Turing Laureate Jack Dongarra discusses HPC challenges: AI's impact on precision computing

For over five decades, Jack Dongarra has been a driving force in high-performance computing (HPC), revolutionizing how we approach scientific computing and data analysis. He discovered a passion for computing during an internship at Argonne National Laboratory. His work at Los Alamos on the Cray-1, one of the first vector supercomputers, was pivotal in shaping his career. He went on to co-create the LINPACK benchmark and the TOP500 list, which has ranked the world's fastest supercomputers since 1993. His innovations, including linear algebra softwares and the Message Passing Interface (MPI), have enabled software to keep pace with rapidly evolving hardware.

Today, Dongarra holds joint positions at the University of Tennessee and Oak Ridge National Laboratory. His contributions power modern scientific research, earning him the 2021 ACM A.M. Turing Award, often called the "Nobel Prize of Computing".

At the PPAM 2024 conference, organized by Czestochowa University of Technology, with IT4Innovations serving as a co-organizer for this special jubilee edition, Lubomír Říha, Head of the Infrastructure Research Lab at IT4Innovations, had the privilege of interviewing the esteemed Jack Dongarra.

Over the course of your career, you've been involved in numerous projects that have become foundational for high-performance computing (HPC) applications and libraries. Reflecting on your work, which project do you consider the most significant or personally interesting?

Dongarra: I have been fortunate to be involved in many projects. Some of those projects have been very successful, but not all of them. And some of them have become part of how we do computing, or we think about computing, or people are used to do computing. The project that has had the most impact is probably MPI. The Message Passing Interface is used by all programs, including scientific programs that use parallel processing. Today, almost everything is done in a parallel way. They all use this interface, which was designed by the community. It's community-driven. It's a standard today, it's a de facto standard. It's not approved by any high organization or entity that approves standards. It was defined in a very, what I'll call organic way. The community defined it. There was a need for a standard, and the community got together and created a very quick, over a very short period, standard that was immediately accepted by the community, the computational community, and the scientific community.

The project was short in duration. It had a lot of smart people involved in building it. It had immediate acceptance, and it has had a tremendous impact on the community. I would say it was fun doing this project and it was also most interesting because of its

impact on the community. But I've been involved in many projects. That's just one of the many that I've been involved in.

And by 'short,' you mean how long exactly?

Dongarra: Year and a half. Within a very short period, basically 18 months, we went from having nothing to having the standard that was put in place and adopted. There were roughly 40 people who were working on it seriously over that period. There was no, what I'll call, funding for the work. They were doing it because they wanted to do it, and they were able to use time at their research office to do this work. It was a tremendous success.

We're here at the PPAM conference, which you've attended 11 times. What is it about this event that keeps drawing you back year after year? And how long have you known Roman Wyrzykowski, the conference's main organiser?

Dongarra: Of course, yeah, Roman I've known since my first PPAM conference. And I think, I'm not sure how we were introduced. There is various paths that we cross in going to meetings. I'm sure I've talked to him. He must have heard me speak and invited me to come to Poland. Why do I keep coming back? Because I keep getting asked to come back. That's one reason. I enjoy seeing different places in Poland. I've been to many cities in Poland that I had never been to before. It's a fun conference in the way it's organized and it usually has a good group of people attending. I just enjoy coming back to seeing old friends like this.

What drives your passion for high-performance computing and numerical algorithms? And looking ahead, how do you foresee the field evolving in the next few years?

Dongarra: I'm trained in numerical linear algebra. That's where my basic formal training is and I think I can make a contribution because of my knowledge of supercomputing. Knowledge of supercomputing means, we want to make things run fast. We want to use the features that are in the architecture to help drive the software that's developed. We want to design algorithms that can run efficiently on today's computers as well as computers in the future. We're motivated to get something which delivers an accurate solution and a solution that can be obtained with utilizing the hardware as best it possibly can. We want to have something which is fast, accurate, and I'll also say portable. We want to have something in which we can move from one environment to another without making major changes to it. Those are all things that go into high-performance computing and numerical algorithms. It's something which I think presents challenges and allows one to use their knowledge of computing.

And how will things change in the future? It's always hard to predict what's going to happen. We see major changes that have taken place over my 50 years. We went from a sequential computer to a computer that had vector instructions to computers that utilized shared memory, parallel computers to machines that had distributed parallelism associated with it to architectures that added graphical processing units to accelerate the performance. What's the next step is the question. I really can't predict what the next step will be. If I had to look at a computer today, we see a, I'll call it conventional CPU, and then something to boost the processing performance. We think of that as an accelerator to boost the performance. And today, that's a GPU. But tomorrow, it can be something different. That is, we can add to that. CPUs and GPUs are going to do something in the near future that will cause them to be more tightly coupled. And then we can think of augmenting that with different kinds of computers, maybe something that specializes in AI-related stuff or something that is using neuromorphic computing where our brain operates or using something that's based on analog computing, something that was done many, many years ago and maybe can have a place today. Augmenting what we have, putting together more things than just accelerators to boost performance. Maybe that's a way forward for the future.

What do you view as the biggest challenges currently facing high-performance computing, both from a technical standpoint and in terms of wider adoption? Moreover, how do you see emerging technologies like quantum computing and AI integrating with future HPC solutions?

Dongarra: The significant technological challenges for high-performance computing relate to how we can effectively use the hardware in place. That's one thing. But there's a much broader thing going on. I think of it in terms of how we buy computers. In the scientific area, we buy a scientific computer using the following process: We're given a certain amount of money, usually given to us by the government. They say you have this money to buy your next supercomputer. Then, we ask the computer vendors to bid on a computer that fits that amount of money and has a certain performance capability. That's the way we buy it. We have an offer that says, submit a bid for a computer with this much computing capability and we have this much money to give you. The vendors then go off and design a computer and try to match the peak performance we asked for with as low a cost as possible. And that usually creates a poorly balanced computer. Theoretically, it has high peak performance, but it's very hard to reach, and it‘s almost impossible to reach. A better way to design a computer would be to get the application people together with the computer architects and the software designers and have them discuss how to build a computer for the application. It's all about the application. We build these computers to solve the applications' challenging problems. Let's get the application people involved in designing a computer.

What happens today is this computer is designed, and it's purchased, and it gets thrown over the fence, is the way I look at it. The vendors throw it over the fence and

the applications people then scramble for the next few years to figure out how to make their applications run effectively on this computer. They finally do. It runs okay on this new computer but by the time they get that figured out, a new computer is given to them because that takes about four or five years to figure out how to use this. And by that time, we have another computer being given to us or thrown over the fence. We need to have a better way of co-designing the computers with the computer architects, the application people, and the software people in the same room designing them. One of the challenges is designing a machine that can effectively be used without major changes to the software. And things like AI are making tremendous impacts on how we solve problems today.

AI is a major contributor to most applications, and it's going to continue to have a positive impact on how we do our applications. I view it as a tool that computational scientists will use to help them solve their problems.

Now, quantum computing is an interesting technology. It's something which I think is overhyped, in the sense that people expect a lot from it, and it probably won't deliver on that expectation, at least not in the near term. I think quantum computing is an area where we should invest in research to understand how we can effectively use that technology. And when we finally get a better understanding, we can then have the technology be ready to be implemented in designing a quantum computer that could be used.

My concern is that there's too much hype and too much effort being spent on quantum computing today. We should step back and view it as a research effort and understand how it could be done and how it can fit into our overall ecosystem of solving things.

In the early 1990s, HPC shifted from using dedicated chips like vector machines to commodity hardware, employing CPUs originally designed for servers and other processing tasks. This transition marked the beginning of massively parallel machines. Recently, AI workloads have driven changes in GPU accelerator architecture, with more silicon being allocated to units that deliver significantly higher performance at lower precision, rather than the double precision traditionally used in HPC. Do you see this shift as a positive or negative development for HPC, which still requires high double precision?

Dongarra: You know, we've had this shift, as you point out, from building dedicated computers to using commodity parts in our computers. That shift took place because the commodity parts became much faster. That happened as a result of Moore's law and Dennard scaling. It just allowed us to put stuff together and develop something that has a broad usage. The result could be that the actual cost per processor is very low due to having those commodity pieces. With the dedicated specialized machines built in the old days, the market wasn't large enough to sustain them. The result is those companies went out of business, and commodity processors became the thing.

What happens in the future as AI becomes more important and architectures shift to being dedicated to servicing AI instead of servicing? Let's call it traditional scientific computing. Well, one thing is AI doesn't need double precision. AI can get by with short precision. The concern is that new processors may give up the higher accuracies seen on chips today. So, 64-bit floating point arithmetic and 32-bit floating point arithmetic, those parts of the chip may not be there. The chips may only serve the AI needs, which are 16-bit, 8-bit, or even 4-bit floating point arithmetic. That would leave the traditional scientific computing community without the ability to carry out computations at higher precision. It may lead to some issues in terms of some of the accuracies that we can obtain. That's the problem.

What happens if that takes place? I don't know. But it's starting to encourage us to look at using lower precision in our numerical computation and having mixed precision. I think that's a great research area, but utilizing both the lower precision and the higher precision selectively and getting that results. I hope that the higher precisions don't go away totally, but I could see where vendors may choose to develop chips which do not have the higher precision, that they use that space for doing other functions that may be related to AI. That may cause the chips with double precision to be more expensive and would limit us somehow. It's an active area of research how we can use mixed precision, how we can get by using shorter precision. I think all of that stuff is a good research area, and I would like to see double precision remain in our hardware.

What do you expect to be the key technology drivers behind the next major leaps we might see in the TOP500 machines?

Dongarra: It's hard for me to predict what's happening. The thing that is around the corner is the ability for this integration to take place between the CPU and the GPU. That's something that will have a big impact. We've seen an impact that's resulted from, as I usually say, the most important aspect of our high-performance computing is not the rate at which we do the computation; it's the rate at which we can move data around to do the computation, so the data's ready to do the computation. The most expensive part is the communication. If we can reduce the communication costs on our computer, then we can save, and it will be more effective. One thing that's helped is having stacked memory. We think about stacking memory today. There's a three-dimensional stack of two-dimensional components that allows data to move in that third dimension so that it's much faster than if we had to spread that out across a two-dimensional surface. What will happen to the integration is today we have a stack of memory and next to it is the CPU. There still is data that has to be transferred from the stack into the CPU, and that's using links that are necessarily small, short, and not very efficient. In the future, the CPU will be integrated with the memory stacks so that the data can flow in that third dimension into the chips at a much higher rate of speed and hopefully improving the overall effectiveness of that. The speed of data movement will be reduced as a result. I think that's a promising area, but I don't know whether that's a

major leap or not. Again, having more things added to the chain of CPU, GPU, and other components that may accelerate parts of the computer may help. I see a day when we might be able to dial up what we put into our computers in the sense of having components that are conventional CPUs, components that are accelerated numerical processors, and maybe having something that relates to an analog computer that we can add into the mix, maybe things that have neuromorphic capability. And depending on the application mix, we can dial up what we buy in terms of our computer and making that to be more effective for the applications that we have.

Looking ahead in the US, the upcoming El Capitan system at Lawrence Livermore National Laboratory is based on APU units – CPU and GPU accelerators integrated into the same chip with shared memory. What impact do you think this fully coherent memory model will have on HPC application and library development?

Dongarra: It certainly simplifies things. We don't have to explicitly send data from one part of the CPU to the accelerator part do the computation, and later retrieve the data. It'll all be available in memory in a very natural way so that the flow of information is going to be sped up in a way that makes us more effective. This is going to be a big change, I think, in the way we do business. It'll simplify the programming stack and the programming components that we use. It will allow us to be more efficient in terms of using those components, and it'll be a more natural way in which the CPU and the part of the machine that does the numerical computations are integrated. Overall, it's going to be a big change, a positive one. It's something that we've long, it's a long time in coming, and we're finally about to see it in place, and I think it'll simplify our use of these machines.

In addition to your work in the US, you've previously reported on Chinese supercomputers. Is there still active collaboration with Chinese labs, or did this diminish when China stopped submitting machines to the TOP500? Could you offer any insights into the current state of HPC development in China?

Dongarra: The Chinese, of course, have an interest in developing high-performance computers, and they use them, as many countries do, for doing scientific computations, and the US government made a decision to restrict certain components from going to China. That restriction imposed limits or boundaries or conditions under which the Chinese can acquire high-performance computing technology. In particular, things like GPUs of a certain range are not allowed to be sold. CPUs of a certain kind are not allowed to be sold in China. You know, I think that initially stopped China because they were developing machines using Western technology; it stopped China and caused them to pivot and develop and invest in their own technology. China today has a number of machines that are at or exceeding the capacity of some of the machines that are in the US. They have some exascale machines, and those exascale machines are

developed using Chinese parts. They have parts that have been designed and put in place that are running in China today. The question is where those parts were fabricated. Were they fabricated in China, Taiwan, or some other place? When I asked that question, my Chinese friends said they were done in Taiwan. When I press them, they say they were done in China. And when I asked them about it, weren't they done in Taiwan? They say, well, Taiwan is part of China. That's their answer to that question.

China has stopped submitting results. There are no new machines on the TOP500, I think, for the last two editions. That's a direct result of the US putting embargoes in place on technology going to China. I think the Chinese are concerned that if it's revealed that they have technology developed in China, then more restrictions might be applied. They don't want to cause more things to happen. I think it's unfortunate that it happened. Unfortunately, the restrictions were put in place. I think the result of that is China developing their own technology. That technology is being used today in China to develop supercomputers. And it's something which we don't, the Western countries, the US in particular, really don't have control over today. And we don't know what precisely they have.

There's a rumor that there are four machines in China. Three of them are what I would consider to be traditional large supercomputers, maybe exceeding what the US has at the exascale level. They are three different computers with different architectures. One machine is, again, a very impressive machine that's being used for AI-related things. So four computers, all with Chinese components, are being used to drive Chinese science and technology.