

Code Together Podcasts

Episode 26: Advancing Bioinformatics with Modern Hardware, HPC Compute + Software

Host: Nicole Huesman, Intel

Guests: Sergio Santander-Jimenez, University of Extremadura; & Ricardo Nobre, INESC-ID

Nicole Huesman (00:04): Welcome to [Code Together](#), an interview series exploring the possibilities of cross-architecture development with those who live it. I'm your host, [Nicole Huesman](#).

Bioinformatics uses computation to understand biological data. It's particularly useful for large complex data sets as used in determining gene and protein functions, establishing evolutionary relationships, and predicting 3D shapes of proteins.

Today, we'll hear from two guests who are working to advance health care through the use of bioinformatics.

[Sergio Santander-Jimenez](#) is an assistant professor in the Department of Computer and Communications Technologies at the [University of Extremadura](#), where he received his PhD in Computer Engineering. He has authored or coauthored over 60 publications, edited three special issues, and served as reviewer for more than 30 JCR indexed journals. His main research interests include evolutionary computation, multi-objective optimization, parallel and distributed computing, and computational biology. So great to have you with us, Sergio.

Sergio Santander-Jimenez (01:30): Thank you for the introduction. It's a pleasure to be here.

Nicole Huesman (01:34): [Ricardo Nobre](#), a researcher at [INESC-ID](#) in Lisbon, Portugal, currently focuses on high-performance computing, compilers, parallel programming and machine learning. He's contributed nearly 20 papers in international journals and conferences. He received his Ph.D. from the Faculty of Engineering from University of Porto. So great to have you with us, Ricardo.

Ricardo Nobre (02:01): It's a pleasure to be here. Thank you for the introduction.

Nicole Huesman (02:04): I'd also like to welcome Sujata Tibrewala, Intel's oneAPI Developer Community manager. Welcome back to the program.

Sujata Tibrewala (02:11): Thank you so much, Nicole, for having me and thank you Ricardo and Sergio for joining this program today. I know we have been working together for quite some time and Ricardo, congratulations to be our [oneAPI Challenge winner](#).

Ricardo Nobre (02:30): Thank you.

Sujata Tibrewala (02:31): So, Sergio and Ricardo, you're doing such important work that's helping advance the medical field. Can you help us understand this work in a little bit more in detail?

Sergio Santander-Jimenez (02:41): For sure. The research work that we are going to descend today is a collaboration between two institutions, INESC-ID, in Lisbon, Portugal, And the [ARCO Research Group](#) from the University of Extremadura in Spain. This collaboration lies in the scope of a Portuguese Research project, what is called a HiPERBio. Where we are investigating emerging hardware and teaching certain technologies to address time-consuming optimization problems. Particularly, we are interested in those problems that belong to the bioinformatics area, which is a hot research topic nowadays. And I'd like to introduce first, the ARCO research group, ARCO means computer architecture and logic design group; and this research group has been in existence since 1995; and is located at the Polytechnic School of the University of Extremadura. The group has to take both your ethical and applied research using vast computing technologies in some of our fields, but other than distributing computing or reconfigurable computing and multi-objective optimization, evolutionary computation and their applications to wireless sense of networks, and of course bioinformatics and biomedicines. Now Ricardo can talk a little bit about INESC.

Code Together Podcasts

Episode 26: Advancing Bioinformatics with Modern Hardware, HPC Compute + Software

Host: Nicole Huesman, Intel

Guests: Sergio Santander-Jimenez, University of Extremadura; & Ricardo Nobre, INESC-ID

Ricardo Nobre (04:03): So, INESC-ID is a research and development institution in Lisbon. In fact, it's one of the most dynamic research institutions in Portugal in the areas of computer science and electrical engineering. While promoting cooperation between academia and industry, research and development at INESC-ID focus on artificial intelligence, graphics and interaction, distributed systems, communication networks, and high-performance computing, just to name a few areas of research.

Sergio Santander-Jimenez (04:44): We're investigating new ways to take advantage of the computing capabilities of modern hardware to improve bioinformatics applications. The solution of informatics problems has important implications for society. We are talking about improving our understanding of biological mechanisms, genetic diseases, even evolution itself. The thing is bio applications are really hard to tackle. They are factored by multiple sources of computational complexity, such as high dimensionality, big data, exponentially increasing search spaces and time-consuming evaluation criteria among other things. So in the end, traditional serial algorithmic approaches are not suitable to satisfy the time to solution requirements of these problems. Therefore, there is a demand for use of intelligent optimization methods and efficient search approaches, approaches that exploit the capabilities of modern hardware. And not only that, an important issue here is to develop at the same time, - both portable and optimized solutions for the wider spectrum of hardware resources that we have nowadays.

Sergio Santander-Jimenez (06:03): So the application that we are most focused in this research is epistasis detection. I am going to explain a little bit the bio background, in a nutshell, and then Ricardo can talk a little bit more about the technical stuff. So, in our chromosomes, we have codified all the information about ourselves, the information that makes human beings be human beings. So, technically the expression of this information is called phenotype the physical manifestation of a genotype. And in our genome, there are some specific positions that have great practical interest because they tend to, show changes - substitutions at the single nucleotide, in a large percentage of the population. These are what we call the SNPs, single nucleotide polymorphisms. Researchers have demonstrated that the interaction between SNPs have an impact in the likelihood of important genetic diseases. For instance, Alzheimer and breast cancer.

Sergio Santander-Jimenez (07:12): So, the discovery of interactions between the SNPs can be modelled as an optimization problem where we study a case control data set to identify which interaction is the most likely to explain the whole picture. Why some individuals have the disease and why others don't have it? The thing is, well, we don't know how many its SNPs are involved in that hiding interaction. And the problem is that the larger the number of SNPs that are in that interaction, the harder the problem is, because the search space tends to increase in an exponential way. So, in order to address this bio background, we have devised some intelligent approaches that Ricardo can talk about now.

Ricardo Nobre (08:01): So, in particular this application that we ported from CUDA to [DPC++](#) targets third order searches. This means that three SNPs are taken into account at a time when searching for associations with a given phenotype, which can be for instance, a disease state. The problem with taking into account three SNPs at a time in contrast with taking into account only two, is that the number of combinations to process is much larger. For instance, in triplet searches, when we are considering only 20,000 SNPs, there is over 1 trillion, so, it's 10^{12} different sets of three SNPs to evaluate. And as you can imagine, this is computationally very demanding, because you will have to also consider that the data sets can have thousands or even millions of different samples, which are basically the patient genotype information. So, in my opinion porting the application to DPC++, has a very big added value.

Ricardo Nobre (08:44): Why? Our application only targeted CUDA compatible GPUs. Which means that it only works with hardware from a single source and it only targets GPUs. And since our interest is to find the best

Code Together Podcasts

Episode 26: Advancing Bioinformatics with Modern Hardware, HPC Compute + Software

Host: Nicole Huesman, Intel

Guests: Sergio Santander-Jimenez, University of Extremadura; & Ricardo Nobre, INESC-ID

possible combination of hardware and software to make epistasis detection searches as fast as possible, we want to be able to target a broader set of architectures and different computing devices

Sujata Tibrewala (09:43): That's really great. That's a great summary that you have trillions of matches to do as part of this application and you want to make it as fast as possible. And so, we wanted to avoid vendor lock-in right? So moving the application to DPC++ actually helped you so that it expands the reach of your application. And then you can find the best device that gives you the best performance. Right? Did I summarize it well?

Ricardo Nobre (10:13): Yes, yes, yes.

Sujata Tibrewala (10:13): I think one question that remains is how was your experience importing the CUDA application to DPC++?

Ricardo Nobre (10:21): Getting the first version compiling and producing the correct results was quite fast. I think I did it in just a few hours. It was less than a day of work. Of course, after that, to improve the performance on the different CPU and GPU targets that we experimented with on DevCloud, it took a little bit more work. But it was expected because we were targeting a single type of device with a focus also on specific architectures. Especially in the case of the execution on CPU in [DevCloud](#), there were small modifications that we needed to do to make the application execute at closer to the potential of the CPU.

Ricardo Nobre (11:02): For instance, I can tell you that just by changing the way that the data is indexed, we were able to improve performance on CPU by close to 6x. So, what this means is that, although the first version that we got from the [DPC++ Compatibility Tool](#) was not the fastest one. To improve its performance significantly didn't take us that much effort. Of course, there are more things to do after that, but we could get very good performance from both types of devices CPUs and GPUs without too much effort.

Sergio Santander-Jimenez (11:38): So in the end, we are talking about advantages that the DPC++ porting tool is a way to increase in a significant way that productivity from a programming perspective, but I'm getting a very satisfying performance so it benefits from two sides, and we are really pleased with the results.

Sujata Tibrewala (11:59): That's awesome. Ricardo touched upon productivity and performance, and you're mentioning even with performance, you're really happy. So, Ricardo, you mentioned that like, just by the way of indexing, you could increase the performance 6x. Where can the developer get these kinds of tips? If they did this kind of porting themselves, how would they know how to treat their port for performance?

Ricardo Nobre (12:26): In the version that is on the GitHub repository, those modifications are not yet there, but I suppose that we can push them to the repository and even put that information in some tutorial or some Wiki.

Sujata Tibrewala (12:57): So thank you, Ricardo, and Sergio for mentioning all the good experiences that you had importing the application. Were there any challenges or any roadblocks that you faced?

Ricardo Nobre (13:10): Regarding porting the application, since the original application was targeting only GPUs from a single source, there were some things that were taken for granted in the source code. So I think that was the part that required a bit more thinking because we needed to rewrite those portions of the code to work with different architectures. Other than that, I think that the output given by the [Intel Compiler](#) and the comments introduced on the source code by the DPC++ compatibility tool were quite good and allowed us to make the necessary code modifications to make the application run, to get the first functional application. After that, it was just an effort of optimizing the code to extract a little bit more performance.

Code Together Podcasts

Episode 26: Advancing Bioinformatics with Modern Hardware, HPC Compute + Software

Host: Nicole Huesman, Intel

Guests: Sergio Santander-Jimenez, University of Extremadura; & Ricardo Nobre, INESC-ID

Sujata Tibrewala ([13:56](#)): Yeah, no, that's a really good point. The compatibility tool can give a functional goal, but for the performance, of course, the developer will have to think through the different architecture that the code is now going to target, right? So it's not the end or a replacement of anything. So one last question, are you looking to expand to more architectures in the future?

Ricardo Nobre ([14:23](#)): Yes, we are planning to expand to FPGAs, in particular to the FPGAs that are available in the DevCloud because of ease of development and the fact that we have access to it for free. And of course, we are also interested in expanding to other architectures when they are supported by the oneAPI software stack, such as neural network processors from Intel, and other architectures that might be released in the future and might be integrated in these oneAPI initiative.

Sergio Santander-Jimenez ([14:50](#)): By using that oneAPI, what is going happen is that we will have a unified quicker solution of this bioinformatics problem, by orchestrating all the hardware resources that was one of most important targets in this research. Because as we mentioned at the beginning that to have the idea is okay. We have the idea, the algorithmic idea. Now we have to port it to the spectrum of resources that we have and orchestrate them in an accurate way to get our solution as fast as possible.

Ricardo Nobre ([15:28](#)): Exactly, and what the DPC++ compatibility tool allowed us to do was to arrive at the point where we only need to care about optimizing the code. If we didn't follow this path, we would have to convert the code by hand, which would take us much more time.

Nicole Huesman ([15:43](#)): The work that you are doing, it's so inspiring. It's really going to advance the medical field of healthcare, and we are so appreciative of your collaboration and what you're doing. So, we look forward to the continued collaboration and to seeing where you go, what you do next, and having you both back on the program. So, thank you. Sergio, as we wrap up today, where can listeners go to learn more?

Sergio Santander-Jimenez ([16:15](#)): More information about this work can be found, for now, in our [DevMesh page on the Intel website](#), that you can search by the keywords 'cross-architecture,' 'epistasis detection' on GPU and CPU devices. There, you got find insights on our methodology, technologies, and also very important our repository with source code.

Nicole Huesman ([16:38](#)): Excellent. Thank you. And Ricardo, any closing thoughts you'd like to leave listeners with?

Ricardo Nobre ([16:45](#)): If listeners want to learn more about this work, they can also take a look at our papers in IPDPS, JSSP 2020 and our paper in TPDS, it's an IEEE journal, that has been published very recently.

Nicole Huesman ([17:02](#)): Thanks Ricardo. We look forward to having you both at the oneAPI Dev Summit and Ricardo. Thanks so much for joining us.

Ricardo Nobre ([17:10](#)): Thank you.

Sujata Tibrewala ([17:11](#)): Thank you so much, Sergio, Ricardo and Nicole.

Sergio Santander-Jimenez ([17:14](#)): Thank you very much.

Nicole Huesman ([17:15](#)): And a big thank you to all of our listeners for joining us. Let's continue the conversation at oneAPI.com.

Code Together Podcasts

Episode 26: Advancing Bioinformatics with Modern Hardware, HPC Compute + Software

Host: Nicole Huesman, Intel

Guests: Sergio Santander-Jimenez, University of Extremadura; & Ricardo Nobre, INESC-ID

Learn more

- [Cross-architecture high-order exhaustive epistasis detection on CPU and GPU devices](#)
- [Accelerating 3-way Epistasis Detection with CPU+GPU processing](#). Paper presented in 23rd Workshop on Job Scheduling Strategies for Parallel Processing (JSSPP), New Orleans, 2020. DOI: 10.1007/978-3-030-63171-0_6
- [GitHub: Epistasis detection using DPC++ on Intel DevCloud source code](#)
- [Intel® DPC++ Compatibility Tool](#)
- [oneAPI.com](#)
- [Intel® oneAPI Toolkits](#)