Parallel Computing Performance

A Comparison of Intel x86 and ARMv8 Processors On Text Analysis Tasks

Yunzhen Ma
yma067@uottawa.ca

Chen Zeng
czeng011@uottawa.ca

School of Electrical Engineering and Computer
University of Ottawa
Ottawa, ON, Canada

***Abstract*—Multi-core processors are becoming increasingly prevalent in the everyday electronic devices that we use such as laptops, cellphones, smart TVs, and smart speakers. Different types of devices use various types of CPUs with all kinds of architectures and technologies. In this paper we compare the performance of text analysis between Intel x86 PC and Raspberry Pi. We compare their processing times and speedups, as well as power consumptions. Our results indicate that the Intel Haswell quad-core CPU has significantly better single-core and multi-core performance, and its equipped PC is more energy efficient than the Raspberry Pi with a quad-core ARMv8 CPU.**

Keywords — processor; parallel computing; multi-core; multithreading; parallelism; text analysis

I. INTRODUCTION

For more than half a century, the demands for faster computers have never been satisfied. Today, these demands are not only for computing speed, but also for mobility, less power consumption and multiprocessing abilities.

Manufacturers have found multi-core processors as the best solution so far. A multi-core processor provides advantages that a single-core processor finds difficult or is unable to achieve. These advantages bring new computing innovations into reality, such as text analysis [1].

Text analysis, or text mining, derives structures data from unstructured text. This technology can be used for media coverage analysis, competitor analysis, insurance claims assessment, texts sentiments analysis and so on [2].

This paper compares the parallel computing performance on tweets sentiment analysis between an Intel x86 PC and Raspberry Pi. Both single-core and multithreading performance are analyzed, as well as the speedups for different numbers of threads used. In addition, the paper also compares the energy efficiency between the two.

II. BACKGROUND

A. Multi-core Processors

A multi-core processor consists of two or more independent processing units – or cores – on a single die. The advantages of multi-core structure over multi-processor in a computer are obvious: lower energy usage and significantly less cost. Also, the communication between separate CPUs causes large latency due to the physical distance of the sockets. Following decades of developing single core processors, the focus has now shifted to multi-core processors

Multi-core processors have been available for over a decade. The first commercially available multiprocessor chip was the IBM POWER4 in 2001 [3]. The POWER4 has two cores with a clock speed of 1.1 GHz to 1.3 GHz. Compared to other commercial single-core CPUs at that time, the frequency of POWER4 is much lower. For example, Intel Pentium 4 Northwood CPUs have a frequency of up to 3.0GHz. This created a new trend in the processor industry: optimizing multi-core performance instead of single core CPU frequency race.

B. Intel Haswell

Intel's Haswell is Intel's fourth generation Core processor based on 22nm 3D tri-gate transistor architecture. The basic chip design of i7 desktop Haswell is 1.4 billion transistors and four processor cores on a 177mm^2 die with a 14-stage pipeline. A GPU is also provided. Each core has a 32KB L1 cache and a 256KB L2 cache. An 8MB L3 cache is shared between all cores along with the GPU. The Haswell comes with two or four physical cores with four or eight logical processors due to Intel HyperThreading technology [4].

Compared to previous versions of Intel platforms, Haswell has several advantages such as: retargeted silicon design, new power management framework and efficient system design. It is about 5% to 15% faster than the third generation Sandy Bridge [4]. One of the noticeable improvements is the newly designed execution engine.

From Intel Core 2 Duo to Ivy Bridge, Intel's Core architecture supported the execution of up to six micro-ops in parallel [4]. The number of execution units have increased but not the number of ports until Haswell. Haswell adds two more execution ports in addition to existing six ports, one for integer math and branches and another for store address calculation [4].

The new execution engine improves heavy integer processing performance, and allows integer work concurrently while ports 0 and 1 are occupied. Moreover, branch heavy code can use both port 0 and 6 at the same time [4].

C. Intel Hyper-Threading

Intel Hyper-Threading Technology (HT) provides multiple logical processors in a processor core. It makes a physical processor appear as two or more logical processors. Each logical processor has a complete set of registers [5]. For example, the x86 CPUs being used for this project were Intel Xeon E3-1270 V3 which has four physical processors with eight threads, and Intel i5-4310U Mobile which has two physical processors with four logical processors show to the OS. The reason an Intel i5 Mobile CPU was used is because measuring power consumption on a desktop PC is not applicable.

The Intel HT feature can be disabled from computer BIOS settings so that the number of logical processors will be the same as the actual number of physical processors. For this project, we are studying the multi-threading performance of common processors for daily use. Thus the HT feature was left enabled since it is a significantly important feature of most Intel CPUs nowadays.

D. Raspberry Pi 3 with Quad-Core Broadcom CPU

Raspberry Pi 3 Model B is the latest Raspberry Pi mini-size computer introduced in 2016. It is equipped with a Quad Core Broadcom BCM2837 64-bit 1.2GHz ARMv8 CPU and 1GB DDR2 RAM. [6]

The BCM2837 CPU consists of four Cortex A53 processors, each with an L1 cache system and a single shared L2 cache. It has a Symmetrical Multiprocessing (SMP) architecture with a single processor cluster, and multiple coherent SMP processor clusters through AMBA 4 technology. It supports TrustZone security technology and DSP & SIMD extensions. [6]

The Cortex-A53 processor provides significantly more performance than Cortex-A7 processor. Some of the new features include 64-bit processing, extended virtual addressing, 8-stage in-order dual issue pipeline and improved integer and floating-point unit. [6]

The Cortex-A53 processor is applicable in a range of devices requiring high performance in power-constrained environments such as smartphones, networking devices and automotive infotainment. [6]

III. METHODOLOGY

There are a variety of approaches to measure the performance of a multithreading system including commercial benchmark suites. For this project, the requirements were given as processing a large number of tweets on each platform and comparing the time of execution as well as speedups.

A. Platforms

This project was to investigate multi-core processor performance on different platforms with different architectures.

As mentioned earlier, we were unable to find a laptop equipped with a Quad-Core Intel CPU so we chose the E3-1270v3 desktop version as our primary testing object since the other object is Raspberry Pi with a Quad-Core CPU as well. While a laptop with Intel Haswell i5 CPU running Ubuntu 16 64bit operating system we selected to measure power consumption and execution time for reference only.

The Intel Xeon CPU has a base frequency of 3.5GHz and 3.7 or 3.9GHz Turbo Boost frequency. The desktop computer has 24GB DDR3 1600MHz memory and a 500GB SSD. Ubuntu 16.04 with latest update was.

The i5 Mobile CPU has a base frequency of 2.0GHz or 3.0GHz if Turbo Boost is enabled, 3MB L3 Cache, dual cores and four threads when HT is enabled. The thermal design power (TDP) is 15W. This computer is equipped with 16GB DDR3 1600MHz memory and an SSD to eliminate read / write wait time as best as possible. The external power supply was connected during the test to avoid battery saving mode. Most background applications such as the auto-update were off. Wifi and Bluetooth were both off so that network applications would not affect the results.

Machine two is Raspberry Pi 3 Model B. The operating system is Raspbian kernel version 4.9. During the test only a keyboard was use, while Wifi and Bluetooth were off all the time. To view the results, a monitor had to be connected via HDMI.

Both computers have Python 2.7 installed, along with necessary libraries.

B. Input Text File

According to the requirement, each computer has to process 5000+ tweets and calculate the sentiment. Due to the limitation of the Twitter 3rd party library, only a small amount of tweets can be obtained at one time. In addition, this project focuses on multithreading performance of different CPUs, but most of the time elapsed would be HTTP request / reply wait time if online tweets are used. Thus an offline text file with 100,000 downloaded tweets was being used. During the test, this text file would be loaded into memory first, then the target processor would read line by line.

C. Test Programs

The program which obtains the tweets through Twitter API – tweepy - was provided. Due to some limitations set by Twitter, third party libraries which access Twitter API can only get 500 tweets within a time window. Thus this piece of program ran multiple times making numerous HTTP requests over a long period of time. After the execution, about 5000 tweets were finally obtained. After the first run of the sample code, it turned out that PC processed the text file too fast. If four threads were used, the processing time was less than a second. Hence this code was modified to repeat 20 times so that the total amount of text strings became 100,000 lines.

The second program which is the main component of the project is the processing python program. There are two parts of the program: first part is serial processing and the second part is parallel processing.

The serial processing program reads the whole text file and store all lines into a temporary list then calculate sentiments from the list. Finally the number of positive and negative tweets are counted and stored. The execution time is recorded at the end of the program.

Unlike the serial processing program, the parallel processing program processes the list of tweets in parallel. The library used is python 'multiprocessing'. It spawns processes with either local or remote concurrency and uses subprocesses instead of threads [7]. This program uses the Pool object which parallelizes the execution of the sentiment processing function with distributing the input data – text lines – across processes.

The serial processing program accepts an argument as the number of processes that can be created. For this project, the program ran 3 set of tests, with each set creating two, three, and four processes accordingly.

D. Tools

This project not only compares the multithreading performance between different platforms, but also their power consumption, as energy efficiency becomes more crucial in modern CPU designs.

Thus in addition to the two computers and related programs, the following power measuring tools were used as well.

1) Digital Clamp Meter: a hinged jaw clamped around a wire and measuing the current through a current sensor. The measure range is 0 ~ 20A with ± 1.5% deviation. Due to the nature of the meter, only one cable can go through the clamp so we used a modified power strip. Thus the meter was placed before the AC/DC conversion. The power adaptor for this laptop is Dell LA65NM130 manufacured by Lite-On which meets the Level V Efficiancy Standard by the United Stated Department of Energy according to the label on it. And Level V Standard states that the enerygy loss has to be less than [0.0626 x Ln(Power)] + 0.622 [8]. If the power is 35W, the power loss due to energy conversion is ≤ 2.792W which is much smaller than 10% of the total power. Therefore the error could be ignored.

2) USB Power Meter: a small size USB device with a monitor which measures voltage, current and power transmitting through the USB port. The measurement range is 0-3.00 A ± 1%

E. Implementation

When running the programs, all background applications including antivirus software were turned off. Firstly the serial process program was executed 10 time. Then the serial process program was executed in 3 groups, each group contained 10 test and ran in different number of processes. Thus there were 40 runs for each platform. At the same time, the power measuring tools were set up. Eventually all the execution times were automatically stored in output file so that we could calculate the average run time. The power consumption was recorded each time as well and the average value was calculated.

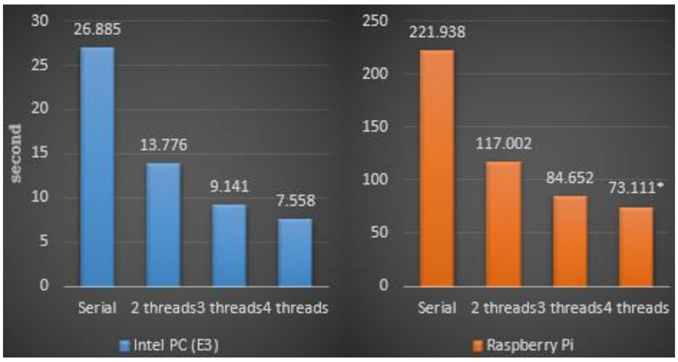

Figure 2. Processing Time
* fluctuating due to high temperature

IV. MEASUREMENT RESULTS

After the executions, the time costs were written to output files and the power consumption results were recorded. The average values were calculated and grouped to several categories: single core performance, multi-core speedup and power efficiency.

A. Single Core Performance

From Figure 2 we can see that the serial computing performance between the two platforms has a significant difference.

The Intel E3-1270 v3 takes about 26.9 seconds to process the texts while the ARMv8 on Raspberry Pi uses about 221.9 seconds. The Intel CPU is approximately 9 times faster than the ARM CPU.

The huge advantage is not only because of the frequency difference. The Intel E3 has base frequency of 3.5GHz and the ARMv8 has 1.2GHz which is only about 66% slower. The fundamental cause of the performance advantage is that Intel E series CPUs are designed for heavy computing while the ARMv8 CPUs are for small size devices aiming for low power consumption and heat control. Furthermore, Intel provides SSE4 instruction set which can be used to speed up text analysis code [9].

B. Multi-Core Speedup

Figure 3 is derived from Figure 2 and it shows the speedups for two, three and four threads accordingly.

From this chart we can observe that when two threads were created, the speedup for both platforms are similar. When the number of threads increases to three, the speedup of the Intel CPU is very close to 3.0 while the ARMv8's speedup drops to about 2.6. The last column shows that when four threads are being used, ie. all CPU cores are being occupied. The speedup of the Intel E3 is near 4.0 while the ARMv8 struggles to reach 3.0.

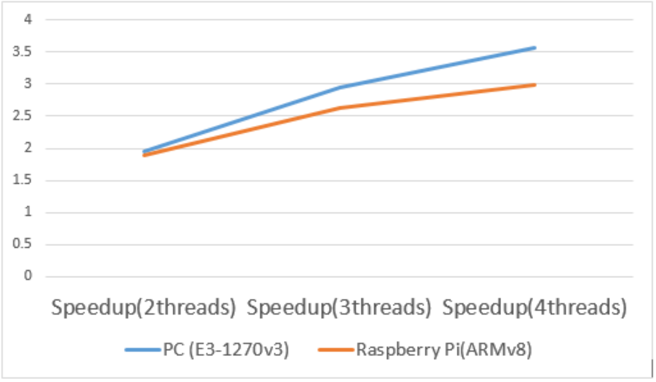

Figure 3. Speedup

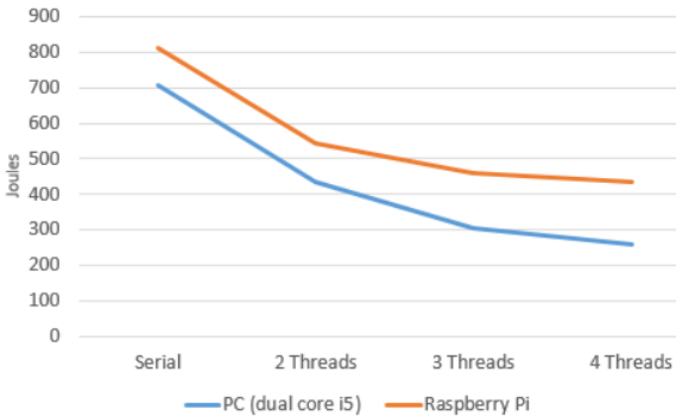

Figure 4. Power Consumption

There are multiple reasons behind the fact that the Intel CPU has much better multiprocessing performance than the ARMv8 such as the Inter-core communication, and operating system. Another factor which can directly affect multi-core performance is system memory and cache coherency.

The PC used for this project has 24GB 1600MHz DDR3 RAM while the Raspberry Pi has only 1GB 900MHz LPDDR2 RAM. Since the tests were based on data parallel which the text lines were stored in memory, the speed of the memory plays a significant role here.

Moreover, the E3-1270v3 CPU has an 8-way 32KB L1 cache and an 8-way 256KB L2 cache for each core, as well as a 16-way 8MB L3 cache shared by all cores. However, the ARMv8 BCM2837 CPU has only a16KB L1 cache for each core and a 512KB L2 cache shared by all cores. Larger L1 cache guarantees a higher cache hit rates and one more level of cache structure makes the read operation from memory happen less often. We were unable to test how much of the test input file could be loaded into the L3 cache of the E3 CPU but its large size makes it reading data from memory as less frequent as possible.

Another observation is that when four threads were created on the Raspberry Pi, the results vary from 70 seconds to over 80 seconds. This is because when the CPU is fully loaded, the temperature becomes very high and the heatsink cannot discharge heat properly. Then the system starts to lower the CPU frequency and thus the performance becomes restricted.

C. Power Efficiency

From Figure 4 we can see that the i5 laptop consumes less energy than Raspberry Pi for all tests.

In particular, then performing serial processing, both PC and Raspberry Pi takes about 700 ~ 800J of energy whereas the PC takes about 12.6% less. As the number of threads increases, the difference of the energy consumption between the two has increased to 67.6%.

Although the ARMv8 processor is designed to use less power (~3W), its overall power efficiency is worse than the Intel CPU. Hence if a processing job is to be done and there is no platform constrains, using PC is preferred rather than a mobile device.

D. Other Observations

When we were measuring the power consumption on the i5 laptop we noticed that the multithreading performance on this dual-core CPU is very poor although the HyperThreading Technology was enabled.

From Figure 5 we can see that the processing time of two threads is close to half of the serial processing. On the other hand, when the number of threads increases, the processing time does not decrease. We can conclude that the Intel HyperThreading Technology does not increase the text analysis performance.

V. RELATED WORK

The competition among CPU manufactures motivates the development of benchmark software. Most of the studies which measure multi-core CPU performance use professional benchmark suites that focus on specific aspects of multiprocessing performance instead of actual applications [10]. Our study compares the performance of multi-core processors

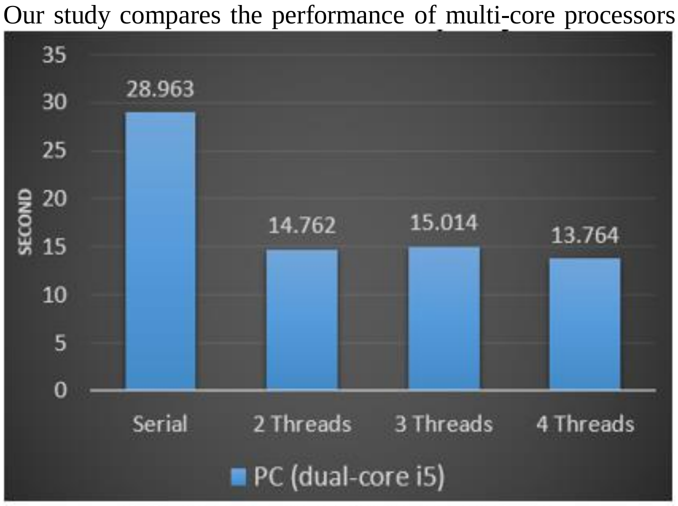

Figure 5. Processing Time of Dual Core i5 4310U

with regards to text analysis between different platforms as a basic usage.

Energy efficiency is as important as computing performance. Many studies only compare the multithreading performance between different processors [11]. In this paper we include power consumptions as well to conclude that a PC is more energy efficient than a much smaller size Raspberry Pi, which challenged and revised our original expectation.

VI. CONCLUSION

This paper compares the multi-core performance of Intel x86 CPU and Raspberry Pi on text analysis. The quad core x86 E3-1270v3 CPU has significantly better single core performance and speedup than the ARMv8 processor with a single core. Our study shows that the Intel CPU is about 9 times faster than the ARM CPU on single core processing. In addition, the Intel CPU has a speedup of 3.56 whereas the ARM CPU has a speedup of less than 3.0 when four threads are created on all cores.

This paper also compares the power consumption of an Intel laptop and Raspberry Pi. The result indicates that although the Raspberry Pi has much less power consumption, it is less energy efficient than the Intel laptop. We conclude that the x86 quad core CPU not only has more advanced multi-core architecture, but also has higher performance-to-energy cost ratio.

ACKNOWLEDGMENT

This work is supported by Professor Miodrag Bolic, of the School of Electrical Engineering and Computer, University of Ottawa.

REFERENCES

[1] D. Gutierrez, Text Analytics: The Next Generation of Big Data

[2] M. Ferguson, What is text analytics?

[3] W. Stallings, Computer Organization and Architecture, Seventh Edition pp. 44, April 1955. (*references*)

[4] A. L. Shimpi, Intel's Haswell Architecture Analyzed: Building a New PC and a New Intel, ANANDTECH, retrieved on Dec 10, 2017, https://www.anandtech.com/show/6355/intels-haswell-architecture

[5] Intel, Intel® 64 and IA-32 Architectures Optimization Reference Manual., 2016

[6] ARM, Cortex-A53, retrieved on Dec 10, 2017, https://developer.arm.com/products/processors/cortex-a/cortex-a53

[7] Python Software Coundation, The Python Standard Library, retrieved on Dec 10, 2017 https://docs.python.org/2/library/multiprocessing.html

[8] CUI Inc., Efficiency Standards for External Power Supplies. Oregon: CUI Inc., 2016, pp. 5.

[9] Intel, Intel® SSE4 Programming Reference, July 2007, pp.2

[10] S. Gal-On and M. Levy, Measuring Multicore Performance, EEMBC, 2009, pp.1.

[11] A. Khajeh-Saeed, S. Poole, J. B. Perot, A Comparison of Multi-Core Processors on Scientific Computing Tasks, San Jose, CA, 2012