Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Perspective:

The rise of high-throughput computing*

Ning-hui SUN‡, Yun-gang BAO, Dong-rui FAN

State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences, University of Chinese Academy of Sciences, Beijing 100080, China

E-mail: snh@ict.ac.cn; baoyg@ict.ac.cn; fandr@ict.ac.cn

Received Aug. 22, 2018; Revision accepted Sept. 14, 2018; Crosschecked Oct. 10, 2018

Abstract: In recent years, the advent of emerging computing applications, such as cloud computing, artificial intelligence, and the Internet of Things, has led to three common requirements in computer system design: high utilization, high throughput, and low latency. Herein, these are referred to as the requirements of 'high-throughput computing (HTC)'. We further propose a new indicator called 'sysentropy' for measuring the degree of chaos and uncertainty within a computer system. We argue that unlike the designs of traditional computing systems that pursue high performance and low power consumption, HTC should aim at achieving low sysentropy. However, from the perspective of computer architecture, HTC faces two major challenges that relate to (1) the full exploitation of the application's data parallelism and execution concurrency to achieve high throughput, and (2) the achievement of low latency, even in the cases at which severe contention occurs in data paths with high utilization. To overcome these two challenges, we introduce two techniques: on-chip data flow architecture and labeled von Neumann architecture. We build two prototypes that can achieve high throughput and low latency, thereby significantly reducing sysentropy.

Key words: High-throughput computing; Sysentropy; Information superbahn
https://doi.org/10.1631/FITEE.1800501 **CLC number:** TP303

1 Introduction

Emerging applications and advanced materials are the two major forces driving computer system technologies (Xie, 2016). Generally, there are three different types of applications, i.e., high-performance computing, desktop, and mobile applications, which have different requirements for computer system technologies. For instance, high-performance applications require high-speed interconnection networks that are used in high-performance computers. Concurrently, desktop applications have propelled microprocessor technologies for PCs, while mobile applications have requirements of low-power technologies (Xu, 2012; Xu et al., 2016).

In recent years, emerging applications, such as cloud computing, artificial intelligence (AI), and the Internet of Things (IoT), have posed three major requirements, i.e., high utilization, high throughput, and low latency. Considering cloud computing as an example, cloud datacenters often cost billions of dollars. For example, Microsoft invested 15 billion US dollars in datacenters till 2015 (Greenberg, 2015), while Alibaba has already spent 2.9 billion dollars in building a datacenter located in Zhangbei (http://www.datacenterdynamics.com/content-tracks/colo-cloud/alibaba-launches-29bn-cloud-base-in-north-China/93689.fullarticle). Conversely, cloud providers are eager to minimize costs by increasing resource utilization. Furthermore, latency is critical to user experience. Specifically, Google has found that advertising revenue will

‡ Corresponding author
* Project supported by the National Key R&D Program of China (No. 2016YFB1000201), the National Natural Science Foundation of China (No. 61420106013), and the Youth Innovation Promotion Association of Chinese Academy of Sciences
ORCID: Ning-hui SUN, http://orcid.org/0000-0002-4179-2660

drop by 20% when the response time of search requests increases by 0.5 s (https://www.zdnet.com/article/googles-marissa-mayer-speed-wins/). Therefore, ensuring good user experience is the top priority for cloud companies. Furthermore, AI and IoT applications are often deployed in clouds, which require high utilization of cloud datacenters, high throughput, and low end-to-end latency. These present new challenges in computer system design.

We propose 'high-throughput computing (HTC)' to refer to the collective requirements of high utilization, high throughput, and low latency. The computer systems that meet these requirements are referred to as 'high-throughput computers'. In fact, HTC is not a new term, and was first proposed in 1996 by Miron Livny (Livny, 2013) and mentioned by Xu (2012). The advent of graphics processing unit (GPU) was considered to be a representative HTC paradigm, which cares mainly about high throughput. In this study, we expand the connotation of HTC and propose that it should simultaneously meet the three requirements. However, traditional computer systems cannot meet all of the three requirements at the same time. Inspired by the thermodynamic entropy, we introduce a new indicator, referred to as 'sysentropy', to measure the degree of latency variation of a system operating at a certain utilization level and throughput. We use sysentropy to evaluate traditional PCs, supercomputers, and datacenters, and show that they all exhibit high sysentropy due to the inherent chaos and uncertainty in traditional computers. In fact, traditional computer systems generally sacrifice sysentropy for high performance and low power/energy consumption. We argue that high-throughput computers should be designed with the aim of achieving low sysentropy.

From the architectural perspective, low sysentropy refers to the need for strong management and control mechanisms. In this study, we introduce two techniques that can achieve low sysentropy: (1) the chip-level dataflow architecture SmarCo (Fan et al., 2018), which is able to fully exploit the parallelism within applications and the concurrency between requests, and could thus increase the utilization and throughput of computer systems; (2) the labeled von Neumann architecture (LvNA) (Ma et al., 2015; Bao and Wang, 2017), which can enhance control over the internal data paths of a computer system through labeling and programmable control logic mechanisms to maintain low latency when resource utilization increases. With the combination of these two technologies, sysentropy can be effectively reduced.

Over the past few years, we have carried out a series of studies on high-throughput computers based on the dataflow and LvNA architectures. We have designed and implemented a high-throughput many-core processor data processing unit (DPU) (Fan et al., 2018), which greatly improves the throughput by 20× with the same power consumption, compared to the traditional x86 processors. We have implemented an RISC-V-based open sourced LvNA prototype, known as 'labeled RISC-V' (Yu, 2017), which can achieve low latency while increasing CPU utilization by up to 4×.

We further evaluate the public transportation system with sysentropy and find that cars and airplanes exhibit high sysentropy, while high-speed rails exhibit low sysentropy. In fact, high-speed rails are highly acclaimed owing to high utilization, high throughput, and low latency variations. We envision that the future information infrastructure should also incorporate a high-speed rail mode, which can be referred to as the 'information high-speed rail (IHSR)' mode. Furthermore, high-throughput computers are at the core of the IHSR. We believe that the IHSR will provide low-cost, high-quality information services for billions of people around the world.

2 Requirements of emerging applications

Although cloud computing, AI, and IoT are considered to belong to different scenarios, they have three common requirements, i.e., high utilization, high throughput, and low latency. We describe these requirements separately below.

High utilization is an indicator used to measure the resource usage of CPUs, memory, storage, and networks in computer systems. Today, cloud computing, AI, and IoT applications are deployed in datacenters to serve online requests from users and IoT devices. A typical datacenter often has tens of thousands of servers and costs hundreds of millions of dollars. However, these resources have not been fully used. In 2013, the CPU utilization of Google's online datacenter was still only 30% on average (Barroso et al., 2013), and many companies did not even

reach this level. For datacenters that cost billions of dollars, it is of great importance to increase resource utilization. In general, a single task often fails to make full use of various resources. Thus, multiple tasks are required to share datacenters to achieve high utilization, which is challenging when taking into account other requirements, such as low latency.

High throughput is an indicator used to measure the number of completed tasks or requests during a time unit. Today, an increasing number of applications are deployed on the smart phone and cloud sides. For instance, IFLYTEK deploys online speech recognition in the cloud to provide AI services for millions of users, and some companies also deploy IoT services in the cloud to serve a large number of end users. It is necessary to fully exploit the parallelism in tasks as well as the parallelism within computer systems so as to increase throughput and minimize costs. Nowadays, a computer system contains various levels of parallelism, including instruction-, thread-, and request-level parallelisms. The identification of methods to make full use of parallelism to improve the throughput of computer systems has become increasingly important.

Low latency is an indicator used to measure the response time of user requests. Many studies have shown that user experience is crucial to online services and directly affects a company's revenue. In addition to Google's example, Amazon found that for every 100-ms increase in loading of its home page, its revenue decreased by 1% (http://highscalability.com/latency-everywhere-and-it-costs-you-sales-how-crush-it). As more and more AI and IoT applications are deployed in the cloud, response time requirements become increasingly stringent. Fig. 1 lists the response time requirements for different types of services deployed in the cloud. For AI services, such as image recognition (img-dnn) and machine translation (moses), the response time is already at the millisecond level, which is a great challenge for computer system design.

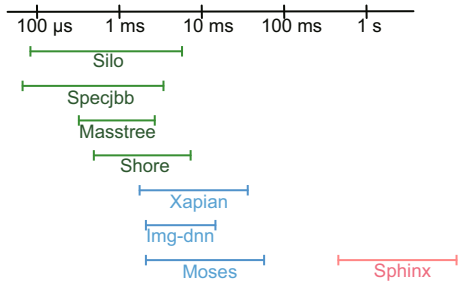

Fig. 1 Response time requirements for different applications. Reprinted from Kasture and Sanchez (2016), Copyright 2016, with permission from IEEE

In summary, the requirements for high utilization, high throughput, and low latency have become an urgent need for emerging scenarios. We will collectively refer to these requirements as HTC, and computer systems that meet these requirements are referred to as high-throughput computers.

3 Sysentropy and low-sysentropy systems

High-throughput computers differ considerably from traditional PCs and supercomputers in terms of their design goals. Traditional computers often target performance, power consumption, or performance-cost ratio, while high-throughput computers target all these three requirements at the same time. To better understand the difference between the design goals of high-throughput computers and traditional computers, we introduce a new indicator, referred to as sysentropy, to evaluate the degree of latency variation of a system at certain utilization levels and throughput:

$$S = \frac{V}{U \cdot T}, \tag{1}$$

where V is the latency variation degree which can be defined according to different scenarios, such as the delay variance σ^2 or the 99^{th} percentile tail latency, U $(0 < U \leq 1)$ represents utilization, which can select the resource object of interest according to the need, such as the CPU resource utilization or network bandwidth, and T represents the throughput, meaning the number of tasks completed in a time unit, which can be calculated by measuring user requests, instructions per cycle (IPC), etc.

Inspired by the thermodynamic entropy, sysentropy is an attempt to measure the degree of chaos and uncertainty within a computer system. It is necessary to emphasize that sysentropy does not have units, and its value is calculated without any unit. Thus, sysentropy makes sense only when it is being used for comparison between different systems with the same metrics for V, U, and T; however, this principle enables great flexibility to the concept of sysentropy.

Consider a memcached application as an example (Fig. 2). We describe V, U, and T using the 99$^{\text{th}}$ percentile tail latency, CPU utilization, and the number of clients, respectively. When the resource utilization is 30% ($U = 0.3$) and the numbers of clients are 46 ($T = 46$) and 161 ($T = 161$), respectively, the 99$^{\text{th}}$ percentile tail latency in both situations is 190 μs ($V = 190$). Thus, we eliminate all units and compute the sysentropy as follows:

$$S_1 = \frac{190}{0.3 \times 46} = 13.78, \tag{2}$$

$$S_2 = \frac{190}{0.3 \times 161} = 3.93. \tag{3}$$

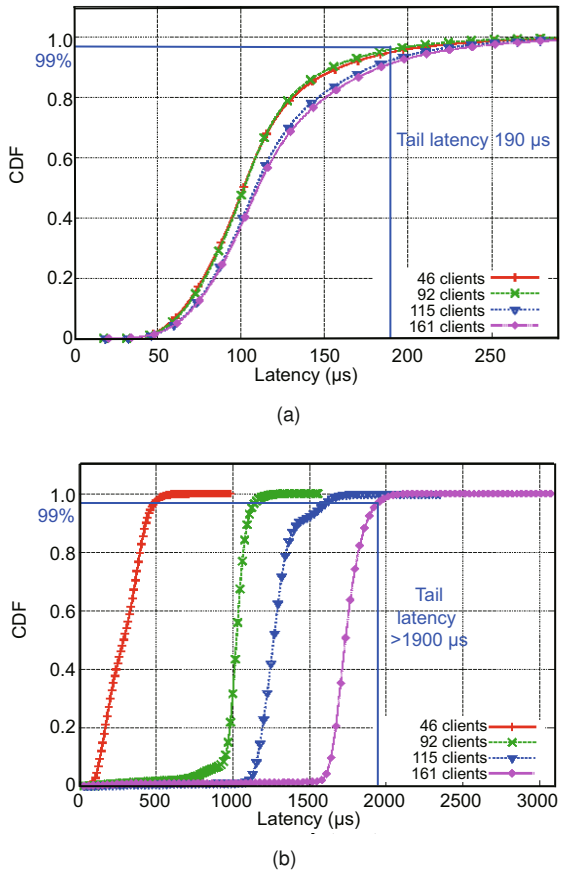

Fig. 2 Tail latency of memcached applications in different scenarios with utilization 30% (a) and 70% (b) (CDF: cumulative distribution function). Reprinted from Kapoor et al. (2012), Copyright 2012, with permission from ACM

The results show that when the resource utilization is low, increasing the throughput can decrease sysentropy. We further calculate the sysentropy when resource utilization is 70% ($U = 0.7$). When there are 46 clients ($T = 46$) and 161 clients ($T = 161$), the 99$^{\text{th}}$ tail latencies are 490 μs ($V = 490$) and 1900 μs ($V = 1900$), respectively. The sysentropy values are then respectively equal to

$$S_3 = \frac{490}{0.7 \times 46} = 15.22, \tag{4}$$

$$S_4 = \frac{1900}{0.7 \times 161} = 16.86. \tag{5}$$

The results demonstrate that when the system utilization reaches a certain level, the sysentropy will increase significantly. Inspired by the concept of entropy in thermodynamics, the term utilization is similar to temperature and determines the basic sysentropy of a system. We further classify the utilization of traditional computers and sysentropy (Fig. 3), and we can clearly see the differences between high-throughput computers and others.

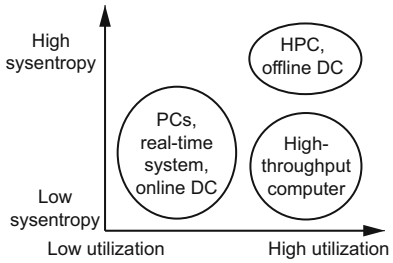

Fig. 3 Difference between high-throughput computers and traditional computer systems

To this date, traditional computer systems cannot achieve low sysentropy at a high utilization. Recently, Xu and Li (2017) proposed low-entropy cloud computing systems. We agree with their argument that it requires some specific technologies to achieve low sysentropy to build high-throughput computers. In particular, we introduce below two techniques that we have investigated in recent years.

1. Chip-level dataflow architecture

The dataflow architecture was originated from the early 1970s and was proposed by Dennis et al. (1974). The traditional dataflow architecture is processor-oriented (or at the core level). The dataflow architecture does not have a unified program counter, and the execution of processing operations is activated by required operands. As long as the operands arrive, the execution is triggered. Therefore, the dataflow architecture naturally has the advantages of parallelization and automatic synchronization. Meanwhile, data can be directly

transferred and calculated between the computing function units after being read from memory, which further simplifies memory access operations. Compared to the traditional control flow, dataflow exhibits an obviously improved efficiency. Generally, the dataflow architecture is beneficial for high utilization and high throughput.

2. Labeled von Neumann architecture (LvNA)

There is a lot of sharing at the hardware level, such as sharing of pipelines by multiple threads, last-level cache (LLC) , memory bandwidth, and the sharing of input/output (I/O) devices by multiple cores. However, the traditional von Neumann architecture lacks effective management and control mechanisms for shared hardware. The weakened control leads to severe contention in the hardware, which further causes large performance variations. LvNA (Bao and Wang, 2017) has been proposed to address this challenge. LvNA adds a new control dimension to the traditional von Neumann architecture, implements a labeling mechanism that conveys high-level software information to the hardware, and introduces programmable control logic at the hardware level to distinguish labels and perform differentiated services. LvNA can effectively achieve high utilization and low latency from the three HTC requirements.

4 Prototypes

Based on the above concepts and design, we have developed a high-throughput processor chip and a labeled architecture FPGA (field-programmable gate array) prototype system based on RISC-V (https://riscv.org/) to advance toward high-throughput computers.

4.1 High-throughput many-core processor DPU

We have built a high-throughput many-core processor DPU prototype chip (Fan et al., 2018), which leverages a global control mechanism for high-throughput data paths. The prototype chip (Fig. 4) is based on the TSMC 40-nm technology and consumes less than 4 W. Based on this chip, we have built an accelerator board that targets high-throughput video processing scenarios on the Internet. Compared to the mainstream Intel processor, the energy efficiency is improved by more than 20 times. Currently, thousands of high-throughput processing systems based on DPU have been deployed in relevant companies and government departments.

Fig. 4 High-throughput many-core processor DPU chip

4.2 Labeled RISC-V

We have implemented LvNA on the open instruction set RISC-V, referred to as 'Labeled RISC-V' (Fig. 5), which has been open sourced (http://sdc.ict.ac.cn/isca2018-tutorial/). The labeled RISC-V adds a label for each memory and I/O request. The label value can be associated with an upper virtual machine (VM) or a process. A programmable control logic is added to the cache and memory controllers to allow these components to process in different manners according to the label.

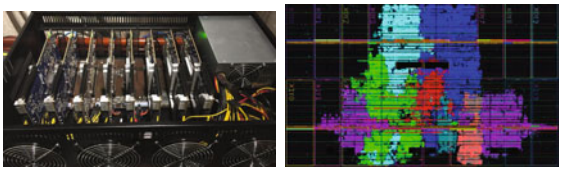

Fig. 5 Labeled RISC-V prototype system based on FPGA

The labeled RISC-V FPGA prototype system has several new features: (1) A physical server can be partitioned into multiple submachines and can directly load the operating system (OS) to run the application without the need for software hypervisors, such as VMware/Xen/KVM; (2) The hardware supports real-time monitoring (<1 ms) without software overhead; (3) The system supports performance isolation, such as dynamic allocation of cache capacity and memory bandwidth. Experimental results show that LvNA is able to improve CPU utilization by 25% to 100% while maintaining low latency. Currently,

the LvNA technology has been applied to enterprise server chips.

5 Conclusions

We have discussed the requirements for HTC. Compared to high-performance computing, desktop, and mobile applications, HTC has distinct features that require high utilization, high throughput, and low latency. However, traditional computer system designs have not yet considered these three requirements at the same time. We have proposed a new indicator, i.e., sysentropy, to better understand the HTC requirements. We introduced two techniques to achieve low sysentropy, namely, the dataflow and LvNA architectures. The prototypes and experimental results preliminarily verified the feasibility of high-throughput computers.

We leveraged sysentropy to roughly evaluate the transportation system (Table 1) and found that cars and airplanes also exhibit increased sysentropy. For instance, when traveling by cars, the throughput increases as the road utilization increases, but the latency variation becomes extremely uncertain owing to traffic jams. Correspondingly, the sysentropy is high. When traveling by plane, the throughput will become lower, which also leads to higher sysentropy. In comparison, only high-speed rails have low sysentropy and have been an indispensable mode of transportation. Thus, we believe that high-throughput computers will also become the core devices of future information infrastructure, and will provide low-cost, high-quality information services for billions of people around the world.

Table 1 Sysentropy of transportation systems

Parameter	Requirement		
	Car	Airplane	High-speed rail
Utilization	Low	High	High
Throughput	High	Low	High
Latency variation	High	High	Low
Sysentropy	High	High	Low

References

Bao YG, Wang S, 2017. Labeled von Neumann architecture for software-defined cloud. *J Comput Sci Technol,* 32(2):219-223. https://doi.org/10.1007/s11390-017-1716-0

Barroso LA, Clidaras J, Hölzle U, 2013. The Datacenter as a Computer: an Introduction to the Design of Warehouse-Scale Machines (2nd Ed.). Morgan & Claypool Publishers, USA. https://doi.org/10.2200/S00516ED2V01Y201306CAC024

Dennis JB, Fosseen JB, Linderman JP, 1974. Data flow schemas. Int Symp on Theoretical Programming, p.187-216. https://doi.org/10.1007/3-540-06720-5_15

Fan DR, Li WM, Ye XC, et al., 2018. SmarCo: an efficient many-core processor for high-throughput applications in datacenters. IEEE Int Symp on High Performance Computer Architecture, p.596-607. https://doi.org/10.1109/HPCA.2018.00057

Greenberg A, 2015. SDN for the cloud. Keynote of SIGCOMM.

Kapoor R, Porter G, Tewari M, et al., 2012. Chronos: predictable low latency for data center applications. Proc 3rd ACM Symp on Cloud Computing, p.9. https://doi.org/10.1145/2391229.2391238

Kasture H, Sanchez D, 2016. Tailbench: a benchmark suite and evaluation methodology for latency-critical applications. IEEE Int Symp on Workload Characterization, p.1-10. https://doi.org/10.1109/IISWC.2016.7581261

Livny M, 2013. From Principles to Capabilities—the Birth and Evolution of High Throughput Computing. Wisconsin Institutes for Discovery, University of Madison-Wisconsin, USA.

Ma JY, Wang HB, Zhang LX, et al., 2015. Supporting differentiated services in computers via programmable architecture for resourcing-on-demand (PARD). *ACM SIGPLAN Not*, 50(4):131-143. https://doi.org/10.1145/2775054.2694382

Xie Y, 2016. Technology-driven architecture innovation: challenges and opportunities. Proc 43rd ACM/IEEE Int Symp on Computer Architecture, Architecture 2030 Workshop.

Xu ZW, 2012. How much power is needed for a billion-thread high-throughput server? *Front Comput Sci,* 6(4):339-346. https://doi.org/10.1007/s11704-012-2071-5

Xu ZW, Li CD, 2017. Low-entropy cloud computing systems. *Sci Sin Inform,* 47(9):1149-1163. https://doi.org/10.1360/N112017-00069

Xu ZW, Chi XB, Xiao N, 2016. High-performance computing environment: a review of twenty years of experiments in China. *Nat Sci Rev,* 3(1):36-48. https://doi.org/10.1093/nsr/nww001

Yu Z, 2017. Labeled RISC-V: a new perspective on software-defined architecture. Proc 6th RISC-V Workshop, p.7.