UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação
ISSN 0103-2569

A Novel Metric for Evaluation of Computer System Heterogeneity

Kalinka Regina Lucas Jaquie Castelo Branco
Marcos José Santana
Regina Helena Carlucci Santana

Nº 177

RELATÓRIOS TÉCNICOS

São Carlos – SP
Nov./2002

A Novel Metric for Evaluation of Computer System Heterogeneity

Kalinka Regina Lucas Jaquie Castelo Branco[1],

Marcos José Santana[1],

Regina Helena Carlucci Santana[1]

[1]Departamento de Ciências Matemáticas e de Computação
ICMC/USP - São Carlos
Caixa Postal 668
13560-970 São Carlos

Abstract

This paper addresses the problem of constructing suitable metrics for the evaluation of computer system heterogeneity. There are several metrics available in the open literature that evaluates the heterogeneity degree of a given system. However, this paper shows that many of these available metrics are not entirely suitable when used to support both load and process scheduling mechanisms. It is shown in this paper that some of the available metrics have a dual behaviour for the same system, depending on the adopted reference. A novel metric, constructed from a new approach using the standard deviation concept, is proposed in this paper. This metric is shown to be adequate for all the case studies adopted and it has potential to support most of the load and process scheduling mechanisms used in parallel/distributed computing. The paper also presents results from many case studies considering both the existing metrics and the new one.

Resumo

Este artigo avalia o problema da construção de métricas satisfatórias para sistemas computacionais heterogêneos. Existem, na literatura aberta, diversas métricas que avaliam o grau de heterogeneidade de um determinado sistema. Porém, este artigo mostra que muitas destas métricas não são completamente satisfatórias quando utilizadas para apoiar mecanismos de escalonamento de processos. É mostrado neste artigo que algumas das métricas disponíveis têm um comportamento duplo para um mesmo sistema, dependendo da referência adotada. Uma nova métrica, construída com base no conceito de dispersão e fazendo uso do desvio padrão, é proposta neste artigo. Esta métrica se mostra adequada para todos os estudos de caso adotados e tem potencial para apoiar a maioria dos mecanismos de escalonamento de processos usados em computação de paralela/distribuída. O artigo apresenta também resultados de muitos estudos de caso que consideram tanto as métricas existentes como a nova métrica.

1. Introduction

A heterogeneous computer system is one that uses processing elements that admit different types of processors, different processing speeds, varied computational modes, nonuniform memory sizes, distinct numbers of processors (even in machines that are essentially parallel), and/or different connectivity paradigms.

When one considers that an application can be executed in a heterogeneous system, one believes it can be executed in a better way provided one knows which of the system machines possesses the best conditions to execute the application. Thus, the use of heterogeneous computing systems provides good opportunities for improving the performance of computer applications by the suitable allocation of computer tasks to the existing processors. Although heterogeneity provides great flexibility, there are new aspects that must be dealt with, such as portability and process scheduling. This flexibility allows for the correct allocation of tasks to be performed by the machines that are better equipped to execute them, which cannot be done in a homogeneous computing system.

Process scheduling must be done carefully in order to achieve optimal utilization of the available computing system. Wrong scheduling decisions can lead to poor overall system performance. Thus, it is important to provide adequate metrics to evaluate the heterogeneity of a system, allowing for a better understanding of the differences in performance of the processors available in a distributed computing system. Heterogeneity metrics can be used in a variety of ways: computer schedulers can use this information to more accurately assess performance indices [1] leading to better scheduling decisions. Also, performance metrics widely used in homogeneous computing systems, such as speed-up, efficiency and response times [2] [3] [4], can be better characterized and tuned with the knowledge of the system level of heterogeneity.

This paper analyzes some of the existing heterogeneity metrics [1] [6] and related models, aiming to evaluate the heterogeneity level [5] [1] [6] [7] of distributed computer systems through both experimental and theoretical evaluations, and gives an overview of the subject as well as several case studies. Two important aspects are discussed: the weaknesses of the existing metrics and a new way to approach and treat the problem of heterogeneity.

2. Related Work

Some models and metrics for heterogeneous systems were proposed by Zhang and Yang [1], in which heterogeneous computing systems can be represented by a graph (M,C), where $M=\{M_1,M_2,M_3,M_4,M_5....,M_n\}$ is considered a set of heterogeneous workstations (each one having its own computing power that can be measured by CPU, disk and memory capacity) and *C* is the communication network linking the workstations (with a homogeneous bandwidth for all the interconnected points).

Aiming to quantify the heterogeneity of a system machines without using complex measurements, Zhang and Yang proposed two metrics to evaluate the relative computing power of a set of workstations (the capacity of each workstation is evaluated in comparison to the fastest one):

$$W_i\ (A) = \frac{S_i\ (A)}{\max_{i=1}^{n}\{S_i\ (A)\}} \qquad (1)$$

where i=1,....,n and $S_i(A)$ represents the speed of M_i to execute application *A* dedicatedly. Speed can be defined by the number of basic operations per time unit, for instance, and the computing power of each workstation is represented by a relative speed.

A second metrics proposed is:

$$W_i(A) = \frac{\min_{i=1}^{n}\{T(A,M_i)\}}{T(A,M_i)} \qquad (2)$$

where i=1,....,n and $T(A,M_i)$ is the time required to execute application *A* at workstation M_i. Grosu [5] extends these metrics so that the computing power is given by the relative speed of the workstation in relation to the slowest one:

$$W_i(A) = \frac{\min_{i=1}^{n}\{S_i\ (A)\}}{S_i\ (A)} \qquad (3)$$

where i=1,....,n and $S_i(A)$ is the speed of workstation Mi to execute application *A* dedicatedly, and the computing power is given by relative speeds. Furthermore, Grosu defines

$$W_i\ (A) = \frac{T(A,M_i)}{\max_{i=1}^{n}\{T(A.M_i)\}} \qquad (4)$$

where i=1,....,n and $T(A,M_i)$ is the time it takes to execute application *A* at workstation M_i.

Thus, equation 1 and 2 now serve as the basis to define the computing power, considering the fastest machine as a reference point, which is renamed W_i^f (f - fast). On the other hand, equations 3 and 4 identify the computing power based on the slowest machine, which is represented by W_i^s (s - slow). Four ways to quantify the heterogeneity level in a system based on the value of W are proposed in [1] and [5]. The first and second use the standard deviation, H_1, (which can be calculated based on the computing powers compared to either the fastest or the slowest workstation):

$$H_1 = \sqrt{\frac{\sum_{i=1}^{n}(W_{med} - W_i)^2}{n}} \qquad (5)$$

or the mean absolute deviation, called H_2, (also calculated based on the fastest or the slowest workstation):

$$H_2 = \frac{\sum_{i=1}^{n}\left|W_{med} - W_i\right|}{n} \qquad (6)$$

where $W_{med} = \frac{\sum_{i=1}^{n} W_i}{n}$.

The values in both H_1 and H_2 are observed and analyzed uniformly, using the average to find the standard deviation and the mean absolute deviation. However, this uniformity invalidates the analysis when there are reasonable differences among the workstations computing powers, since the standard deviation cannot reflect computer systems.

Based on this weakness of the H_1 and H_2 metrics, Zhang and Yang [1] proposed a third metric, H_3, evaluated from the computing power of the fastest workstation in the computing system:

$$H_3 = \frac{\sum_{i=1}^{n}(1 - W_i^f(A))}{n} \tag{7}$$

Similarly, Grosu [5] defines H_4 based on the computing power of the slowest workstation in the computing system:

$$H_4 = \frac{\sum_{i=1}^{n}(1 - W_i^s(A))}{n} \tag{8}$$

In H_3, the computing power of the fastest workstation is equal to 1 while, in H_4, the slowest machine has a computing power value of 1. Thus, H_4 represents the difference (in terms of computing power) between each machine and the fastest machine and H_3 calculates the same difference between each machine and the slowest one.

Based on his experiments, Grosu [5] states that the metric H_4 is more suitable than H_3. However, several case studies presented herein demonstrate the fallacy of that statement in some situations, as shown in tables 1, 2 and 3 and in the corresponding figures 1, 2 and 3.

Table 1- The machines computing power is initially close to 1, but gradually decreases

*	$W_{(M1)}$	$W_{(M2)}$	$W_{(M3)}$	$W_{(M4)}$	$W_{(M5)}$	$W_{(M6)}$	$W_{(M7)}$	$W_{(M8)}$	$W_{(M9)}$	$W_{(M10)}$	H_1	H_2	H_3	H_4
1	1	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.03	0.018	0.09	0.01
2	1	0.75	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.06	0.029	0.105	0.161
3	1	0.75	0.65	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.1	0.068	0.13	0.24
4	1	0.75	0.65	0.55	0.9	0.9	0.9	0.9	0.9	0.9	0.14	0.111	0.165	0.321
5	1	0.75	0.65	0.55	0.4	0.9	0.9	0.9	0.9	0.9	0.19	0.158	0.215	0.452
6	1	0.75	0.65	0.55	0.4	0.35	0.9	0.9	0.9	0.9	0.23	0.194	0.27	0.46
7	1	0.75	0.65	0.55	0.4	0.35	0.25	0.9	0.9	0.9	0.27	0.225	0.34	0.54
8	1	0.75	0.65	0.55	0.4	0.35	0.25	0.15	0.9	0.9	0.3	0.25	0.41	0.64
9	1	0.75	0.65	0.55	0.4	0.35	0.25	0.15	0.1	0.9	0.31	0.26	0.49	0.67

* Configuration

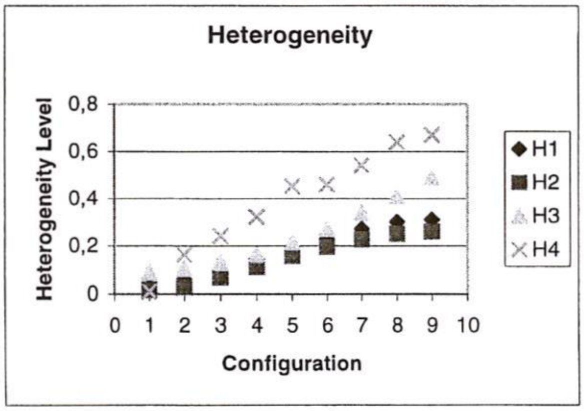

Figure 1 – Heterogeneity levels when all the system computing powers are close to 1

Figure 1 shows that all the metrics are satisfactory when most of the computing powers tend to 1, i.e., the level of heterogeneity increases as the computing power shows greater variations.

Table 2 - All the workstations computing powers are initially far below that of the fastest machine. Although this difference gradually decreases, their powers still remain below that of the fastest machine.

*	$W_{(M1)}$	$W_{(M2)}$	$W_{(M3)}$	$W_{(M4)}$	$W_{(M5)}$	$W_{(M6)}$	$W_{(M7)}$	$W_{(M8)}$	$W_{(M9)}$	$W_{(M10)}$	H_1	H_2	H_3	H_4
1	1	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.15	0.09	0.45	0.05
2	1	0.75	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.16	0.12	0.425	0.084
3	1	0.75	0.65	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.17	0.13	0.41	0.107
4	1	0.75	0.65	0.55	0.5	0.5	0.5	0.5	0.5	0.5	0.17	0.12	0.405	0.116
5	1	0.75	0.65	0.55	0.4	0.5	0.5	0.5	0.5	0.5	0.17	0.13	0.415	0.271
6	1	0.75	0.65	0.55	0.4	0.35	0.5	0.5	0.5	0.5	0.19	0.14	0.43	0.333
7	1	0.75	0.65	0.55	0.4	0.35	0.25	0.5	0.5	0.5	0.21	0.15	0.46	0.474
8	1	0.75	0.65	0.55	0.4	0.35	0.25	0.15	0.5	0.5	0.25	0.18	0.49	0.614
9	1	0.75	0.65	0.55	0.4	0.35	0.25	0.15	0.1	0.5	0.28	0.22	0.53	0.664

* Configuration

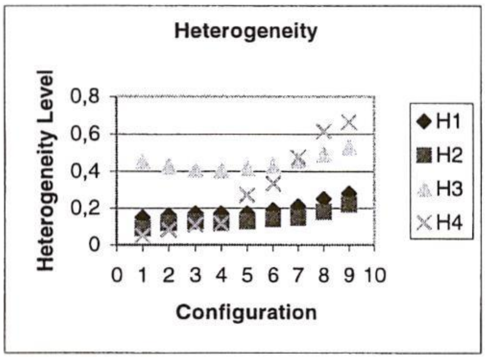

Figure 2 – Heterogeneity levels when all the computing powers are below 1

In this case, equation H_3 does not adequately represent the system heterogeneity; equation H_4, however, produces a good result. From these results, it can be incorrectly inferred that H_4 is the equation that best characterizes the heterogeneity of a computing system. This inference is incorrect because it is a particular case whose result cannot be generalized.

Table 3 – The configuration of the machines leads to a situation in which half the machines have less than average computer powers while the other half have above average powers. Two distinct cases are also presented in this table: in case 1, one machine has a high load and all the others have low loads, and in case 2, one machine has a low load and the others have high loads.

*	$W_{(M1)}$	$W_{(M2)}$	$W_{(M3)}$	$W_{(M4)}$	$W_{(M5)}$	$W_{(M6)}$	$W_{(M7)}$	$W_{(M8)}$	$W_{(M9)}$	$W_{(M10)}$	H_1	H_2	H_3	H_4
1	1	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.15	0.09	0.45	0.05
2	1	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.16	0.12	0.425	0.084
3	1	1	1	1	1	0.5	0.5	0.5	0.5	0.5	0.17	0.13	0.41	0.107
4	1	1	1	1	1	0.3	0.3	0.3	0.3	0.3	0.17	0.12	0.405	0.116

* Configuration

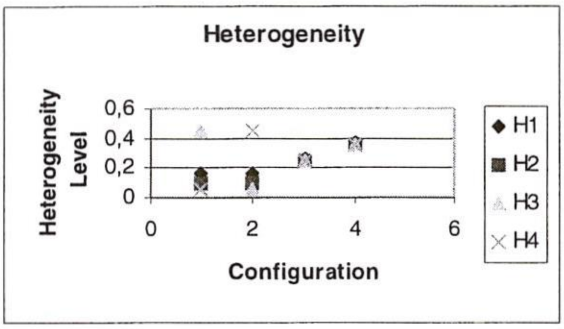

Figure 3 - Heterogeneity levels in which five situations are imposed

Figure 3 shows two cases ([a] 9 machines with computing powers equal to 1 and 1 machine with computing power equal to 0,5; and [b] 9 machines with computing powers equal to 0,5 and 1 machine with computing power equal to 1) in which the heterogeneity levels (in H_3 and H_4) should be similar but, instead, they are contradictory. For this reason, this paper discusses new studies that took into account the two evaluation equations, since the results obtained did not reflect reality.

3. Modeling the Heterogeneity

Since the aforementioned studies proved inadequate, we decided to use another metrics. The metrics used in this study is a simple one, serving to check the dispersion of machines around a standard, as discussed in (Zhang & Yang, 1995) and (Grosu, 1999), though seeking a more suitable reference that replaces the fastest and the slowest machines.

3.1. Model to obtain the (desired) degree of heterogeneity

Taking as standard the average virtual machine, we found that the use of the absolute standard deviation (which corresponds to the mean difference between the computational powers of the machines and the standard machine) in relation to the standard machine led to a suitable degree of heterogeneity for the purposes of this study.

Hence, the degree of heterogeneity/homogeneity of a given system can be found based on eq. (9) where *HL* is the Heterogeneity Level:

$$HL = \frac{\sum_{i=1}^{n} \left| X_i - \overline{X} \right|}{n.\overline{X}} \qquad (9)$$

The use of the average speed as a standard leads to the adoption of a virtual standard machine, adjusted automatically and dynamically for each set, thus accounting for the variations occurring when additional machines are inserted into the system (whether these machines are high or low speed). That is precisely what does not occur with the metrics proposed previously, since they all consider the lowest or highest speed machine, restricting the flexibility of the standard and, hence, the correct adjustment of the degree of heterogeneity. The case studies discussed below aim to elucidate the metrics proposed herein.

3.2. Behavior of the Model

Several tests were conducted to represent the behavior of the proposed model. The first test, which began with 3 machines with different computing speeds, i.e., 10, 100 and 1000, consisted of adding (one at a time) machines identical to the highest speed machines. In the initial configuration, the heterogeneity level is high, but as more machines identical to the fastest one are added, this degree of heterogeneity tends toward zero, as shown in Figure 4.

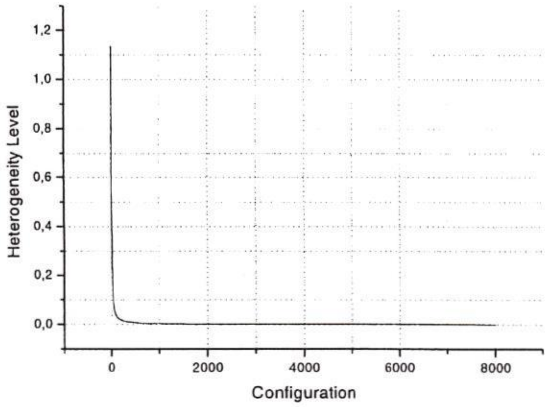

Figure 4 – Behavior of the degree of heterogeneity when machines identical to the system fastest machine are added (initial speeds of 10, 100 and 1000)

The behavior of the degree of heterogeneity is coherent since, as similar high-speed machines are added, the system heterogeneity level decreases. This type of system is dealt with coherently by the metrics proposed by Zhang (Zhang & Yang); however, it is not dealt with by Grosu's metrics (Grosu, 1996), precisely because of the influence in the selection of the standard.

In the second study, the initial configuration again consists of 3 machines with speeds of 10, 100 and 1000, but this time machines are added, one by one, which are identical to the system slowest one. The initial degree of heterogeneity is identical to that of the first case study, but the behavior is entirely different as each machine is added, illustrating exactly the impact of the system heterogeneity, taking into account the substantial differences in the machines speeds, Figure 5.

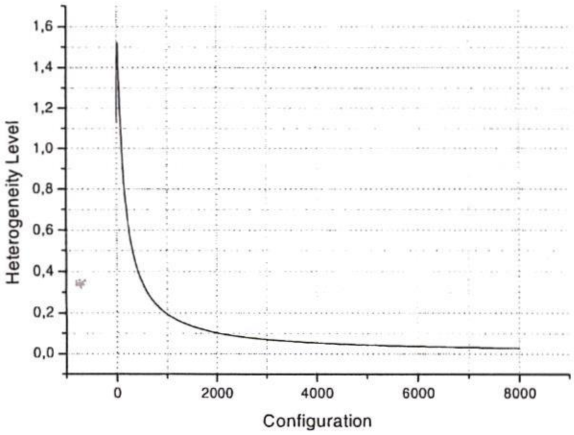

Figure 5 – Behavior of the degree of heterogeneity when machines identical to the system slowest machine are added (initial speeds of 10, 100 and 1000)

When machines that are identical to the system slowest one are inserted, the behavior of the degree of heterogeneity, according to the proposed metrics, shows that up to a certain number of additional machines, the impact of removing them from the system in order to render the system homogeneous is slight. It also shows that, starting from a certain number of machines, even if those machines are identical to the system slowest one, the impact of removing them from the system is great (since the computational power of these slower machines, collectively, is greater than that of the faster ones) and may damage the system performance.

Analogously to the former case, this type of system is dealt with by Grosu's metrics (Grosu, 1996), but not by Zhang's metrics (Zhang & Yang).

Repeating the two aforementioned studies, and altering only the initial configuration of the machines speeds to 10, 1000 and 100000, the insertion of machines identical to the system fastest one, and then the slowest one, leads to behaviors similar to those depicted in Figures 4 and 5, the only changes being the peak values and the peak of the number of machines required to alter the possibility of disposal, as shown in Figures 6 and 7.

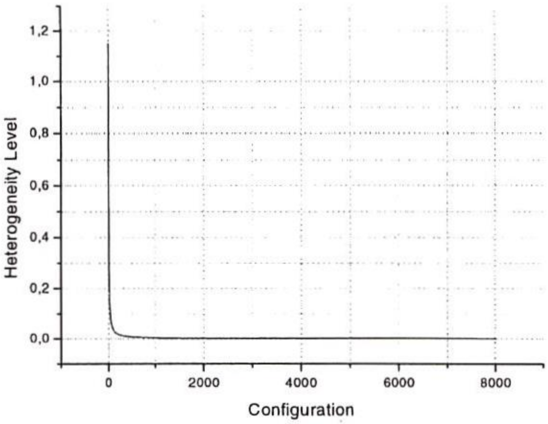

Figure 6 – Behavior of the degree of heterogeneity when machines identical to the system fastest machine are added (initial speeds of 10, 10000 and 100000)

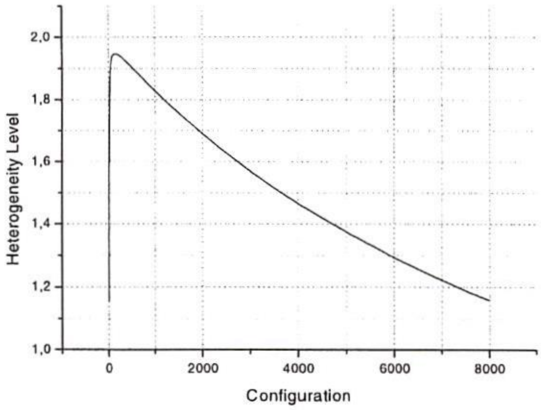

Figure 7 – Behavior of the degree of heterogeneity when machines identical to the system slowest machine are added (initial speeds of 10, 10000 and 100000)

Another case study sought to identify the behavior of the degree of heterogeneity achieved by the proposed metrics when the system is started with a given processing capacity (in terms of total processing speed), keeping this processing capacity and simply altering the machines configuration. In other words, the sum of computational powers, taken in terms of speed, is kept constant throughout the experiment.

The first configuration consists of no machine with a speed of 10 and 10 machines with speeds of 100, making a total speed of 1000. Successive changes are made in the configuration so as to have 1 machine with a speed of 10 and 99 machines with speeds of 100, until the configuration is 100 machines with speeds of 10 and no machine with a speed of 100.

The behavior of the metrics is illustrated in Figure 8.

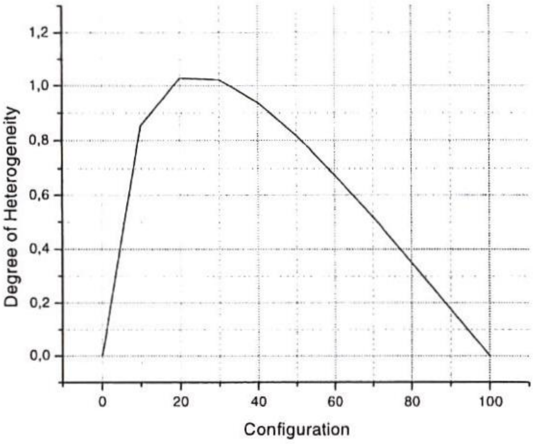

Figure 8 – Behavior of the degree of heterogeneity when machines with high and low speeds are inserted into the system, maintaining the system same total speed and altering only the number of high and low speed machines (initial configuration of 0 machines with speeds of 10 and 10 machines with speeds of 100)

The graph shows precisely that the degree of heterogeneity varies according to the presence of machines with identical speeds or with different speeds, reflecting how far these machines are from a standard and what the influence is if the system is composed of a greater number of slow machines or fast machines, in terms of the concept of heterogeneity.

In contrast, the metrics proposed earlier and presented in the literature (Zhang & Yang) (Grosu, 1996) present values that are conflicting and even complementary for configurations that should show the same degree of heterogeneity.

Figure 9 depicts the same experiment conducted earlier, but with 0 machines having the same speed of 10 and 100 machines with speeds of 100, and so on, until the system contains 1000 machines with speeds of 10 and 0 machines with speeds of 100.

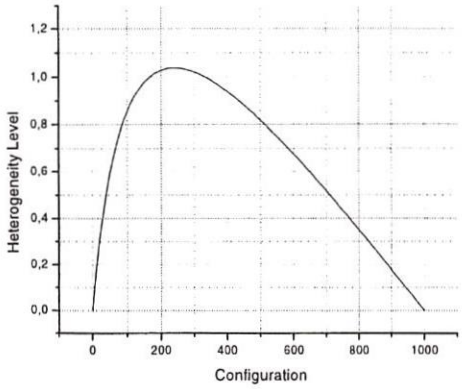

Figure 9 – Behavior of the degree of heterogeneity when machines with high and low speeds are inserted into the system, maintaining the system same total speed and altering only the number of high and low speed machines (initial configuration of 0 machines with speeds of 10 and 100 machines with speeds of 100)

Two additional studies were conducted, using as initial configurations, in the first case, two machines, one with a speed of 1 and the other with a speed of 1000 and, in the second case, one machine with a speed of 1 and the other with a speed of 10000. Starting from this initial configuration, the speed of the fastest machine was divided by the number of machines in the system and this value was inserted into a new machine. In each successive step, another machine was added until a heterogeneous system was reached which, in the first case, contained 1000 machines with a difference of one unit of speed from each other while, in the second case, the system contained 1000 machines with differences among each other of 10 units of speed. These experiments are illustrated in Figures 10 and 11.

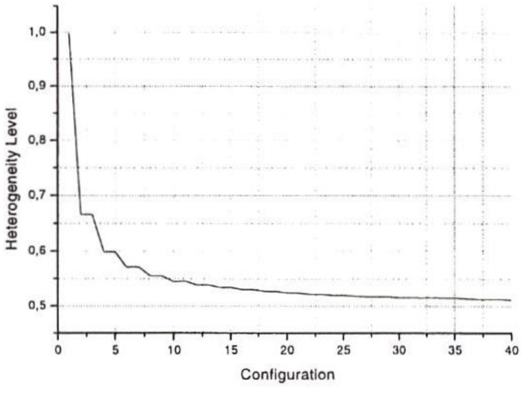

Figure 10 – Behavior of the degree of heterogeneity when machines having intermediary speeds are inserted into the system, until these additions reach values of one unit of speed from one machine to the next (initial configuration of one machine with a speed of 1 and one machine having a speed of 1000)

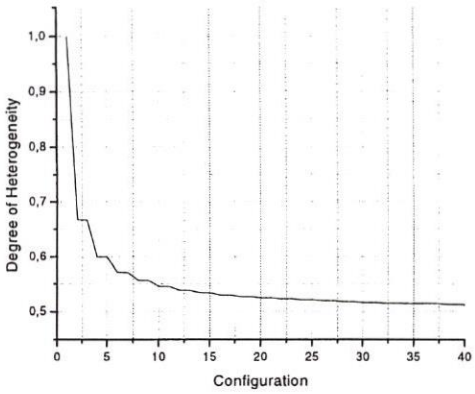

Figure 11 – Behavior of the degree of heterogeneity when machines having intermediary speeds are inserted into the system, until these additions reach values of 10 units of speed from one machine to the next (initial configuration of one machine with a speed of 1 and one machine having speeds of 10000)

This experiment revealed that the average of machines in any of the situations remains practically the same, even with different configurations and numbers of machines. This demonstrates that, as the average distance between machines diminishes, the degree of heterogeneity also diminishes, until a degree of heterogeneity of 0,5 is reached (a relatively high heterogeneity).

This behavior was not observed in the other metrics (Zhang & Yang, 1995) (Grosu, 1996), since the machine considered standard is not flexible but fixed as the fastest machine in the first case and the slowest one in the second.

4. Case Studies

There are systems characterized by the variety of CPU architectures, different execution speeds, different operational systems, distinct input and output resources, different storage capacities, and different means of interconnection.

This heterogeneity of resources produces a considerable variation in computing power, leading to new difficulties that require management. Models of performance evaluation and traditional performance indices in homogeneous systems have a reduced viability in heterogeneous systems. Moreover, the process schedulers must incorporate effective mechanisms for the management of resources that are potentially dissimilar.

The main purpose of our experiments is to define when a system can be considered homogeneous, partially homogeneous, or heterogeneous. To this end, the system degree of heterogeneity was compared to the relations of response time when the same heterogeneous system is treated homogeneously and heterogeneously according to the way the processes are schedule.

A heterogeneous system was considered and treated as a homogeneous one (simply by distributing the loads in a round-robin mode). The same system was then treated appropriately as heterogeneous (attributing weights in the distribution according to each machine speed), as illustrated in Figures 12 and 13.

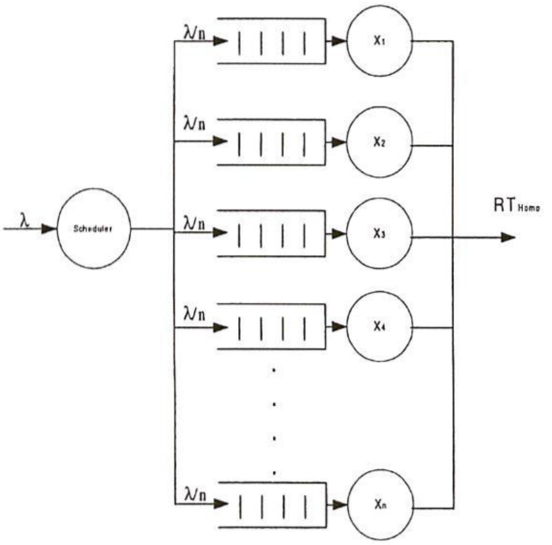

Figure 12 – Queue model for a system composed of heterogeneous machines treated by the scheduler as if it were homogeneous

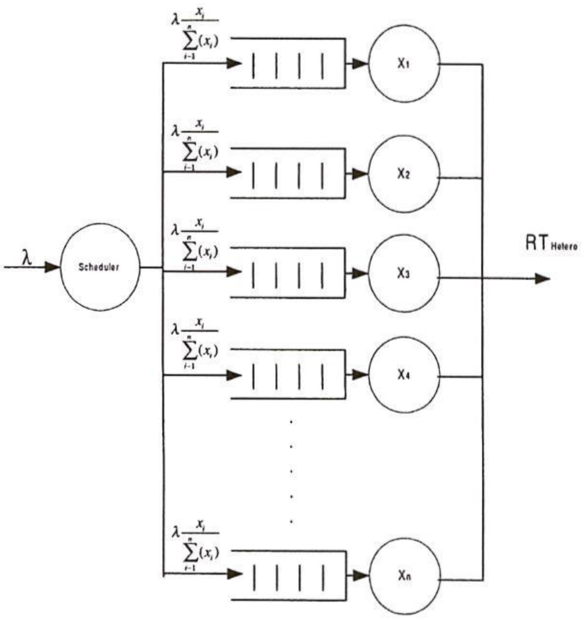
λ
Scheduler
$\lambda \frac{x_i}{\sum_{i=1}^{n}(x_i)}$
$\lambda \frac{x_i}{\sum_{i=1}^{n}(x_i)}$
$\lambda \frac{x_i}{\sum_{i=1}^{n}(x_i)}$
$\lambda \frac{x_i}{\sum_{i=1}^{n}(x_i)}$
$\lambda \frac{x_i}{\sum_{i=1}^{n}(x_i)}$
X1
X2
X3
X4
Xn
RT Hetero

Figure 13 – Queue model for a system composed of heterogeneous machines treated by the scheduler as if it were heterogeneous

In both models, X_1, X_2,... Xn correspond to the machines speeds, n represents the number of machines that make up the system, λ is the number of applications to be schedule, RT_{Homo} and RT_{Hetero} are the response times of the system treated, respectively, as homogeneous and heterogeneous.

4.1 Results

Assuming the applications to be distributed were all homogeneous and interactive, the response time was used to evaluate the system performance. The response time was used because it is a measure of quantity fixed according to the system configuration. The degree of heterogeneity was thus achieved in all the case studies, according to the metrics defined in the previous section, and the system response times were treated first homogeneously and then heterogeneously for each configuration.

In the first study, we sought to identify the behavior of the degree of heterogeneity in relation to the heterogeneous and homogeneous response time when the system is started with ten machines with identical speeds of 100 and subsequently the machines speeds are gradually altered, though keeping the number of machines constant. Thus, the configurations begin with 10 machines having the same processing speed (100), and are subsequently modified to 9 machines having speeds of 100 and 1 machine with a speed of 200, 8 machines having speeds of 100, 1 machine with a speed of 200 and 1 machine with a speed of 300 and so on, successively, until the system ends with 10 machines with different speeds ranging from 100 to 1000, to obtain a heterogeneous system. The above-described behavior is illustrated in Figure 14.

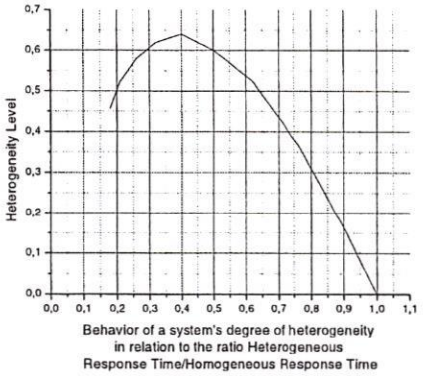

Figure 14 – Representation of the behavior of a system degree of heterogeneity in relation to the ratio Heterogeneous Response Time/Homogeneous Response Time, with the configurations alternating from 10 machines with identical speeds to 10 machines with different speeds, replacing the machines so as to add the values of the speeds.

As can be seen, when the system is treated as homogeneous (but is heterogeneous), the system total response time is always given by the response time of the slowest machine. In contrast, when the system is treated as heterogeneous, the response time improves significantly.

The metrics proposed here allows for the evaluation of systems that should display the same behavior, since their standard is the average virtual machine rather than the fastest or slowest one.

Figure 14 shows that the degree of heterogeneity begins with a value of 0 and, as this value increases, the ratio between the heterogeneous and homogeneous response times decreases. This represents the possible loss of performance caused by a system that, being heterogeneous, is treated as homogeneous.

In the metrics previously presented in the literature, the behavior of the graph for the same set of machines shown in Figure 14 is contradictory. In Zhang's metrics (Zhang & Yang), the degree of heterogeneity, like that depicted in Figure 14, increases as the ratio between the heterogeneous and homogeneous response times decreases. However, based on Grosu's metrics (Grosu, 1996), as the degree of heterogeneity decreases, the ratio between the heterogeneous and homogeneous response times increases.

Another interesting factor of the degree of heterogeneity proposed here is that it allows one not only to demonstrate the impact of the system heterogeneity but also to identify whether this heterogeneity is positive or negative for the system. A heterogeneity is considered in this paper as positive when there are several machines with identical speeds (or computing powers) and one machine with a higher speed than the others. In this case, this machine improves the system instead of spoiling it. This can be observed in the graph, when a machine with a speed of 200 is added to a set of machines with speeds of 100; the degree of heterogeneity increases, but the loss resulting from treating the system as being homogeneous does not substantially decrease the system performance.

On the other hand, when the degree of heterogeneity increases, to consider the system as homogeneous, one or more machines must be removed from the set. It is normally the slower machines that should be removed even though there are a large number of them once the depending on the difference between them and the fastest machine, their removal may result in a considerable performance improvement.

Analogously to Figure 14, Figure 15 depicts the behavior of the degree of heterogeneity, also in relation to the response times in heterogeneous and homogeneous environments, in a configuration of machines that begins with machines having different speeds (starting from 100 and ending with 1000) and reaches a configuration in which all the machines are identical to the system fastest one.

The machines are substituted, starting from the slowest one and progressing to the fastest one, with the purpose of demonstrating the behavior of the degree of heterogeneity when slow machines are added to and removed from the system.

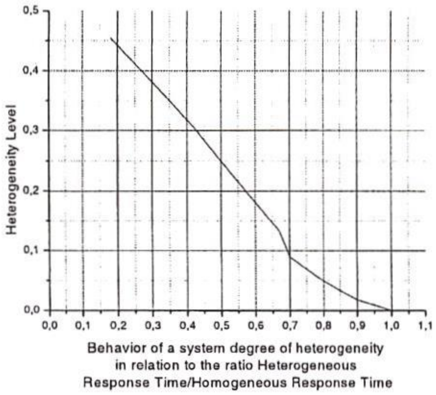

Figure 15 – Representation of the behavior of a system degree of heterogeneity in relation to the ratio Heterogeneous Response Time/Homogeneous Response Time, with the configurations alternating from 10 machines with identical speeds to 10 machines with different speeds, replacing the machines with the slowest speeds.

As can be seen, as the number of faster machines increases, the degree of heterogeneity decreases, demonstrating that the impact of removing slower machines leads not to a loss of performance but to an increase thereof.

The results portrayed in Figures 16 and 17 once again confirm the values observed earlier, further corroborating the conclusions reached herein.

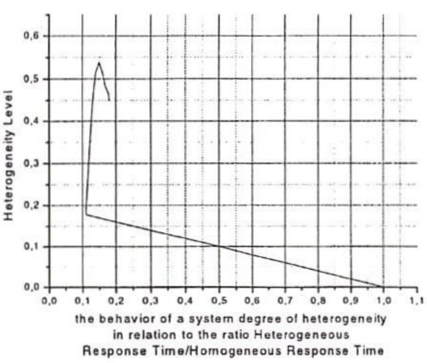

Figure 16 – Representation of the behavior of a system degree of heterogeneity in relation to the ratio Heterogeneous Response Time/Homogeneous Response Time, with configurations alternating from 10 machines with identical speeds to 10 machines with different speeds, replacing them, one by one, for very slow machines, beginning with the slowest one.

In the example illustrated in Figure 16, the initial configuration consists of 10 machines with the same speed of 1000, which is subsequently altered to 9 machines with speeds of 1000 and 1 machine with a speed of 100. The configuration continues to be modified successively until it ends with a set of machines with different speeds ranging from 10 to 1000.

In Figure 17, the initial configuration has 10 machines with the same speed of 100, which are gradually replaced by faster machines, starting from the fastest and proceeding to the slowest, until a configuration of machines with different speeds varying from 10 to 1000 is achieved.

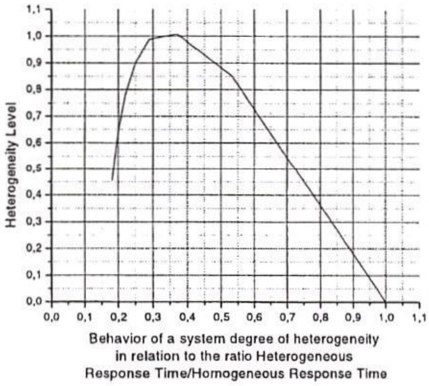

Figure 17 – Representation of the behavior of a system degree of heterogeneity in relation to the ratio Heterogeneous Response Time/Homogeneous Response Time, with configurations alternating from 10 machines with identical speeds to 10 machines with different speeds, and gradually replacing the slower speed machines.

It is therefore evident that, like in the previous examples, as the number of faster machines increases, the degree of heterogeneity decreases, demonstrating that the effect of removing the slower machines leads to a gain, rather than a loss, in the system performance.

In another case study, the system was started with a given processing capacity (in terms of total processing speed) and this processing capacity was maintained, although changes were made not only in the machines configuration but also in the number of machines in the system. The first configuration consisted of zero machines with speeds of 10 and 10 machines with speeds of 100, making a total speed of 10000. Successive changes were made in the configuration, ending with 100 machines with speeds of 10 and zero machines with speeds of 100. The behavior of this ratio is illustrated in Figure 18.

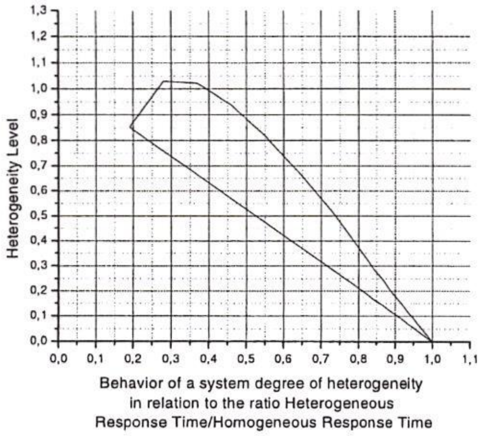

Figure 18 – Representation of the behavior of a system degree of heterogeneity in relation to the ratio Heterogeneous Response Time/Homogeneous Response Time.

This study demonstrates precisely the possibility of identifying the impact of the heterogeneity (positive or negative) based on the degree of that heterogeneity.

When one has ten machines with speeds of 90 and one machine with a speed of 100, the degree of heterogeneity is low because the impact caused by the latter machine is positive, since it helps increase the performance when the system is treated heterogeneously. Similarly, when the system contains ten machines with speeds of 10 and nine machines with speeds of 100, although there are more slow machines, it is better, from the standpoint of performance, to remove these slow machines from the system, since they represent a negative homogeneity.

Figure 19 shows some of the results of yet another case study involving an application that implements the Traveling Salesman problem. These results, obtained from a real application, confirm the findings of all the theoretical case studies presented herein.

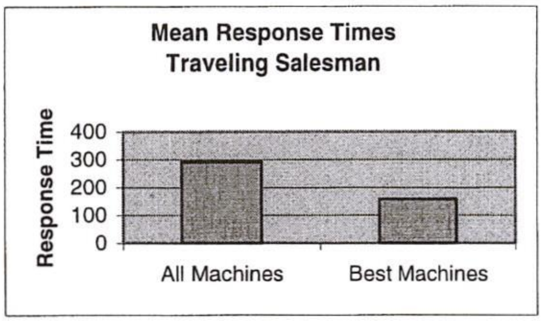

Figure 19 – Comparison of the use of the AMIGO environment with the minimum resource approach, using all the system machines.

In the new example above, a process scheduling environment called AMIGO (Souza, 2000) is used. The approach adopted in the process scheduling is that of minimum resources, which involves a minimum of resources stipulated by the user to allow a machine to be inserted into the scheduling process. The same application was scheduler for the AMIGO environment, making use of all the machines in the system and, subsequently, using the best machines. The gain in performance, in terms of response time, was far greater when the minimum resource approach was used, demonstrating that the removal of slower machines from the system can, potentially, improve the system performance, depending on the positivity or negativity of the degree of heterogeneity.

5. Conclusion

This paper examines the effects caused by the appropriate choice of a metric for obtaining the heterogeneity level of a distributed computing system. It is emphasized that, given the broad interest in distributed systems use and that the evolution of these systems has taken to a large degree of heterogeneity, further investigations aiming at building appropriate metrics to measure this degree of heterogeneity are need.

The paper approaches some metrics proposed in the literature and points out the weaknesses of those metric, when compared to a novel metric proposed in this work. The new metric has as objective to measure the dispersion of the machines around a virtual pattern.

Results from empirical experiments were presented as case studies, wich investigated the real usefulness of the heterogeneity level. Despite the case studies used for validation of the metric adopted as parameters the speed of the machines, this metric can assume differents parameters such as the size of memory, the processing capacity, etc showing that this metric is more flexible and generic than those available in the literature.

The metric proposed in this paper removes the restrictions observed in the metrics proposed by Zhang & Yan [1] and Grosu [5], allowing a coherent use of the heterogeneity level of a given computer system for scheduling proposes.

The paper also showed the importance of the knowledge of the heterogeneity level of a system, also through the case studies that related the heterogeneity levels and the response times when a heterogeneous system is treated as being homogeneous.

6. References

[1] X. Zhang, Y. Yan. Modeling and Characterizing Parallel Computing Performance on Heterogeneous Networks of Workstations. *Proceedings of the Seventh IEEE Symposium on Parallel and Distributed Proceeding*, October, 1995, 25-34.

[2] D. Ferrari, S. Zhou. An Empirical Investigation of Load Indices for Load Balancing Applications. In: *Proceedings of Performance'87, the 12th Int'l Symposium on Computer Performance Modeling, Measurement, and Evaluation*, 1987, 515-528.

[3] S. Zhou, X. Zheng, J. Wang, P. Delisle. Utopia: a Load Sharing Facility for Large, Heterogeneous Distributed Computer Systems. *Software: Practice and Experience*, *23*(12), December 1993, 1305-1336.

[4] P. Mehra, B. W. Wah. Automated Learning of Load-Balancing Strategies for a Distributed Computer System. *(These)* University of Illinois at Urbana-Champaign, 1993.

[5] D. Grosu. Some Performance Metrics for Heterogeneous Distributed Systems. *Proceedings of PDPTA´99*. Las Vegas, June, 1999.

[6] M. Kafil, I. Ahmad. Optimal Task Assignment in Heterogeneous Computing Systems. *6th Heterogeneous Computing Workshop (HCW´97)*. April, 1997, 135-146.

[7] R.F. Freund. Optimal selection theory for superconcurrency. *Proceedings of Supercomputing´89*, November 1989, 699-703.

[8] P. S. L. SOUZA, AMIGO: Um Ambiente de Escalonamento Flexível e Dinâmico. Tese de Doutorado, Instituto de Física de São Carlos da Universidade de São Paulo, São Carlos-SP.

[9] T. KUNZ, The Influence of Different Workload Descriptions on a Heuristic Load Balancing Scheme. IEEE Transactions on Software Engineering, v.17, n.7, p.725-730, julho, 1991.

[10] M. HARCHOL-BALTER, A. DOWNEY, Exploiting Process Lifetime Distributions for Dynamic Load Balancing. ACM Transactions on Computer Systems v.15, n.3, agosto, 1997.

[11] A. P. F. ARAÚJO, DPWP – Uma nova abordagem para o Escalonamento Dinâmico em Computação Paralela Virtual. Dissertação de mestrado, ICMC/USP, São Carlos, 1999.

[12] I. EKEMECIC, I. TARTALJA, V. MILUTINOVIC. A Survey of Heterogeneous Computing: Concepts and Systems. Proc. IEEE, 84: 1127-1144, 1996.

[13] H. J. SIEGEL, S. ALI. Techniques for Mapping Tasks to Machines in Heterogeneous Computing Systems. Special Issues on Heterogeneous Distributed and Parallel Architectures: Hardware, Software and Design Tools. June, 1999.