(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
30 December 2015 (30.12.2015)

WIPO | PCT

(10) International Publication Number
WO 2015/198286 Al

(51) International Patent Classification:
*G06F 1/32* (2006.01)

(21) International Application Number:
PCT/IB2015/054835

(22) International Filing Date:
26 June 2015 (26.06.2015)

(25) Filing Language: Italian

(26) Publication Language: English

(30) Priority Data:
GE2014A000062 26 June 2014 (26.06.2014) IT

(71) Applicants: **CONSIGLIO NAZIONALE DELLE RICERCHE** [IT/IT]; Piazzale Aldo Moro 7, 1-00185 Rome (IT). **UNIVERSITA' DEGLI STUDI DI CAGLIARI** [IT/IT]; Via Universita' 40, 1-09124 Cagliari (IT).

(72) Inventors: **CAVIGLIONE, Luca;** Via Piacenza 21/1 1, I-16138 Genova (IT). **PISANO, Alessandro;** Via delle Libellule 12, 1-09134 Cagliari (IT).

(74) Agents: **BORSANO, Corrado** et al; C/o Metroconsult S.r.L, Foro Buonaparte 51, 1-20121 Milan (IT).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available):* AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

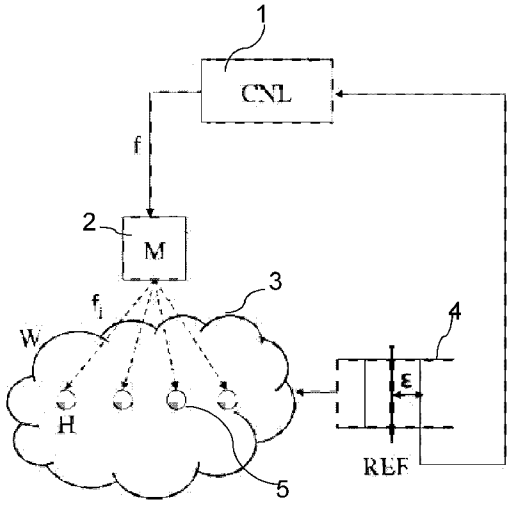
(84) Designated States *(unless otherwise indicated, for every kind of regional protection available):* ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

**Published:**

— *with international search report (Art. 21(3))*

(54) **Title:** METHOD AND SYSTEM FOR REGULATING IN REAL TIME THE CLOCK FREQUENCIES OF AT LEAST ONE CLUSTER OF ELECTRONIC MACHINES

FIG. 1




(57) **Abstract:** Method for regulating in real time the clock frequencies of at least one cluster of electronic machines, characterized in that it provides for carrying out the following steps: a) defining a finite number of discrete virtual capacity values f[l], f[2],.... f[K], as global performance indices, of said cluster of machines; b) calculating by means of a randomized optimization procedure, for each value of said virtual capacity, a set of 1 vectors containing clock frequency values for each machine in said cluster; c) defining a reference queue value, related to the number of processing requests received by said cluster; and provides for iteratively carrying out the following steps: d) measuring the deviation between a current queue value, related to the number of processing requests in said current queue, and said reference queue value; e) selecting one of said discrete virtual capacity values on the basis of said measured deviation, said selection occurring on the basis of a procedure that, at each iteration, analyzes said measured deviation, compares it with the measured deviation value obtained at the previous iteration, and chooses whether to keep the current virtual capacity value or to adopt one of the two adjacent or non-adjacent admissible virtual capacity values of said finite number of discrete values; f) selecting, based on said selected virtual capacity value, a vector of clock frequency values for each machine from said set of 1 vectors, so as to optimize a multi-target performance index (J), and then setting the clock frequency of each machine in the cluster.

WO 2015/198286 A1

# METHOD AND SYSTEM FOR REAL-TIME ADJUSTMENT OF THE CLOCK FREQUENCIES OF AT LEAST ONE CLUSTER OF ELECTRONIC MACHINES

## DESCRIPTION

### Field of the invention.

The present invention relates to a method and a system for regulating in real time the clock frequencies of at least one cluster of electronic machines operating under a variable load.

### State of the art.

Network accessible services have become increasingly complex and are simultaneously used by millions of people (e.g. online social networks, data storage and cloud computing systems), thus requiring the use of Internet-scale infrastructures.

The latter are characterized by a large number of machines, which may also have heterogeneous or specialized hardware and functionalities (e.g. entities for database management o entities offering routing services).

As a consequence, the energy requirements of these installations is critical, since they also imply high running and accessory costs (e.g. for dissipating the generated heat). It should also be pointed out that such installations operate under variable loads (since the load is determined by the instantaneous requests coming from the service users), and it is therefore possible to adopt energy saving policies that deactivate a part of the host computers or reduce the operating frequency and/or the power voltage of the CPU under low-load conditions. In parallel, the current consumer technology implements the Advanced Configuration and Power Interface (ACPI) standard, which allows changing the consumption

profile of the host computers, mainly by acting upon the operating frequency and/or the power voltage of the CPU (and of some peripherals) .

However, the ACPI interface only offers a management mechanism, without providing any control algorithm.

The low-level control mechanisms for energy saving purposes currently known in the art act upon the operating frequency and/or the power voltage (Dynamic Voltage Frequency Scaling - DVFS technique) , without however taking into account any global performance index .

Moreover, the management of a complex system at individual machine level is not scalable, and makes the control problem difficult to be solved in real time.

The use of a distributed approach is a known palliative; however, since it acts locally, this cannot take into account the delays introduced both by the turning on/off of the machines and by the changes in the operating frequency. This also implies the impossibility of ensuring global optimization indices .

The invention has been developed within the frame of a research activity on the reduction of the energy consumption of ICT infrastructures responsible for providing complex services.

From a technical viewpoint, the following aspects or problems of the prior art are of fundamental importance :

i) presence of a standard (whether *de-jure* or de-facto) for controlling the frequency of commercial CPUs and peripherals, i.e. the ACPI standard;

ii) wide literature on the use of DVFS techniques for consumption optimization and for adapting the hosts' clock frequencies to the workload;

iii) numerous cases of use of feedback-based control techniques for optimizing multi-CPU systems;

iv) absence of a two-layer system that allows control at individual machine level in installations equipped with a large number of hosts;

v) absence of an architecture where machines are partitioned into virtual clusters (also referred to as groups or agglomerates), considering concurrence in a common purpose as a joining element;

vi) diffusion of the "Opencompute" standard.

US-8,301,925-B2 describes a method for regulating in real time the operating frequencies of at least one cluster or group of machines, wherein frequencies are considered on two abstraction layers, i.e. an upper layer and a lower layer.

The upper layer is implemented by direct measurement of instantaneous power consumption, and therefore it does not take into account the workload of groups or clusters of machines.

In addition, the lower layer (i.e. the one that manages individual machines, as opposed to "aggregates" of the same) has not been developed with characteristics that ensure system scalability.

US-2009/328055-Al describes a logic for improving the energetic efficiency of Multi-Processor-Systems-On-Chips (MPSPOC) architectures through a shut-down logic for the individual cores and a thread re-allocation strategy. The method proposed therein uses a metrics that also takes into account the number of threads in the queues, but does not provide a "fine" adjustment of the operating frequencies of the single CPUs nor any scalability mechanism for managing large-scale architectures.

**Summary of the invention**

It is therefore the object of the present invention

to propose a method and a system for regulating in real time the clock frequencies of at least one cluster of electronic machines, which can overcome the above-mentioned problems .

The method of the invention solves the above-described problems of the methods currently known in the art, by ensuring a scalable implementation and real-time operation.

The present invention aims at overcoming the above-mentioned drawbacks of the methods currently known in the art by providing a method for regulating in real time the clock frequencies of at least one cluster of electronic machines, which provides for the execution of the following steps:

a ) defining a finite number of discrete virtual capacity values f[l], f[2], …, f[K], as global performance indices in terms of computational capacity, of said cluster of machines;

b ) calculating by means of a randomized optimization procedure, for each value of said virtual capacity, a set of 1 vectors containing clock frequency values for each machine in said cluster;

c ) defining a reference queue value, related to the number of processing requests received by said cluster;

and provides for iteratively carrying out the following steps:

d ) measuring the deviation between a current queue value, related to the number of processing requests in said current queue, and said reference queue value;

e ) selecting one of said discrete virtual capacity values on the basis of said measured deviation, said selection occurring on the basis of a procedure that, at each iteration, analyzes said measured deviation,

compares it with the measured deviation value obtained at the previous iteration, and decides whether to keep the current virtual capacity value or to adopt one of the two directly or non-directly adjacent admissible virtual capacity values of said finite number of discrete values;

f) selecting, based on said selected virtual capacity value, a vector of clock frequency values for each machine from said set of 1 vectors, so as to optimize a multi-target performance index (J), and then setting the clock frequency of each machine of the cluster .

Preferably, said multi-target performance index (J) takes into account the characteristics of the frequencies to be associated with the individual machines, and in particular it allows reducing the number of frequency changes for each machine while avoiding an excessive number of on/off operations for the machines that make up the cluster.

Advantageously, the calculation, for each virtual capacity value, of the set of vectors of operating frequency values for each machine (step b), from which the optimal one will be chosen from time to time, is made during an offline step that precedes the online step comprising the iterative execution of steps d) to f).

The invention allows:

i) defining a specific performance index to be optimized for the service being provided, thanks to the possibility of *ad-hoc* parametrization of said index;

ii) using a high-level control scheme for defining "virtual" capacities (i.e. independent of the underlying technology) ;

iii) dynamically changing in real time the operating

frequencies of each physical entity (host) in use, for the purpose of reducing the energy consumption.

The present invention also relates to a system for regulating in real time the clock frequencies of at least one cluster of electronic machines, comprising:

a unit for defining a finite number of discrete virtual capacity values f[l], f[2], …, f[K], as indices of global performance, of said cluster of machines;

a unit for calculating, by means of a randomized optimization procedure, for each value of said virtual capacity, a set of 1 vectors containing clock frequency values for each machine in said cluster;

a unit for defining a reference queue value, related to the number of processing requests received by said cluster;

a measurer for measuring a reference queue value related to the number of processing requests received by said cluster, adapted to iteratively measure a deviation between a current queue value, related to the number of requests in said current queue, and said reference queue value;

a non-linear controller adapted to iteratively determine- a current virtual capacity value, among said finite number of discrete values, on the basis of said measured deviation, wherein at each iteration it analyzes said measured deviation, compares it with the measured deviation value obtained at the previous iteration, and chooses whether to keep the current virtual capacity value or to adopt one of the two directly or non-directly adjacent admissible virtual capacity values of said finite number of discrete values ;

a mapping unit adapted to select, based on said selected virtual capacity value, a vector of clock

frequency values for each machine from said set of 1 vectors, choosing from a set of admissible configurations for said operating frequencies so as to optimize said multi-target performance index (J), and then setting the clock frequency of each machine in the cluster.

The machines are operating units cooperating with one another to form one or more virtual clusters, considering concurrence in a common purpose as a joining element.

The cluster is inputted a queue of operation requests, which must be processed by the machines that make up the cluster.

The method and the system of the invention allow minimizing the energy consumed by the cluster for processing the requested operations.

The request queue is dynamic, i.e. it changes over time, and the method allows overcoming the limitations of simply acting upon every individual machine, by considering the cluster as a whole.

The iterative part of the method uses a feedback algorithm, which, by controlling the request queue, dynamically regulates first the virtual capacity of the cluster of machines, and then the clock frequency of the individual machines in the cluster in such a way as to minimize the absorbed energy.

A publication entitled: "A Control Theoretic Approach for Energy-Efficient Management of Online Social Network Services" by L. Caviglione, A. Pisano, 978-1-4799-0756-4/13, IEEE, 2013, [TYRRENIAN2013] is known which describes a method for regulating the frequencies of a set of machines, which is only limited to the upper layer of the two-layer scheme of the present invention, and which is specialized for use as a tool for controlling a hardware installation

intended for providing an Online Social Network service. Unlike the present invention, which uses a more general concept of "virtual capacity" in order to make the scheme applicable to broader contexts, said method uses, as a reference variable, a "virtual frequency". According to such a method, the algorithm for virtual capacity selection operates with a constant timing (every T seconds), whereas the present invention offers the possibility of adaptive timing of said algorithm. Furthermore, said method does not include any mechanism for allocating the individual clock frequencies of the individual hosts that make up the various aggregates (clusters) of machines.

It is a particular object of the present invention to provide a method and a system for regulating in real time the clock frequencies of at least one cluster of electronic machines as set out in the claims, which are an integral part of the present description.

**Brief description of the drawings**

Further objects and advantages of the present invention will become apparent from the following detailed description of a preferred embodiment (and variants) thereof referring to the annexed drawings, which are only supplied by way of non-limiting example, wherein:

Fig. 1 is a block diagram of the system according to the invention;

Fig. 2 illustrates the management of a plurality of queues, in accordance with an aspect of the present invention.

In the drawings, the same reference numerals and letters identify the same items or components.

**Detailed description of embodiments of the invention.**

At each activation instant $T_i$, $i=1,2,\ldots$, to be sized

according to the required reactivity of the system, the non-linear controller CNL receives from the measurer M the deviation, measured at point d), between the current queue value and the reference queue value.

Then the non-linear controller calculates a reference virtual capacity value *f*, as a global performance index, for the cluster of machines, as per the above step e). The virtual capacity is chosen from a set of admissible values f[l], f[2], …, f[K].

Based on the calculated reference virtual capacity value *f*, the mapping unit selects, among the admissible vectors, the "best" vector (which optimizes a multi-target performance index, hereafter referred to as J) containing the frequencies $f_i$, $i=1,...,N$, to be assigned to each machine, as per step f).

At each activation instant $T_i$, i=l,2,..., the measurer acquires the difference between the current queue value and the reference queue value, and a new deviation is calculated, which is then provided again to the non-linear controller CNL in order to carry out a subsequent iterative step.

One option regarding the choice of the activation instants $T_i$ is to choose them equally spaced in time (Ti=i T), de *facto* activating the algorithm CNL every T seconds.

According to one possible alternative, the algorithm uses an adaptive timing which is load-dependent, i.e. dependent on the measured value of the request queue. In this case, in addition to detecting the deviation ε (defined below) from the reference queue value, the non-linear controller CNL must also evaluate the entire contents of the queue.

To this end, it is possible to define a number Q of

operations, such that the non-linear controller will only be activated and will only execute the algorithm when the queue increases or decreases by the value Q of requests.

The cluster of machines is the abstraction of N entities, each operating at an operating frequency $f_i$ of its own.

The operating frequency "status" of the cluster is thus described by a frequency vector $f_i$ having the following structure: vector=[fi, $f_2$, …, $f_N$]

Due to technologic constraints, the frequency $f_i$ can take a limited number of values. For example, when using the ACPI standard, the frequency $f_i$ can typically take three/four values. Generally such values are equally distributed within the ranges $[f_{max}/2, f_{max}]$ or $[f_{max}/3_f f_{max}]$.

By convention, a turned-off or in-idle machine has $f_i = 0$.

Then the non-linear controller CNL produces, as a reference value, a "virtual" capacity $f$, meaning that it abstracts the single allocations into a single value not bound to the underlying technology, i.e. the vector that contains the operating frequencies $f_i$ of the individual machines.

Of course, $f$ must meet the constraint $f \leq f_{max}$; however, the presence of idle machines also allows $f \leq f_{max}/2$ or $f \leq f_{max}/3$. In this case, it will be necessary to turn off an appropriate number of machines for reaching such a value. Note that this innovative approach allows condensing into a single parameter also the presence of inactive nodes.

In one example of embodiment, the calculation of the virtual capacity value on the basis of the measured deviation provides for comparing the current measured deviation with the deviation measured at the previous

step.

If the current measured deviation corresponds to a longer queue compared to the previously detected value, then the virtual capacity value will be increased to the immediately higher adjacent value.

On the contrary, if the current measured deviation corresponds to a shorter queue, then the virtual capacity value will be decreased to the immediately lower adjacent value.

In this manner, the non-linear controller will consider the reference queue value and, if the queue is longer than the reference queue value and the variation trend goes towards a further increase in the queue value, it will calculate an increased virtual capacity value, so as to increase the computational capacity of the cluster of machines. This indicates, in fact, that the queue is growing and that the machines are not carrying out the requests quickly enough. If, on the contrary, the queue is shorter than the reference value, and is progressively getting even shorter, indicating an oversized operating frequency for the current workload, then the non-linear controller will calculate a lower virtual capacity value, corresponding to a lower speed of the machines.

In an improved embodiment, the calculation of the virtual capacity value based on the measured deviation provides for comparing the current measured deviation with a hysteresis parameter.

The hysteresis parameter is introduced in order to reduce the number of variations of $f$ to be made, in that it allows changing the virtual capacity value $f$ only when significant variations of the measured deviation occur. This means that the non-linear controller is prevented from changing the virtual

capacity *f* upon any small variation of the measured deviation .

The hysteresis parameter can be sized at will on the basis of the characteristics of the cluster of machines.

Let Δ be the hysteresis parameter. Let ε[j] be the error or deviation measured at the j-th step. Let *f* be the virtual capacity that one wishes to set for the cluster of machines, through the allocation calculated by the mapping unit. Let *f* be in the range of $f_{min} \leq f \leq f_{max}$, where $f_{min}$ represents the minimum virtual capacity to be imposed on the cluster W (e.g. such value may represent a mix of turned-off machines and active machines operating at the minimum ACPI frequency) . The maximum number of admissible turned-off (idle) machines is a design parameter, and $f_{min}=0$ will correspond, at most, to a fully deactivated cluster (i.e. with all machines in idle condition) . *f* is assumed to be discrete, i.e. selectable among K possible values f[l], f[2], …, f[K] equally distributed within the range $[f_{min}, f_{max}]$. For finer control, the number K is advantageously greater than the number of frequency steps allowed by the ACPI standard. Let the operations of "increasing" and "decreasing" the virtual capacity *f* be forced to use only the previous/next steps of f.

In one example of embodiment, the non-linear controller CNL uses, for the calculation of the reference frequency value based on the measured deviation, the following algorithm:

At each activation instance it executes;

*if* [(ε[j] < - Δ) AND (e[j] ≤ e[j-l]) *then* f is increased *to the adjacent admissible value*

*if* [(s[j] > Δ) AND (e[j] ≥ e[j-l]) *then* f is decreased *to the adjacent admissible value*

It is however possible to adopt different forms of the non-linear controller algorithm, provided that it implements a negative feedback logic on the deviation $\varepsilon$ (e.g. a P.I.D. algorithm or variants thereof). It is also possible, in the presence of very fast variations of the deviation $z$, to modify the algorithm in such a way as to increase or decrease $f$ towards admissible values not immediately adjacent to the current value.

In a preferred embodiment, the mapping unit or mapper also acts as a computing unit for calculating, for each reference virtual capacity value, a set of vectors of operating frequency values for each machine. However, these may also be two separate units.

The mapping unit is responsible for finding an allocation vector that reflects, as accurately as possible, the current virtual capacity value $f$.

However, this mapping problem is computationally exacting for the following reasons:

1) the number of machines that make up the cluster may be high, e.g. more than 10,000 units;

2) the standard mechanism for energetic management of consumer machines is based on ACPI, which provides a limited and discrete number of operating frequencies: numerically, this implies the use of integer variables ;

3) the on/off condition of a machine is also an integer (binary) variable.

4) at the same time, it is desirable to minimize the number of machines that need to be turned on/off and for which the operating frequency needs to be changed, since such operations introduce delays.

To this end, the mapping unit uses an algorithm divided into two parts: an offline step for

calculating, for each admissible virtual capacity value, a set of vectors of operating frequency values for each machine, to be carried out only once at the system design stage (or in the event of significant structural changes) , and an online step for selecting the vector of operating frequency values for each machine on the basis of the calculated virtual capacity value, which is carried out at every variation of $f$.

However, also the offline process may require high computational resources. Therefore, in order to make the proposed method widely accessible, inexpensive, and executable on simple hardware, a randomized procedure is introduced for sampling all possible allocation vectors.

According to one example of embodiment, the calculation of a set of vectors of operating frequency values for each machine (offline step) for each virtual capacity value includes:

i ) defining a number of turned-off machines $M_0[i]$, i=l,2,...,K for each admissible virtual capacity value f[i], and an additional parameter $M_0[K+1]$ that defines the minimum admissible number of turned-off machines .

ii) randomly generating, for each admissible virtual capacity value, a set of vectors of operating frequencies for the individual machines. The randomized generation procedure is carried out by taking into account the variation of the number of turned-off machines between the adjacent virtual capacity values. In particular, a sub-set of the calculated vectors must have a number of turned-off machines equal to the number of turned-off machines associated with the adjacent virtual capacity values. In this manner, constraints are imposed on the

variation of the number of turned-on or turned-off machines among the admissible configuration vectors related to adjacent values of virtual capacity *f* by entering appropriate "link" values, for the purpose of reducing the number of machines with alternating on and off states.

The link is advantageously effected for both the lower adjacent virtual capacity value and the upper adjacent virtual capacity value.

If the modified version of the virtual capacity calculation algorithm is adopted, which can increase or decrease *f* towards admissible values not immediately adjacent to the current value, then the randomized generation of the admissible configuration vectors related to the discrete virtual capacity values will have to provide a link, in terms of number of turned-off machines, also to virtual capacity values that are not directly adjacent.

Advantageously, the number of turned-off machines for each virtual capacity value may decrease linearly as the virtual capacity values grow, thus following a simple proportionality constant with a subsequent rounding off.

This reflects the fact that greater virtual capacities require smaller numbers of turned-off machines .

The method then provides a random procedure for sampling all possible allocation vectors.

A random generation is followed by a selection of the vectors composed of values that generate a mean as close as possible to the virtual capacity value taken into account.

This dramatically reduces the time required for calculating the set of vectors of operating frequency values for each machine for each virtual capacity

value: said calculation, in fact, would be excessively costly in computational terms for very large clusters, if the vectors had to be calculated, starting from the virtual capacity values, as a linear combination of frequency values the mean of which corresponds to the virtual capacity under consideration .

One example of embodiment of the algorithm for the randomized selection of the configuration vectors includes the following operations:

Let the admissible values f[l], f[2], …, f[K] for f be sorted in increasing order. Let L be the number of attempts, and let 1 be the number of values to be considered for the next online procedure. Let 1 << L .

For the offline step, the vector calculation uses the following algorithm (executed by the mapping unit (mapper) M ):

Let $M_0$=[$M_0$[1], Mo[2], …, $M_0$[K]] be the vector that contains the number of turned-off machines for each value f[l], f[2], …, f[K] of f, and let $M_0$[K+1] be the minimum admissible number of turned-off machines.

Calculation of the vectors associated with the frequency f[1]:

*set* the number of turned-off machines to $M_0$[l]

*generate* in a random manner L/2 admissible values for the remaining M-Mo[l] machines

*calculate* $\mathbf{err} = \left| f - \frac{1}{M - M0[1]} \sum_{i=1}^{M-M0[1]} f_i \right|$

*choose the* 1/2 values that have a minimum **err**

*set* the number of turned-off machines to $M_0$[2]

*generate* in a random manner L/2 admissible values for the remaining M-$M_0$[2] machines

*calculate* $\mathbf{err} = \left| f - \frac{1}{M - M0[2]} \sum_{i=1}^{M-M0[2]} f_i \right|$

*choose the* 1/2 values that have a minimum **err**

Calculation of the vectors associated with the frequency f[i] (i=2, 3,..., K-l) :

set the number of turned-off machines to Mo[i-1]

*generate* in a random manner L/2 admissible values for the remaining $M-M_0[i-1]$ machines

*calculate* ***err*** = $\left| f - \frac{1}{M - M0[i-1]} \sum_{i=1}^{M-M0[i-1]} f_i \right|$

*choose the* 1/2 values that have a minimum **err**

*set* the number of turned-off machines to $M_0[i]$

*generate* in a random manner L/2 admissible values for the remaining $M-M_0[i]$ machines

*calculate* ***err*** = $\left| f - \frac{1}{M - M0[i]} \sum_{i=1}^{M-M0[i]} f_i \right|$

*choose the* 1/2 values that have a minimum **err**

Calculation of the vectors associated with the frequency $f_{[K]}$ :

set the number of turned-off machines to $M_0[K]$

*generate* in a random manner L/2 admissible values for the remaining $M-M_0[K]$ machines

*calculate* ***err*** = $\left| f - \frac{1}{M - M0[K]} \sum_{i=1}^{M-M0[K]} f_i \right|$

*choose the* 1/2 values that have a minimum **err**

set the number of turned-off machines to $M_0[K+1]$

*generate* in a random manner L/2 admissible values for the remaining $M-M_0[K+1]$ machines

*calculate* ***err*** = $\left| f - \frac{1}{M - M0[K+1]} \sum_{i=1}^{M-M0[K+1]} f_i \right|$

*choose the* 1/2 values that have a minimum **err**

The procedure returns 1*K vectors. As L grows, the error **err** decreases at the cost of a higher computational load.

The output of this procedure is indicated in the

following table:

| Values of $f$ calculated by CNL | Offline pre-allocation |
|---|---|
| $f[1]$ | $Vector_1$ of values $f_i$ for $f[1]$<br>$Vector_2$ of values $f_i$ for $f[1]$<br>…<br>$Vector_l$ of values $f_i$ for $f[1]$ |
| $f[2]$ | $Vector_1$ of values $f_i$ for $f[2]$<br>$Vector_2$ of values $f_i$ for $f[2]$<br>…<br>$Vector_l$ of values $f_i$ for $f[2]$ |
| … | … |
| $f[K]$ | $Vector_1$ of values $f_i$ for $f[K]$<br>$Vector_2$ of values $f_i$ for $f[K]$<br>…<br>$Vector_l$ of values $f_i$ for $f[K]$ |

where Vectori is a vector of operating frequency values $f_i$ for each machine.

At each value of f calculated by CNL, there are 1 vectors available. The mapper M chooses the optimal one to be used. The vector also contains the information about the connection between a single frequency and the machine involved that will have to use it, also in the case of f=0, which means that the machine is off.

The output of the offline procedure is then stored into a memory unit of the mapping unit, e.g. a flash memory or the like.

For this reason, M may be a very simple device, since there is a tradeoff between computational power and storage capacity (for storing the 1*K values).

According to a further embodiment, the setting of the operating frequency of each machine (online step) is

performed by selecting, among the vectors related to the calculated virtual capacity value, the vector that provides the number of turned-off machines closest to the number of turned-off machines of the vector selected at the previous iterative step, or that provides the smallest number of changes to the operating frequencies of the individual machines with respect to the vector selected at the previous iterative step.

According to a further and more general embodiment, the setting of the operating frequency of each machine (online step) is made in such a way as to optimize a multi-target performance index (J) by choosing, among the vectors related to the virtual capacity value, the one that minimizes a cost index that "weighs", with arbitrary coefficients (hereafter referred to as al, a2, a3), the deviation between the mean value of the vector and the virtual capacity f, the number of machines in the cluster that need to be turned on/off, and the number of machines in the cluster for which the operating frequency needs to be changed. Of course, these requests are in conflict with each other, and the weights can be chosen according to the case, in such a way as to obtain the desired tradeoff (e.g. to give priority to minimizing the number of machines to be turned on/off over the necessary number of frequency changes) .

The vector selection procedure, which is the online part (executed by the mapper) , uses the following algorithm:

Let Vector be the current allocation, and let f[i] be the virtual capacity chosen by the algorithm (executed by CNL) :

- Calculate, for each $Vector_j$ (j=1,2,…,1) associated with f[i], the difference between

the number of idle machines in the configuration Vectorj and the number of idle machines in the current configuration Vector. Let $D_j$ be such quantities.

- Calculate, for each $Vector_j$ associated with f[i], the number of machines the frequency of which should be changed in order to bring the cluster or worker into the configuration $Vector_j$ starting from the current configuration Vector. Let $v_j$ be such quantities .
- Calculate, for each $Vector_j$ associated with f[i], the difference between f[i] and the algebraic mean $\frac{1}{N}\sum_{i=1}^{N} f_i$ of the frequencies contained in the vector $Vector_j$. Siano $S_j$ tali quantita.Let $s_j$ be such quantities .Calculate, for each Vectorj associated with f[i], the index $J=a1\ |D_j|+ a2\ |v_j|+ a3\ |s_j|$, where a1, a2, a3 are non-negative "weights" chosen arbitrarily.
- Choose the Vectorj corresponding to the minimum value of J.

The control method proposed herein is based on the architecture shown in Figure 1. For simplicity, the following will describe the case wherein there is only one group of machines to be controlled. .

However, the mechanism is designed for working in a scalable manner with multiple groups.

The non-linear controller 1 (CNL) is responsible for calculating the virtual or high-level capacity, designated as *f*. This value is the reference for the cluster of machines 3, the consumption of which needs be optimized.

The mapping unit 2, or Mapper (M), is the entity responsible for converting the value $f$ into a frequency allocation for an individual machine, designated as $f_i$.

The cluster 3, or Worker (W), is a homogeneous group of machines 5 concurring in a common target. The cluster 3 is therefore populated by individual hosts/machines 5 (H). For example, the cluster 3 may consist of N nodes responsible for providing a database service, or N devices implementing network functionalities .

The queue 4 of the requests / processing load offered to W is essentially virtual, meaning that it is a measure of the volume of requests, or it is quantified by monitoring the queuing in the buffers of a controller/dispatcher. By way of example, in the case wherein W abstracts a Web Farm, the queue 4 will be represented by the backlog of the requests of the Web front-ends.

REF designates a value of the queue 4 that needs to be maintained.

$\varepsilon$ designates the error, i.e. the deviation between the current value of the queue 4 and the desired value REF.

Said value is then returned to the CNL 1, which will use it to make the appropriate decision.

The architecture of Figure 1 can be extended to the case of multiple clusters 3, as shown in Figure 2.

Due to the separate mapping units 4, the CNL 1 can consistently manage a variety. of heterogeneous machines.

The invention pursues the idea of abstracting a complex system as a chain of Workers (W, Wl, W21....W24) interconnected by means of queues (Ql, Q21....Q24, Q31...Q34), by using non-linear control

techniques for the energetic optimization of multi-CPU systems.

In view of a possible implementation of the system, the "Opencompute" standard has also been taken into account, which offers all the functionalities necessary for collecting data for building the reference queues and the remote interfaces for managing the servers (hence effectively applying the control results to the hardware) .

Furthermore, "Opencompute" emphasizes the use of unified hardware also for the creation of network devices, thus making the "partitioning into workers" mechanism excellent for the integrated management of green-computing/networking policies .

The present invention can advantageously be implemented through a computer program, e.g. written in C or Java language, which comprises coding means for implementing one or more steps of the method when said program is executed by a computer. It is therefore understood that the protection scope extends to said computer program as well as to computer-readable means that comprise a recorded message, said computer-readable means comprising program coding means for implementing one or more steps of the method when said program is executed by a computer.

The above-described non-limiting examples may be subject to further variations without departing from the protection scope of the present invention, including all equivalent embodiments known to a man skilled in the art.

The elements and features shown in the various preferred embodiments may be combined together without however departing from the protection scope of the present invention.

The advantages deriving from the application of the present invention are apparent.

i) Availability of a scalable and runtime-implementable architecture which allows real-time dynamic variations of the operating frequencies of each physical entity (host) in one or more agglomerates or groups or clusters, for the purpose of reducing the energy consumption and the delays due to the turning on/off of the hosts and to the variations of their clock frequencies.

ii) Possibility of defining performance indices which are specific for the service being provided, thanks to the possibility of *ad-hoc* parametrization of said indices.

From the above description, those skilled in the art will be able to produce the object of the invention without introducing any further details.

**CLAIMS**

1. Method for regulating in real time the clock frequencies of at least one cluster of electronic machines, characterized in that it provides for carrying out the following steps:

a) defining a finite number of discrete virtual capacity values f[l], f[2], …, f[K], as global performance indices, of said cluster of machines;

b) calculating by means of a randomized optimization procedure, for each one of said virtual capacity values, a set of 1 vectors containing clock frequency values for each machine in said cluster;

c) defining a reference queue value, related to the number of processing requests received by said cluster;

and provides for iteratively carrying out the following steps:

d) measuring the deviation between a current queue value, related to the number of processing requests in said current queue, and said reference queue value;

e) selecting one of said discrete virtual capacity values on the basis of said measured deviation, said selection occurring on the basis of a procedure that, at each iteration, analyzes said measured deviation, compares it with the measured deviation value obtained at the previous iteration, and chooses whether to keep the current virtual capacity value or to adopt one of the two directly or non-directly adjacent admissible virtual capacity values of said finite number of discrete values;

f) selecting, based on said selected virtual capacity value, a vector of clock frequency values for each machine from said set of 1 vectors, so as to optimize a multi-target performance index (J), and then

setting the clock frequency of each machine in the cluster .

2. Method according to claim 1, wherein the calculation of the virtual capacity value based on the measured deviation provides for comparing the current measured deviation with the deviation measured at the previous step, and, if the current measured deviation corresponds to an increase in said current queue value, then the virtual capacity value is increased to the next discrete value, or, if the current measured deviation corresponds to a decrease in said current queue value, then the virtual capacity value is decreased to the previous discrete value .

3. Method according to claim 2, wherein the calculation of the virtual capacity value based on the measured deviation provides for comparing the current measured deviation with a hysteresis parameter .

4. Method according to one or more of the preceding claims, wherein said calculation, for each virtual capacity value, of a set of vectors of clock frequency for each machine provides for defining a number of turned-off machines for each virtual capacity value and a minimum admissible number of turned-off machines, and for generating a set of vectors such that a predetermined number of vectors provide a number of turned-off machines corresponding to the number of turned-off machines related to the adjacent virtual capacity values.

5. Method according to one or more of the preceding claims, wherein the setting of the clock frequency of each machine is performed by selecting, among the vectors related to the calculated virtual capacity value, the vector that provides the number of turned-

off machines closest to the number of turned-off machines of the vector selected at the previous iterative step, or that provides the smallest number of changes to the operating frequencies of the individual machines with respect to the vector selected at the previous iterative step.

6. Method according to one or more of the preceding claims, wherein the step of optimizing a multi-target performance index is carried out by selecting, among said vectors related to the virtual capacity value, the one that minimizes a cost index that weighs, with arbitrary coefficients, the deviation between the mean vector value and said virtual capacity f, the number of machines in the cluster that need to be turned on/off, and the number of machines in the cluster for which the clock frequency needs to be changed.

7. System for regulating in real time the clock frequencies of at least one cluster of electronic machines, characterized in that it comprises:

a unit for defining a finite number of discrete virtual capacity values f[l], f[2], …, f[K], as indices of global performance, of said cluster of machines;

a unit for calculating by means of a randomized optimization procedure, for each value of said virtual capacity, a set of 1 vectors containing clock frequency values for each machine in said cluster;

a unit for defining a reference queue value, related to the number of processing requests received by said cluster;

a measurer for measuring a reference queue value related to the number of processing requests received by said cluster, adapted to iteratively measure a deviation between a current queue value, related to

the number of requests in said current queue, and said reference queue value;

a non-linear controller (CNL) adapted to iteratively determine a current virtual capacity value, from said finite number of discrete values, on the basis of said measured deviation, wherein at each iteration it analyzes said measured deviation, compares it with the measured deviation value obtained at the previous iteration, and chooses whether to keep the current virtual capacity value or to adopt one of the two directly or non-directly adjacent admissible virtual capacity values of said finite number of discrete values ;

a mapping unit (M) adapted to select, based on said selected virtual capacity value, a vector of clock frequency values for each machine from said set of 1 vectors, choosing among a set of admissible configurations for said operating frequencies so as to optimize said multi-target performance index (J) , and then setting the clock frequency of each machine in the cluster.

8 . System according to claim 8, characterized in that it comprises means for operating according to the method according to one or more of claims 1 to 6 .

9 . Computer program comprising program coding means adapted to implement the steps of claims 1 to 6 when said program is executed on a computer.

10. Computer-readable means comprising a recorded program, said computer-readable means comprising program coding means adapted to implement the steps of claims 1 to 6 when said program is executed on a computer .

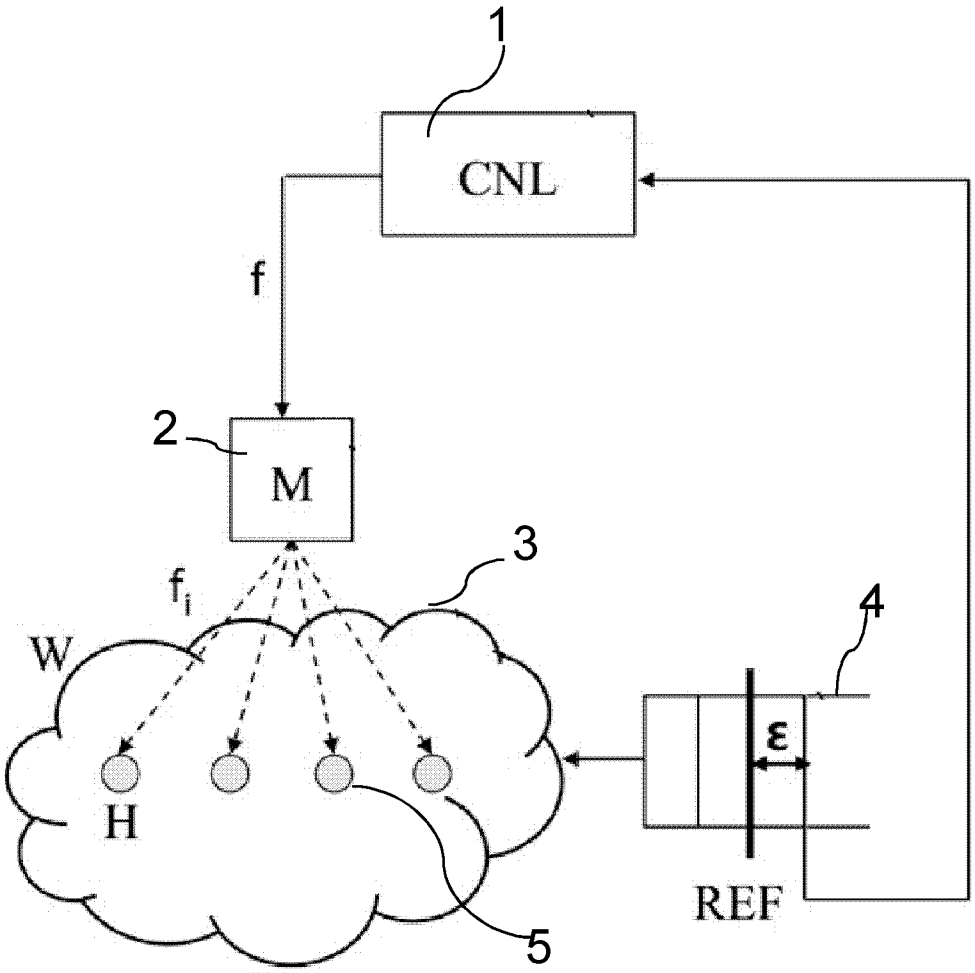



FIG. 1

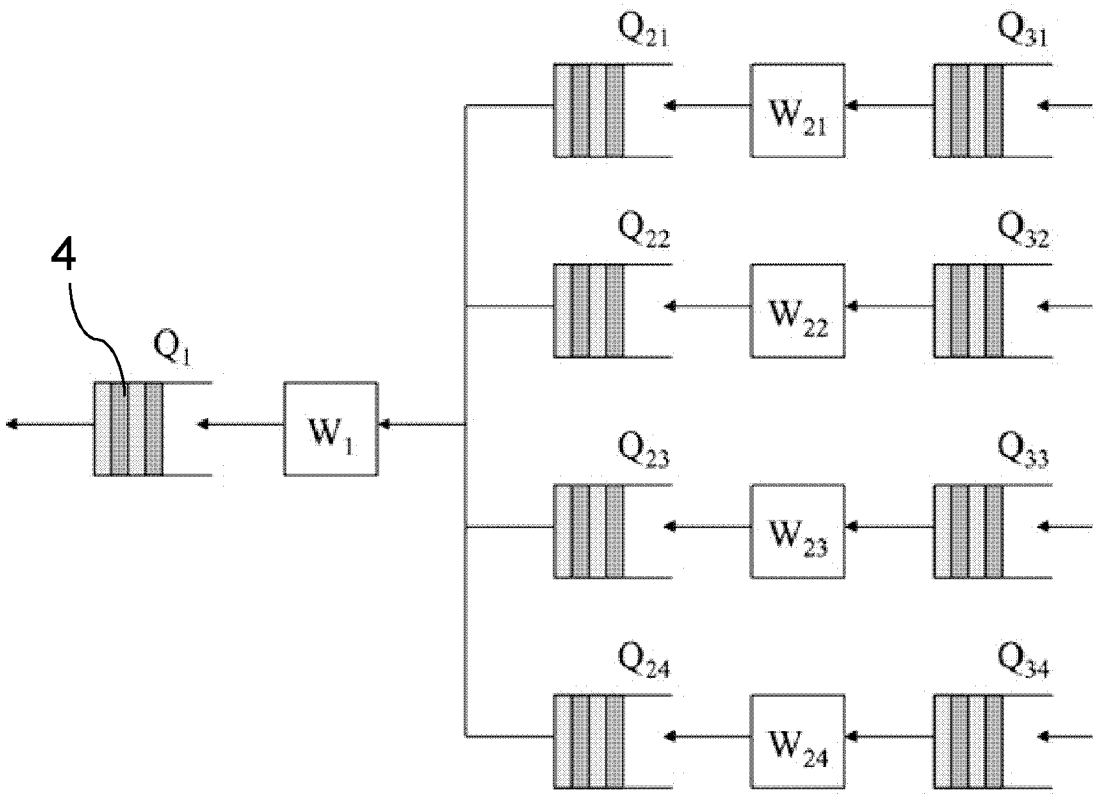



FIG. 2

# INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2015/054835

**A. CLASSIFICATION OF SUBJECT MATTER**
INV. G06F1/32
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal , WPI Data

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| X | WO 2012/170214 A2 (QUALCOMM INC [US] ; THOMSON STEVEN S [US] ; REGINI ED0ARD0 [US] ; MONDAL) 13 December 2012 (2012-12-13) paragraph [0015] - paragraph [0114] ; figures 1-16 | 1-10 |
| X | US 2009/328055 Al (B0SE PRADIP [US] ET AL) 31 December 2009 (2009-12-31) paragraph [0019] - paragraph [0087] ; figures 1-4 | 1-10 |
| X | US 2011/289329 Al (B0SE SUMIT KUMAR [IN] ET AL) 24 November 2011 (2011-11-24) paragraph [0031] - paragraph [0068] ; figures 1-4 | 1-10 |

-/ -

[X] Further documents are listed in the continuation of Box C. [X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) orwhich is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
|---|---|
| 6 October 2015 | 14/10/2015 |

| Name and mailing address of the ISA/ | Authorized officer |
|---|---|
| European Patent Office, P.B. 5818 Patentlaan 2<br>NL - 2280 HV Rijswijk<br>Tel. (+31-70) 340-2040,<br>Fax: (+31-70) 340-3016 | Vertua, Arturo |

1

Form PCT/ISA/210 (second sheet) (April 2005)

# INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2015/054835

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| X | US 2002/169990 Al (SHERBURNE ROBERT WARREN [US] SHERBURNE J R ROBERT WARREN [US] ) 14 November 2002 (2002-11-14) paragraph [0008] - paragraph [0070] ; f i gures 1-5 | 1-10 |
| A | US 2008/028249 Al (AGRAWAL PARAG V [IN] AGRAWAL PARAG VIJAY [IN] ) 31 January 2008 (2008-01-31) the whole document | 1-10 |
| A | US 2002/099964 Al (ZDRAVKOVIC ANDREJ [CA] ) 25 July 2002 (2002-07-25) the whole document | 1-10 |
| A | US 2009/094437 Al (FUKUDA MASAHI R0 [JP] ) 9 Apri l 2009 (2009-04-09) the whole document | 1-10 |
| A | US 2012/066526 Al (SALSBERY BRIAN J [US] ET AL) 15 March 2012 (2012-03-15) the whole document | 1-10 |

1

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/IB2015/054835

| Patent document cited in search report | Publication date | Patent family member(s) | Publication date |
|---|---|---|---|
| WO 2012170214 A2 | 13-12-2012 | US 2013060555 Al | 07-03-2013 |
| | | WO 2012170214 A2 | 13-12-2012 |
| US 2009328055 Al | 31-12--2009 | NONE | |
| US 2011289329 Al | 24-11--2011 | AU 2011255552 Al | 13--12--2012 |
| | | CA 2799985 Al | 24--11--2011 |
| | | EP 2572254 A2 | 27--03--2013 |
| | | US 2011289329 Al | 24--11--2011 |
| | | WO 2011146731 A2 | 24--11--2011 |
| US 2002169990 Al | 14-11--2002 | US 2002169990 Al | 14--11--2002 |
| | | US 2006080566 Al | 13--04--2006 |
| US 2008028249 Al | 31-01--2008 | NONE | |
| US 2002099964 Al | 25-07-2002 | EP 1237067 A2 | 04--09--2002 |
| | | US 2002099964 Al | 25--07--2002 |
| | | US 2005044435 Al | 24--02--2005 |
| | | US 2007208962 Al | 06--09--2007 |
| US 2009094437 Al | 09.04.-2009 | JP 5182792 B2 | 17-.04-2013 |
| | | JP 2009093383 A | 30--04--2009 |
| | | US 2009094437 Al | 09-.04.-2009 |
| US 2012066526 Al | 15-03--2012 | US 2012066526 Al | 15--03--2012 |
| | | WO 2012036779 Al | 22--03--2012 |