose compute architectures promise to be much more power efficient than general-purpose systems. However, leveraging these novel architectures is time-consuming and thus expensive due to the porting effort it takes to port existing code to a new architecture and the increasing complexity and specialization of hardware components.

In this vision paper, we present **SuperCode** (SUstainability PER AI-driven CO-DEsign), a novel approach to improve effective co-design of hardware and software. since this is essential to ensure that the resulting hardware and software combination is fit for purpose and able to run efficiently. We hypothesize that recent advances in code generation with AI-based Large Language Models (LLMs, e.g., ChatGPT) can be a catalyst for this process. We propose a systematic AI-driven co-design methodology that can drastically reduce the turn-around time to evaluate emerging technologies for data-intensive science, with sustainability as Key Performance Indicator (KPI). To validate our novel approach, we will explore two radio astronomy science cases and investigate their most optimal and sustainable emerging technology platform. With the partners in our project, we will explore opportunities in other domains like climate research, remote sensing and earth observation. The primary contributions in this paper can be summarized as follows:

- We present our vision for a novel AI-driven co-design methodology that promises to greatly improve the turn-around time for the evaluation of emerging and new technologies for data-intensive science.
- In the spirit of Open Science, team science and accountability, we publish a modified version of the project proposal for the accepted SuperCode project.
- We define sustainability as first class citizen, and use the methodology we introduced in previous work [6] to explicitly reason about this.

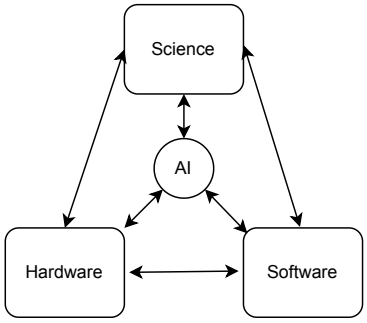

Figure 1: Adding science to the hardware/software co-design loop and leveraging AI to facilitate this process

2 PROBLEM STATEMENT

Data- and compute-intensive science, or eScience, is now firmly established as the fourth science paradigm [23], as introduced by Tony Hey and Jim Gray. While this opens exciting new abilities to explore new science in vast amounts of data, this comes at very significant processing and energy cost [39, 56]. This cost is often overlooked and poorly understood or appreciated. In the current climate crisis we can no longer accept that world-leading science has an unknown, and more importantly, potentially unconstrained environmental impact [39]. We argue that a careful and deliberate consideration needs to be made whether the scientific impact outweighs the potential environmental impact by the processing required. Optimization and a better mapping of software to hardware can shift this balance in our favour.

3 VISION AND METHODOLOGY

The primary transverse goal of SuperCode is to reduce the environmental impact of data-intensive applications, through a novel approach of AI-driven co-design. In contrast to normal co-design, where computational performance or efficiency is used as Key Performance Indicator (KPI), we introduce a sustainability score instead. This approach not only benefits our radio astronomy use-cases, but also immediately benefits our commercial partners.

Data-intensive science, and in particular radio astronomy, thrives by virtue of the availability of abundant and affordable computing to process the vast amount of data generated by modern instruments. Current generation instruments produce data at petascale [8], this is expected to increase to exascale in the near future [7, 9, 18, 52]. Processing such data for primary science cases is already challenging, mining for serendipitous discoveries is likely infeasible due to prohibitive energy costs. The current climate crisis makes it important to visualize and minimize the environmental impact of the processing done. Generative AI, and in particular large language models like ChatGPT, have already shown a remarkable ability to generate code for well-known architectures [4, 21, 30, 38]. We will investigate if we can push the boundaries of these AI models to generate code for more energy efficient emerging hardware platforms. Our method will thus face an extremely challenging problem: AI needs to generate code in a language or paradigm that does not know about. In this project we will develop and test a novel AI-assisted hardware-software-science co-design methodology (see Figure 1).

3.1 Sustainability

Scientific discovery is moving at an unprecedented pace, and many areas of research are limited in their potential by the lack of sufficient signal and data processing capacity [7]. However, the age of data-driven scientific discovery potentially comes at a very significant environmental impact. Signal processing for the LOFAR telescope, excluding science processing done by the astronomer, exceeds 500 MWh per year [29]. With increased capabilities this is expected to increase over the next couple of years. Future telescopes, like the Square Kilometre Array (SKA), are scaled such that procuring sufficient compute capacity is initially infeasible [18]. Even the partial system in that design will likely require MW-scale power. Efforts are underway to gain insights into the environmental impact that groundbreaking research infrastructures have [29, 31, 37]. However, there is currently no measure for the environmental impact of a scientific discovery.

Part of our vision is to visualize the resources required to make science possible. While energy consumption is the most obvious parameter in this context, it is by no means the only one. This is offset by the science output and/or economic impact, this is the societal value created. The latter was studied for e.g. LOFAR by the Rathenau institute [46]. While the definition of science value is bound to be controversial, this may be defined in terms of peer-reviewed publications, scientific discoveries, or prestigious prizes. We optimize for relative science value, which we define as the total value created (total value of ownership, TVO) divided by the resources consumed (total cost of ownership, TCO), shown in equation 1.

$$M_S = \frac{TVO}{TCO} \tag{1}$$

More detail about this method and its application in science can be found in our earlier work [6]. Using this relation, we will design hardware and software combinations that maximize science per unit of environmental impact (the used unit is flexible, and can be energy, CO_2, water, etc., or combinations of those). While some resources will inevitably be consumed, we need to be conscious of both the cost of those resources, the value that can be created, and be responsible enough to maximize the science we do with those. Using the proposed methodology, we aim to create a process that provides tailored advice for different science cases.

4 SUSTAINABILITY AS KPI

In this project we distinguish two separate classes of KPIs. At the micro level we evaluate the effectiveness of our AI-models by measuring the effort required to evaluate a particular emerging technology for our two use-cases. This involves an estimate of the quality of the produced prototype code (i.e., does it work as is, what performance is achieved, and does it produce accurate results), as well as the accuracy of the sustainability estimate produced by the AI model. Furthermore, we track the effort required and time needed to evaluate complex emerging technologies. We will thus use concrete measurements to validate our hypothesis.

At macro level our co-design KPIs focus on sustainability. We aim to minimize the environmental impact of data-intensive science by leveraging emerging technologies to optimize the efficiency of signal processing needed to turn instrument data into scientific data products. Traditionally we would optimize for computational performance or efficiency, in this project we rather focus on sustainability, the exact definition and metric of which is to be defined in the first stage of the project. Likely this is a combination of scientific potential, energy consumed, time to answer, resources and environmental impact required for production, recycling potential and cost, and others, combined to a sustainability score. We note that the environmental impact of technology will be an estimate based on public information and best effort estimates.

We refer to notable efforts by companies like Fairphone to be more transparent with their annual environmental impact and the progress made to reduce this [19]. Defining scientific potential of an instrument or technology will be challenging and is bound to be controversial. This will be the focus of the first part of the project but is likely to include potential for peer-reviewed publications, ground-breaking discoveries, instrument mean time between failure and mean time to recovery and ability to host multiple experiments. We acknowledge that this will be difficult to define but it is an important and often forgotten metric. We note that as a fallback scenario value can be defined as a constant, provided we ensure all technologies deliver similar scientific potential.

5 AI-DRIVEN CO-DESIGN

Co-design generally is complex and time consuming, often requiring hand-optimizing computational hotspots and a vast amount of domain- and platform-specific knowledge. Therefore, it is currently unfeasible to quickly react to changing scientific requirements or emerging more efficient hardware solutions, nor to discuss unforeseen opportunities with the researchers. Essentially, the scientist (our end-user) currently is not an integral part of the co-design loop. Exploiting the latest generative AI developments, we aim to turn this around.

Generative AI, and in particular large language models like Qwen, Llama3, ChatGPT, DeepSeek and many others have shown a remarkable ability to solve straightforward coding problems. This allows the programmer to focus on functional correctness, performance and scalability. Our hypothesis is that the use of AI in co-design can drastically reduce the effort needed to evaluate emerging technologies, making their use in smaller niche use-cases far more viable. This in turn leads to more sustainable science through more efficient use of energy and/or other resources. While we will test closed models, our approach is generic and we will favour open models like Llama 3 [22], DeepSeek-R1 [16] and Qwen hui2024qwen2 to ensure reproducibility and (to a degree) explainability. While previously open models performed worse than many closed modes, this gap has recently been nearly eliminated with the release of efficient reasoning models like DeepSeek.

A well-known property of LLMs is that they can hallucinate [59, 63], and thus generate unrelated or incorrect output. This is commonly seen as a bug. In this project, we see this as a feature instead, and aim to tap into this creativity to generate novel solutions. I.e., we want to use this emerging knowledge for emerging technologies.

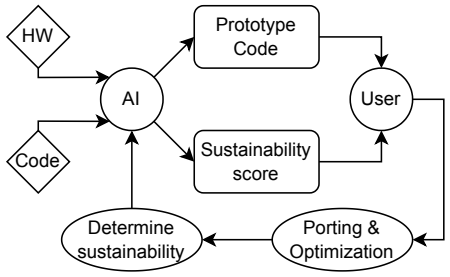

Figure 2: A high-level representation of our AI-driven co-design vision

By combining AI with the concept of the human in the co-design loop, the creativity exhibited by large language models is both constrained and leveraged.

By evaluating existing generative models and tailoring and testing our own models based on existing foundation models, we will build an AI co-design companion that will assist both programmer and system architect to design a sustainable hardware and software combination. Initially we will prompt an existing AI large language model with a combination of software and a detailed technical description of an emerging technology under investigation. Figure 2 shows a high-level overview of our AI-driven co-design concept. The methodology we envision works as follows.

We will add more information to the foundation models through matrix factorization techniques like low-rank adaptation (LoRA [25], LoHA [26], LoKr [17]) or other emerging compute-efficient training and/or fine-tuning techniques (DyLoRA [47], GloRA [14] or $(IA)^3$ [33]). This way, we tailor the AI for our needs. We will train these for every emerging technology we investigate, providing the knowledge needed to generate code and estimate the sustainability of the platform, and the two use-cases at the heart of the project. Recent successful work in using LLMs for Chip design indicates that LLMs are indeed capable of grasping new architectures [34]. Additionally, we will investigate prompt tailoring and retrieval-augmented generation on modern models [20, 32, 61], where we take advantage of longer context lengths achieved by newer LLMs that allow ever larger prompts that would ultimately allow us to provide both the entire architecture description as well as the reference code. We will compare these two approaches. These techniques will form the heart of our AI-accelerated co-design process. The trained specializations will be publicly released.

We will first train generative foundation models with reference implementations of the scientific data algorithms we use (in our case signal processing algorithms like FIR filter banks [48], FFTs, beam forming [45], correlation [8, 41, 50, 53], dedispersion [43, 44], etc.). These reference implementations are not optimized for performance, but for explainability and correctness. Using the existing base of open-source radio astronomy code, which is most of the code running current telescopes, we will train our own model. To validate our approach, as a first step we will use LLMs to translate existing Nvidia GPU code to use AMD GPUs. Resulting code

and performance will be compared to those generated by the HIPify [24] tool provided by AMD, which functions as the ground truth, making this an attractive first step that allows us to validate our methodology in an early stage. These ported codes will be publicly released.

Next, we will fine-tune the models with existing hand-optimized implementations, including CPU codes with vector instructions (AVX-512), and GPU codes. We currently use auto-tuning [43, 44, 55] to generate and test executable code for many different optimization options. In this project, we will feed all these generated codes into the AI model, providing us with sufficient training data. What makes this unique is that we measure the energy efficiency of the generated codes with unprecedented accuracy [55], allowing us to construct a cost function, in turn enabling us to use reinforcement learning to steer the AI towards more energy efficient, and thus potentially more sustainable, solutions [30]. Additionally, we can train the model on a host of technical documentation of existing and emerging hardware, for so far as publicly available. The fully trained and fine-tuned trained model will then be tasked to generate sustainability estimates and prototype codes for emerging architectures, based on its learned representations and the new hardware description. The sustainability estimate will be compared to the final measured index. We note here that the field of generative AI is evolving at an unprecedented pace. Any technologies and solutions mentioned in this paper may become obsolete during the SuperCode project. We will monitor the field carefully and use an agile approach to adjust our solutions as needed.

5.1 Emerging Technologies

To reduce the environmental impact of data-intensive science, we turn to emerging technologies. The inevitable demise of Moore's law scaling has given rise to a host of alternative technologies that aim to offer improved performance and efficiency at lower cost. One of the earliest examples and one that is now firmly in the mainstream, is General Purpose computing on Graphics Processing Units (GP-GPU) [35, 36]. Many techniques leverage specialization, where special purpose components perform specific tasks more efficiently than general purpose hardware. This is very costly due to non-recurring engineering expenses, and data-intensive science is not expected to procure sufficient volume to recoup that investment, even at the scale of the SKA. However, we do have the opportunity to re-use specialized hardware tailored for other applications, AI currently being the most obvious specialized hardware target. Examples are the tensor core correlator that leverages AI-focused cores on modern GPUs to improve efficiency over older GPUs by several factors [41][1].

In the short term we will target specialized cores on proven hardware solutions for use in our use-cases. Specifically, AVX-512 extensions in CPUs [12], tensor cores in NVIDIA GPUs, and matrix core engines on AMD GPUs. This will validate our approach, while simultaneously reducing project risk. Next, we will apply the approach on emerging but conventional silicon-based hardware platforms, like the planned accelerators developed in the European Processor Initiative (EPI), the Intel Ponte Veccio accelerator, and currently not publicly described products by startup companies like NextSilicon[2] and Accelera AI [3]. SURF's open innovation lab[4], and in some cases DAS-6 [2] or its successor will enable access to these emerging architectures. Finally, we will test our approach on more esoteric emerging technologies, such as neuromorphic or memristor based systems, without material changes to the underlying methodology. This will test our hypothesis that the AI-driven approach should be flexible enough to accommodate such very different platforms transparently, greatly reducing the effort and turnaround needed to evaluate such systems for our use-cases.

[1] https://git.astron.nl/RD/tensor-core-correlator

Figure 3: The core of the LOFAR telescope.

6 USE-CASES

We will apply our methodology to two use-cases: one terrestrial and one space-based. We have selected these because they are both challenging but put vastly different constraints and requirements on the processing platform. Moreover, compared to data-intensive applications in general, these are at extreme ends of the scale. If we have validated these, it is plausible that our methodology also is applicable for less data intensive applications.

6.1 Use-case 1: Terrestrial large-scale distributed radio telescopes

Our first use-case focuses on terrestrial large-scale distributed radio aperture synthesis arrays. These are at the forefront of low- and mid-frequency radio astronomy. Specifically, we will use The LOw Frequency ARray (LOFAR) [49], designed and built by ASTRON (See also Figure 3), and Square Kilometre Array (SKA) [10, 42] telescopes, currently under construction by a multi-national consortium including ASTRON. Most partners in our user committee are also involved in the construction of the software pipelines of the SKA.

Aperture synthesis arrays create a virtual telescope by combining multiple geographically separated sensors. These all sample the electromagnetic spectrum that, according to the Van Cittert-Zernike theorem [62] can be considered to come from the same distant source. Correlating many combinations of sensors gives us a sparse set of points that have a Fourier-relation with the sky image. We generally take advantage of the earth's rotation as well

[2] https://www.nextsilicon.com/
[3] https://axelera.ai
[4] https://www.surf.nl/lab

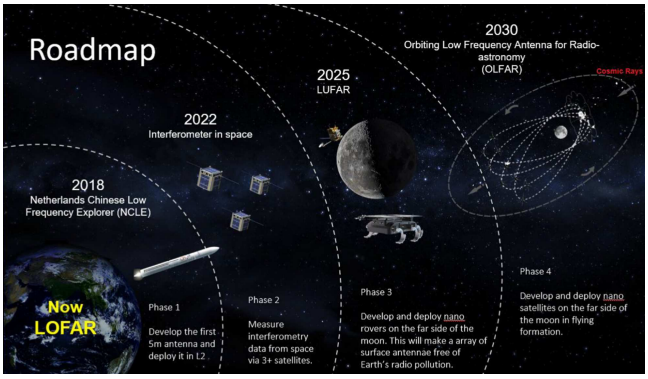

Figure 4: The OLFAR roadmap, picture courtesy [5].

as the coherent nature of the signal of interest over incoherent noise to fill in the sparse image and amplify the weak astronomical signals. Radio astronomy requires a lot of signal processing to turn what is essentially noise into a scientific data product. The computational requirements for such instruments scale dramatically ($O(n^3)$ - $O(n^4)$) with the size of the telescope, both due to increased data volumes and more processing required per unit of data, which is also a key driving factor in sensitivity and resolution.

LOFAR, a state-of-the-art low-frequency array, requires processing at tera-scale. The SKA, which is about an order of magnitude larger in terms of receivers, is expected to require peta-scale processing [10]. The initial procurement of compute infrastructure will not cover that requirement, mostly due to budget and energy constraints. The scientific potential of current and future radio telescopes is limited by the availability of sufficient affordable data processing capacity and software. In use-case 1, we will evaluate the sustainability aspects of signal processing algorithms for terrestrial telescopes, like FIR filter banks [48], FFTs, beam forming [45], correlation [8, 41, 50, 53], dedispersion [43, 44], etc.

6.2 Use-case 2: Space-based radio telescopes

Use-case 2 is more speculative. The promise of higher sensitivity thanks to less ionospheric disturbance and Radio Frequency Interference (RFI) [51] and an unexplored low-frequency band due to the opaqueness of the atmosphere to frequencies around 10MHz, is generating interest for space-based interferometers. Such instruments include both orbiting swarms (OLFAR [5, 54], ALO [28] and DSL [15], as well as lunar surface instruments (Farside [11] and LuSEE'Night' [3]). The OLFAR roadmap is presented in Figure 4. The hostile extra-terrestrial environment and limited available energy, volume, mass and ability to dissipate heat, lead to unique and quite different concepts and challenges [40] compared to terrestrial telescopes, making this an interesting use-case.

Where terrestrial radio telescopes favour the transport of as much data as feasible to a central location for processing, this is not the case for space-based telescopes. Here, due to excessive bandwidth constraints and costs, edge processing is key. Furthermore, in space, energy is not abundant and heat dissipation of signal processing systems may be challenging. Cosmic radiation requires space-hardened systems, which are based on older (i.e., with larger gate sizes) production technologies, whereas from a sustainability viewpoint we would favour the newest processes. Note that these considerations are immediately applicable to space-based earth observation, where similar constraints are encountered. Project partners Sioux and S[&]T have extensive experience in earth observation and have expressed a keen interest in this similarity.

The signal processing is similar for space-based radio interferometry. That said, sustainability will be very different, considering energy in space is sustainably generated. Thus, production and launch will be much more dominant in the equation. Furthermore, we need to consider the environmental impact after the useful life of the instrument. If in orbit, this means de-orbiting to avoid creating space junk. In use-case 2, we will evaluate which emerging architectures are most suitable and sustainable for space-based telescopes.

7 SCIENTIFIC AND SOCIETAL IMPACT

We firmly believe that the use of LLMs will fundamentally and disruptively change the way both academia and industry develop software. The role of software developers will not disappear but will require developers to use and understand this novel AI technology to be more productive and efficient. The Supercode project has six industrial partners who already collaborate with us in designing and constructing software pipelines for the SKA. The partners were carefully selected on what is their core business: software development in business and industrial applications. Therefore, they will be very much affected by LLM-based code generation. By using the technologies developed in this project in their own use-cases during the workshops we will organize, they will be better prepared for the future, thus significantly enlarging the societal impact of our research.

To ensure that our methods and results are transferred to our industrial partners we will organize workshops and hackathons, with their broader organizations and stakeholders beyond the user committee. These workshops are broader in terms of the audience and also include the software development teams constructing the SKA software pipelines. In these workshops we will discuss our results, show demos, but will also take a hands-on approach to tackle concrete use cases brought in by the participants. The advantage of our methodology is that it enables quick prototyping and experimentation, allowing the users to quickly get a feel for what is possible. Towards the end of the project, we will develop open training materials in Software Carpentry style, enabling an even broader community to implement the methods we will develop. This ensures long lasting impact, also after the project has finished.

We will develop tools and instrumentation to maximize science output per unit of environmental impact. The sustainability score, the KPI in this project, is a trade-off of value (science output) and cost (environmental impact), clearly indicating that societal and scientific impact are of comparable focus. The energy usage of data centres, often to a large extent caused by AI applications, is unsustainable, and will have to be reduced. By not training full models, but instead using and finetuning and extending existing pruned foundation models, we aim to show that much more efficient approaches are feasible, in science, but also in industrial applications.

By implementing our work in concrete use cases related to the SKA, and the direct involvement of the SKA software developer teams, we embed our work in the SKA community. We will publish the results of this project in top conferences and journals. Because of the interdisciplinary nature of the work, we aim to publish in both computer science and astronomy. We take the approach of first publishing extended abstracts in astronomy, mostly to disseminate and validate the work with our peers. Next, we publish the methodological aspects in top computer science venues (e.g., supercomputing, PASC, IPDPS, ICS). Finally, we publish the applied results in astronomy journals (A&A, MNRAS, astronomy & computing, etc.). With this triple-target publication strategy, we reach different audiences, maximizing our scientific impact.

8 RELATED WORK

AI-supported code generation has recently become possible [1, 58] and has seen only very limited application in high-performance computing. So far, no research has been done on generating and porting code to emerging architectures. This is challenging since there are no concrete code examples on the emerging architecture, so the LLMs must effectively do transfer learning to port existing codes to new architectures. LLMs have not been used with the specific focus of generating code for exploring more energy efficient combinations of hardware architecture and software implementation.

Godoy et al. [21] evaluate AI-assisted generative capabilities on several numerical HPC kernels. They generated codes for a variety of programming models and languages, including C++/OpenMP, OpenACC, Kokkos, SyCL, CUDA, Fortran and Julia. However, all algorithms tested were well known (e.g., the models were trained on many examples), and they did not generated code for emerging architectures. Nevertheless, this does indicate that our proposed approach is feasible.

Currently, LLMs do not understand program semantics, and offer no guarantees about quality of the generated code. Jain et al. [27] demonstrated that augmenting LLMs with syntax and semantics-aware program analysis and user feedback improves the output. The interactive properties of LLM models (via the prompt) have not been used for co-design research in this context. Our project will, for the first time, apply the emerging abilities of generative AI to reduce the challenges that are faced in effective co-design.

IBM is currently using generative AI to modernize legacy applications [13]. Their work focuses mostly on translation from COBOL to Java, while our approach translates to different architectures. We must translate between programming languages, but also need to exploit different forms of parallelism (semi-)automatically.

EcoOptiGen [57] is a hyperparameter optimization framework for LLMs. EcoOptiGen leverages cost-based pruning to reduce the (energy) cost of LLMs. Like EcoOptiGen, we will also explore the AutoGen [60] framework. While EcoOptiGen focuses on inference, our work focuses on training and finetuning aspects.

CodeRL [30] is a framework for program synthesis through pretrained LLMs and deep reinforcement learning. It uses a critical sampling strategy to automatically generate programs based on feedback from example unit tests. Our envisioned approach extends this with full reference codes and codes for other architectures. Moreover, our philosophy is (semi)-supervised and based around co-design, keeping an expert in the loop.

9 COLLABORATION

The SuperCode project, started in January 2025, is funded through a Dutch Public Private Partnership programme. Seven Dutch and international companies, as well as representatives from the Dutch scientific community already collaborate in this project. The project will organise several workshops and hackathons, where we work with our partners to apply this methodology on their own codes. Furthermore, the project will follow open science best practice, allowing broad adoption of the methodology and results.

Because of extensive expressed interest from industry and scientists alike, we invite the international scientific community and interested international industry representatives, to consider partnering in this project. We would especially be interested in collaborating with disciplines beyond the natural sciences.

10 CONCLUSION

We have presented a novel AI-driven co-design concept that aims to revolutionise the way we design and build data-intensive science infrastructure and code. The use of generative AI allows us to more quickly adopt new and emerging technologies, making it feasible to optimise for sustainability instead of functionality. We invite collaboration to explore this concept beyond its original vision and use cases. The full proposal is available as a pre-print publication[5].

ACKNOWLEDGMENTS

This work was funded in 2024, by the Open Technology Programme call of the Technical and Applied Sciences department of the Netherlands Research Council NWO, under grant number 21356. We would like to thank the following partner organizations in the SuperCode project for their support, in-kind funding and valuable discussions: SURF (Raymond Oonk), CGI (Alexandra Zevenbergen), TriOpSys (Joy Ong), [S&T] (Erik van Mulligen), Netherlands eScience Center (Patrick Bos), and Sioux Technologies (Bas van der Linden). Finally, we would like to thank the SuperCode advisory committee (Mark Bentum, Aske Plaat, Suzan Verberne, Ben van Werkhoven and Albert-Jan Boonstra) for their insightful feedback.

REFERENCES

[1] Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. 2021. Program Synthesis with Large Language Models. https://doi.org/10.48550/arXiv.2108.07732 arXiv:2108.07732 [cs].

[2] Henri Bal, Raoul Bhoedjang, Rutger Hofman, Ceriel Jacobs, Thilo Kielmann, Jason Maassen, Rob van Nieuwpoort, John Romein, Luc Renambot, Tim Rühl, Ronald Veldema, Kees Verstoep, Aline Baggio, Gerco Ballintijn, Ihor Kuz, Guillaume Pierre, Maarten van Steen, Andy Tanenbaum, Gerben Doornbos, Desmond Germans, Hans Spoelder, Evert-Jan Baerends, Stan van Gisbergen, Hamideh Afsermanesh, Dick van Albada, Adam Belloum, David Dubbeldam, Zeger Hendrikse, Bob Hertzberger, Alfons Hoekstra, Kamil Iskra, Drona Kandhai, Dennis Koelma, Frank van der Linden, Benno Overeinder, Peter Sloot, Piero Spinnato, Dick Epema, Arjan van Gemund, Pieter Jonker, Andrei Radulescu, Cees van Reeuwijk, Henk Sips, Peter Knijnenburg, Michael Lew, Floris Sluiter, Lex Wolters, Hans Blom, Cees de Laat, and Aad van der Steen. 2000. The distributed ASCI Supercomputer project. *ACM SIGOPS Operating Systems Review* 34, 4 (Oct. 2000), 76–96. https://doi.org/10.1145/506106.506115

[3] Stuart D. Bale, Neil Bassett, Jack O. Burns, Johnny Dorigo Jones, Keith Goetz, Christian Hellum-Bye, Sven Hermann, Joshua Hibbard, Milan Maksimovic,

[5] https://arxiv.org/abs/2412.08490/

Ryan McLean, Raul Monsalve, Paul O'Connor, Aaron Parsons, Marc Pulupa, Rugved Pund, David Rapetti, Kaja M. Rotermund, Ben Saliwanchik, Anze Slosar, David Sundkvist, and Aritoki Suzuki. 2023. LuSEE 'Night': The Lunar Surface Electromagnetics Experiment. https://doi.org/10.48550/arXiv.2301.10345 arXiv:2301.10345 [astro-ph].

[4] Shraddha Barke, Michael B. James, and Nadia Polikarpova. 2023. Grounded Copilot: How Programmers Interact with Code-Generating Models. *Proceedings of the ACM on Programming Languages* 7, OOPSLA (April 2023), 85–111. https://doi.org/10.1145/3586030

[5] M. J. Bentum, M. K. Verma, R. T. Rajan, A.-J. Boonstra, C. J. M. Verhoeven, E. K. A. Gill, A. J. van der Veen, H. Falcke, M. Klein Wolt, B. Monna, S. Engelen, J. Rotteveel, and L. I. Gurvits. 2020. A Roadmap towards a Space-based Radio Telescope for Ultra-Low Frequency Radio Astronomy. *Advances in Space Research* 65, 2 (Jan. 2020), 856–867. https://doi.org/10.1016/j.asr.2019.09.007 arXiv:1909.08951 [astro-ph].

[6] P Chris Broekema, Verity Allan, Rob V van Nieuwpoort, and Henri E Bal. 2020. On optimising cost and value in compute systems for radio astronomy. *Astronomy and Computing* 30 (2020), 100337.

[7] P Chris Broekema, Albert-Jan Boonstra, Victoria Caparrós Cabezas, Ton Engbersen, Hanno Holties, Jens Jelitto, Ronald P Luijten, Peter Maat, Rob V Van Nieuwpoort, Ronald Nijboer, et al. 2012. DOME: towards the ASTRON & IBM center for exascale technology. In *Proceedings of the 2012 Workshop on High-Performance Computing for Astronomy Date*. 1–4.

[8] P. Chris Broekema, J. Jan David Mol, R. Nijboer, A. S. van Amesfoort, M. A. Brentjens, G. Marcel Loose, W. F. A. Klijn, and J. W. Romein. 2018. Cobalt: A GPU-based correlator and beamformer for LOFAR. *Astronomy and Computing* 23 (April 2018), 180–192. https://doi.org/10.1016/j.ascom.2018.04.006

[9] P Chris Broekema, Rob V Van Nieuwpoort, and Henri E Bal. 2012. Exascale high performance computing in the square kilometer array. In *Proceedings of the 2012 workshop on High-Performance Computing for Astronomy Date*. 9–16.

[10] P Chris Broekema, Rob V van Nieuwpoort, and Henri E Bal. 2015. The square kilometre array science data processor. Preliminary compute platform design. *Journal of Instrumentation* 10, 07 (2015), C07004.

[11] Jack Burns, Gregg Hallinan, Tzu-Ching Chang, Marin Anderson, Judd Bowman, Richard Bradley, Steven Furlanetto, Alex Hegedus, Justin Kasper, Jonathan Kocz, Joseph Lazio, Jim Lux, Robert MacDowall, Jordan Mirocha, Issa Nesnas, Jonathan Pober, Ronald Polidan, David Rapetti, Andres Romero-Wolf, Anže Slosar, Albert Stebbins, Lawrence Teitelbaum, and Martin White. 2021. A Lunar Farside Low Radio Frequency Array for Dark Ages 21-cm Cosmology. https://doi.org/10.48550/arXiv.2103.08623 arXiv:2103.08623 [astro-ph, physics:hep-ex].

[12] André Ramos Carneiro, Matheus S. Serpa, and Philippe O. A. Navaux. 2021. Lightweight Deep Learning Applications on AVX-512. In *2021 IEEE Symposium on Computers and Communications (ISCC)*. 1–6. https://doi.org/10.1109/ISCC53001.2021.9631464 ISSN: 2642-7389.

[13] Kyle Charlet. 2023. Harnessing Generative AI for Application Modernization at Speed and Scale - IBM Z and LinuxONE Community. https://community.ibm.com/community/user/ibmz-and-linuxone/blogs/kyle-charlet/2023/08/22/harnessing-generative-ai-for-modernization

[14] Arnav Chavan, Zhuang Liu, Deepak Gupta, Eric Xing, and Zhiqiang Shen. 2023. One-for-All: Generalized LoRA for Parameter-Efficient Fine-tuning. https://doi.org/10.48550/arXiv.2306.07967 arXiv:2306.07967 [cs].

[15] Xuelei Chen, Jingye Yan, Li Deng, Fengquan Wu, Lin Wu, Yidong Xu, and Li Zhou. 2020. Discovering the sky at the longest wavelengths with a lunar orbit array. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 379, 2188 (Nov. 2020), 20190566. https://doi.org/10.1098/rsta.2019.0566 Publisher: Royal Society.

[16] DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948 [cs.CL] https://arxiv.org/abs/2501.12948

[17] Ali Edalati, Marzieh Tahaei, Ivan Kobyzev, Vahid Partovi Nia, James J. Clark, and Mehdi Rezagholizadeh. 2022. KronA: Parameter Efficient Tuning with Kronecker Adapter. https://doi.org/10.48550/arXiv.2212.10650 arXiv:2212.10650 [cs].

[18] Ton Engbersen, A Boonstra, A Anghel, C Broekema, R Dangel, G van Diepen, G Dittmann, A Doering, C Hagleitner, H Holties, et al. 2014. SKA a bridge too far, or not? *Exascale Radio Astronomy* 2 (2014), 10101.

[19] Fairphone. 2023. Impact Report - A challenge to the electronics industry. https://www.fairphone.com/en/impact-report/

[20] Andrew Gao. 2023. Prompt Engineering for Large Language Models. https://doi.org/10.2139/ssrn.4504303

[21] William F. Godoy, Pedro Valero-Lara, Keita Teranishi, Prasanna Balaprakash, and Jeffrey S. Vetter. 2023. Evaluation of OpenAI Codex for HPC Parallel Programming Models Kernel Generation. In *Proceedings of the 52nd International Conference on Parallel Processing Workshops*. 136–144. https://doi.org/10.1145/3605731.3605886 arXiv:2306.15121 [cs].

[22] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu,

Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The Llama 3 Herd of Models. https://doi.org/10.48550/arXiv.2407.21783 arXiv:2407.21783 [cs].

[23] Anthony J. G. Hey (Ed.). 2009. *The fourth paradigm: data-intensive scientific discovery*. Microsoft Research, Redmond, Washington.

[24] Brian Homerding and John Tramm. 2020. Evaluating the Performance of the hipSYCL Toolchain for HPC Kernels on NVIDIA V100 GPUs. In *Proceedings of the International Workshop on OpenCL (IWOCL '20)*. Association for Computing Machinery, New York, NY, USA, 1–7. https://doi.org/10.1145/3388333.3388660

[25] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models. https://doi.org/10.48550/arXiv.2106.09685 arXiv:2106.09685 [cs].

[26] Nam Hyeon-Woo, Moon Ye-Bin, and Tae-Hyun Oh. 2021. FedPara: Low-rank Hadamard Product for Communication-Efficient Federated Learning. *arXiv preprint arXiv:2108.06098* (Aug. 2021).

[27] Naman Jain, Skanda Vaidyanath, Arun Iyer, Nagarajan Natarajan, Suresh Parthasarathy, Sriram Rajamani, and Rahul Sharma. 2022. Jigsaw: large language models meet program synthesis. In *Proceedings of the 44th International Conference on Software Engineering (ICSE '22)*. Association for Computing Machinery, New York, NY, USA, 1219–1231. https://doi.org/10.1145/3510003.3510203

[28] Sebastian Jester and Heino Falcke. 2009. Science with a lunar low-frequency array: From the dark ages of the Universe to nearby exoplanets. *New Astronomy Reviews* 53, 1 (May 2009), 1–26. https://doi.org/10.1016/j.newar.2009.02.001

[29] Gert Kruithof, Cees Bassa, Irene Bonati, Wim van Cappellen, Anne Doek, Nico Ebbendorf, Marchel Gerbers, Michiel van Haarlem, Ronald Halfwerk, Hanno Holties, Simone Kajuiter, Vlad Kondratiev, Henri Meulman, Roberto Pizzo, Timothy Shimwell, and John Swinbank. 2023. The energy consumption and carbon footprint of the LOFAR telescope. *Experimental Astronomy* (July 2023). https://doi.org/10.1007/s10686-023-09901-z

[30] Hung Le, Yue Wang, Akhilesh Deepak Gotmare, Silvio Savarese, and Steven Chu Hong Hoi. 2022. CodeRL: Mastering Code Generation through Pretrained Models and Deep Reinforcement Learning. *Advances in Neural Information Processing Systems* 35 (Dec. 2022), 21314–21328. https://papers.nips.cc/paper_files/paper/2022/hash/8636419dea1aa9fbd25fc4248e702da4-Abstract-Conference.html

[31] Baolin Li, Rohan Basu Roy, Daniel Wang, Siddharth Samsi, Vijay Gadepally, and Devesh Tiwari. 2023. Toward Sustainable HPC: Carbon Footprint Estimation and Environmental Implications of HPC Systems. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC '23)*. ACM, New York, NY, USA, 1–15. https://doi.org/10.1145/3581784.3607035

[32] Zhicheng Lin. 2023. Ten Simple Rules for Crafting Effective Prompts for Large Language Models. https://doi.org/10.2139/ssrn.4565553

[33] Haokun Liu, Derek Tam, Muqeeth Mohammed, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. 2022. Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning. *Advances in Neural Information Processing Systems* 35 (May 2022), 1950–1965. https://openreview.net/forum?id=rBCvMG-JsPd

[34] Mingjie Liu, Teo Ene, Robert Kirby, Chris Cheng, Nathaniel Pinckney, Rongjian Liang, Jonah Alben, Himyanshu Anand, Sanmitra Banerjee, Ismet Bayraktaroglu, Bonita Bhaskaran, Bryan Catanzaro, Arjun Chaudhuri, Sharon Clay, Bill Dally, Laura Dang, Parikshit Deshpande, Siddhanth Dhodhi, Sameer Halepete, Eric Hill, Jiashang Hu, Sumit Jain, Brucek Khailany, Kishor Kunal, Xiaowei Li, Hao Liu, Stuart Oberman, Sujeet Omar, Sreedhar Pratty, Ambar Sarkar, Zhengjiang Shao, Hanfei Sun, Pratik P. Suthar, Varun Tej, Kaizhe Xu, and Haoxing Ren. 2023. ChipNeMo: Domain-Adapted LLMs for Chip Design. https://doi.org/10.48550/arXiv.2311.00176 arXiv:2311.00176 [cs].

[35] Souley Madougou, Ana Varbanescu, Cees de Laat, and Rob van Nieuwpoort. 2016. The landscape of GPGPU performance modeling tools. *Parallel Comput.* 56 (2016), 18–33.

[36] Souley Madougou, Ana Lucia Varbanescu, Cees de Laat, and Rob van Nieuwpoort. 2014. An empirical evaluation of GPGPU performance models. In *Euro-Par 2014: Parallel Processing Workshops: Euro-Par 2014 International Workshops, Porto, Portugal, August 25-26, 2014, Revised Selected Papers, Part I 20*. Springer, 165–176.

[37] Pierrick Martin, Sylvie Brau-Nogué, Mickael Coriat, Philippe Garnier, Annie Hughes, Jürgen Knödlseder, and Luigi Tibaldo. 2022. A comprehensive assessment of the carbon footprint of an astronomical institute. *Nature Astronomy* 6, 11 (Nov. 2022), 1219–1222. https://doi.org/10.1038/s41550-022-01771-3 Number: 11 Publisher: Nature Publishing Group.

[38] Nhan Nguyen and Sarah Nadi. 2022. An Empirical Evaluation of GitHub Copilot's Code Suggestions. In *2022 IEEE/ACM 19th International Conference on Mining Software Repositories (MSR)*. 1–5. https://doi.org/10.1145/3524842.3528470 ISSN: 2574-3864.

[39] The Shift Project. 2019. Lean ICT: Towards Digital Sobriety. https://theshiftproject.org/en/article/lean-ict-our-new-report/

[40] Raj Thilak Rajan, Albert-Jan Boonstra, Mark Bentum, Marc Klein-Wolt, Frederik Belien, Michel Arts, Noah Saks, and Alle-Jan van der Veen. 2016. Space-based aperture array for ultra-long wavelength radio astronomy. *Experimental Astronomy* 41, 1 (Feb. 2016), 271–306. https://doi.org/10.1007/s10686-015-9486-6

[41] John W. Romein. 2021. The Tensor-Core Correlator. *Astronomy & Astrophysics* 656 (Dec. 2021), A52. https://doi.org/10.1051/0004-6361/202141896 Publisher: EDP Sciences.

[42] Richard T Schilizzi, Peter EF Dewdney, and T Joseph W Lazio. 2008. The square kilometre array. In *Ground-based and Airborne Telescopes II*, Vol. 7012. SPIE, 603–615.

[43] Alessio Sclocco, Henri E Bal, Jason Hessels, Joeri Van Leeuwen, and Rob V Van Nieuwpoort. 2014. Auto-tuning dedispersion for many-core accelerators. In *2014 IEEE 28th International Parallel and Distributed Processing Symposium*. IEEE, 952–961.

[44] Alessio Sclocco, Joeri van Leeuwen, Henri E Bal, and Rob V van Nieuwpoort. 2016. Real-time dedispersion for fast radio transient surveys, using auto tuning on many-core accelerators. *Astronomy and computing* 14 (2016), 1–7.

[45] Alessio Sclocco, Ana Lucia Varbanescu, Jan David Mol, and Rob V van Nieuwpoort. 2012. Radio astronomy beam forming on Many-Core architectures. In *2012 IEEE 26th International Parallel and Distributed Processing Symposium*. IEEE, 1105–1116.

[46] Sue-Yen Tjong Tjin Tai, Jos van den Broek, and Jasper Deuten. 2019. *De impact van grootschalige onderzoeksinfrastructuren | Rathenau Instituut*. Technical Report. Rathenau Instituut. https://www.rathenau.nl/nl/werking-van-het-wetenschapssysteem/de-impact-van-grootschalige-onderzoeksinfrastructuren

[47] Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. 2023. DyLoRA: Parameter-Efficient Tuning of Pre-trained Models using Dynamic Search-Free Low-Rank Adaptation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, Dubrovnik, Croatia, 3274–3287. https://doi.org/10.18653/v1/2023.eacl-main.239

[48] Karel van der Veldt, Rob van Nieuwpoort, Ana Lucia Varbanescu, and Chris Jesshope. 2012. A polyphase filter for GPUs and multi-core processors. In *Proceedings of the 2012 workshop on High-Performance Computing for Astronomy Date*. 33–40.

[49] Michael P van Haarlem, Michael W Wise, AW Gunst, George Heald, John P McKean, Jason WT Hessels, A Ger de Bruyn, Ronald Nijboer, John Swinbank, Richard Fallows, et al. 2013. LOFAR: The low-frequency array. *Astronomy & astrophysics* 556 (2013), A2.

[50] Rob van Nieuwpoort and John W Romein. 2010. Building correlators with many-core hardware. *IEEE Signal Processing Magazine* 27, 2 (2010), 108–117.

[51] R van Nieuwpoort, J van Leeuwen, A Sclocco, H Spreeuw, and C Williams. 2018. Real-Time RFI Mitigation for LOFAR, Apertif and SKA. In *2018 2nd URSI Atlantic Radio Science Meeting (AT-RASC)*. IEEE, 1–1.

[52] Rob V van Nieuwpoort. 2016. Towards exascale real-time RFI mitigation. In *2016 Radio Frequency Interference (RFI)*. IEEE, 69–74.

[53] Rob V van Nieuwpoort and John W Romein. 2011. Correlating radio astronomy signals with many-core hardware. *International Journal of Parallel Programming* 39 (2011), 88–114.

[54] Pieter van Vugt, Arjan Meijerink, and Mark Bentum. 2016. Calibration of the OLFAR space-based radio telescope using an alternating least squares approach. In *2016 IEEE Aerospace Conference*. 1–8. https://doi.org/10.1109/AERO.2016.7500559

[55] Ben van Werkhoven. 2019. Kernel Tuner: A search-optimizing GPU code auto-tuner. *Future Generation Computer Systems* 90 (Jan. 2019), 347–358. https://doi.org/10.1016/j.future.2018.08.004

[56] Centraal Bureau voor de Statistiek. 2021. Elektriciteit geleverd aan datacenters, 2017-2020. https://www.cbs.nl/nl-nl/maatwerk/2021/50/elektriciteit-geleverd-aan-datacenters-2017-2020 Last Modified: 2021-12-15T10:21:00+01:00.

[57] Chi Wang, Susan Xueqing Liu, and Ahmed H. Awadallah. 2023. Cost-Effective Hyperparameter Optimization for Large Language Model Generation Inference. https://doi.org/10.48550/arXiv.2303.04673 arXiv:2303.04673 [cs].

[58] Yue Wang, Weishi Wang, Shafiq Joty, and Steven C.H. Hoi. 2021. CodeT5: Identifier-aware Unified Pre-trained Encoder-Decoder Models for Code Understanding and Generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 8696–8708. https://doi.org/10.18653/v1/2021.emnlp-main.685

[59] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682* (2022).

[60] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, Ahmed Hassan Awadallah, Ryen W. White, Doug Burger, and Chi Wang. 2023. AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation. https://doi.org/10.48550/arXiv.2308.08155 arXiv:2308.08155 [cs].

[61] Benfeng Xu, An Yang, Junyang Lin, Quan Wang, Chang Zhou, Yongdong Zhang, and Zhendong Mao. 2023. ExpertPrompting: Instructing Large Language Models to be Distinguished Experts. https://doi.org/10.48550/arXiv.2305.14688 arXiv:2305.14688 [cs].

[62] F. Zernike. 1938. The concept of degree of coherence and its application to optical problems. *Physica* 5, 8 (Aug. 1938), 785–795. https://doi.org/10.1016/S0031-8914(38)80203-2

[63] Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models. https://doi.org/10.48550/arXiv.2309.01219 arXiv:2309.01219 [cs].