Performance Optimisation and Productivity

A Centre of Excellence in Computing Applications

POP Newsletter 6 – Issue September 2017

Welcome to the sixth newsletter from the EU POP Centre of Excellence. Our free of charge services help EU organisations improve performance of parallel software.

This issue includes:

- POP Webinar **Understand the Performance of your Application with just Three Numbers** on 11th October.
- **Vote for POP in the European Commission's Innovation Radar Prize 2017**
- Top Tips - analysing and understanding low IPC, and profiling Python in VTune

For information on our services and past editions of the newsletter see the POP website.

POP Webinar - Understand the Performance of your Application with just Three Numbers

Wednesday 11 October 2017 - 13:00 BST | 14:00 CEST

Measuring application performance often results in a large amount of profile data or traces that are difficult to handle or interpret beyond some trivial first observations. These analyses often do not provide the kind of insight that would really help a code developer determine the most appropriate direction to follow to improve the code. But is it possible to compute a very limited number of metrics for a parallel MPI application to explain its behaviour in terms of fundamental properties?

For more information and to register click here.

About the presenter: Jesús Labarta has been a full professor of Computer Architecture at the Technical University of Catalonia (UPC) since 1990, and since 2005 he has been responsible for the Computer Science Research Department within the Barcelona Supercomputing Center (BSC). His current work focusses on performance analysis tools, programming models and resource management.

Vote for POP

POP is competing in the 'Best early stage innovation' category in the European Commission's Innovation Radar Prize 2017. This initiative identifies high potential innovations and innovators in EU-funded research and innovation projects, and voting takes place until 15 October 2017.

Please vote for POP here.

The POP webinar series

Our popular series of live webinars began in June with a talk on *How to Improve the Performance of Parallel Codes.* This presented a systematic approach to optimising codes, whilst pointing out various factors that should be considered. As well as a live demo of performance tools from Barcelona Supercomputing Center, the talk was illustrated with practical examples from various POP performance assessments.

In August, our second webinar looked at *Getting Performance from OpenMP Programs on NUMA Architectures*. This webinar covered the characteristics of cc-NUMA architectures, the OpenMP thread affinity model and the operating system mechanisms of memory placement. It also explained how to use this understanding to achieve performance optimization.

The October webinar will explain how just three metrics can characterise the behaviour of an application, and provide a degree of insight into how the application might be improved. Future webinars include *Using OpenMP Tasking* (4th December – 14:00 GMT | 15:00 CET) and *Profiling the I/O characteristics of HPC applications* (early 2018).

Further details can be found on the POP website, including recordings of our previous webinars.

Top Tip – analysing Python code with VTune

Intel VTune Amplifier is a powerful tool for profiling code written in a variety of languages including C/C++, Fortran and Python. It can be used on parallel code that uses paradigms such as OpenMP, MPI and Intel Threading Building Blocks (TBB), as well as on serial code.

One useful feature of VTune is that it supports a set of Instrumentation and Tracing Technology (ITT) APIs that allow developers to programmatically control and augment the tracing of their applications. These are native C/C++ functions that can easily be called from code written in these languages. It is also possible to call them from Fortran code with little extra work. However, even though VTune supports profiling Python code, there are no native Python ITT bindings provided out-of-the-box by Intel.

This shortcoming has been addressed by the National Energy Research Scientific Computing Center (NERSC), who have made Python bindings for a subset of the ITT APIs available as the itt-python package, via their Github page. Their contribution covers two of the most useful APIs: Collection Control and Task.

For more information on using ITT APIs see our recent blog post**.**

POP joins SESAME Net

POP is adding our expertise and free of charge services to SESAME Net to help SMEs make the best use of HPC.

SESAME Net contains a mix of internationally-respected Competence Centres and organizations which have joined forces to build an open and inclusive network, with the common aim of raising SMEs' awareness of HPC and to demonstrate its features and benefits. One of the most important deliverables of the project is the HPC4SME Assessment Tool which offers SMEs the opportunity to discover IF and HOW their organization can benefit from supercomputing services.

SMEs interested in more information on SESAME Net can be found on their website after registering here. Resources on the website are aimed at SMEs and include a technical forum, training material, and the HPC4SME Assessment Tool.

Top Tip – Analysing low IPC

One source of inefficiency we routinely investigate in our free performance analyses is low IPC (Instructions per Cycle). Low IPC can be the cause of load imbalance and of slow down, as core count is increased. Low IPC often indicates computational throughput has reduced because the CPU core is waiting unnecessarily before executing instructions, usually because the core is waiting for data from main memory. Hence low IPC needs further investigation using hardware counters, e.g. PAPI. And although on a modern x86 based Intel server chip an IPC of 4 is the theoretical peak, in our experience an IPC above 1 is good.

Hardware counters can be used to measure how much data is transferred over the memory bus per second. Multiplying the number of memory instructions (or last level cache misses) by the cache line size for a region of computation, and dividing by time in seconds gives the bandwidth in GB/s. If this value is close to peak, then memory bandwidth is a bottleneck for the application. Perhaps blocking can help provide more cache data reuse and this bottleneck can be circumvented. However sometimes this is just a property of the algorithm and has to be accepted.

Ideally data is aligned in memory such that all data in a cache line is used. This can be tested by comparing cache accesses and cache misses. For example, if 8 double values (i.e. 8 bytes each) are loaded into L1 cache with a cache line size of 64 bytes, then a ratio of 1 cache miss to 8 cache accesses signifies that all values in the cache line are used. If the ratio is 1 to 1 then every access corresponds to a cache miss, i.e. only one value in the cache line is used. In such a case, it may be possible to achieve better performance with a different data layout in the algorithm, for example changing arrays-of-structures into structures-of-arrays.

And on servers with two or more sockets every processor typically has its own local memory, these systems are known as NUMA (non-uniform memory access) architectures. The application can access any memory in the system with the same load and store instructions, but accessing local memory is faster than accessing memory attached to a different socket. Hardware counters can be used to find out if a lot of remote memory accesses are present during execution. If this value is high, for example 50% of all accesses go to remote memory, it is likely a performance problem, and the programmer should optimize the data distribution during initialization or migrate data between sockets. For further information on NUMA issues see the recording of the recent POP webinar.

POP users can contact us via the POP helpdesk for advice on measuring and analysing IPC.

Meet POP at upcoming events

POP @ International CAE Conference

The International CAE Conference (6th-7th November in Vicenza, Italy) is the principal conference in the area of simulation based engineering and science. POP will be presenting the project within the conference's research agorà, a space where project consortia can exhibit their achievements to a wider audience. As well as a poster and other promotional materials, we will be giving live demonstrations of the profiling tools in our booth.

Apply for free help with code optimisation

We offer a range of free services designed to help EU organisations improve the performance of parallel software. If you're not getting the performance you need from parallel software, please apply for help via the short Service Request Form, or email us to discuss further.

These services are funded (until end of March 2018) by the European Union Horizon 2020 research and innovation programme - there's no direct cost to our users! If you're interested in our services, please contact us soon to express an interest.

The POP Helpdesk

Past and present POP users are eligible to use our email helpdesk (pop-helpdesk@bsc.es). Please contact our team of experts for help analysing code changes, to discuss your next steps, and to ask questions about your parallel performance optimisation.

https://pop-coe.eu

This project received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 676553