PGI®

SUCCESS STORY

UQAC Leverages CUDA Fortran to Optimize Innovative Aluminum Welding Techniques

UQAC

CHALLENGE

Write code to simulate friction stir welding (FSW) and enable engineers to determine the optimal parameters for the FSW process in less than 24 hours.

SOLUTION

Employed parallel programming methods to develop the code, leveraging CUDA Fortran compiler.

RESULTS

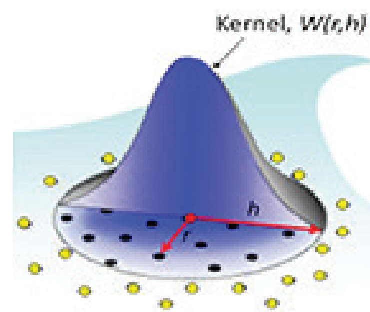
- Simulations run up to 45X faster
- Simulation code makes FSW easier to understand and use
- Able to run multiple simulations at once on multiple NVIDIA GPUs

Project Overview

At the University of Quebec at Chicoutimi (UQAC), many researchers are interested in one thing: aluminum. That's because UQAC is at the center of the aluminum valley in the heart of the province of Quebec—one of the largest producers of aluminum in the world. Most of the research at the University revolves around aluminum processing.

One group of researchers, the friction stir welding research group (FSWRG), is focused on developing important advancements for the aluminum industry. Friction stir welding (FSW) is a relatively new process developed in the early 90s, and has gained popularity for its use in high-end manufacturing processes. "FSW is well-suited for high strength aluminum alloys, such as those used to manufacture spacecraft," said Kirk Fraser, a Ph.D. student at UQAC. "It is capable of producing full-penetration welds in aluminum plates in a fraction of the time of conventional welding."

According to Fraser, the main difference between FSW and traditional welding is the aluminum doesn't actually melt; rather, it comes very close to melting temperature, but stays intact. A highly specialized tool rotates into the aluminum and stirs the metal, enabling the two pieces to join at a molecular level. Fewer defects result, compared to using conventional welding processes. It also makes a stronger joint and is a much faster process.



Fraser and his team are developing code that simulates the FSW process, to enable more precise, successful welds. Using an advanced modeling approach that employs smoothed particle hydrodynamics (SPH), Fraser is leveraging NVIDIA's graphics processing units (GPUs) and his numerical algorithm to achieve significant speed ups in the welding process. "When the code is finished, it will perform a numerical simulation of the friction stir welding process and determine the optimal operation parameters."

> "I was able to start writing productive CUDA Fortran code quickly, with very little prior experience with CUDA extensions."
>
> Kirk Fraser, a Ph.D. student at the University of Quebec at Chicoutimi (UQAC)

© 2015 NVIDIA Corporation, All rights reserved.

Challenge

However, the FSW process is very difficult to simulate. One of the main complications is the enormous amount of plastic deformation that occurs during the process.

"Because of the mixing behavior, most mesh-based simulation approaches cannot easily simulate the whole FSW process," said Fraser. "A number of research groups have attempted to simulate the FSW process with other numerical methods, but capturing all the physics of the process, from the start to the end of the weld, including cool-down to predict residual stresses, has not been possible."

Another challenge is the time it takes to run simulations. "Typically, a single simulation could take 5-7 days," said Fraser. "With multiple simulations to run per development cycle, simulation time can really extend the timeframe for your research."

And for manufacturers of aluminum products, time is money. "The FSW welding machines cost millions, but can significantly increase time-to-market for new products," said Fraser. "We want to provide code that helps engineers understand how fast the weld can be completed, and a specialized tool that enables them to get the biggest return on their investment. My goal is to enable them to run a simulation that provides optimal parameters in 24 hours or less."

Solution

Fraser decided to employ parallel programming methods to develop the code, leveraging PGI's CUDA Fortran compiler.

PGI is at the forefront in developing programming models for GPU accelerators. In addition to CUDA Fortran, PGI is a leader in the development of OpenACC, a directive-based accelerator programming model targeted at scientists, engineers and other domain experts who are not full-time software developers.

Although he evaluated other languages, CUDA Fortran was the best choice for the project. "CUDA enables code to run in parallel on NVIDIA graphics cards much faster than standard Fortran on a CPU," he said. For help with compiling the code, he turned to PGI. "I was able to start writing productive CUDA Fortran code quickly, with very little prior experience with CUDA extensions."

Fraser secured a license of the PGI suite that interfaces into Microsoft Visual Studio. NVIDIA provided the GeForce Titan Black GPU for testing the code.

"I was surprised at how easy it was to use the PGI compiler," said Fraser. "I thought compiling the CUDA Fortran code would be a nightmare, but it only required slightly more work than coding for Standard Fortran."

Kirk Fraser, a Ph.D. student at the University of Quebec at Chicoutimi (UQAC)

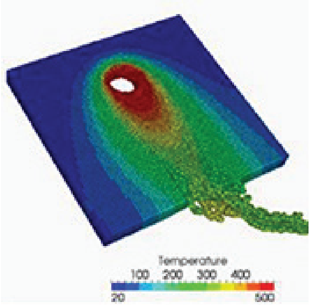
According to Fraser, who has used open-source solutions in the past, writing and compiling the code typically requires the developer to toggle between systems. PGI eliminates that step. "Because the solution is integrated with Visual Studio, I don't have to jump around from screen to screen—I can compile the code right from within the Visual Studio environment," he said.

Results/Benefits

Simulations run up to 45X faster

Fraser and his team have implemented a fully coupled thermo-mechanical large deformation solid mechanics code using smooth-particle hydrodynamics on the GPU. Using the PGI compiler to run his code on the NVIDIA GPU has significantly decreased the time it takes to run simulations.

"The simulation process used to be really tedious," he said. "You'd press 'enter' on the computer and come back in five days, only to find the simulation failed for some reason and you have to run it again. With PGI's solution, what took more than a day before only takes an hour."

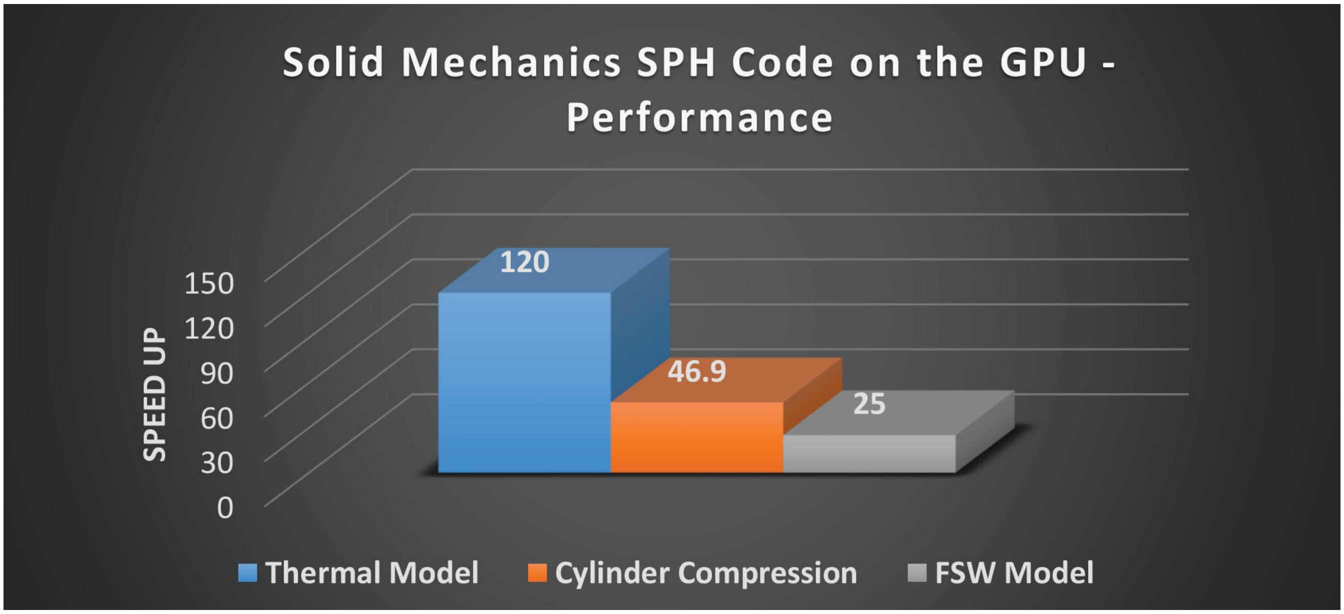


Additionally, Fraser can now run multiple simulations at once on multiple GPUs. "We can have one computer with three GPUs run three simulations in five hours," he said. "That's a huge time savings."

© 2015 NVIDIA Corporation, All rights reserved.

As a result, the team has achieved full code speed ups of over 45x compared to the same code on one core of a central processing unit. All of this was accomplished without any previous knowledge of CUDA Fortran.

Speed-up obtained from different model test cases when running on a GPU



All numbers are versus a single CPU core and include data transfer times. GPU is an NVIDIA GeForce Titan Black.

Simulation code makes FSW easier to understand and use

Fraser's simulation tool will make the FSW process easy for the engineer to understand and use. "The process is a high-end and expensive one that's not well understood, but with the simulation code, it's fast and accessible for engineers who want to understand how it works," he said.

Fraser said his experience working with PGI has been rewarding. "My interaction with the support team at PGI has been phenomenal," he said. "The people at PGI really enjoy what they do and are passionate about helping their clients succeed."

If you would like to learn more about the FSW project at UQAC, read Kirk Fraser's report at https://www.researchgate.net/publication/280255101_CUDA_Fortran_Success_Story.

© 2015 NVIDIA Corporation, All rights reserved.

PGI®