

Article

Initial Results of Testing Some Statistical Properties of Hard Disks Workload in Personal Computers in Terms of Non-Extensive Entropy and Long-Range Dependencies

Dominik Strzałka 

Department of Complex Systems, Faculty of Electrical and Computer Engineering, Rzeszów University of Technology, Al. Powstańców Warszawy 12, Rzeszów 35-959, Poland; strzalka@prz.edu.pl; Tel.: +48-17-865-1104

Received: 14 April 2017; Accepted: 23 June 2017; Published: 5 July 2017

Abstract: The aim of this paper is to present some preliminary results and non-extensive statistical properties of selected operating system counters related to hard drive behaviour. A number of experiments have been carried out in order to generate the workload and analyse the behaviour of computers during man–machine interaction. All analysed computers were personal ones, worked under Windows operating systems. The research was conducted to demonstrate how the concept of non-extensive statistical mechanics can be helpful in the description of computer systems behaviour, especially in the context of statistical properties with scaling phenomena, long-term dependencies and statistical self-similarity. The studies have been made on the basis of perfmon tool that allows the user to trace operating systems counters during processing.

Keywords: complex systems; non-extensive entropy; operating systems; human computer interaction; workload; physics of processing

1. Introduction

One of the main areas of interest in computer engineering is the description of behaviour of selected parts in modern computer systems. The understanding of complex processes that appeared in computer systems is a crucial challenge not only for better understanding of physical and logical phenomena that exist on hardware and software levels, but also to make a significant progress in the design of new solutions and improvement of their performance [1]. It will be a trivial statement, but the level of complexity of modern computer systems is considerable and the same can be said about the complexity level of processed tasks. The multi-task processing and very sophisticated access interfaces incorporated in operating systems enable personal computer users to work in a very convenient and efficient way. It is obvious that these solutions require significant amount of available computer resources, but it should be remembered that these resources are managed by operating systems in such a way that both efficiency and convenience should be guaranteed [2]. This is not easy taking into account the fact that the amount of available resources is always limited [3]. It is worth noting that the structural complexity of computer systems (hardware level) is connected with software complexity that is processed. Therefore, it is difficult to guarantee that these complex systems will work efficiently assuming that most of this work is done as a combination of algorithmic and interactive processing [4]. The aforementioned combination may pose a problem, especially if physical and technological constraints are considered.

The statement that computer systems are complex is not a new one, it has its roots in the works by Dijkstra [5] in the seventies of the 20th century and a few years later by Gell-Man [6]. The term complex systems is usually referred to the pioneering works by von Bertalanffy (a father of General

Systems Theory) and his followers [6]. Currently, it is believed by many researchers that this is the science of the XXIst century [7] and a lot of attention is paid to recognising, describing, modelling and stimulating a number of natural and artificial systems in order to understand their real behaviour [8]. This is done in a spatial and time domain looking for emergent phenomena that are not visible at a first glance because they appear as a consequence of many interactions between different parts of the systems analysed. However, it is not common practice to use this interesting approach in the field of computer science and engineering. In this work, we would like to emphasize that complex systems approach seems to be interesting and helpful in better understanding of computer systems behaviour similarly to approaches presented in sociology and social networks [9], economy [10,11], ecology [12], biology [13], hydrology [14], medicine [15], etc.

There are many important aspects of system analysis in terms of complex systems. This paper focuses on analysis of time series that represent behaviour of two selected operating system counters. We would like to present their complex properties including power-law behaviour of probability distributions, long-term dependencies in time series and scaling. Conclusions presented here are based on statistical methods that are related to the concept of non-extensive entropy and statistics.

The approach proposed in this paper is simultaneously an extension to the results which have already been demonstrated in literature. For example, as shown in [16], the authors have collected data for the UNIX file server, proposed a method for simulation of time and spatial correlations with analysis of Boltzmann–Gibbs entropy trying at the same to understand the real nature of bursty disk traffic. The approach presented in paper [17] is similar to the one in question; however, the authors have collected data of the disk arrival patterns focusing on three computer systems: *cello*—time sharing system (similar to [16]); *snake*—a file server; and *hplajw*—one personal workstation. All systems working under a UNIX operating system. They have used three methods for calculations of the Hurst parameter, yet they have neither included information about how the data traces were collected, nor about the analysed processes probability distributions and their stationarity. Hence, the calculations of the H parameter should be improved. Ref. [18] deals with a study of self-similarity (based on methods given by SELFIS tool) of RAID-arrays in various systems (web-servers, e-mail servers, game consoles). There is no information, however, how data sets were collected (due to Non-Disclosure Agreement). Furthermore, the study does not include any information about the processes probability distributions and stationarity. The researchers used two methods that allow calculations of Hurst exponent; however, they did not propose any information about statistical properties of probability distributions. Therefore, the obtained results are incomplete—see the details in Section 2.3. There is an extension of previous results in paper [19]; however, the traces were collected from enterprise systems with different resolutions. The authors analysed the dynamics of the read and write traffic, the disks utilization out of the entire drives family. The analysis presented in [20] refers to Windows NT systems (seven different servers), but the workload was also generated with regard to different benchmarks. The collected records were based on Event Tracing for Windows (ETW) tool. On the other hand, in Ref. [21], it was noted that the existence of statistical self-similarity at large time scales does not significantly affect disk behaviour with respect to response times.

The purpose of this paper is to confirm the results presented in [17–20] and references therein, as well as focus on personal computers (workstations) working under the Windows 7 operating system where the workload was generated only during long-term human–computer interaction. The perfmon tool, which enabled to trace the results, is a part of every Windows system from Windows ME. Therefore, it is not necessary to have any special tracing components of computer programs. The collected traces were recorded with sampling resolution 1 s for two important system counters, namely Average Disk sec/Transfer and Disk Transfers/s, facilitates the extension of the already existing approaches. Owing the analysis of power-law behaviour of probability distributions, it is worth noting that the non-extensive entropy formalism may be adopted in the quest for better understanding of complex hard disk behaviour in relation to operating system management.

The paper is divided into five sections. The foreword in the Introduction, Section 2, where mathematical background with necessary definitions have been presented, Section 3, which includes a short description of the experiment that has been conducted, Section 4 where all the calculation results have been proposed, and Section 5 is devoted to Conclusions.

2. Non-Extensive Statistical Mechanics and Long-Range Dependencies

The concept of entropy non-extensivity has its history in literature, dating back to 1988 when Tsallis published a research paper on the then-new idea [22]. Since then a gradual development of this theory has been observed and it is quite commonly accepted that there is no possibility to understand the behaviour of complex systems without the idea of non-extensive entropy [23,24]. The term non-extensive is understood as a feature of the system in which its properties entirely differ from the sum of the characteristics of system components and cannot be inferred on the basis of a simple summation of these components. In this work, we take into consideration some applications of this theory in statistics (statistical mechanics) called non-extensive statistics mainly used for analysis of time series.

2.1. Definitions of Non-Extensive Entropy

Non-extensive formalism is based on the concept of q -generalization of logarithm given by

$$\ln_q(x) = \frac{x^{1-q} - 1}{1-q} \quad q \in \mathbb{R}, \quad \ln_1(x) = \ln(x). \quad (1)$$

Based on (1), Tsallis proposed in [22] to extend classical definition of Boltzmann-Gibbs entropy given by

$$S_{BG} = -k \ln(W), \quad (2)$$

where k is Boltzmann constant and W stands for the number of system microstates to the form:

$$S_q = -k \ln_q(W). \quad (3)$$

Equation (1) has a pseudoadditivity property

$$\ln_q(x \cdot y) = \ln_q(x) + \ln_q(y) + (1-q) \ln_q(x) \ln_q(y), \quad (4)$$

which means that the principal property of entropy, called additivity, is strictly related to values of q parameter giving for $q > 1$ pseudo-additivity and for $q < 1$ super-additivity.

System microstates W are given with the set of probabilities p_i and in discrete cases result in:

$$S_{BG} = -k \sum_{i=1}^W p_i \ln(p_i) \quad \text{with} \quad \sum_i p_i = 1 \quad \text{and} \quad p_i \in \langle 0, 1 \rangle, \quad (5)$$

which, for equiprobable states ($p_i = 1/W, \forall i$), gives (2). In the case of (3), we have:

$$S_q = k \frac{1 - \sum_{i=1}^W p_i^q}{q-1} \quad \text{and} \quad \sum_i p_i^q \neq 1. \quad (6)$$

2.2. Probability Distributions in S_q Entropy

In the continuous form, the set of p_i probabilities is given by the function $p(x)$. The extremization of (5) in continuous form

$$S_{BG} = -k \int_{-\infty}^{+\infty} p(x) \ln p(x) dx, \quad (7)$$

with the constraints $\int_{-\infty}^{+\infty} p(x)dx = 1, \langle x \rangle \equiv \int_{-\infty}^{+\infty} xp(x)dx = 0, \langle x^2 \rangle \equiv \int_{-\infty}^{+\infty} x^2p(x)dx = \sigma^2$, leads to the Gaussian equilibrium probability distribution given by:

$$p_{BG}(x) = \frac{e^{-\beta x^2}}{\int_{-\infty}^{\infty} e^{-\beta x^2} dx'} \tag{8}$$

where $\beta > 0$ is the Lagrange parameter determined by σ^2 and (8) is a ground state of the Central Limit Theorem (CLT) [25]. Cognately, the same can be applied to (6) and as a result the stationary $p_q(x)$ distribution called q -Gaussian can be obtained:

$$p_q(x) = \frac{e_q^{-\beta x^2}}{\int_{-\infty}^{\infty} e_q^{-\beta x^2} dx} \quad (q < 3; \beta > 0), \tag{9}$$

where e_q is called the q -exponential function given by:

$$e_q^x = [1 + (1 - q)x]^{1/(1-q)} \quad \text{with} \quad e_1^x = e^x \tag{10}$$

as the inverse function to Equation (1).

The most surprising point in (9) appears when we have a set of N such random variables and their sum given by

$$X_N = \sum_{i=1}^N p_i \tag{11}$$

is calculated because, for $q < 5/3$ and $N \rightarrow \infty$, it converges to a (stable) Gaussian distribution (variance of distribution (9) is limited), but, for $q > 5/3$, relation (9) converges to a set of α -stable Lévy distributions [26]. It is a well-known fact that α -stable distributions for many years haven't got commonly accepted reference to statistical mechanics (paper [27] has elucidated this situation), but, on the other hand, they were used in statistics to generalize CLT for probability distributions with power laws.

Owning the non-extensive entropy the α parameter can be described as

$$\alpha = \frac{3 - q}{q - 1} \quad \text{with} \quad q \in \left(\frac{5}{3}, 3\right). \tag{12}$$

It can be also worth noting that for $q < 5/3$, one has distributions with finite variance, whereas, for $q \geq 5/3$, the variance is infinite.

Tsallis's definition of entropy also offered a new kind of formalism for probability distributions in stationary states; according to [28], we can define q -normal distributions:

$$P(x) = N_q \left(1 + B_q x^2 (q - 1)\right)^{\frac{1}{1-q}}, \tag{13}$$

where

$$B_q = \left((5 - 3q) \sigma^2\right)^{-1}, \tag{14}$$

with σ^2 as a data variance. N_q is given as

$$N_q = \frac{\Gamma\left(\frac{1}{q-1}\right)}{\Gamma\left(\frac{3-q}{2q-2}\right)} \sqrt{\frac{q-1}{\pi}} B_q, \tag{15}$$

with Γ Gamma function. When $|x| \rightarrow \infty$ Equation (13) takes the form: $P(x) \sim x^{2/(1-q)}$, and for cumulative distribution, there is $P(|a| > x) \sim x^{-(3-q)/(q-1)}$ as it is in Equation (12).

The existence of power-law distributions in the analysed system indicates that the system:

- is out of equilibrium [29];
- some of its statistical properties (especially second moment) are difficult to be interpreted [30];
- is governed by long-term (time domain) and long-range (spatial domain) dependencies [31];
- is described by multifractals and scaling phenomena [32];
- has complex spatial structure and collective dynamics [33].

The concept of non-extensive (mechanical) statistics, outlined above, has become one of the most interesting ideas in recent years leading to many important achievements and results—see [34] and references therein.

The existence of power-law distributions is not the only manifestation of non-extensive statistics. For example, regarding the results given in [35,36], a relation between the non-extensive q parameter and the statistical self-similarity Hurst exponent H , which is used to measure long-range dependencies in complex systems leading to $H = 1/(3 - q)$, can be suggested.

2.3. Long-Range Dependencies

Statistical self-similarity and multifractals are well-known features of many complex systems enabling a better understanding of their behaviour. The existence of long-term dependencies in different time series was first observed by Hurst [37].

Generally, the process $X(t)$ can be seen as self-similar if for some $H > 0$ a relation (16) holds:

$$X(at) \stackrel{d}{\approx} a^H X(t) \quad \text{for every } a > 0. \quad (16)$$

Equation (16) states that if the process is self-similar, its invariant under suitable translations of time and scale [38]. Usually, t is time and $X(t)$ is a process space—relation (16) shows that change of time scale $a > 0$ corresponds to change of space scale a^H . There are two main classes of self-similar processes: fractional Brownian motions (fBm), where H exponent is as a parameter that reflects the process memory effects (called long-term (long-range) dependencies) measured by d , $d \in \langle -0.5, 0.5 \rangle$ and $H = d + 1/2$, but also gives information of time series persistence. The second class of such processes are—Lévy-stable processes (with power-like distributions), where $H = 1/\alpha$, and Hurst parameter is responsible for space scaling. In the most complex case, there are Lévy processes with memory effects described by d parameter and $H = 1/\alpha + d$.

Since Hurst's discovery, a lot of interesting, similar examples have been shown where the ubiquity of such phenomena is very surprising; additionally, it enabled a better understanding of the real nature of the complex world. In the case of computer systems, the existence of long-range dependencies can lead to the decrease of throughput [39]. There are plenty of statistical methods that are used for calculations of statistical self-similarity—their detailed description and analysis is shown in [40]. In our study, we show the results of experiment where 10 different computers were traced by a perfmon tool. There is a list of necessary details for each computer hardware configuration in Table 1. The results were obtained on the basis of proven statistical methods used for analysis of probability distributions and long-range dependencies. The analysis of every single computer enables to obtain a set of statistics that consists of: determination of the slope of heavy tail probability distribution and Hurst parameter H with information about long-range dependencies measured by d parameter.

Table 1. Configurations of computers systems used in the experiment.

| Id | CPU | RAM | HDD | OS | Number of Records |
|----|--|------------------|---------------------------------------|-------|-------------------|
| 1 | AMD Athlon X2 Dual-Core QL-65 2.10 GHz | 2.0 GB DDR2 | Hitachi HTS5432225L9A300 ATA | Win 7 | 698848 |
| 2 | DualCore Intel Core i5 450 M, 2.666 GHz | 4.0 GB DDR3 | Western Digital WD5000BEVT-22A0RT0 | Win 7 | 872891 |
| 3 | Intel Core i5 CPU M 520 2.4 GHz | 8.0 GB DDR3 | Seagate ST9500420AS ATA | Win 7 | 439615 |
| 4 | DualCore AMD Athlon II X2 250, 2.952 GHz | 4.0 GB DDR2 | Seagate ST3500418AS ATA | Win 7 | 557020 |
| 5 | QuadCore AMD Phenom II X4 Black Edition 955, 3.2 GHz | 8.0 GB DDR2 | Seagate ST31000528AS ATA | Win 7 | 684518 |
| 6 | Intel Pentium Dual-Core E5200 2.5 GHz | 2.0 GB PC800 CL4 | Seagate ST3500320AS ATA | Win 7 | 506599 |
| 7 | Intel Core 2 Duo CPU, T5450, 1.66 GHz | 3.0 GB DDR2 | Seagate ST9250827AS ATA | Win 7 | 1048569 |
| 8 | Intel Core 2 duo P7350 2.00 GHz | 3.0 GB DDR2 | Fujitsu MHZ2320BH G2 ATA | Win 7 | 309766 |
| 9 | AMD Athlon 64 X2 Dual-Core TK-55 1.80 GHz | 3.0 GB DDR2 | Hitachi HTS542512K9SA00 ATA | Win 7 | 696955 |
| 10 | Intel Core i5 650 3.20 GHz | 4.0 GB DDR2 | Seagate ST3500320AS ATA | Win 7 | 519946 |

3. Experiment Details

A typical approach for modelling different problems of workload and processing in computer systems is usually referred to the theory of queues. This gives a very comprehensive and useful approach including many analytical solutions and simulation results that have practical applications. This is a natural consequence of the approach proposed by Kendall, Kleinrock and their followers. However, it should be remembered that most of these solutions can be calculated according to some assumptions that may be too restrictive to be able to model real-world situations precisely [41]. The most important limitations include: the possibility that the waiting space may be limited, the arrival rate is state dependent, the arrival process is not stationary (peak, slack and bursty periods), and the queue discipline may not be first come first served (FCFS). Moreover, queuing models give steady state giving a lot of information, for instance whether the queuing system operates long enough excluding unsteady states of the system. In a typical service system, we denote by A the number of income tasks (requests to be served) and by X the number of served tasks C during time T and ($X = C/T$), then $\lambda = A/T$ parameter can set as the requests intensity. Only if $\lambda \leq X$ is the system in a stable state [42]—for the rest of the cases, the system is out of equilibrium.

If we denote by N the number of served tasks and by R the system response time, it can be written, according to Little's law [43], that $NT = CR$, which gives $N = RX$ for $\lambda \leq X$. For closed service systems (served tasks come again into the system), there is also user waiting time Z , thus $N = (R + Z)X$. There are certain limitations incorporated in service systems that describe performance in terms of throughput $X(N)$ (the number of served tasks per one unit of time) and response time $R(N)$. They are expressed as follows [42]:

$$\begin{aligned} N/(ND + Z) \leq X(N) &\leq \min\{1/D_{max}, N/(D + Z)\} \\ ND \geq R(N) &\geq \max\{ND_{max} - Z, D, \} \end{aligned} \quad (17)$$

where D_{max} is the maximal service time at service node i . If there is only one served task in the system, ($n = 1$), the left part of first formula in Equation (17) is given as

$$X_{max}(N) = 1/(D + Z). \quad (18)$$

If the number of served tasks N increases, $X(N) = N/(D + Z)$ (the right part of formula (17)) and according to [44] (Equation (6)) it can be written that

$$X(N) = dN(t)/dt = rN(t), \quad (19)$$

where r is the rate of system performance growth—in the simplest case $r = 1/(D + Z) = const.$

In the Boltzmann–Gibbs entropy, the size frequency distribution function $N(x)$ is given as:

$$\frac{dN(x)}{dx} = -\lambda N(x) \quad \lambda = const, \quad (20)$$

with the solution:

$$N(x) = N_0 \exp(-\lambda x). \quad (21)$$

Similarly, Tsallis q -entropy proposes the following equation:

$$\frac{dN(x)}{dx} = -\lambda N^q(x) \quad \lambda = const; \quad (22)$$

thus, we can finally observe that

$$N(x) = N_0 e_q(-\lambda x) = \frac{N_0}{[1 + (1 - q)\lambda x]^{\frac{1}{1-q}}}, \quad (23)$$

with $e_q(x) = (1 + (1 - q)x)^{\frac{1}{1-q}}$ and $q \neq 1, 1 + (1 - q)x \geq 0$. If $q > 1$, Equation (23) results with power-law behaviour of distribution $N(x)$ when $x \rightarrow \infty$.

Comparatively, following the same way of thinking, Equation (19) can be extended to Equation (22), exposing the possible connection of Tsallis entropy and service system performance. A similar approach was proposed, for instance, by [45].

A literature review in the field of hard disk behaviour analysis (as it presents, for example, in the Introduction) shows that most of the data gathered so far focuses on the analysis of statistical self-similarity without any references to statistical properties of distribution probabilities. Figures 1 and 2 illustrate easily noticeable bursty periods and high peaks of measured parameters. As shown in Table 2, all values of mean and standard deviation suggest that there is no evidence of possible existence of heavy tailed distributions in the analysed process. Tsallis's entropy presents a convincing background for such distributions. Thus, the entropy in question may be considered as a "denominator" for different approaches presented so far.

The authors focus in this paper on real measurements that were obtained in different personal computer systems that worked under Windows family of operating systems (Windows 7). These measurements were done on the basis of the solutions that are inside this system, especially those that are available in the system performance monitor called perfmon. This useful administration tool is included in Windows systems and can be used for tracing different parts of the system [46]. In most cases, the results obtained due to this monitor are used by many system administrators to get the overview of the computer systems (especially servers) and are helpful in ongoing management. There are many technical and online reports that explain how to gather data from the perfmon and how the recorded findings may be used to monitor and optimize system performance and reliability. On one hand, a great number of these reports give exact advice; on the other hand, they do not focus on the possibility to make advanced statistical analysis. In this paper, such a possibility is taken because

perfmon allows for gathering data as time series with different sampling time—the highest is 1 s. Some online forums report that this is the best solution for data analysis [47]. Additionally, an official Microsoft Technet Support believes that [48] *If you do set it to 1 s, it could tax your system*, but it is also written that: *the short period of time is for being able to see items such as something kicking off that triggers the problem that you are seeing*. For 1 s sampling resolution, it is always easy to have lower resolutions (minutes, hours, days) using sums or averages over the period of time. Resolution 1 s allowed to catch as much as possible details in counter behaviour and helped to achieve very long time series (see Table 1).

Table 2. Calculated means, standard deviations and min/max values for time series of Average Disk sec/Transfer counter.

| Id | μ | σ | Min. | Max. |
|----|---------|----------|--------------------------|--------------|
| 1 | 0.15315 | 27.84289 | 1.99998×10^{-4} | 9293.79366 |
| 2 | 0.0905 | 18.10498 | 1.50105×10^{-4} | 7757.53781 |
| 3 | 0.00582 | 0.03968 | 2.50004×10^{-4} | 2.5388 |
| 4 | 0.00288 | 0.01095 | 3.17642×10^{-4} | 2.19522 |
| 5 | 0.00261 | 0.01501 | 7.37×10^{-5} | 1.71806 |
| 6 | 0.00315 | 0.03627 | 1.99997×10^{-4} | 25.07448 |
| 7 | 0.00879 | 0.01669 | 0 | 2.13655 |
| 8 | 0.08761 | 15.45299 | 1.49874×10^{-4} | 7599.39 |
| 9 | 0.24258 | 35.72358 | 1.49984×10^{-4} | 15,152.58086 |
| 10 | 0.00293 | 0.01852 | 0 | 3.6076 |

The key issue when using perfmon tool is the fact that performance system merely generates any additional workload. It needs to be remembered that tracing of computer system is not easy, as it is necessary to have a special computer program that allows data collection. However, such a program is also processed the on investigated system and influences obtained results. Perfmon presents data that is usually gathered by operating systems in order to guarantee its work. Nonetheless, perfmon does not create any performance data per se, and it only shows data provided by other Windows subsystems. In [49], it was calculated that if perfmon stores data on hard drive for computers with Windows ME and CPU 550 MHz, during 40 k of Input/Output operations per second, it adds about 5% (~ 2 k) of additional workload; thus, its possible influence on final results can be neglected.

It was also assumed in our experiment that computer workload was made by normal computer users (humans). Presented considerations are based on long-term tracing of 10 different (with different hardware configurations) personal computers that were used for *normal, typical work*. This is the case in a situation when computer users are generating workload that is mostly based on office programs, Internet browsers, multimedia, email clients, etc. The experiment was hands-off where the users could use the computer and the Internet freely, without any scenarios. However, it was emphasized that, during one session, which should last at least 1 h, they should browse through webpages, check their mail several times, use word processor or spreadsheet, play games, etc. It was also assumed that normal work with the computer permits short breaks, thus no computer–user interaction, which should last no longer than fifteen minutes. It was not necessary to conduct any additional tests or benchmarks. The approach described here has its strengths and weaknesses. Firstly, it is clearly noticeable that its weakest point is the fact that every computer user chooses his or her unique ways of interaction as well as preferred computer programs. Some common features can be found, however generally, the behaviour of one user cannot be copied by another. This can be guaranteed if special tests based on benchmarks are used. Even so, such tests are not realistic and do not reflect the whole possible range of different situations. Extreme (yet repeatable) values of workload are commonly generated by benchmarks due to artificial (unrealistic) tests, which does not reflect normal computer work. The question is, however, how many times does a typical computer users force the computer to process such workload? Moreover, according to [20], it should be remember

that: the real-world behaviour drawn from analyses of benchmark traces are only as accurate as the benchmarks' accuracy in representing real-world environments.

As it was noted before, the typical approach for modelling queuing systems is based on Kendall's notation and efforts to find analytical solutions. After the development of first simple (Poisson) models, most of the more complex obtained results were related to (computer) networks with (heavy) traffic models (especially those with *fractal* nature) [50,51]. As far as hard disks are concerned, the most popular measure of their performance is the average access time expressed by waiting time (understood as the time until the disk head is over the track to serve the request) plus rotational latency (mostly governed by rotational disk speed) and transfer time (necessary to read/write requested data) [52]. One can also find another term: "seek time" considered as the amount of time needed to the reposition of head during request service, but this is the part of wait time, and "seeking" stands as a synonym for process of moving the read/write head [53]. In regard to the queuing theory, usually a hard disk drive is modelled as a FIFO M/G/1 queue, and this is connected with an implicit assumption that the speed of seeking and the task of scheduling algorithm do not influence the overall performance [53]. Nonetheless, this is not necessarily the truth because hard disks have special requests' scheduling schemes that unnecessarily reflect the FIFO regime. Among them, there are: SCAN, C-SCAN, Shortest Seek Time First, Shortest Positioning Time First (both better than SCAN/C-SCAN in average access time but have higher variance), First Come First Served, the latter being the worst one. Moreover, the term "average access time" is used as a meaningful measure, but it is important only when the system is ergodic and has well established distribution over states (is in equilibrium state).

In this paper, a statistical analysis of the following system counters is presented:

1. Average Disk sec/Transfer (that consists of a sum of counters Average Disk sec/Read and Average Disk sec/Write). It represents the average time the disk transfers (reads/writes, I/O requests) took to complete, in seconds (the counter has a millisecond precision). It does not include the time that is necessary to be spent in the system queue but is the most important counter that reflects the physical disk properties; they are usually related to the disk speed, and, for many computer systems, one can find some recommendations about their suggested or critical values [52].
2. Disk Transfers/s. It is a counter that shows the number of transfers (consisting of disc read/write) during a time unit (1 s for the purpose of this paper). It shows how many different application requests are necessary to be handled by the disk.

In perfmon, for physical disk, the data is captured at the Partition Manager level in the storage stack (Figure 3). Whenever the application demands an I/O request, it uses the Windows I/O Subsystem as an intermediate stage (see the top of the stack in Figure 3) to serve it. Depending on the request type, it can be sent further to the Files System, which imposes the structure of the files in the operating system, and to the Volumes Manager, which presents disk volumes like C:, D:, E:, etc., which decides, for example, which physical hard drive is taken and then to the Partitions Manager that manages logical disks (partitions). Below the Partitions Manager, there is a Device Classes level that acts as a manager of the device type (hard disk, tape, CD, etc.)—and the Port/Miniport level that is responsible for the transport protocol (SCSI, FC, SATA, etc.) along with the device driver for the Storage Adapter (supplied by the vendor of the device). Disk Subsystem is a physical level of this structure considered as a simple cable connected to a single physical hard disk or even more complex solution like a Storage Area Network (SAN). Average Disk sec/Transfer counter measures the whole time spent below the Partitions Manager level. However, the important issue is not only how long each request will be served, but also how many requests should be served. This is the role of the Disk Transfers/s counter, which gives profound insights into a given problem. The analysis of the findings on this counter permit a better understanding of system non-stationarity or long-range dependencies (the existence of peak, slack and bursty periods).

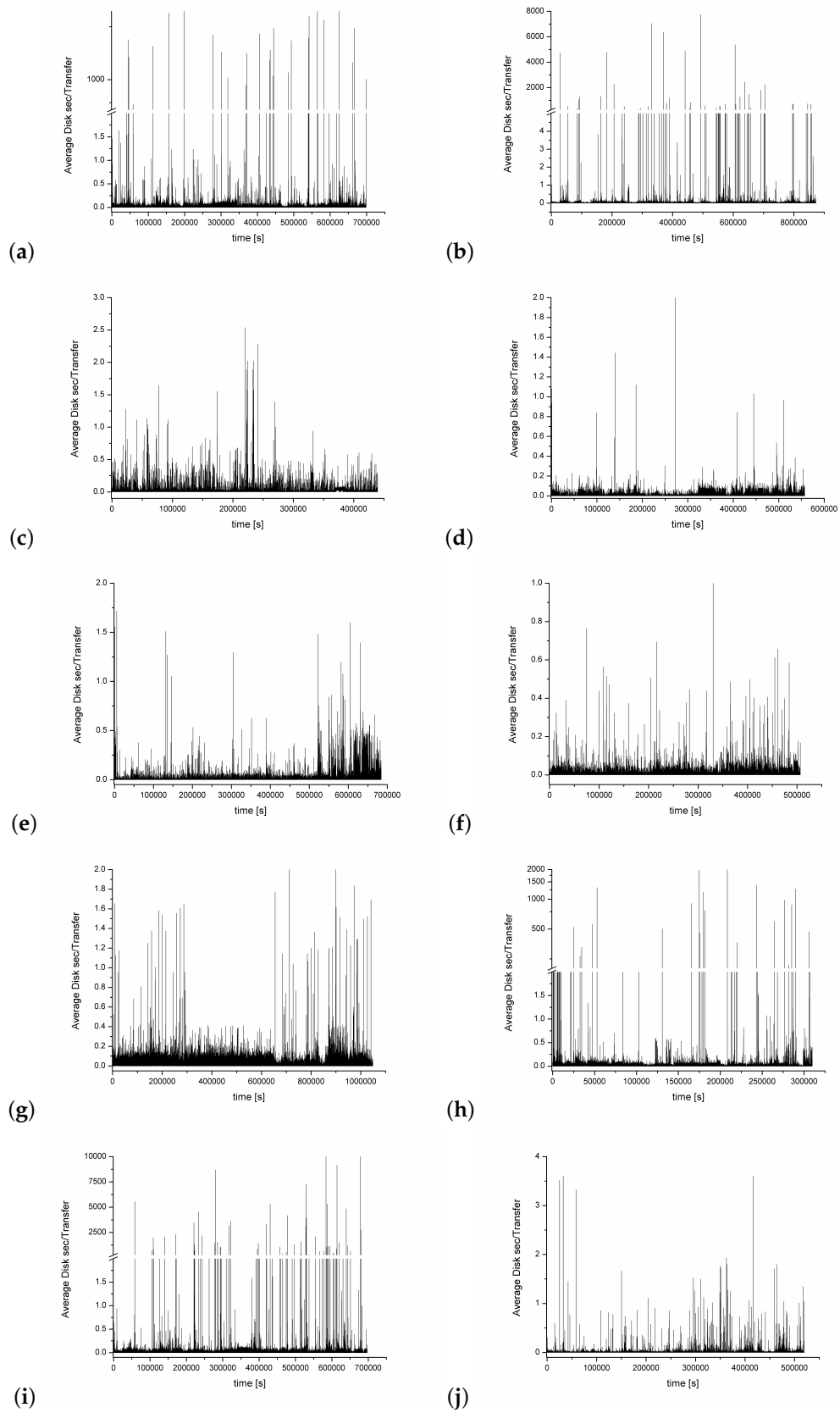


Figure 1. Behaviour of Average Disk sec/Transfer counters for computers with Id 1 ÷ 10 (subfigures (a–j)). In the case of (a,b,h,i) 0 y-axis has a break and the second part is plotted in log scale.

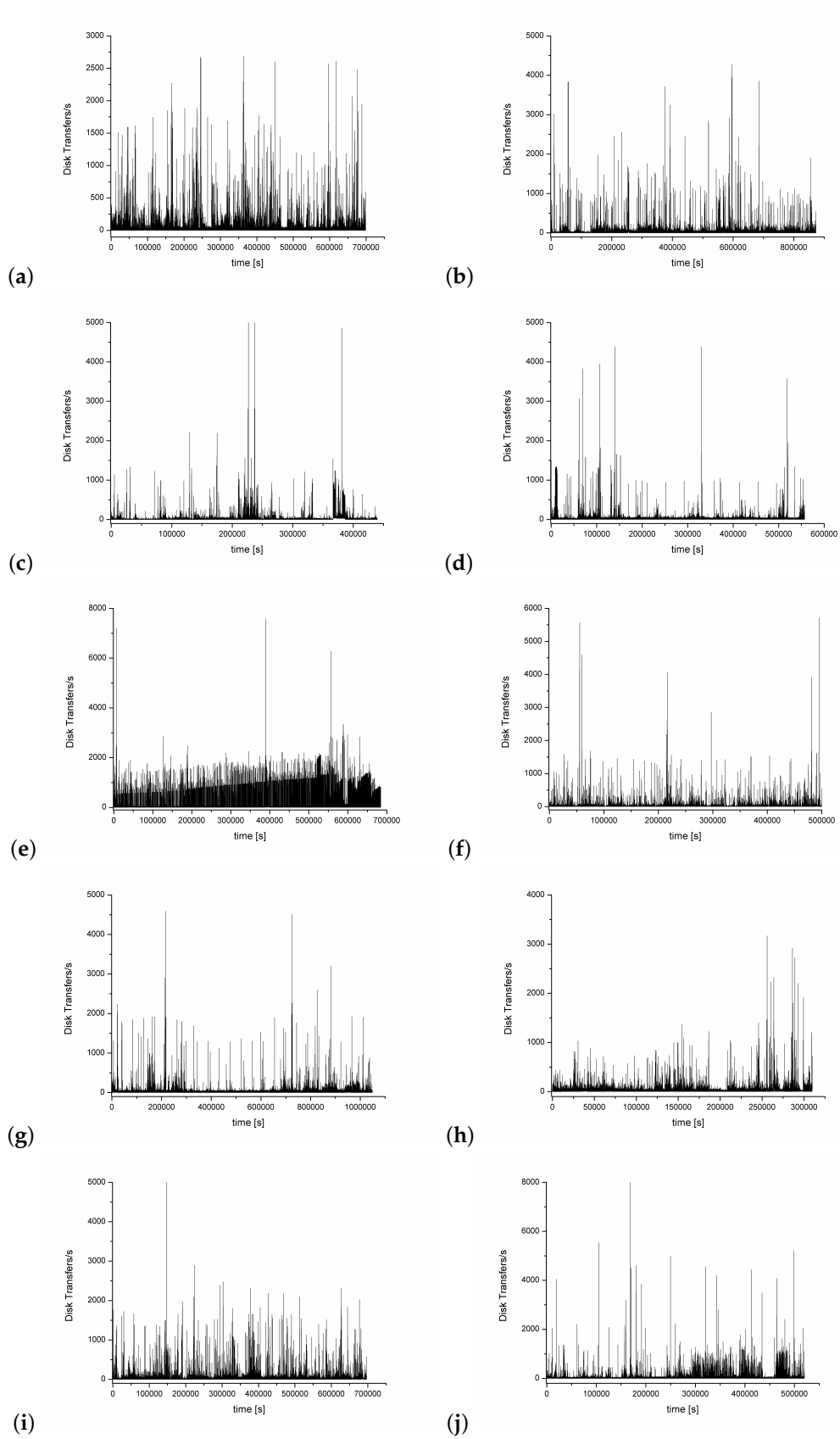


Figure 2. Behaviour of counter Disk Transfers/s for computers with Id 1÷10 (subfigures (a–j)).

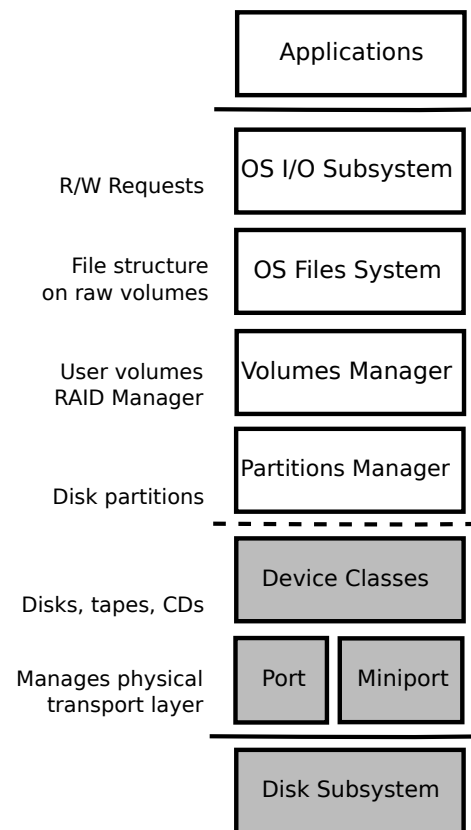


Figure 3. Windows storage stack.

4. Experiment Results

In this paper, we focus mainly on a real computer-user behaviour traced for some periods of time. The shortest available data set has 309,766 observations, whereas the longest >1 M (see details in Table 1). In every case, the collected data is considered as a time series that permit statistical analysis. Figure 1 shows the behaviour of Average Disk sec/Transfer counter of each traced system. Taking into account the data presented in Table 1, it is worth mentioning that, in the case of non-Seagate disks (Hitachi—Id. 1 and Id. 9, Western Digital—Id. 2, Fujitsu—Id. 8), the range of the counter fluctuations is much bigger than in the case of others. On the basis of information gathered in Table 2, which shows calculated mean μ and standard deviation σ for all disks, it is easy to notice that Seagate disks have the smallest values of μ and σ . The same principle can be applied to min and max values. If we try to compare their observations with the amount of available RAM memory in computers, we can pose a question about whether there is any relation between the amount of the available memory and the hard disk type. In order to prove this, further investigations need to be conducted, and a greater number of computers with hardware configurations are also necessary. Nevertheless, in order to present a dynamical nature of this counter for Figure 1a,b,h,i, we put a break in the 0 y -axis. This is caused by max values for these Ids, and it reaches more than 7 k.

The counters Disk Transfers/s are shown in Figure 2. Again, the attentive observer would be able to see the periods of bursty behaviour. The existence of high peaks in these time series enables to make an assumption that the Tsallis formalism plays an important role in understanding of counters behaviour; however, this will be confirmed when the processes of distribution probability are tested.

Our analysis starts with statistics that refer the probability distributions of time series to Tsallis proposals. It was assumed that these probabilities can have slowly vanishing tails for large counter values. The simplest approach assumes that estimation of the distributions slope is done on a log-log probability distribution plot with the fit of allometric equation $y = \beta x^\alpha$, where α is the slope. In addition, a more accurate graphical method is suggested referring to log-log plot of $P(|a| > x) \sim x^{-(3-q)/(q-1)}$ and allometric fit for large values of x . However, both of these methods may not necessarily lead to convincing results of tail behaviour—they show its power-law property, but the estimation of α parameter is not necessarily exact [54]. Other methods that can be used are based on Hill estimator or maximum likelihood estimator [55]. The estimation of the heavy tail index $\hat{\alpha}$ by means of the Hill method is quite often used by researchers. It is believed that this is one of the best methods [56]. The estimator is based on the approach where, for a set of Y_i of recorded data, a new ordered, in a decreasing fashion, set X_i is given, with X_1 being the highest value from Y_i . Then, the procedure given by Equation (24) is done, and, as a result, a plot of tail distribution is given:

$$\hat{\alpha}_k = \left[\frac{1}{k} \sum_{i=0}^{k-1} \ln \frac{X_{n-i,n}}{X_{n-k,n}} \right]^{-1} \quad \text{for } k \in \{1, 2, \dots, n\}. \quad (24)$$

The use of Hill estimator poses a problem when choosing the appropriate number k . This is a critical issue because this estimator is very sensitive when it comes to this choice. The most common technique applied to solve this problem is the use of plot $\hat{\alpha}$ vs. k as well as choosing a part of the plot that looks stable. There are also effective techniques and approaches based on statistical testing [56].

The estimation of an α index according to the Hill method is given in Figures 4 and 5. We based on a graphical approach having plots that show behaviour of α vs. sample size n . For Average Disk sec/Transfer counter (Figure 4), almost in all cases with the exception of (c), (d) and (e), the obtained values of α for large n are lower than 2. This suggests that probability distributions of analysed counters may have the property of infinite variance and description of counters by standard deviation (see Table 2) may lead to some misunderstandings, especially in the case of standard deviation. This is also exceptionally important if the property of statistical self-similarity is analysed because it is present in spatial and temporal domains. Heavy-tailed distributions are a manifestation of long-range spatial correlations. If the α stability index is lower than 2, we not only obtain processes with infinite variance, but also evidence that suggest that the system is in out-of-equilibrium state and the use of Tsallis thermostats should be preferred. In all cases of Disk Transfers/s, counter Hill plots show (see Figure 5) that the spectrum of α values are in the range of $\sim 1.6 \div \sim 2.3$ for large n . This particular counter represents the number of requests generated by applications that were handled by the disk. Taking into account that Windows I/O subsystem (Figure 3) is responsible for serving these demands, a more detailed view on a more complex nature of operating system behaviour may be observed. Again, this problem requires more time series (this case focuses only on 10) and other statistical tests and comparisons of results for different estimators. We assume that a rough estimation is enough in order to confirm or deny whether probability distributions are closer to Gaussian or power-law ones—some similar results were also given in [57], yet dealing mostly with cache memory.

According to Equation (12) having the slope α , it is possible to calculate the q index as

$$q = \frac{\alpha + 3}{\alpha + 1}. \quad (25)$$

Table 3 shows calculated slopes and q -indexes for all computer systems hardware configurations in the case of both counters. It is clear that the q index differs from value 1 in all cases—non-extensive entropy seems to be the right choice. Moreover, in most cases, $q > 5/3$, thus the power-law behaviour of distributions can be expected for both counters. A more detailed analysis is needed in order to confirm and define the obtained results; however, they indicate the existence of spatial scaling phenomena.

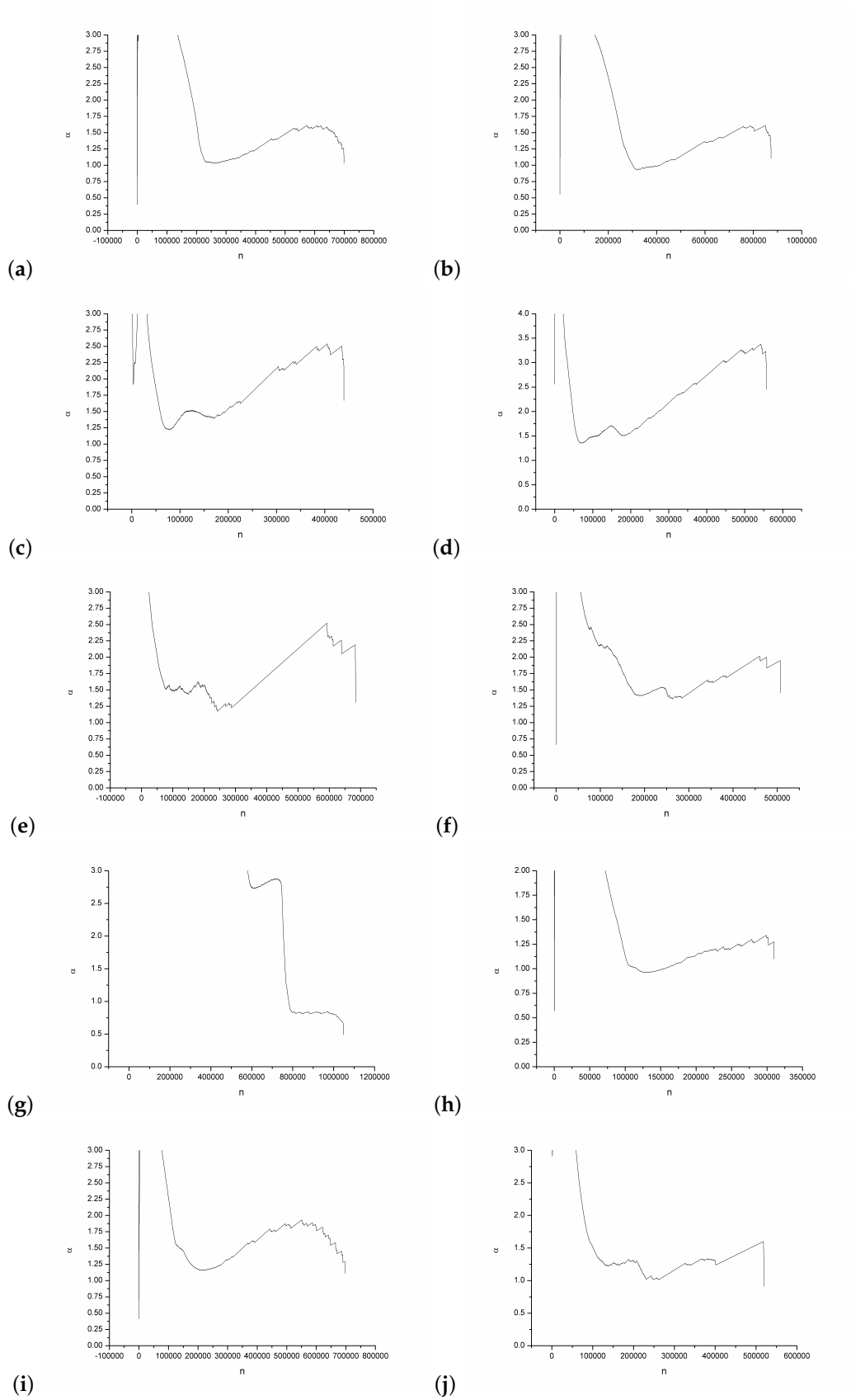


Figure 4. Hill estimator of counters Average Disk sec/Transfer for computers with Id 1 ÷ 10 (subfigures (a-j)).

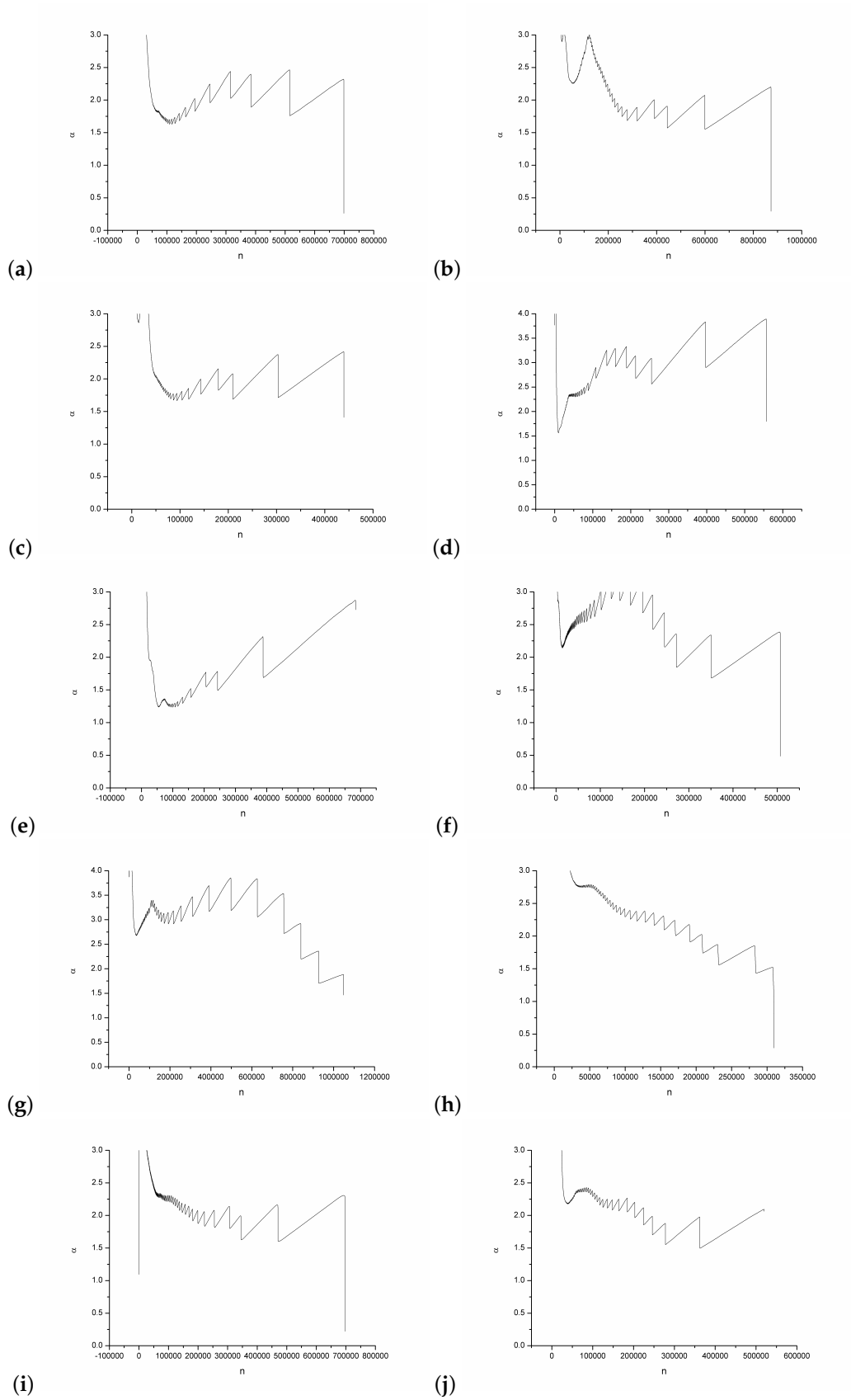


Figure 5. Hill estimator of counters Disk Transfers/s for computers with Id 1 ÷ 10 (subfigures (a–j)).

Table 3. Tsallis q -indexes for analysed counters.

| Id | Average Disk Sec/Transfer | | Disk Transfers/s | |
|----|---------------------------|------|------------------|------|
| | α | q | α | q |
| 1 | 2.19 | 1.62 | 1.54 | 1.78 |
| 2 | 1.88 | 1.69 | 1.67 | 1.74 |
| 3 | 2.37 | 1.59 | 1.79 | 1.71 |
| 4 | 2.01 | 1.66 | 1.23 | 1.89 |
| 5 | 2.14 | 1.63 | 1.64 | 1.75 |
| 6 | 2.28 | 1.61 | 1.63 | 1.75 |
| 7 | 2.15 | 1.63 | 1.95 | 1.67 |
| 8 | 1.64 | 1.75 | 1.85 | 1.70 |
| 9 | 1.88 | 1.69 | 2.03 | 1.66 |
| 10 | 1.67 | 1.74 | 1.74 | 1.73 |

The possible existence of long-range dependencies in analysed time series is checked by the use of two different methods (DFA and spectral density), due to which calculations of Hurst parameter H are feasible. There are other methods that can be used in this field; however, both of the selected methods work very well even with time series that can have non-stationary parts. The problem of time series stationarity analysis can be solved, for example, by the approach based on quantile lines test [58]. Here, we can assume that analysed time series are at least weak stationary owing to the fact that they have well-defined probability distributions. This tacit assumption is connected with the use of the above-mentioned methods, which enabled the production of convincing and reliable results. Details are given in Table 4.

Table 4. Long-range dependencies in analysed counters measured by H parameter.

| Id | Average Disk Sec/Transfer | | Disk Transfers/s | |
|----|---------------------------|----------|------------------|----------|
| | DFA | Spectrum | DFA | Spectrum |
| 1 | 0.501 | 0.53 | 0.92 | 0.91 |
| 2 | 0.494 | 0.5 | 1.04 | 0.98 |
| 3 | 0.988 | 0.99 | 0.96 | 0.95 |
| 4 | 0.858 | 0.78 | 0.98 | 0.99 |
| 5 | 0.791 | 0.77 | 0.86 | 0.79 |
| 6 | 0.654 | 0.55 | 0.94 | 0.89 |
| 7 | 0.878 | 0.83 | 0.84 | 0.88 |
| 8 | 0.488 | 0.5 | 0.90 | 0.93 |
| 9 | 0.509 | 0.51 | 0.77 | 0.8 |
| 10 | 0.773 | 0.74 | 0.70 | 0.91 |

For Average Disk sec/Transfer counter, six cases have a value of $H > 0.5$. The other four with Ids: 1, 2, 8, 9 have value H very close to 0.5. The most surprising fact is that they have exactly the same Ids like in Table 2 with high values of μ and σ . These cases can be a confirmation of a situation described and analysed in details in [59], where it was proved that, in the case of the Internet, if load increases, the Internet traffic tends to Poisson models. If there is little or no queueing on links, the nature of traffic is long-range dependent with bursty periods. Still, if the workload increases, the behaviour of network traffic (due to the superposition of marked point processes) pushes the statistical properties of traffic toward Poisson process, where long-range dependencies are non-existent. Obviously, the hypothesis needs further investigations. Another important issue may be the research in the field of multifractal spectrum analysis. The MF-DFA method (details and its improvement are given, for example, in [60]) can be used in order to obtain fractal spectrum of analysed time-series.

5. Conclusions

In this paper, it is shown that the mathematical formalism related to the non-extensive concept of entropy can give valuable outcomes. Our experiment and statistical analysis lead to some results, when referred to the concept of complex systems, scaling and long-range time and spatial dependencies can have interesting applications in the description of computer systems behaviour and be used in computer systems management, where we always have a problem of limited resources. We can easily notice that the computer operating systems is responsible for management of system limited resources—the main goal is to find such solutions that would hinder this fact for computer users. Usually, the simplest solution would be to buy a computer with a higher amount of RAM memory; however, our results show that there are certain phenomena on a hardware level that should be better recognized.

In order to confirm and expand obtained outcomes, further experiments and more detailed statistical analysis are needed. It is obvious that a set of experiments with a higher number of computers is necessary. It will be also very interesting to confirm obtained results with experiments mostly based on tests with benchmarks, but the challenge is even to find such benchmarks that will last for weeks.

In comparison to the previous works, which presents similar results in this field, we were not only able to show the self-similarity property measured by H parameter, but were also able to give a description of probability distributions analysis in terms of non-extensive statistics. It was possible due to experiments performed on personal computers during user–computer interaction. Table 1 gives information that these were mainly purpose computers working under Windows 7. The power-law behaviour of the analysed counters with estimation of α indexes can be the evidence that non-extensive statistics may play a crucial role in further development of so far obtained results, and the use of the complex system approach becomes more popular.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Wescott, B. *Every Computer Performance Book*; Create Space Independent Publishing Platform: Charleston, SC, USA, 2013.
2. Silberschatz, A.; Galvin, P.B.; Gagne, G. *Operating System Concepts*; John Wiley & Sons, Inc.: Somerset, NJ, USA, 2012.
3. Grabowski, F. Nonextensive model of self-organizing systems. *Complexity* **2013**, *18*, 28–36.
4. Wegner, P.; Goldin, D. Computation beyond Turing Machines. *Commun. ACM* **2003**, *46*, 100–102.
5. Wegner, P. Research paradigms in computer science. In Proceedings of the 2nd International Conference on Software Engineering, San Francisco, CA, USA, 13–15 October 1976; pp. 322–330.
6. Waldrop, M.M. *Complexity: The Emerging Science at the Edge of Order and Chaos*; Simon and Schuster: New York, NY, USA, 1992.
7. Dum, R. Science of Complex Systems for Tackling Challenges of the 21st Century: A Brief Overview. *Eur. Manag. Rev.* **2007**, *4*, 73–76.
8. Jacobson, M.J.; Wilensky, U. Complex Systems in Education: Scientific and Educational Importance and Implications for the Learning Sciences. *J. Learn. Sci.* **2006**, *15*, 11–34.
9. Benham-Hutchins, M.; Clancy, T.R. Social networks as embedded complex adaptive systems. *J. Nurs. Adm.* **2010**, *40*, 352–356.
10. Arthur, W.B.; Durlauf, S.N.; Lane, D.A. (Eds.) *The Economy as an Evolving Complex System, II, Santa Fe Institute Studies in the Sciences of Complexity Proceedings*; Addison-Wesley: Reading, UK, 1997; Volume XXVII.
11. Arthur, W.B. *Complexity Economics: A Different Framework for Economic Thought*; SFI Working Paper; Oxford University Press: Oxford, UK, 2013.
12. Anand, M.; Gonzalez, A.; Guichard, F.; Kolasa, J.; Parrott, L. Ecological Systems as Complex Systems: Challenges for an Emerging Science. *Diversity* **2010**, *2*, 395–410.

13. Rosenthal, S.B.; Twomey, C.R.; Hartnett, A.T.; Wu, H.S.; Couzin, I.D. Revealing the hidden networks of interaction in mobile animal groups allows prediction of complex behavioral contagion. *Proc. Natl. Acad. Sci. USA* **2015**, doi:10.1073/pnas.1420068112.
14. Khan, S.; Luo, Y.; Ahmad, A. Analysing complex behaviour of hydrological systems through a system dynamics approach. *Environ. Model. Softw.* **2009**, *24*, 1363–1372.
15. Lipsitz, L.A. Understanding Health Care as a Complex System. *J. Am. Med. Assoc.* **2012**, *308*, 243–244.
16. Wang, M.; Ailamaki, A.; Faloutsos, C. Capturing the spatio-temporal behavior of real traffic data. In Proceedings of the Performance 2002, IFIP International Symposium on Computer Performance Modeling, Measurement and Evaluation, Rome, Italy, 23–27 September 2002; pp. 147–163.
17. Gómez, M.E.; Santonja, V. Self-Similarity in I/O Workload: Analysis and Modelling. In Proceedings of the Workload Characterization: Methodology and Case Studies, Based on the First Workshop on Workload Characterization, Dallas, TX, USA, 29 November 1998; pp. 97–104.
18. Riska, A.; Riedel, E. Long-Range Dependence at the Disk Drive Level. In Proceedings of the Third International Conference on the Quantitative Evaluation of Systems—(QEST'06), Riverside, CA, USA, 11–14 September 2006; pp. 41–50.
19. Riska, A.; Riedel, E. Evaluation of disk-level workloads at different time-scales. In Proceedings of the 2009 IEEE International Symposium on Workload Characterization (IISWC), Austin, TX, USA, 4–6 October 2009; pp. 158–167.
20. Kavalanekar, S.; Worthington, B.; Zhang, Q.; Sharda, V. Characterization of storage workload traces from production Windows Servers. In Proceedings of the 2008 IEEE International Symposium on Workload Characterization, Seattle, WA, USA, 14–16 September 2008; pp. 119–128.
21. Hong, B.; Madhyastha, T.M. The relevance of long-range dependence in disk traffic and implications for trace synthesis. In Proceedings of the 22nd IEEE/13th NASA Goddard Conference on Mass Storage Systems and Technologies (MSST'05), Monterey, CA, USA, 11–14 April 2005; pp. 316–326.
22. Tsallis, C. Possible generalization of Boltzmann–Gibbs statistics. *J. Stat. Phys.* **1988**, *52*, 479–487.
23. Di Matteo, T. *The Physics of Complex Systems (New Advances and Perspectives)*; Mallamace, F., Stanley, H.E., Eds.; IOS Press: Amsterdam, The Netherlands, 2004.
24. Mohazzabi, P.; Mansoori, G.A. Nonextensivity and Nonintensity in Nanosystems: A molecular dynamics simulation. *J. Comput. Theor. Nanosci.* **2005**, *2*, 138–147.
25. Tsallis, C.; Baldovin, F.; Cerbino, R.; Pierobon, P. Introduction to Nonextensive Statistical Mechanics and Thermodynamics. In *The Physics of Complex Systems (New Advances and Perspectives)*; Mallamace, F., Stanley, H.E., Eds.; IOS Press: Amsterdam, The Netherlands, 2004.
26. Tsallis, C. Nonadditive entropy and nonextensive statistical mechanics—An overview after 20 years. *Braz. J. Phys.* **2009**, *39*, 337–355.
27. Tsallis, C. Nonextensive statistical mechanics, anomalous diffusion and central limit theorems. *Milan J. Math.* **2005**, *73*, 145–176.
28. Tsallis, C.; Levy, S.V.F.; Souza, A.M.C.; Maynard, R. Statistical-mechanical foundation of the ubiquity of Lévy distributions in nature. *Phys. Rev. Lett.* **1995**, *75*, 3589.
29. Gallet, F.; Arcizet, D.; Bohec, P.; Richert, A. Power Spectrum of Out-of-Equilibrium Forces in Living Cells: Amplitude and Frequency Dependence. *Soft Matter* **2009**, *5*, 2947–2953.
30. Buchanan, M. Laws, power laws and statistics. *Nat. Phys.* **2008**, *4*, 339.
31. Makowiec, D.; Gałaska, R.; Dudkowska, A.; Rynkiewicz, A.; Zwierz, M. Long-range dependencies in heart rate signals-revisited. *Phys. A Stat. Mech. Appl.* **2006**, *369*, 632–644.
32. Mandelbrot, B.B. Multifractal Power Law Distributions: Negative and Critical Dimensions and Other “Anomalies” Explained by a Simple Example. *J. Stat. Phys.* **2003**, *110*, 739–774.
33. Tsallis, C. The Nonadditive Entropy S_q and Its Applications in Physics and Elsewhere: Some Remarks. *Entropy* **2011**, *13*, 1765–1804.
34. Borland, L. Option Pricing Formulas Based on a Non-Gaussian Stock Price Model. *Phys. Rev. Lett.* **2002**, *89*, 098701.
35. Feder, J. *Fractals*; Plenum Press: New York, NY, USA, 1988.
36. Cajueiro, D.O.; Tabak, B.M. Is the expression $H = 1/(3 - q)$ valid for real financial data? *Phys. A* **2007**, *373*, 593–602.
37. Hurst, H.E. Long-term storage capacity of reservoirs. *Trans. Am. Soc. Civ. Eng.* **1951**, *116*, 770.

38. Weron, A.; Burnecki, K.; Mercik, S.; Weron, K. Complete description of all self-similar models driven by Lévy stable noise. *Phys. Rev. E Stat. Nonlinear Soft* **2005**, *71*, 016113.
39. Grabowski, F. Logistic equation of arbitrary order. *Phys. A Stat. Mech. Appl.* **2010**, *389*, 3081–3093.
40. Taqqu, M.S.; Teverovsky, V. On Estimating the Intensity of Long-Range Dependence in Finite and Infinite Variance Time Series. In *A Practical Guide to Heavy Tails: Statistical Techniques and Applications Book Contents*; Birkhauser Boston Inc.: Cambridge, MA, USA, 1998; pp. 177–217.
41. Shahzad, F.; Mushtaq, M.F.; Ullah, S.; Siddique, M.A.; Khurram, S.; Saher, N. Improving Queuing System Throughput Using Distributed Mean Value Analysis to Control Network Congestion. *Commun. Netw.* **2015**, *7*, 21–29.
42. Lazowska, E.D.; Zahorjan, J.; Graham, G.S.; Sevcik, K.C. *Quantitative System Performance: Computer System Analysis Using Queueing Network Models*; Prentice-Hall Inc.: Upper Saddle River, NJ, USA, 1984.
43. Simchi-Levi, D.; Trick, M.A. Introduction to “Little’s Law as Viewed on Its 50th Anniversary”. *Oper. Res.* **2013**, *59*, 535.
44. Strzałka, D.; Grabowski, F. Processes in systems with limited resources in the context of non-extensive thermodynamics. *Fundam. Inform.* **2008**, *85*, 455–464.
45. Wilk, G.; Włodarczyk, Z. Nonextensive information entropy for stochastic networks. *Acta Phys. Pol. B* **2004**, *35*, 871.
46. Overview of Windows Performance Monitor. Available online: <https://technet.microsoft.com/en-us/library/cc749154.aspx> (accessed on 26 June 2016).
47. How to Measure IOPS for Windows. Available online: <http://blog.synology.com/?p=2086> (accessed on 26 June 2017).
48. How Often Should Perfmon Sample? Available online: <https://blogs.technet.microsoft.com/yongrhee/2011/11/13/how-often-should-perfmon-sample/> (accessed on 26 June 2017).
49. Windows Performance Monitor. Available online: <https://technet.microsoft.com/en-us/library/cc749249.aspx> (accessed on 26 June 2017).
50. Boxma, O.J.; Cohen, J.W. Heavy-traffic analysis for the GI/G/1 queue with heavy-tailed distributions. *Queueing Syst.* **1999**, *33*, 177–204.
51. Roughan, M.; Veitch, D.; Rumsewicz, M. Computing queue-length distributions for power-law queues. In Proceedings of the IEEE of Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies, San Francisco, CA, USA, 29 March–2 April 1998; pp. 356–363.
52. Troubleshooting Slow Disk I/O in SQL Server. Available online: <https://blogs.msdn.microsoft.com/askjay/2011/07/08/troubleshooting-slow-disk-io-in-sql-server/> (accessed on 26 June 2017).
53. Cady, F.; Zhuang, Y.; Harchol-Balter, M. A Stochastic Analysis of Hard Disk Drives. *Int. J. Stoch. Anal.* **2011**, *2011*, 390548.
54. Crovella, M.E.; Taqqu, M.S. A Tool for Estimating the Heavy Tail Index from Scaling Properties. *Methodol. Comput. Appl. Probab.* **1999**, *1*, 55.
55. Kim, J.H.T.; Kim, J. A parametric alternative to the Hill estimator for heavy-tailed distributions. *J. Bank. Financ.* **2015**, *54*, 60–71.
56. Nguyen, T.; Samorodnitsky, G. Tail inference: Where does the tail begin? *Extremes* **2012**, *15*, 437.
57. Strzałka, D. Non-Extensive Statistical Mechanics—A Possible Basis for Modelling Processes in Computer Memory System. *Acta Phys. Pol. Ser. A Gen. Phys.* **2010**, *117*, 652.
58. Janicki, A.; Weron, A. *Simulation and Chaotic Behavior of Alpha-Stable Stochastic Processes*; Marcel Dekker: New York, NY, USA, 2000.
59. Cao, J.; Cleveland, W.S.; Lin, D.; Sun, D.X. Internet Traffic Tends Toward Poisson and Independent as the Load Increases. In *Lecture Notes in Statistics, Nonlinear Estimation and Classification*; Springer: New York, NY, USA, 2003; Volume 171, pp. 83–109.
60. Rak, R.; Zięba, P. Multifractal Flexibly Detrended Fluctuation Analysis. *Acta Phys. Pol. B* **2015**, *46*, 1925.

