No. 565 April 2017

The ICARUS white paper: A scalable, energy-efficient, solar-powered HPC center based on low power GPUs

M. Geveler, D. Ribbrock, D. Donner,
H. Ruelmann, C. Höppke, D. Schneider,
D. Tomaschweski, S. Turek

ISSN: 2190-1767

The ICARUS white paper: A scalable, energy-efficient, solar-powered HPC center based on low power GPUs

Markus Geveler, Dirk Ribbrock, Daniel Donner, Hannes Ruelmann, Christoph Höppke, David Schneider, Daniel Tomaschewski, and Stefan Turek

Institute for Applied Mathematics, TU Dortmund
Vogelpothsweg 87, 44227 Dortmund, Germany
markus.geveler@math.tu-dortmund.de
http://www.icarus-green-hpc.org

Abstract. We present a unique approach for integrating research in High Performance Computing (HPC) as well as photovoltaic (PV) solar farming and battery technologies into a container-based compute center designed for a maximum of energy efficiency, performance and extensibility/scalability. We use NVIDIA Jetson TK1 boards to build a considerably dimensioned cluster of 60 low-power GPUs, attach a 7.5 kWp solar farm and a 8 kWh Lithium-Ion battery power supply and integrate everything into a single-container, standalone housing. We demonstrate the success of our system by evaluating the performance and energy efficiency for common versatile dense and sparse linear algebra kernels as well as a full CFD code. By this work we can show, that with current technology, energy consumption-induced follow-up cost of HPC can be reduced to zero.

Keywords: energy-efficient HPC, ARM cluster, GPGPU, solar power, battery power supply

1 Introduction

In the age of transitioning from nuclear- and fossil-driven energy supplies to renewables, besides energy harvesting and energy grids adapting to this decentralized energy production, *energy consumers* (such as computer hardware) have to be adapted, which in principle means a necessary increase in *energy efficiency*. Today's HPC centers mostly rely on massively parallel distributed memory clusters whose compute nodes are also multi-level parallel and heterogeneous. The nodes usually comprise one or more high-end server CPUs based on the x86, Power, or SPARC architectures optionally accelerated by GPUs or other (accelerator) hardware. Large HPC sites of this type have substantial energy requirements so that the associated expenses over the lifetime of the system may reach the same order of magnitude as the initial acquisition costs. In addition, the energy supply for supercomputers is not always an integral part of its overall

design - consumers (such as the compute-cluster, cooling, networking, management hardware) are often developed independently from the key technologies of the energy revolution, e.g. renewable energy sources, battery- and power-grid techniques. The Power Wall has been accepted to be one of the major challenges in high scale computing. However, as a consequence of decades of performance-centric hardware development, there is a huge gap between pure performance and energy efficiency in these designs: The Top500 list's best performing HPC system (dissipating power in the 20 mega-watts range making a power supply by local solar farming for instance an impossible-to-achieve aim) is only ranked 84th on the corresponding Green500 list, whereas the most energy-efficient system in place only performs 160th in the metric of raw floating point performance [17, 4]. The most obvious feature all Green500 top ten systems share is, that they rely on accelerators - mostly GPUs, but the top three even on an unconventional micro architecture. From an HPC center's point of view, there are two possible ways to tune the energy efficiency: For a given HPC installation, an optimal reduced processor voltage and frequency can be found [23, 24], or – at the hardware-design stage – more energy efficient hardware components can be selected. Recently, power and energy metrics started being included into performance models for numerical software [12, 2, 1, 15]. However, developers of scientific software can (if at all) only control the energy efficiency of their 'production'-code, while hardware of the targeted HPC centers is out of their influence. The most impacting reason for this is the fact, that the cluster design is prone to principles of mass markets or in other words, HPC users do not determine the properties of available compute hardware. The users are literally trapped between very 'traditional' chip vendor- and HPC center construction markets concentrating on raw performance and being as much versatile as possible on the one hand and relatively application-oblivious acquisition processes on the HPC-site level (i.e. university-level- or even regional resources) on the other. Hence, there is a huge potential in energy savings in HPC. Recently, a game-changing impulse in this regard for HPC may come from mobile/embedded computing with devices featuring a long history of being developed under one major aspect: they have had to be operated with a (limited) battery power supply. Hence, as opposed to x86 and other commodity designs (with a focus on chipset compatibility and performance), the resulting energy efficiency advantage can be made accessible to the HPC community. In our earlier work [10] we demonstrated reductions in the energy-to-solution of simulations by using ARM-based processors. Those findings were obtained on a cluster prototype built with NVIDIA Tegra 2 and continued later with Tegra 3 micro-architecture [21]. Both chips are based on the Cortex-A9 processor; our current work employs NVIDIA Tegra K1 with Cortex-A15 CPUs [6] and – focused in this paper – the embedded GPU. In the meantime, using low-power (ARM) hardware in the HPC context, especially as a 'low energy-to-solution' alternative to commodity CPUs, has become an active research topic [3]. With the NVIDIA Tegra K1, even a programmable embedded low power Kepler GPU becomes accessible alongside the ARM cores on one System-on-Chip (SoC), making a huge jump in theoretical peak perfor-

mance whilst preserving minimum power requirements. This may hence offer a way to change hosting of simulations, making them accessible to more universities/enterprises/data centers. Also, we believe that in order to make a change it is necessary to take a look at the problem of too much overall energy consumption (and therefore carbon dioxide pollution) from a greater angle than any scientific field alone can provide. Our idea is to bring to life a lighthouse project, that overcomes the limits regarding (energy) efficiency of scientific software development on the one hand side and standard HPC center construction on the other. Our system combines the high ends in energy-efficient floating point hardware, renewable energies and battery storage with a self-made housing and cooling. Normally, we are concerned with hardware-oriented simulation software. In this paper, we deliberately switch angles designing a versatile, extensible and scalable HPC resource *at zero follow-up cost after installation*. Our approach comprises 60 NVIDIA Tegra K1 SoC that is 240 ARM CPU cores and 60 GPUs offering a theoretical peak performance of more than 21 TFlop/s at a total power dissipation of less than 1 kW with no additional energy costs due to an insular solar power supply and battery system. We show for a range of very versatile numerical kernels, that compared to commodity CPUs and -accelerators, energy efficiency is enhanced to a great extend. Also, we demonstrate, that such a system can be built by means of mass-market components and that it works properly with a 7.5 kWp solar power supply and a 8 kWh battery. The remainder of this paper is organised as follows: In Section 2 we provide a deep-as possible insight into all components of the project. We then dedicate Section 3 to evaluating the system, putting a clear focus on the HPC aspects but also presenting first results concerning the whole system. Finally, we conclude in Section 4.

2 System design

ICARUS is short hand for **I**nsular **C**ompute center for **A**pplied Mathematics, powered by **R**enewables, built upon **U**nconventional hardware combined with high-end **S**imulation Software. It is intended to be a system integration pilot project covering two pillars of the energy revolution, namely renewable energies and energy-efficient consumers [9].

2.1 System overview

There are several basic design principles for ICARUS: All energy consumers have to fulfill the latest standards regarding energy efficiency. For the digital components such as switches for instance, the IEEE 802.3az [13] standard has to be applicable. The system has to be independent, which means in particular, independence from the public energy grid and any architectural constraints. The reason for this choice is to free it from any infrastructural necessities in order to maintain versatility of operation. For instance with its holistic design, ICARUS can be used standalone in areas with little or no power grid development. The supercomputer component as well as its housing, cooling, management hardware,

(a) primary PV modules (b) secondary PV modules (c) assembled helix (d) full cluster

Fig. 1: ICARUS system construction site in March 2016 and cluster assembly.

solar power supply and battery storage must be able to be used in parallel, without inducing super linear cost in any regard (such as space, monetary- and energy cost) in order to be scalable. With respect to these paradigms, ICARUS is aggregated by the following key components: (1) A prototype of a compute-cluster built solely from compute nodes with mobile SoCs featuring programmable floating point accelerators. This is our main focus and is described in Section 2.2. (2) A state-of-the-art photovoltaic solar farm that is sufficiently dimensioned to provide power for operating the cluster under full load whole day plus charging the battery both *in summer* that is, with sufficient sun harvesting at weather in Dortmund, Germany. (3) For operation at night, a sufficiently sized battery rack is employed that is capable to power the cluster under full load after full charge for 8 hours without sunlight. (4) A simple housing that contains everything (except the solar modules of the PV farm). We achieve the goal of scalability by the design of a housing implemented by a modified oversees cargo container see section 2.3. Images of the fully assembled system can be found in Figure 1. Years after the ICARUS project started in 2013, there are several comparable approaches nowadays. Recently, NASA published a data center in a container, in order to be movable and scalable [18]. Another container-based data center is commercially available as a standalone, fuel- and battery-power supply driven resource [14]. Using mobile SoCs in the context of HPC and building small clusters of unconventional hardware [5, 11] as well as exploring Jetson TK1 for this purpose has also been performed [20] or at least considered [16] by others. However, to the best of our knowledge there is currently no group or enterprise that has driven this kind of system integration this far and ICARUS is the only container-based system combined with customised renewable energies power supply.

2.2 The Tegra K1 cluster

The system's core component is the NVIDIA Jetson TK1 development board released in late 2014. The Tegra K1 chip is a SoC hosting a quad-core 32 Bit ARM Cortex-A15 CPU and a programmable Low-Power Kepler GPU sharing the DRAM. The chip is of special interest because of the CUDA-capable GPU promising a theoretical (single precision) performance of around 300 GFlops/s at a power dissipation of ca. 10 W. The Jetson is a carrier board intended as

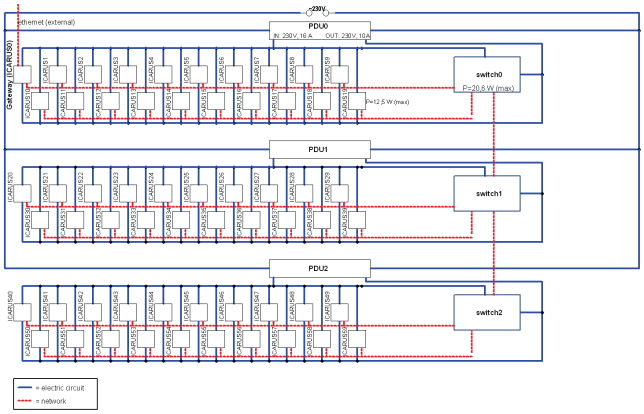


Fig. 2: Power-(blue) and network (red) topology of the cluster.

development environment for the Tegra K1 SoC. It includes everything to be used as a standalone, 'single-circuit' computer, featuring (inter alia) a GigaBit Ethernet adapter, a small fan for cooling the SoC, an SD-card slot (which we use for secondary storage) and a Ubuntu-based Linux OS [19]. In the course of this paper, we denote a single Jetson board to be one compute node in ICARUS. For comparison in Section 3, we employ two workstations representing different hardware generations, featuring (1) a Haswell CPU and a GeForce 980 Ti, representing the high-end in commodity (desktop) computer hardware. (2) An older IvyBride CPU alongside GeForce GTX660 and Tesla K20x GPUs, representing an average workstation with desktop- and compute GPUs. Hardware details can be obtained from Tables 1 and 2. It must be noted, that the Jetson TK1 is not exactly intended to serve as a cluster node. A slightly over sized fan and (for the purpose of HPC) unwanted board components such as I/O pins stemming from the intention to be used in embedded systems both induce a power dissipation malus. The greatest drawback of the board is its comparatively small RAM (2 GB). However, recently, the Tegra K1 has also been released as a card-sized compute module [22]. In addition, the 64 Bit follow up to the Tegra K1, called Tegra X1 has become available in 2016, featuring the augmented 1.9 GHz ARM Cortex-A57 CPU, a 1 GHz Maxwell GPU, almost doubling the theoretical peak performance via its much better LPDDR4 memory interface.

The network in ICARUS is composed of three 28 port GiB Ethernet switches (Cisco SG300-28) with a switching capacity of 56 GB/s and a power dissipation of 19-20 W peak only due to fanless cooling. We depict the network topology in Figure 2. Note that for technical reasons, we provide access to the cluster via a dedicated gateway node. The additional Ethernet port on that board is provided by a compatible Mini-PCI-e-to-Ethernet adapter. The on-board eMMC memory (16 GB) is used for the operating system and primary data. In addition, we provide each with a 128 GB Ultra SDXC 128 GB 40MB/s Class 1 SD-card. For mass storage, the Max-Planck Institute for the Dynamics of Complex

	i5-3470	i5-4690K	Jetson TK1
micro-architecture	Ivy Bridge	Haswell	Cortex-A15 (Tegra K1)
N_{cores}	4	4	4
clock speed	3.20 GHz (turbo 3.60 GHz)	3.50 GHz (turbo 3.9 GHz)	2.3 GHz
L1-cache	4x 32 KB + 4x 32 KB	4x 32 KB + 4x 32 KB	32 KB + 32 KB
L2- / L3-cache	4x 256 KB / 6 MB	4x 256 KB / 6 MB	2 MB / –
memory type	DDR3	DDR3	LPDDR3
peak memory bandwidth	25.6 GByte/s	25.6 GByte/s	14.9 GByte/s
P_{base}	51 W (Intel chipset)	41 W (Intel chipset)	3.9 W (Jetson TK1)
release date	Q2'12	Q2'14	Q2'14

Table 1: CPU Hardware details and measured base (idle-) power of carrier environments.

	GTX 660 / Tesla K20x systems	GTX 980 system	Jetson TK1
micro-architecture	Kepler	Maxwell	Kepler
memory type	GDDR5	GDDR5	LPDDR3
peak memory bandwidth	144.2/250 GByte/s	336.5 GByte/s	14.9 GByte/s
peak performance (SP)	1881/3935 GFlop/s	6054 GFlop/s	326 GFlop/s
peak performance (DP)	78/1312 GFlop/s	189 GFlop/s	13 GFlop/s
P_{base}	41/45 W (Intel chipset)	51 W (Intel chipset)	3.9 W (Jetson TK1)
release date	Q3'12	Q2'15	Q2'14

Table 2: GPU Hardware details and measured base (idle-) power of carrier environments. Note: first row is for two systems identical except for the GPU.

Systems has developed an energy-efficient RAID system intended to be used within ICARUS. This system is based on the BananaPi board and with its mere 50 W of peak power dissipation, it is a perfect device for ICARUS. All compute hardware and switches together (plus management hardware and power loss in the converters) ICARUS is calculated to be a less-than 1 kWp system. The boards (and PDUs, see below) are built into a single, modified rack unit whose side-panels have been removed for a maximum of passive cooling. The boards have been aligned in a 'double-helix' layout, which has proved itself to be very effective for avoiding heat-nests. This unique construction can be assembled using commercially available metal or plastics standoffs of different lengths. Full cluster images are depicted in Figure 5. Due to its new and unique design, some compounds had to be constructed from scratch, such as a mount for the Jetson TK1 power adapter which we constructed using 3D-printing.

2.3 Power supply, housing, cooling

The photovoltaic farming is implemented by 30 solar modules (Heckert Nemo 60 P) with a single peak power generation of 255 W each, resulting in a 7.65 kWp solar farm. The high output is needed due to the need of charging the battery whilst providing an additional 1 kW of power for the (peaked-out) cluster. DC/AC conversion is done by 2 converters (SMA Sunny Boy) and the energy-buffering (i.e. control of battery charge/discharge in conjunction with providing solar power to the consumers) is performed by an island converter (SMA Sunny Island). As power distribution units (PDU), we employ 3 rack PDUs for vertical installation (APC Rack PDU 2 G AP8959, see Figure 2) with 24 outlets each. To one of these, we attach a sensor for temperature and humidity. These PDUs can be remotely used for monitoring and control the different banks/outlets. In addition, for the purpose of double checking, to each AC-inlet, we attach a

high-sampling-rate energy meter that connects via Bluetooth to a central management unit (SMA Sunny HomeManager). This way, we can monitor power dissipation levels even 'in front of' the PDUs. Both, climate and power data is collected by a dashboard-system that runs on a RaspberryPi, adding only negligible power consumption. For energy storage, we use a lithium ion battery rack (HOPPECKE sun powerpack premium), scaled for providing close to 8 kWh of energy for the night (and day times with weather providing too low power levels from the solar system). In order to be independent from any architectural infrastructure, we designed all subsystems for being able to be packed into a heavily modified overseas cargo container a so called Steel Dry Cargo Container (High Cube) with dimensions $20 \times 8 \times 10$ feet. Here, the main task is to provide a climate-proof isolation in order to keep the hardware cool in summer and warm in winter times. For this purpose, we lined the walls, roof and floor with a 120 mm commodity heat-isolation. In addition, we provide it with fans (three inlets, three outlets), powered by secondary PV units in order to induce a proper airflow within the container for ventilation and cooling, see Figure 1(b). In winter, these fans can also be used to heat up the airflow at the inlet.

3 Exploring the system's limits

3.1 Hardware- and energy efficiency, scalability

In the following, we will provide energy- as well as performance measurements. Energy measurements are provided via taking the power P at the AC inlet of the carrier system, multiplied by execution time T. In the case of cluster benchmarks, we include total power consumption that is, including dissipation induced by all electric consumers of the system such as switches, converters, etc ... For energy measurement, in this study, we consider an ideal race-to-idle situation, where a core is either 'on' (i. e., operating at a preset peak frequency) or 'off' (i. e., cut off from the system clock) and neglect frequent adjustments of voltage and clock speed as well as any dynamic power dissipation due to heat.

Results for the single node measurements for the general matrix matrix multiply kernel are depicted in Figures 3. In the (two leftmost) plots denoted CPU, we show the results (log scale) for different numbers of cores: with increasing core count, performance increases (data point 'moves' to the right), and E decreases (data point 'moves' downwards) since power behaves like $P = P_{\text{base}} + kP_{\text{core}}, k = 1 \dots N_{\text{cores}}$. We employ kernels based on the newest versions of OpenBLAS on the CPUs and cuBLAS , respectively, on the GPUs. What we can find first is, that the Cortex-A15 cannot compete with its x86 counterparts for computationally intense tasks (as expected). Note that this is not the case for memory-bound codes, that is less computationally intense kernels with lower flop per byte ratio. All three CPU architectures behave as expected for this type of task, when increasing the number of threads used (i.e. good scaling) with the exception of the Cortex-A15 on 4 threads suffering due to its comparatively thin memory interface. However, the primary design paradigm of ICARUS was the exploitation of the GPUs. Hence, in the remainder of the single node benchmarks in

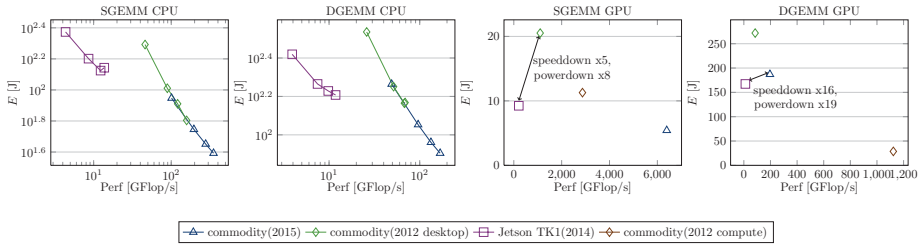
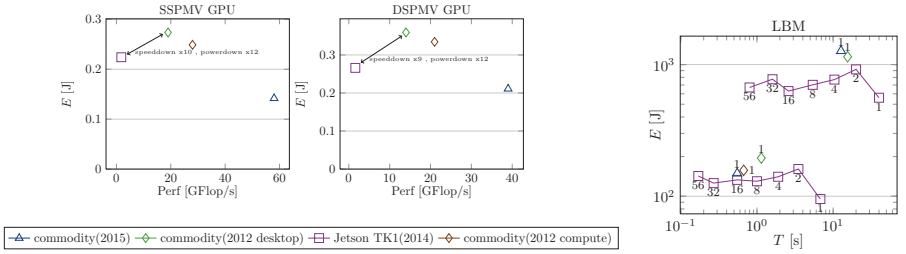


Fig. 3: Total energy consumption (E) and performance (Perf) of the dense matrix matrix product in single (SGEMM) and double (DGEMM) precision for all covered hardware architectures.

this paper, we concentrate on the GPGPU architectures. Here, for S/DGEMM we can find that the Tegra K1 can beat the GTX660 GPU easily in terms of energy to solution (as a metric for energy efficiency). This is due to the mobile chip achieving 210 GFlop/s in single and 12 GFlop/s in double precision respectively, both at approximately only 14 W power dissipation in its host system. The GTX660 on the other side can offer 1000 GFlop/s in single at around 171 W and – with slightly more power – 1.6 GFlop/s using 64Bit precision. In the plots, we provide speedup a well as power-down values between the Tegra K1 and the respective other systems, that ultimately lead to this higher energy efficiency. Note that concerning energy-to-solution, the low-power Kepler GPU can even outperform a compute card of that time, the Tesla K20x in single precision. Taking the high-end GTX980 Ti into account, the Tegra has to surrender to its tremendous more than 6000 GFlop/s in single precision sustainable performance at an average overall power dissipation of 271 W. Surprisingly, with DGEMM, the relation between performance and power favors the Tegra K1, which can be addressed to the 980 Ti being almost 30 times slower with double precision than with 32Bit data. This phenomenon however is not present in the comparison with the Tesla model. However, the fact, that the Tegra can even compete with (slightly outdated) commodity floating point specialists on this 'far end' of the range of computational intensities is promising when taking the advances on this segment of the chip market into account, even already with the Tegra X1, that virtually doubles performance at constant power.

As a common member of the class of memory-bound operations (i.e. low flop per byte ratio) we examine the sparse matrix vector multiply (SpMV). This kernel is versatile (especially in the context of PDE-based simulations) and very well understood regarding optimisation for GPUs. In previous work we have demonstrated how very sophisticated multigrid solvers can be constructed out of combinations from calls to SpMV based on ELLPACK-type storage and kernels [7]. Benchmark results are given in Figures 4 analogously to those in the GPU part of the S/DGEMM results. Modelling the relative performance of this type of kernel on different architectures boils down to the comparison of the respective memory interfaces. Here, only more on-chip memory bandwidth can



(a) SpMV (b) LBM

Fig. 4: SpMV and CFD benchmarks. Left: SpMV performance and energy to solution. Right: LBM solver time- and energy to solution (upper data series: CPU, lower series: GPU).

generate speedup. As one can see in the results, the speedups perfectly align with the factor that lies between the values for memory bandwidth: The LPDDR3 memory of the Tegra SoC can only a tenth of that of the GTX660. With its 12 times lower power dissipation however, the Jetson board remains more energy-efficient than its desktop counterpart as well as the Tesla card, regardless if computing in single or double precision. However now, the Tegra system stands no chance against the advanced 340 GByte/s interface of the GTX980 Ti.

As a final benchmark, we demonstrate the effectiveness of the full ICARUS Tegra K1 cluster with a sophisticated CFD solver based on the Lattice-Boltzmann method, optimised for GPU as well as CPU execution [8]. In Figure 4 we depict how energy and time to solution behave in a strong scaling test in single precision (note, that this time, a smaller value on the x-axis means higher performance). We give the used number of nodes for each data-point and, in the CPU case, use four threads per node. We also relate the cluster results to the competitor workstations as in the S/DGEMM benchmark. Concerning the total energy consumption, we add the measured energy consumed by the switches needed for the respective number of nodes (that is every 20 nodes add the energy value of the switch). This can be seen for instance in the rise of the energy level when going from 16 to 32 nodes. In both CPU and GPU configurations, the ICARUS systems scales well and provides higher energy efficiency then the respective architecture with the host-workstations. Note, that the increase in power when using additional nodes is very small and is dominated by the necessity to use an additional switch. The potential for scaling up the cluster is therefore quite high. We can also determine the number of ICARUS nodes needed for beating the reference workstations in terms of time to solution: For the Cortex-A15, we can see, that with 4 or more ICARUS nodes, lower execution time is needed than with the commodity hardware, at a considerably lower energy consumption. This state is reached with 16 ICARUS GPUs, where the combined GK20a beat even the most augmented floating point accelerator at the time of writing this paper in both, performance and energy efficiency.

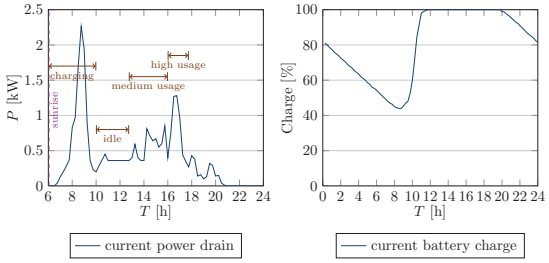


Fig. 5: Typical daytime solar power provision and nighttime battery discharge cycles.

3.2 Energy supply, temperature and humidity

For solar systems, the solar cycle is of major importance and it is elemental to know the time-spread of the hours of sunshine, which additionally includes charging breaks effected by cloudy conditions. Figure 1 (a, left) shows power P over time T in April, with sunrise at 6 am and sunset at 9 pm. The complete charging power of the solar system can be used, because the energy spent over night needs to be recovered, and the battery charging status is entering a hot-loading-phase in which it reaches a peak at 2.6 kW (this value can rise up to 7.5 kW). After fully charging the battery, the power decreases to the usage of the compute cluster in idle mode at approximately 0.36 kW between 11 am and 1 pm. Afterwards, the energy consumption of the cluster increases due to some calculations performed on it. Figure 1 (a, right) shows the percentaged charge of the battery in May for two different load intensities on the respective previous day. Here it can be seen, that even on slightly cloudy days, it is possible to reach the full charge of the battery, proofing that the dimensioning of the power supply system is correct for the current cluster size. Concerning cooling of the system, currently we observe that the climate in the server room is very stable and beneficial for the cluster: on the warmest day in July (with 31 degrees Celsius external temperature and around 50% relative humidity) we measure an average ambient temperature of 33 degrees Celsius and an ambient relative humidity of around 35% within the conmtainer. The Tegra boards are usually as cool as 39-43 degrees in idle mode and up to 53-68 degrees under load, which prooves our custom made cooling system to be sufficiently dimensioned.

4 Conclusion, discussion, and future work

Since starting operation in March 2016, ICARUS has passed all our expectations. Even almost three years after starting its design, we were able to show, that the Tegra K1 can compete with state-of-the art commodity hardware. In this paper, we are the first to publish a system-integration success that combines a technology-mixture from these very different fields. However, we have only just begun to explore the limits of the cluster and its power supply systems. Also,

the dynamics of the mobile compute hardware market is so fast, that hardware from a current generation, i.e. Tegra X1 must be added. All together, we find our approach for energy-efficient HPC based on unconventional embedded hardware to be well worth the effort.

Acknowledgments. ICARUS hardware is financed by MIWF NRW under the lead of MERCUR. This work has been supported in part by the German Research Foundation (DFG) through the Priority Program 1648 'Software for Exascale Computing' (grant TU 102/48). We thank the participants of student project Modeling and Simulation 2015/16 at TU Dortmund for initial support. We also want to thank Markus Borowski at Borowski GmbH for advice regarding the solar farming and battery supply as well as Björn Henkel at Bloedorn Containers for his advice in designing the container unit.

References

1. Anzt, H., Quintana-Ortí, E.S.: Improving the energy efficiency of sparse linear system solvers on multicore and manycore systems. Phil. Trans. R. Soc. A 372(2018) (2014)
2. Benner, P., Ezzatti, P., Quintana-Ortí, E., Remón, A.: On the impact of optimization on the time-power-energy balance of dense linear algebra factorizations. In: Aversa, R.e.a. (ed.) Algorithms and Architectures for Parallel Processing, Lect Notes Comput Sc, vol. 8286, pp. 3–10. Springer (2013)
3. Castelló, A., Duato, J., Mayo, R., Peña, A., Quintana-Ortí, E., Roca, V., V, S.: On the Use of Remote GPUs and Low-Power Processors for the Acceleration of Scientific Applications. In: ENERGY 2014, The 4. Int Conf on Smart Grids, Green Commu and IT Energy-aware Tech. pp. 57–62 (2014)
4. Feng, W., Cameron, K., Scogland, T., Subraumaniam, B.: Green500 list (jul 2015), http://www.green500.org/lists/green201506
5. Fürlinger, K., Klausecker, C., Kranzlmüller, D.: Information and Communication on Technology for the Fight against Global Warming: First International Conference, ICT-GLOW 2011, Toulouse, France, August 30-31, 2011. Proceedings, chap. Towards Energy Efficient Parallel Computing on Consumer Electronic Devices, pp. 1–9. Springer Berlin Heidelberg, Berlin, Heidelberg (2011), http://dx.doi.org/10.1007/978-3-642-23447-7_1
6. Geveler, M., Reuter, B., Aizinger, V., Göddeke, D., Turek, S.: Energy efficiency of the simulation of three-dimensional coastal ocean circulation on modern commodity and mobile processors – a case study based on the haswell and cortex-a15 microarchitectures. In: Workshop on Energy-Aware HPC. LNCS, ISC '16, Springer (June 2016), accepted
7. Geveler, M., Ribbrock, D., Göddeke, D., Zajac, P., Turek, S.: Towards a complete FEM–based simulation toolkit on gpus: Unstructured grid finite element geometric multigrid solvers with strong smoothers based on sparse approximate inverses. Computers and Fluids 80, 327–332 (Jul 2013), doi: 10.1016/j.compfluid.2012.01.025
8. Geveler, M., Ribbrock, D., Mallach, S., Göddeke, D., Turek, S.: A simulation suite for Lattice–Boltzmann based real–time–CFD applications exploiting multi–level parallelism on modern multi– and many–core architectures. Journal of Computational Science (2), 113–123 (Jan 2011), doi 10.1016/j.jocs.2011.01.008

9. Geveler, M., Turek, S.: Icarus project homepage (2016), http://www.icarus-green-hpc.org
10. Göddeke, D., Komatitsch, D., Geveler, M., Ribbrock, D., Rajovic, N., Puzovic, N., Ramirez, A.: Energy efficiency vs. performance of the numerical solution of PDEs: an application study on a low–power arm–based cluster. J Comput Phys 237, 132–150 (2013)
11. Grasso, I., Radojkovic, P., Rajovic, N., Gelado, I., Ramirez, A.: Energy efficient hpc on embedded socs: Optimization techniques for mali gpu. In: Proceedings of the 2014 IEEE 28th International Parallel and Distributed Processing Symposium. pp. 123–132. IPDPS '14, IEEE Computer Society, Washington, DC, USA (2014), http://dx.doi.org/10.1109/IPDPS.2014.24
12. Hager, G., Treibig, J., Habich, J., Wellein, G.: Exploring performance and power properties of modern multi-core chips via simple machine models. Concurrency Computat.: Pract. Exper. 28(2) (2016)
13. IEEE: Ieee 802.3 standard (2015), http://standards.ieee.org/getieee802/download/802.3bm-2015.pdf
14. InfoTech: Mobile Data Center MDC40 (2015), https://www.infotech.de/2_MDC40/2015_Oktober/Data%20sheet.pdf
15. Malas, T.M., Hager, G., Ltaief, H., Keyes, D.E.: Towards energy efficiency and maximum computational intensity for stencil algorithms using wavefront diamond temporal blocking. CoRR abs/1410.5561 (2014), http://arxiv.org/abs/1410.5561
16. Mantovani, F.: High performance computing based on mobile embedded processors. International conferences, Mont-Blanc Project (2015), url: https://www.montblanc-project.eu/sites/default/files/publications/Mont-Blanc-EMiT15-lq-public.pdf
17. Meuer, H., Strohmeier, E., Dongarra, J., Simon, H., Meuer, M.: Top500 list (jul 2015), http://top500.org/lists/2015/06/
18. NASA: High End Computing Capability, Project Status Report (2015), https://www.nas.nasa.gov/hecc/assets/monthlies/pdf/HECC_10-15.pdf, modular Supercomputing Facility
19. NVIDIA Corp: NVIDIA Jetson TK1 Development Kit - Bringing GPU-accelerated computing to Embedded Systems (2014), http://developer.download.nvidia.com/embedded/jetson/TK1/docs/Jetson_platform_brief_May2014.pdf
20. Paolucci, P.S., Ammendola, R., Biagioni, A., Frezza, O., Cicero, F.L., Lonardo, A., Martinelli, M., Pastorelli, E., Simula, F., Vicini, P.: Power, energy and speed of embedded and server multi-cores applied to distributed simulation of spiking neural networks: ARM in NVIDIA tegra vs intel xeon quad-cores. CoRR abs/1505.03015 (2015), http://arxiv.org/abs/1505.03015
21. Rajovic, N., Rico, A., Vipond, J., Gelado, I., Puzovic, N., Ramirez, A.: Experiences with mobile processors for energy efficient hpc. In: Design, Automation Test in Europe Conference Exhibition (DATE), 2013. pp. 464–468 (March 2013)
22. Toradex: Tegra K1 System on Module - Pressemitteilung (2016), https://www.toradex.com/de/news/toradex-embedded-computer-nvidia-tegra-k1
23. Treibig, J., Dolz, M.F., Guillen, C., Navarrete, C., Knobloch, M., Rountree, B.: Tools and methods for measuring and tuning the energy efficiency of HPC systems. J Scientific Programming 22, 273–283 (2014)
24. Wittmann, M., Hager, G., Zeiser, T., Wellein, G.: An analysis of energy-optimized lattice-boltzmann CFD simulations from the chip to the highly parallel level. CoRR abs/1304.7664 (2013), http://arxiv.org/abs/1304.7664