

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
9 July 2009 (09.07.2009)

PCT

(10) International Publication Number
WO 2009/086326 A1

- (51) International Patent Classification:
G06F 9/45 (2006.01)
- (21) International Application Number:
PCT/US2008/088062
- (22) International Filing Date:
22 December 2008 (22.12.2008)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
61/015,615 20 December 2007 (20.12.2007) US
- (71) Applicant (for all designated States except US): **AKORRI NETWORKS, INC.** [US/US]; 305 Foster Street, Littleton, MA 01460 (US).
- (72) Inventors; and
- (75) Inventors/Applicants (for US only): **SMIRNOV, George** [US/US]; 305 Foster Street, Littleton, MA 01460 (US). **FAIRBANKS, Leon** [US/US]; 305 Foster Street, Littleton, MA 01460 (US). **HU, Kenneth** [US/US]; 305

Foster Street, Littleton, MA 01460 (US). **KAELI, David** [US/US]; 305 Foster Street, Littleton, MA 01460 (US).

(74) Agent: **JACOBS & KIM LLP**; 1050 Winter Street, Suite 1000, #1082, Waltham, MA 02451 (US).

(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, BR, BW, BY, BZ, CA, CH, 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, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, 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, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL,

[Continued on next page]

(54) Title: EVALUATING AND PREDICTING COMPUTER SYSTEM PERFORMANCE USING KNEEPOINT ANALYSIS

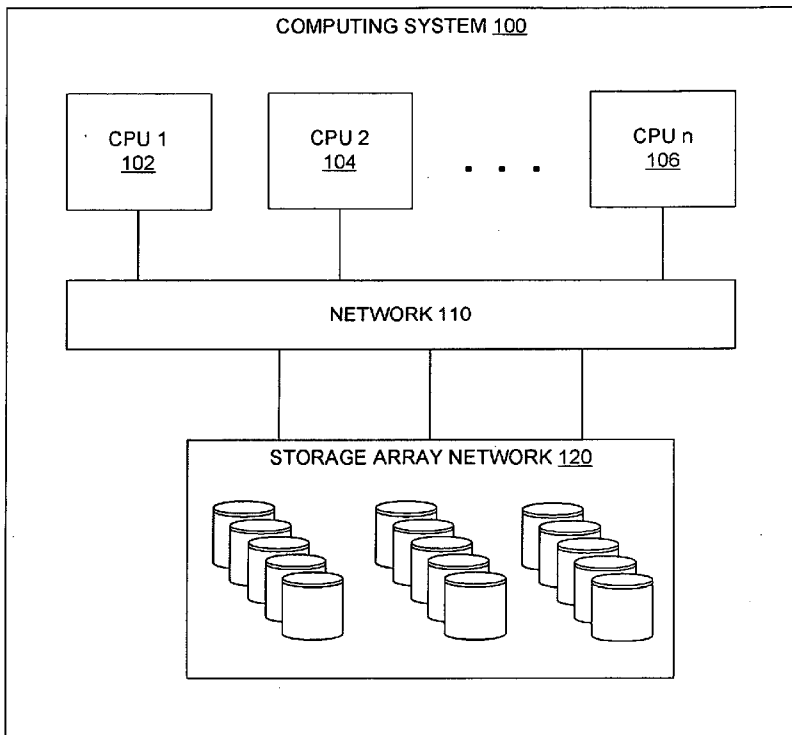


FIG. 1

(57) Abstract: The present invention provides methods, systems, apparatus, and computer software/program code products adapted for operating in, or in conjunction with, an otherwise conventional computing system, and which enable evaluating, monitoring and predicting the performance of computer systems and individual elements or groups of elements within such computer systems.

WO 2009/086326 A1



NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG,
CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

5

EVALUATING AND PREDICTING COMPUTER SYSTEM PERFORMANCE USING KNEEPOINT ANALYSIS

Cross-Reference to Related Applications

This application for patent claims the priority benefit of United States Provisional Patent Application Serial No. 61/015,615 filed December 20, 2007 (Attorney Docket: AKR-115-PR), entitled "Evaluating and Predicting Computer System Performance Using Kneepoint Analysis," which is incorporated by reference herein as if set forth in its entirety.

Also incorporated herein by reference in their entirety are United States Patent Application Serial No. 11/773,825, filed July 5, 2007 (Attorney Docket: AKR-110-US), entitled "Managing Application System Load," and United States Patent Application Serial No. 11/961,186, filed December 20, 2007 (Attorney Docket: AKR-114-US), entitled "Methods and Systems for Identifying Application System Storage Resources."

Field of the Invention

The present invention relates to methods, systems, apparatus, and computer software/program code for evaluating, monitoring and predicting the performance of computer systems or individual elements or groups of elements within such computer systems.

Background of the Invention

In many computing environments, including, for example, network-based computing systems that may include multiple computing or processing units, storage units, storage networks, and other system resources, it is desirable to monitor the performance of the computing system or the performance of individual elements or groups of elements within the system. (The terms "computer system" and "computing system" are utilized substantially interchangeably herein unless otherwise indicated in particular circumstances.)

Computer system performance monitoring, evaluation and even prediction can be useful for a variety of reasons and purposes, such as determining whether additional storage, processing or other system resources may need to be added or brought online to meet quality

of service or reliability requirements, to enable system balancing, and to provide application tuning, among other possible applications.

By way of example, FIG 1 is a schematic diagram depicting general aspects of a conventional computing system 100 that might be found in the prior art. A computing system
5 such as that shown in FIG 1 can include a Storage Array Network (SAN) 120, a Network (such as a Local Area Network (LAN), Wide Area Network (WAN) or other form of network(s)), and a number of Central Processing Units (CPUs) 102, 104, ... 106, etc.

Referring now to FIG 2, it would be desirable to be able to monitor, or even predict, aspects of the performance of the computing system such as that shown in FIGS. 1 and 2, or
10 the performance of individual elements or groups of elements in the computing system, such as SAN 120, network 110, or CPUs 102, 104, ... 106, etc.

In FIG 2, the functions of monitoring or predicting the performance of the computing system, or elements of the computing system (and corresponding devices, systems or computer software/program code products adapted for providing such functions), are
15 represented schematically by performance monitoring meter, element, module or function 130. Referring again to FIG 2, element 130 is shown in communication with the various elements of the computing system of FIG 2, including SAN 120, network 110, or CPUs 102, 104, ... 106, etc.

While various forms of elements, structures and computer software/program code
20 products adapted to monitor the performance of computing systems are known in the prior art, it is the novel aspects of the structure, function and operation of element, module or function 130, utilizing kneepoint analysis (KPA) that form the subject of the present invention.

Summary of the Invention

25 The present invention provides methods, systems, apparatus, and computer software/program code products adapted for operating in, or in conjunction with, an otherwise conventional computing system (such as that depicted in FIG 1 attached hereto), and which enable evaluating, monitoring and predicting the performance of computer systems and individual elements or groups of elements within such computer systems.

In accordance with one aspect of the present invention, a computer system performance evaluation and prediction mechanism is based on utilization measurements of a computer system or its components.

5 In a further aspect of the present invention, a computer system performance evaluation and prediction mechanism is based on a comparison of utilization of the computer system or its components to a coordinate of the kneepoint of the operating curve of the computer system or its components.

10 Another aspect of the present invention provides a method comprising: characterizing the throughput produced by a queuing network model (the throughput representing the utilization of each element in the queuing network), collecting runtime performance data from the computing system (the runtime performance data representing the present response time and resource utilization on individual elements on the computing system), using the utilization data for each element in the queuing network to define an operating curve, evaluating the operating curve to identify a “kneepoint” in the operating curve, and using kneepoint analysis
15 to classify non-linear performance response in the computing system.

Another aspect of the invention provides a method of determining whether the performance of a computing system is likely to scale exponentially or faster than exponentially with respect to increased system load, the method comprising: (a) measuring utilization of one or more computing system components or groups of components to generate
20 utilization information, wherein the utilization may vary over time; (b) generating, using the utilization information, an operating curve characteristics of the computing system; (c) determining at least one utilization value on the operating curve to be an x-coordinate of a kneepoint of the operating curve, the kneepoint being a unique property of the operating curve and being substantially constant for a performance queuing model representing one or more
25 aspects of the performance of the computing system; (d) comparing a given utilization value to the x-coordinate of the kneepoint; (e) setting a bound based on the x-coordinate of the kneepoint; and (f) indicating an alarm condition when the given utilization value is above the bound based on the x-coordinate of the kneepoint.

30 In a further aspect of the invention, a kneepoint is determined using a set of kneepoint parameters, and the kneepoint parameters are calculated in closed analytical form for the linear operating region of the logarithm of the operating curve.

In another aspect of the invention, the linear operating region is a function solely of the queuing model; and in a further aspect of the invention, the linear operating region is calculated in a closed analytical form.

In still another aspect of the invention, the set of kneepoint parameters comprises curvature of the logarithm of the operating curve in the linear operating region, and bounds of the linear operating region comprise two points on the x-axis of operating curve.

A further aspect of the invention comprises evaluating system element performance by comparing system element utilization measurements with at least one operating curve kneepoint.

Each of these aspects will be discussed in greater detail below, along with various examples, embodiments and practices of the present invention.

Those skilled in the art will appreciate that while the following detailed description provides sufficient detail to enable one skilled in the art to practice the present invention, the various examples, embodiments and practices of the present invention that are discussed and described below, in conjunction with the attached drawing figures, are provided by way of example, and not by way of limitation. Numerous variations, additions, and other modifications or different implementations of the present invention are possible, and are within the spirit and scope of the invention.

Brief Description of the Drawings

Those skilled in the art will readily understand the present invention based on the following Detailed Description, taken in connection with the attached drawings, in which:

FIG 1 is a schematic diagram depicting general aspects of a conventional computing system that might be found in the prior art.

FIG 2 is a schematic diagram in which a monitor has been connected to selected components of the FIG 1 system.

FIGS. 3A and 3B are flowcharts of techniques according to aspects of the invention for characterizing the performance of a computer system or its components.

FIG 4 is a plot of operating curve data for an exemplary computer system or its components.

FIG 5 is a plot of the behavior curve for the FIG 4 operating curve data.

FIG 6 is a plot of the curvature of the FIG 5 behavior curve.

FIG 7 is a plot of the first derivative of the FIG 6 curvature function.

FIG 8 is a plot identifying the two points on the FIG 7 plot where the derivative of the curvature function is zero.

5 FIG 9 is a plot identifying the values on the x-axis that define the linearity region.

FIG 10 is a plot projecting the linearity region back on the original operating curve shown in FIG 4.

FIG 11 is a generalized block diagram of an embodiment of the present invention.

10 FIG 12A (Prior Art) is a schematic diagram of a conventional workstation or PC (personal computer) digital computing system, on which the present invention may be implemented; or which may form a part of a networked digital computing system on which the present invention may be implemented.

FIG 12B (Prior Art) is a schematic diagram of a networked digital computing system on which the present invention may be implemented.

15 FIG 12C (Prior Art) is a schematic diagram of components of a conventional workstation or PC environment like that depicted in FIG 1.

Detailed Description of the Invention

20 The following description sets forth numerous specific details to provide an understanding of the invention. However, those skilled in the art will appreciate that the invention may be practiced without these specific details. In other instances, well-known methods, procedures, components, protocols, algorithms, and circuits have not been described in detail so as not to obscure the invention. The following discussion describes various aspects of the invention.

The present Detailed Description is organized into the following sections:

- A. Kneepoint Analysis
- B. Examples and Embodiments
- 5 C. Digital Processing Environment in Which the
Invention Can Be Implemented
- D. Conclusion

A. Kneepoint Analysis

10 The concept of a kneepoint, and more particularly its application as in the present invention to monitoring, evaluating and even predicting the performance of computer or computing systems and their components, is a significant and novel aspect of the present invention.

15 The term "kneepoint," as used in the prior art, has been accorded various definitions in various fields of engineering. Some examples from various engineering fields include the following:

(1) The kneepoint or effective point of saturation is defined by ANSI/IEEE Standard as the intersection of the curve with a 45 degree tangent line.

20 (2) The International Electrotechnical Commission (IEC) defines the knee as the intersection of the straight lines from the non-saturated and saturated parts of the exciting curve.

In another example, in the prior art, U.S. Patent No. 5,780,944 defines the detection of an inflection point for the purposes of identifying the current charging status of a battery. The prior art describes finding the first derivative of the battery's characteristic curve.

25 The present invention differs from the foregoing prior art examples, however, in a number of substantive and significant aspects. Unlike the prior art, methods, systems, devices, and computer software/program code products in accordance with the present invention look for a point of maximum curvature on a characteristic function generated from measurements of a computing system or its components.

The methods of the present invention are novel and unique to computer performance, and further, do not use the first derivative of the original operating curve, but instead use the curvature of the logarithm of the original operating curve.

In the field of mathematics the concepts of curvature are well known from ancient
5 times and developed by many prominent mathematicians including Sir Isaac Newton and
Gottfried Wilhelm Leibniz. However, these concepts of curvature have not heretofore been
applied to analysis of computer system performance.

As applied to the context of a conventional computer system performance analysis, the
present invention introduces a kneepoint and a kneepoint analysis for the first time, and does
10 so in a novel manner that represents a significant performance advantage over the prior art. It
will be understood that the present invention can be used in connection with otherwise
conventional computer systems; information about those systems can be obtained using
discovery techniques, such as those disclosed, for example, in United States Patent
Application Serial No. 11/773,825, filed July 5, 2007 (Attorney Docket: AKR-110-US),
15 entitled "Managing Application System Load," and United States Patent Application Serial
No. 11/961,186, filed December 20, 2007 (Attorney Docket: AKR-114-US), entitled "Methods
and Systems for Identifying Application System Storage Resources," both of which are
incorporated by reference herein in their entirety.

In contrast to the use of the term "kneepoint" in the prior art, wherein the term is used
20 to indicate a point where a system, object, substance, or the like undergoes a critical change
resulting in a dramatic change in behavior, properties or appearance of the complete system,
object, or substance (e.g., the vulcanization of rubber upon reaching a particular kneepoint
temperature), the present invention defines a kneepoint more specifically to the performance
of a computer system or its components, and we consider curvature as a way to identify
25 changes in computer performance.

As discussed in greater detail below, definition of such a kneepoint for underlying
computer systems or their components has the advantage of simplifying performance
evaluation and prediction, by reducing analysis to a simple comparison of utilization of the
computer system or its components vs. the underlying kneepoint.

Such a kneepoint then becomes a characteristic property of a computer system or its components, and can be calculated in the same, identical, straightforward, finite-time manner for substantially any computer system or its components.

We also define herein an operating curve of a computer system or computer system component, which is a plot of data points that show the response time of the computer system or computer system component as a function of the utilization of the computer system or computer system component. The utilization can also be referred to as "load."

B. Embodiments and Examples

Referring now to FIG 4, plot 250 presents a typical operating curve of an individual computer system component or combination of computer system components. Most operating curves follow some form of common queuing discipline (e.g., the known M/M/1, M/D/1, M/G/1), though the present invention does not depend upon any particular queuing discipline and can more generally be applied to any performance curve that compares response time to load.

In accordance with the present invention, the operating curve is analyzed using kneepoint analysis to identify the kneepoint of the computer system and/or computer system component being analyzed. In the example of FIG 4, the horizontal axis denotes the normalized utilization of a particular computer system or its components (e.g., bandwidth for a network, percent of the maximum transfer rate for a disk). The vertical axis denotes some measure of response time, queue time, or performance. (Those skilled in the art will understand that of course, a different arrangement of axes could be used in a practice of the present invention.)

Thus, the present invention provides a new definition of a kneepoint (KP) and kneepoint analysis (KPA).

The following illustrates, in accordance with aspects of the present invention, how to apply KPA to characterize the performance of a computer system or its components when processing a fixed amount of work (input) and producing some discrete amount of output.

As shown in FIG 3A, a method 200 according to the present invention comprises the following:

201: Measure utilization of one or more computing system components or groups of components to generate utilization information (wherein the utilization may vary over time);

5 202: Generate, using utilization information for the computing system, an operating curve characteristic of the computing system;

203: Determine kneepoint of operating curve, using a set of kneepoint parameters, wherein the kneepoint parameters are calculated in closed analytical form for a linear operating region of the logarithm of the operating curve (wherein the linear operating region is a function solely of a performance queuing model representing one or more aspects of the performance of the computing system); (wherein the linear operating region is calculated in a closed analytical form); (wherein the set of kneepoint parameters comprises curvature of the logarithm of the operating curve in the linear operating region, and wherein bounds of the linear operating region comprise two points on the x-axis of operating curve);

15 204: Determine at least one utilization value on the operating curve to be an x-coordinate of a kneepoint of the operating curve, the kneepoint being a unique property of the operating curve and being substantially constant for a performance queuing model representing one or more aspects of the performance of the computing system;

205: Compare given utilization value to the x-coordinate of the kneepoint (evaluate system element performance by comparing system element utilization measurements with at least one operating curve kneepoint);

206: Set a bound based on the x-coordinate of the kneepoint;

207: Indicate alarm condition when the given utilization value is above the bound based on the x-coordinate of the kneepoint.

25 As shown in FIG 3B, a further method 220 according to the present invention comprises the following:

221: Obtain actual performance measurement data points of a computer system or computer system component(s) (e.g., CPU, disk, network) that capture response time as a function of utilization.

30 a) Utilization in this context is the amount of input in measured units, normalized by the maximum permitted or possible input.

b) Response time is the amount of time elapsed, processing the fixed amount of input mentioned above in preferred units of time.

222: It is assumed that the output is a function of the input, and satisfies the following constraints:

- 5 a) If the rate of input is zero, then the rate of output is also zero.
b) The rate of output is monotonic as a function of input.
c) Output has a vertical asymptote at the maximum permitted input value mentioned above.

223: Convert the raw data points that represent the response time (plotted, for
10 example, on the vertical axis) as a function of utilization or load (plotted, for example, on the horizontal axis) to a continuous mathematical function using a selected curve fitting model (e.g., least-squares curve fitting). Curve fitting techniques are well known, and the choice of curve fitting model, whether least-squares curve fitting or other, is left to the implementer. We will refer to this fitted curve as the operating curve for the computer system or computer
15 system component(s).

224: Compute the logarithm of the operating system curve. We will refer to this new curve as the behavior curve. Note that the operating curve begins at zero and passes to infinity at a vertical asymptote. This implies that behavior curve monotonically rises from negative infinity to positive infinity and so its derivative goes from positive infinity to positive infinity
20 and thus has only one minimum in between.

225: Calculate the curvature function of the behavior curve. A curvature function of a plane curve is well defined in mathematics. Because the behavior curve possesses a concave-upwards region on the left half of the curve and a concave-downward region on the right half of the curve, the curvature is assured to possess maximal negative and maximal positive
25 curvature points.

226: Find two points on the behavior curve of maximal curvature. We can compute this by taking the first derivative of the curvature and identify zero values (i.e., inflection points). We can use these points to define the endpoints of the linearity region.

227: Project these two points back onto the operating (and behavior) curve. These two
30 points define the span of the operating (behavior) curve linearity region.

228: Find the best fitting exponential function for operating curve over the operating curve linearity region using a selected curve fitting algorithm (e.g., least-squares).

229: Find rightmost point of intersection between the operating curve and the best-fitted exponential function found in the previous.

5 230: Project this point back onto the x-axis. This point uniquely defines the kneepoint of the operating curve for a computer system or computer system component. The respective value on the y-axis is the operation response time for this kneepoint.

FIG 5 is a plot 300 of the behavior curve for the original operating curve data 250 shown in FIG 4, in accordance with an aspect of the invention. As can be seen by a review of
10 FIG 5, the curve exhibits strong linearity in the middle of the curve, as well as nonlinearity on the left and right portions of the curve. A wide range of classical queuing models, as well as observed real life queuing functions, exhibit this same behavior, though the graphs may be scaled.

FIG 6 is a plot 350 of the curvature of the behavior curve.

15 FIG 7 is a plot 400 of the first derivative of the curvature function.

FIG 8 is a plot 450 identifying the two points on the FIG 7 plot 400 where the derivative of the curvature function is zero.

FIG 9 is a plot 500 identifying the values on the x-axis that define the linearity region.

FIG 10 is a plot 550 projecting the linearity region back on the original operating
20 curve 250 presented in FIG 4. FIG 10 also fits a best-fit exponential curve to the points in the linearity region. The rightmost intercept of the best-fit exponential curve and the operating curve identifies the kneepoint of the operating curve.

FIG 10 further projects the kneepoint onto the x-axis to identify the kneepoint
25 utilization. A similar projection can be applied to the y-axis to identify the response time kneepoint.

It will be seen that these aspects track the method aspects of the flowchart of FIG 3.

Using these techniques of the present invention, it will be seen that the invention can be implemented within, or in conjunction with, an otherwise conventional computer system, to provide, for example, a method of determining whether the performance of a computing
30

system is likely to scale exponentially or faster than exponentially with respect to increased system load. Such a method comprises the following:

measuring utilization of one or more computing system components or groups of components to generate utilization information, wherein the utilization may vary over time;

5 generating, using the utilization information, an operating curve characteristics of the computing system;

determining at least one utilization value on the operating curve to be an x-coordinate of a kneepoint of the operating curve, the kneepoint being a unique property of the operating curve and being substantially constant for a performance queuing model representing one or
10 more aspects of the performance of the computing system;

comparing a given utilization value to the x-coordinate of the kneepoint;

setting a bound based on the x-coordinate of the kneepoint; and

indicating an alarm condition when the given utilization value is above the bound based on the x-coordinate of the kneepoint.

15 Also in accordance with the invention, a kneepoint can be determined using a set of kneepoint parameters, which can be calculated in closed analytical form for the linear operating region of the logarithm of the operating curve.

In accordance with these aspects of the present invention, the linear operating region is a function solely of the queuing model; and the linear operating region is calculated in a
20 closed analytical form. The set of kneepoint parameters comprises curvature of the logarithm of the operating curve in the linear operating region, and bounds of the linear operating region comprise two points on the x-axis of operating curve.

Methods in accordance with the invention can also comprise evaluating system element performance by comparing system element utilization measurements with at least one
25 operating curve kneepoint.

FIG 11 is a generalized block diagram of one embodiment of the present invention. As shown in FIG 11, a kneepoint analysis module 601 can be part of a performance monitoring module 130 (FIG 2) within a digital computing system; or alternatively, it can be a separate element. As also shown in FIG 11, the kneepoint analysis module 601 can include
30 hardware or software modules in electronic communication with each other for

executing functions necessary for kneepoint analysis in accordance with the present invention, including:

Module 610 for measuring utilization of one or more computing system components or groups of components to generate utilization information (utilization may vary over time);

5 Module 611 for generating, using the utilization information, an operating curve characteristics of the computing system;

Module 612 for determining at least one utilization value on the operating curve to be an x-coordinate of a kneepoint of the operating curve, the kneepoint being a unique property of the operating curve and being substantially constant for a performance queuing model
10 representing one or more aspects of the performance of the computing system;

Module 613 for comparing a given utilization value to the x-coordinate of the kneepoint;

Module 614 for setting a bound based on the x-coordinate of the kneepoint; and

Module 615 for indicating an alarm condition when the given utilization value is above
15 the bound based on the x-coordinate of the kneepoint.

Those skilled in the art will appreciate that these modules can be implemented in computer software in a general purpose digital computer, or in other contexts.

20 C. Digital Processing Environment in Which the Invention
 Can Be Implemented

Having described the present invention, the following is a discussion, to be read in connection with FIGS. 12A-C, of digital processing structures and environments in which the invention may be implemented and practiced.

25 It will be understood by those skilled in the art that the present invention provides methods, systems, devices and computer program products that enable more efficient application execution on applications commonly found on compute server class systems. These applications include database, web-server and email-server applications. These applications are commonly used to support a medium to large group of computer users
30 simultaneously. These applications provide coherent and organized access and sharing by multiple users to a shared set of data. The applications can be hosted on multiple or a single shared set of digital computing systems. The set of tasks carried out on each application

dictates the patterns and loads generated on the digital computing system, which can be managed through a set of configurable application parameters.

The present invention can thus be implemented in a number of different contexts including, but not limited to, a separate software application, part of the computer system operating system software, or the like. The present invention may be implemented as a separate, stand-alone system. The implementation may include user interface elements such as a keyboard and/or mouse, memory, storage, and other conventional user-interface components. While conventional components of such kind are well known to those skilled in the art, and thus need not be described in great detail herein, the following overview indicates how the present invention can be implemented in conjunction with such components in a digital computer system.

More, particularly, those skilled in the art will understand that the present invention can be utilized in the profiling and analysis of digital computer system performance and application tuning. The techniques described herein can be practiced as part of a digital computer system, in which performance data is periodically collected and analyzed adaptively, or in other like contexts.

The above detailed description illustrates examples of methods, structures, systems, and computer software products in accordance with these techniques. It will be understood by those skilled in the art that the described methods and systems can be implemented in software, hardware, or a combination of software and hardware, using conventional computer apparatus such as a personal computer (PC) or an equivalent device operating in accordance with (or emulating) a conventional operating system such as Microsoft Windows, Linux, or Unix, either in a standalone configuration or across a network. The various processing aspects and means described herein may therefore be implemented in the software and/or hardware elements of a properly configured digital processing device or network of devices. Processing may be performed sequentially or in parallel, and may be implemented using special purpose or re-configurable hardware.

As an example, FIG 12A attached hereto depicts an illustrative computer system 1010 that can run server-class applications such as databases and mail-servers. With reference to FIG 12A, the computer system 1010 in one embodiment includes a processor module 1011 and operator interface elements comprising operator input components such as a keyboard

1012A and/or a mouse 1012B (or digitizing tablet or other analogous element(s), generally identified as operator input element(s) 1012) and an operator output element such as a video display device 1013. The illustrative computer system 1010 can be of a conventional stored-program computer architecture. The processor module 1011 can include, for example, one or
5 more processor, memory and mass storage devices, such as disk and/or tape storage elements (not separately shown), which perform processing and storage operations in connection with digital data provided thereto. The operator input element(s) 1012 can be provided to permit an operator to input information for processing. The video display device 1013 can be provided to display output information generated by the processor module 1011 on a screen 1014 to the
10 operator, including data that the operator may input for processing, information that the operator may input to control processing, as well as information generated during processing. The processor module 11 can generate information for display by the video display device 13 using a so-called “graphical user interface” (“GUI”), in which information for various applications programs is displayed using various “windows.”

15 The terms “memory,” “storage” and “disk storage devices” can encompass any computer readable medium, such as a computer hard disk, computer floppy disk, computer-readable flash drive, computer-readable RAM or ROM element or any other known means of encoding digital information. The term “applications programs,” “applications,” “programs,” “computer program product” or “computer software product” can encompass any computer
20 program product consisting of computer-readable programs instructions encoded and/or stored on a computer readable medium, whether that medium is fixed or removable, permanent or erasable, or otherwise. As noted, for example, in block 1122 of the schematic block diagram of FIG 12C, applications and data can be stored on a disk, in RAM, ROM, on other removable or fixed storage, whether internal or external, and can be downloaded or uploaded, in
25 accordance with practices and techniques well known in the art. As will also be noted in this document, the present invention can take the form of software or a computer program product stored on a computer-readable medium, or it can be in the form of computer program code that can be uploaded or downloaded, or it can take the form of a method or a system for carrying out such a method. Although the computer system 1010 is shown as comprising particular
30 components, such as the keyboard 1012A and mouse 1012B for receiving input information from an operator, and a video display device 1013 for displaying output information to the

operator, it will be appreciated that the computer system 1010 may include a variety of components in addition to or instead of those depicted in FIG. 12A.

In addition, the processor module 1011 can include one or more network ports, generally identified by reference numeral 1014, which are connected to communication links
5 which connect the computer system 1010 in a computer network. The network ports enable the computer system 1010 to transmit information to, and receive information from, other computer systems and other devices in the network. In a typical network organized according to, for example, the client-server paradigm, certain computer systems in the network are designated as servers, which store data and programs (generally, "information") for processing
10 by the other, client computer systems, thereby to enable the client computer systems to conveniently share the information. A client computer system which needs access to information maintained by a particular server will enable the server to download the information to it over the network. After processing the data, the client computer system may also return the processed data to the server for storage. In addition to computer systems
15 (including the above-described servers and clients), a network may also include, for example, printers and facsimile devices, digital audio or video storage and distribution devices, and the like, which may be shared among the various computer systems connected in the network. The communication links interconnecting the computer systems in the network may, as is conventional, comprise any convenient information-carrying medium, including wires, optical
20 fibers or other media for carrying signals among the computer systems. Computer systems transfer information over the network by means of messages transferred over the communication links, with each message including information and an identifier identifying the device to receive the message.

In addition to the computer system 1010 shown in the drawings, methods, devices or
25 software products in accordance with the present invention can operate on any of a wide range of conventional computing devices and systems, such as those depicted by way of example in FIGS. 12B and 12C (e.g., network system 1100), whether standalone, networked, portable or fixed, including conventional PCs 1102, laptops 1104, handheld or mobile computers 1106, or across the Internet or other networks 1108, which may in turn include servers 1110 and
30 storage 1112.

In line with conventional computer software and hardware practice, a software application configured in accordance with the invention can operate within, e.g., a PC 1102 like that shown in FIGS. 12A-C, in which program instructions can be read from ROM or CD ROM 1116 (FIG 12C), magnetic disk or other storage 1120 and loaded into RAM 1114 for execution by CPU 1118. Data can be input into the system via any known device or means, including a conventional keyboard, scanner, mouse, digitizing tablet, or other elements 1103. As shown in FIG 12C, the depicted storage 1120 includes removable storage. As further shown in FIG 12C, applications and data 1122 can be located on some or all of fixed or removable storage or ROM, or downloaded.

Those skilled in the art will also understand that method aspects of the present invention can be carried out within commercially available digital processing systems, such as workstations and personal computers (PCs), operating under the collective command of the workstation or PC's operating system and a computer program product configured in accordance with the present invention. The term "computer program product" can encompass any set of computer-readable programs instructions encoded on a computer readable medium. A computer readable medium can encompass any form of computer readable element, including, but not limited to, a computer hard disk, computer floppy disk, computer-readable flash drive, computer-readable RAM or ROM element, or any other known means of encoding, storing or providing digital information, whether local to or remote from the workstation, PC or other digital processing device or system. Various forms of computer readable elements and media are well known in the computing arts, and their selection is left to the implementer.

D. Conclusion

Those skilled in the art will appreciate that while the foregoing detailed description provides sufficient detail to enable one skilled in the art to practice the present invention, the various examples, embodiments and practices of the present invention that are discussed and described herein, in conjunction with the attached drawing figures, are provided by way of example, and not by way of limitation. Numerous variations, additions, and other modifications or different implementations of the present invention are possible, and are within the spirit and scope of the invention.

We claim:

1. A method of determining whether the performance of a computing system is likely to scale exponentially or faster than exponentially with respect to increased system load, the method comprising:

5 measuring utilization of one or more computing system components or groups of components to generate utilization information, wherein the utilization may vary over time; generating, using the utilization information, an operating curve characteristics of the computing system;

10 determining at least one utilization value on the operating curve to be an x-coordinate of a kneepoint of the operating curve, the kneepoint being a unique property of the operating curve and being substantially constant for a performance queuing model representing one or more aspects of the performance of the computing system;

 comparing a given utilization value to the x-coordinate of the kneepoint;

 setting a bound based on the x-coordinate of the kneepoint; and

15 indicating an alarm condition when the given utilization value is above the bound based on the x-coordinate of the kneepoint.

2. A method in accordance with claim 1 further wherein a kneepoint is determined using a set of kneepoint parameters, and wherein the kneepoint parameters are calculated in closed analytical form for the linear operating region of the logarithm of the operating curve.

20 3. A method in accordance with claim 2 wherein the linear operating region is a function solely of the queuing model.

4. A method in accordance with claim 3 wherein the linear operating region is calculated in a closed analytical form.

25 5. A method in accordance with claim 2 wherein the set of kneepoint parameters comprises curvature of the logarithm of the operating curve in the linear operating region, and wherein bounds of the linear operating region comprise two points on the x-axis of operating curve.

6. A method in accordance with claims 1, 2 or 4 further comprising:

30 evaluating system element performance by comparing system element utilization measurements with at least one operating curve kneepoint.

7. A system for determining whether the performance of a computing system is likely to scale exponentially or faster than exponentially with respect to increased system load, the system comprising:

5 means for measuring utilization of one or more computing system components or groups of components to generate utilization information, wherein the utilization may vary over time;

means for generating, using the utilization information, an operating curve characteristics of the computing system;

10 means for determining at least one utilization value on the operating curve to be an x-coordinate of a kneepoint of the operating curve, the kneepoint being a unique property of the operating curve and being substantially constant for a performance queuing model representing one or more aspects of the performance of the computing system;

means for comparing a given utilization value to the x-coordinate of the kneepoint;

means for setting a bound based on the x-coordinate of the kneepoint; and

15 means for indicating an alarm condition when the given utilization value is above the bound based on the x-coordinate of the kneepoint.

8. A system in accordance with claim 7 further wherein a kneepoint is determined using a set of kneepoint parameters, and wherein the kneepoint parameters are calculated in closed analytical form for the linear operating region of the logarithm of the operating curve.

20 9. A system in accordance with claim 8 wherein the linear operating region is a function solely of the queuing model.

10. A system in accordance with claim 9 wherein the linear operating region is calculated in a closed analytical form.

25 11. A system in accordance with claim 8 wherein the set of kneepoint parameters comprises curvature of the logarithm of the operating curve in the linear operating region, and wherein bounds of the linear operating region comprise two points on the x-axis of operating curve.

12. A system in accordance with claims 7, 8 or 10 further comprising:

30 means for evaluating system element performance by comparing system element utilization measurements with at least one operating curve kneepoint.

13. A computer program product comprising comprising computer-readable program instructions encoded on a computer-readable medium and executable by a computer to enable the computer to determine whether the performance of a computing system is likely to scale exponentially or faster than exponentially with respect to increased system load, the computer program product comprising:

- 5
- A. first computer-readable program instructions, encoded on the computer-readable medium and executable by the computer to enable the computer to measure utilization of one or more computing system components or groups of components to generate utilization information, wherein the utilization may vary over time;
 - 10 B. second computer-readable program instructions, encoded on the computer-readable medium and executable by the computer to enable the computer to generate, using the utilization information, an operating curve characteristics of the computing system;
 - C. third computer-readable program instructions, encoded on the computer-readable medium and executable by the computer to enable the computer to determine at least one
15 utilization value on the operating curve to be an x-coordinate of a kneepoint of the operating curve, the kneepoint being a unique property of the operating curve and being substantially constant for a performance queuing model representing one or more aspects of the performance of the computing system;
 - D. fourth computer-readable program instructions, encoded on the computer-readable medium and executable by the computer to enable the computer to compare a given
20 utilization value to the x-coordinate of the kneepoint;
 - E. fifth computer-readable program instructions, encoded on the computer-readable medium and executable by the computer to enable the computer to set a bound based on the x-coordinate of the kneepoint; and
 - 25 F. sixth computer-readable program instructions, encoded on the computer-readable medium and executable by the computer to enable the computer to indicate an alarm condition when the given utilization value is above the bound based on the x-coordinate of the kneepoint.

14. The computer program product of claim 13 further wherein a kneepoint is
30 determined using a set of kneepoint parameters, and wherein the kneepoint parameters are

calculated in closed analytical form for the linear operating region of the logarithm of the operating curve.

15. The computer program product of claim 14 wherein the linear operating region is a function solely of the queuing model.

5 16. The computer program product of claim 15 wherein the linear operating region is calculated in a closed analytical form.

17. The computer program product of claim 14 wherein the set of kneepoint parameters comprises curvature of the logarithm of the operating curve in the linear operating region, and wherein bounds of the linear operating region comprise two points on
10 the x-axis of operating curve.

18. The computer program product of claims 13, 14 or 16 further comprising:
seventh computer-readable program instructions, encoded on the computer-readable medium and executable by the computer to enable the computer to evaluate system element performance by comparing system element utilization measurements with at least one
15 operating curve kneepoint.

19. In a computing system, an improvement for determining whether the performance of a computing system is likely to scale exponentially or faster than exponentially with respect to increased system load, the improvement comprising:
means for measuring utilization of one or more computing system components or
20 groups of components to generate utilization information, wherein the utilization may vary over time;

means for generating, using the utilization information, an operating curve characteristics of the computing system;

means for determining at least one utilization value on the operating curve to be an x-
25 coordinate of a kneepoint of the operating curve, the kneepoint being a unique property of the operating curve and being substantially constant for a performance queuing model representing one or more aspects of the performance of the computing system;

means for comparing a given utilization value to the x-coordinate of the kneepoint;

means for setting a bound based on the x-coordinate of the kneepoint; and

30 means for indicating an alarm condition when the given utilization value is above the bound based on the x-coordinate of the kneepoint.

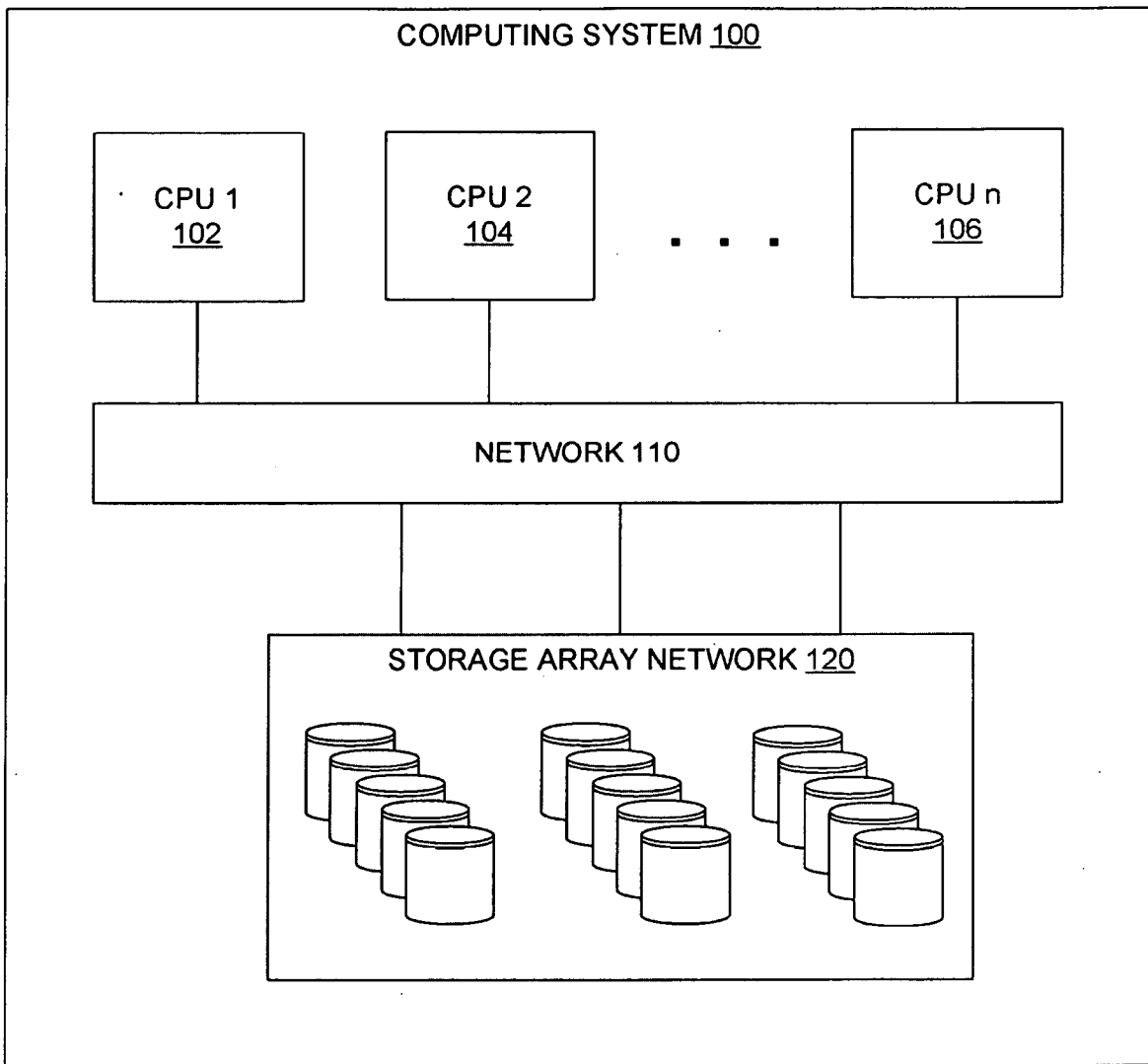


FIG. 1

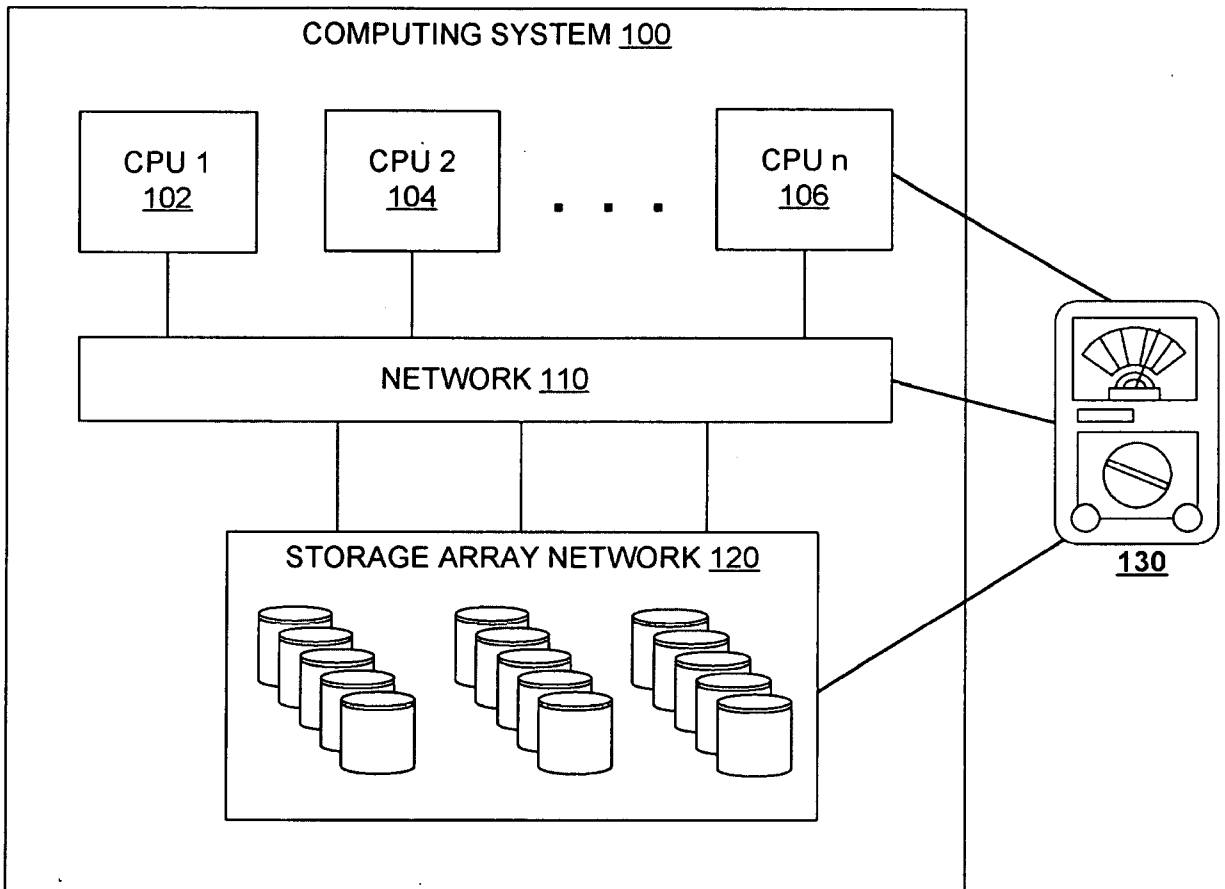


FIG. 2

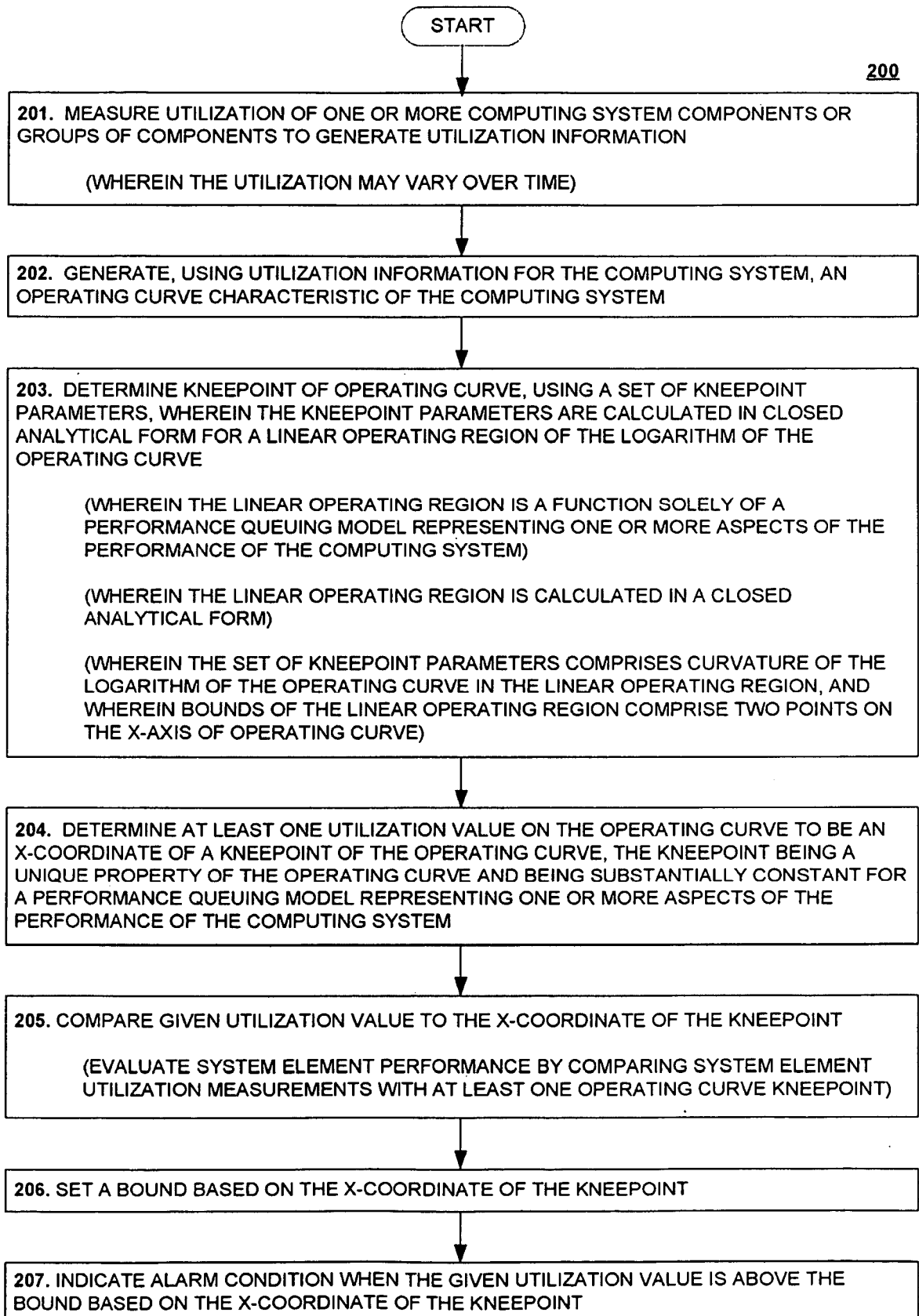


FIG. 3A

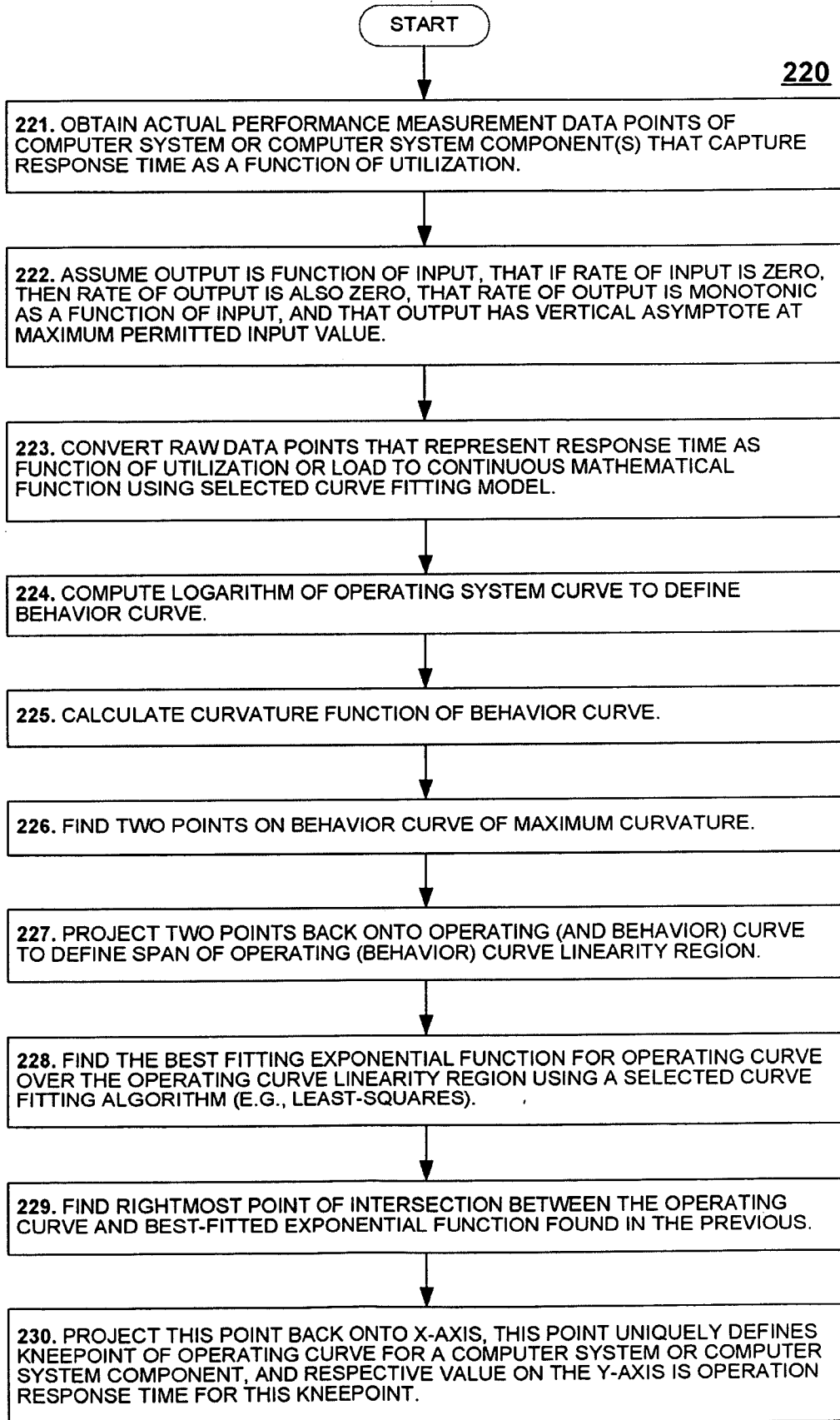


FIG. 3B

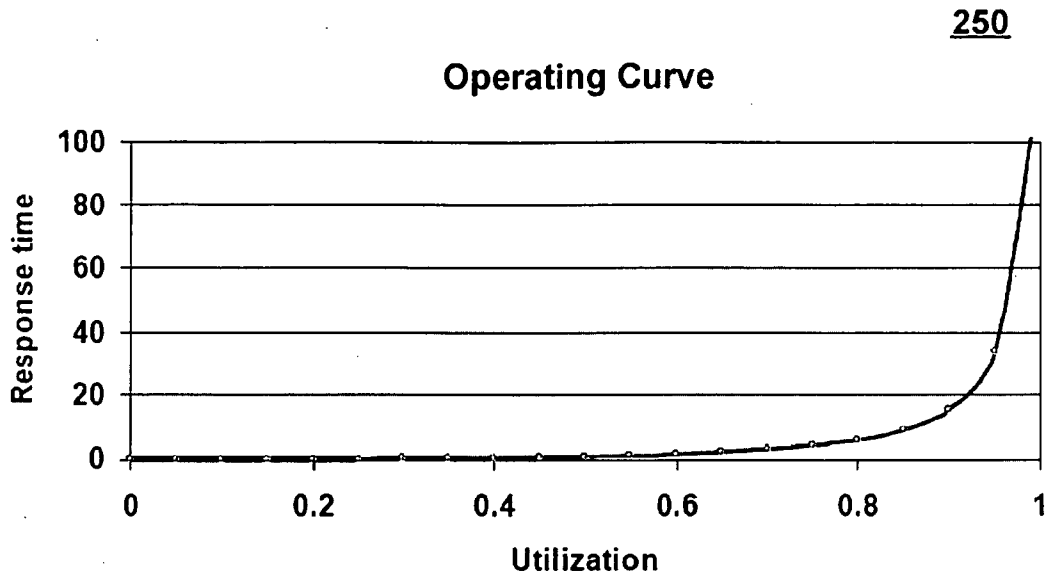


FIG. 4

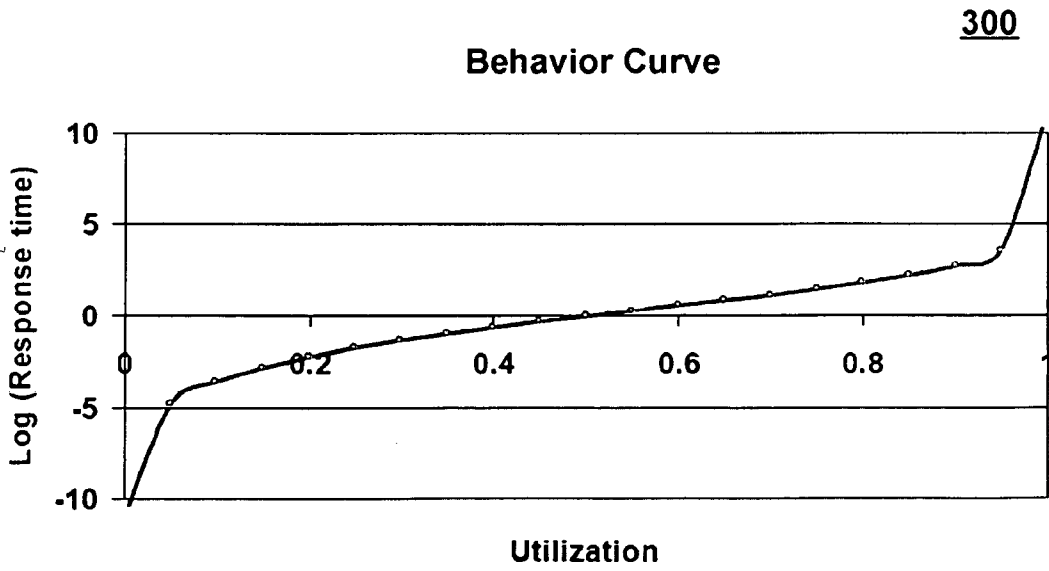


FIG. 5

350

Curvature of the Behavior Curve

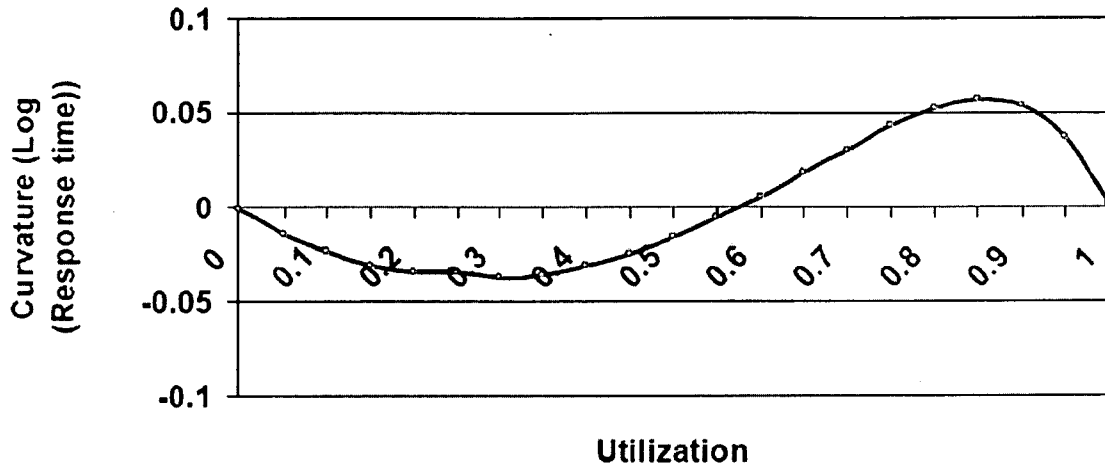


FIG. 6

400

Derivative of Curvature of the Behavior Curve

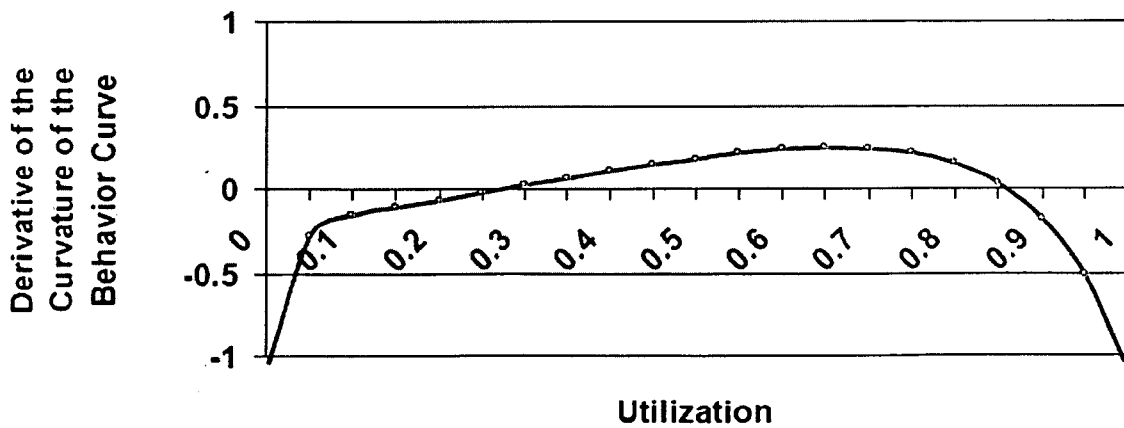


FIG. 7

450

Definition of the End-Points of the Linearity Region

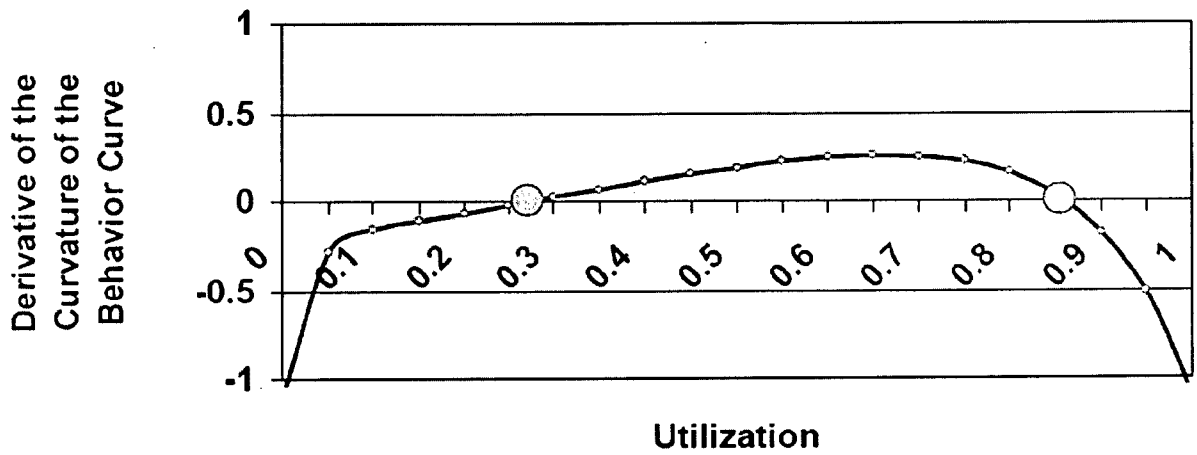


FIG. 8

500

Definition of the Linearity Region

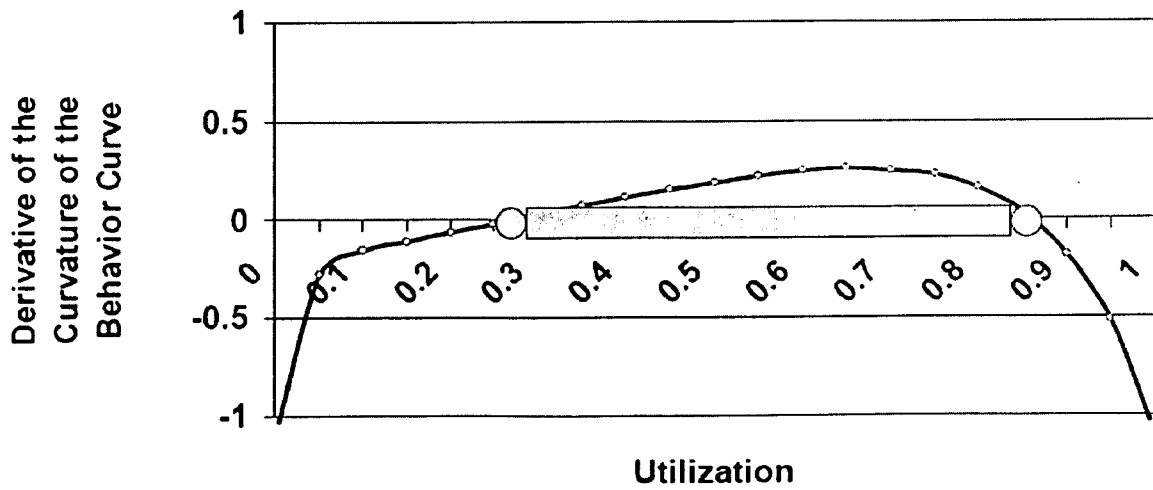


FIG. 9

550

Operating Curve with Fixed Exponent at Endpoints of Linear Region

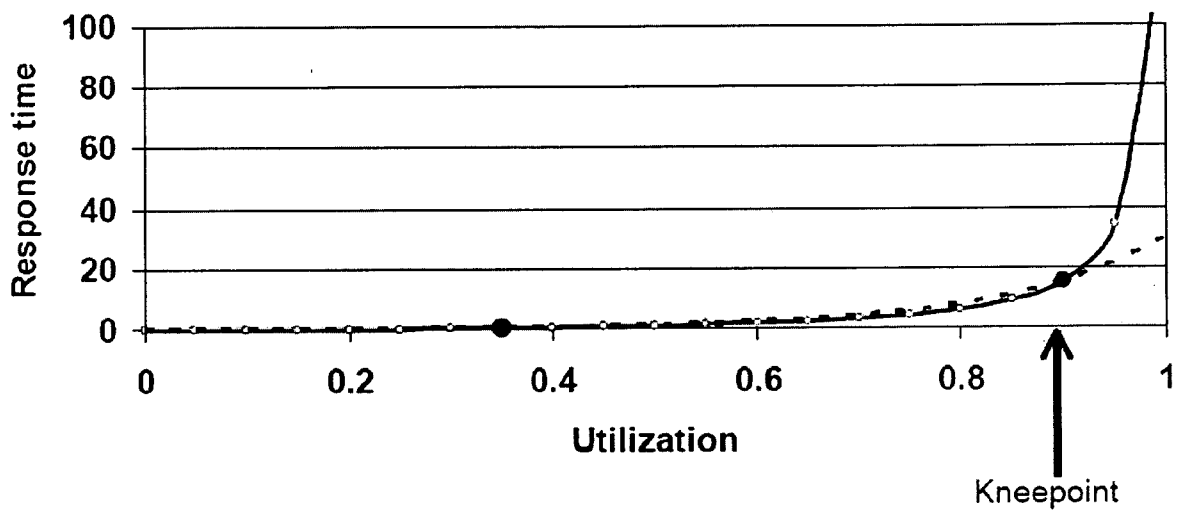


FIG. 10

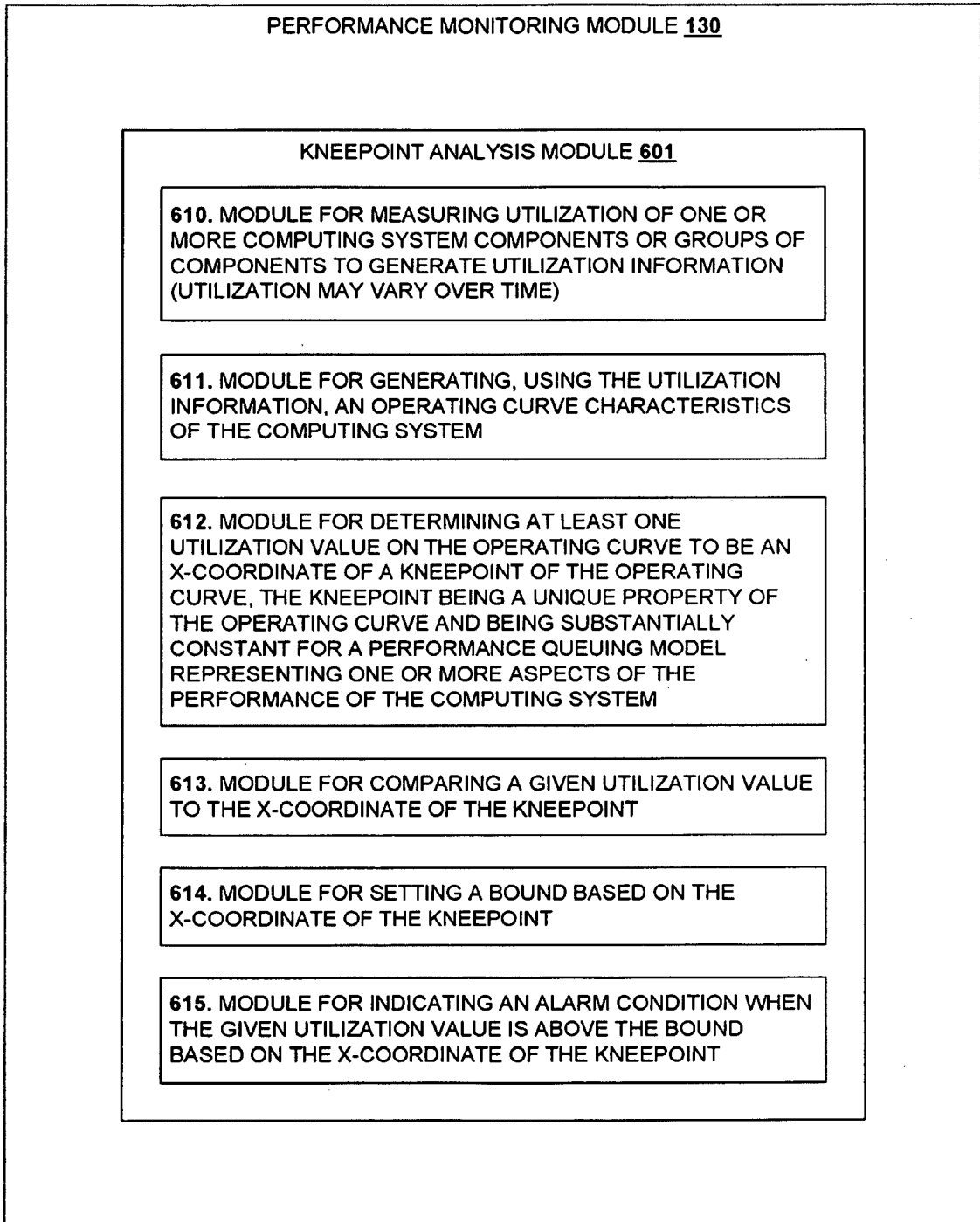


FIG. 11

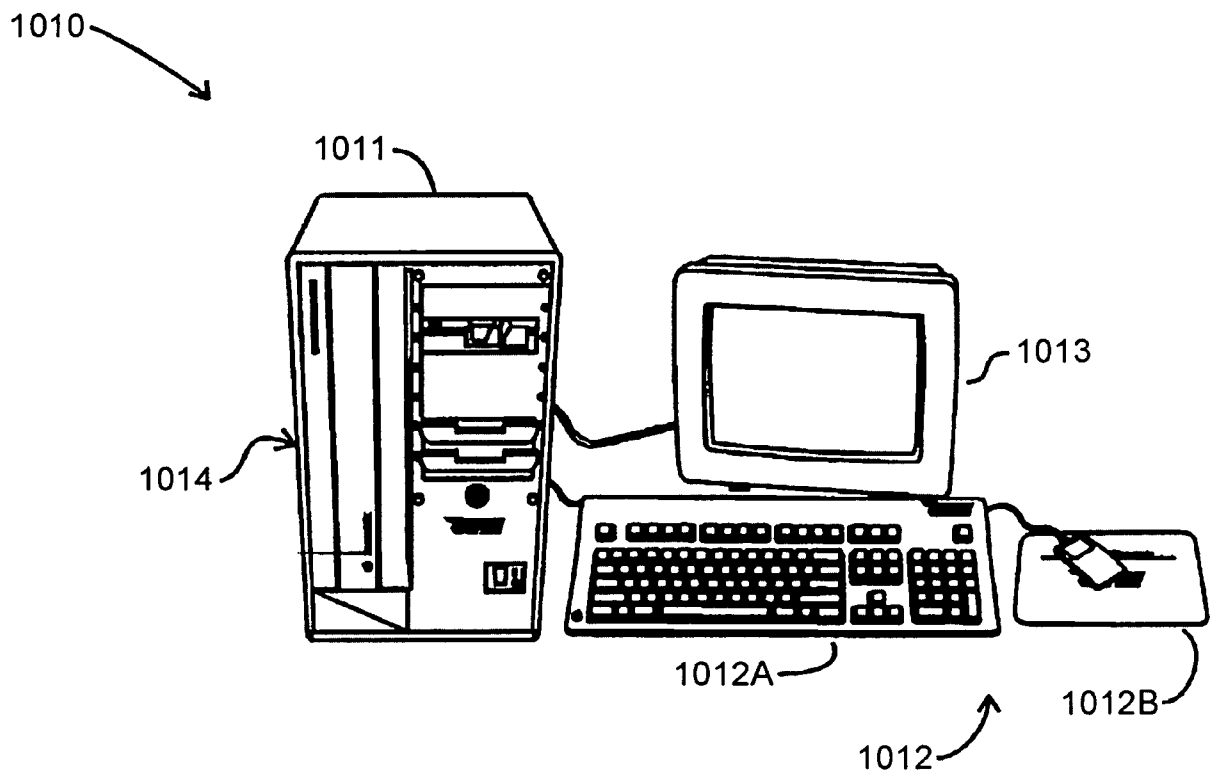


FIG. 12A

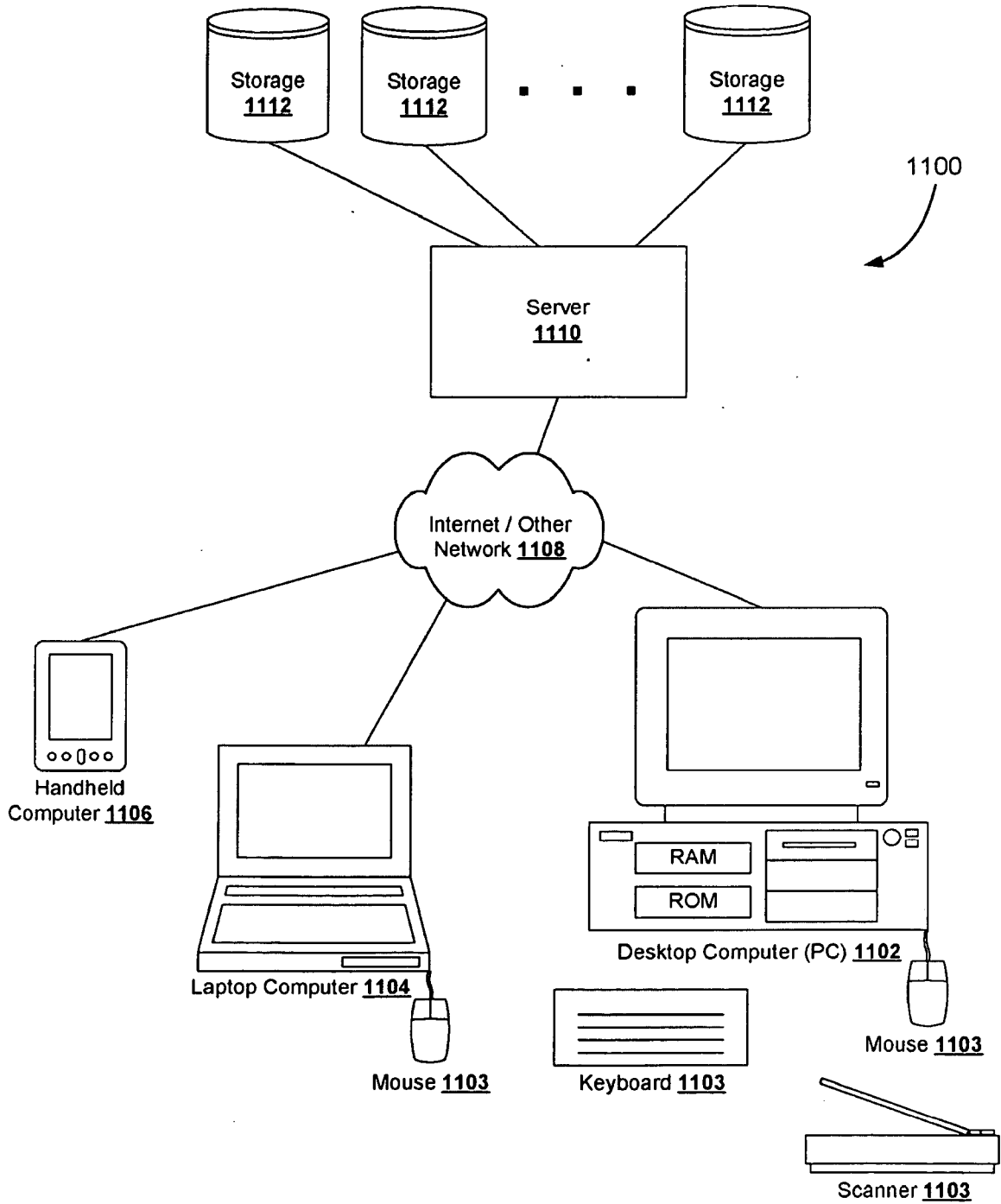


FIG. 12B

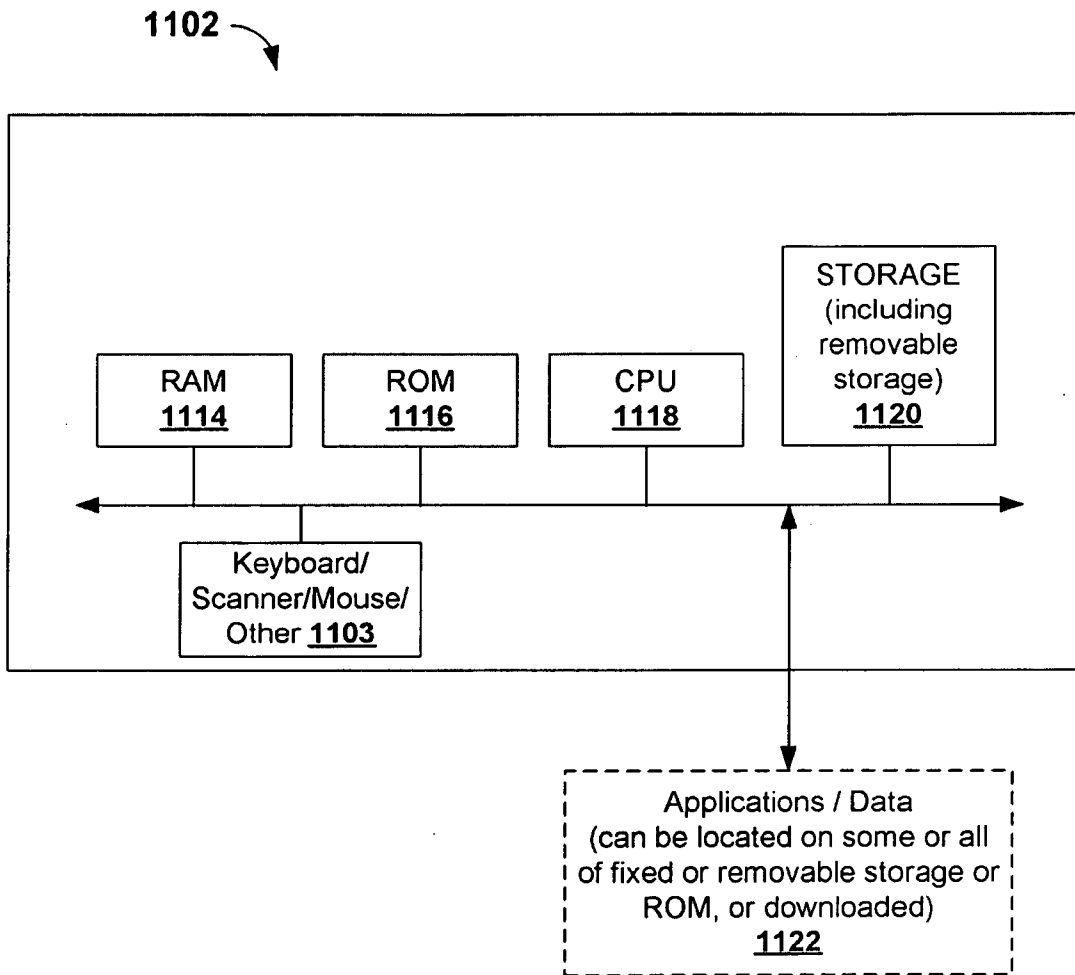


FIG. 12C
(Prior Art)

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US 08/88062

A. CLASSIFICATION OF SUBJECT MATTER IPC(8) - G06F 9/45 (2009.01) USPC - 703/22 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) USPC-703/22 Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched USPC-703/21; 708/100; 708/800 Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) Search Terms Used: computer, utilization information, linear operating region, evaluate/predicting , performance, system load, storage resources, kneepoint (KPA), x coordinate, operating curve, maximum curvature, logarithm, closed analytical form		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,854,754 A (Cabrera, et al.) 29 December 1998 (29.12.1998) Abstract, Fig. 3, Fig. 4, col 1 ln 50-col 2 ln 10, col 7 ln 66-col 8 ln 45, col 9 ln 56- col 10 ln 17,	1, 6, 7, 12, 13, 18, 19
Y	col 14 ln 57-col 16 ln 8	2-5, 8-11, 14-17
Y	US 2004/0122647 A1 (Monroe, et al.) 24 June 2004 (24.06.2004) para [0011], [0020]-[0025], [0027]	2-5, 8-11, 14-17
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/>		
* 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) or which 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 23 March 2009 (23.03.2009)		Date of mailing of the international search report 03 APR 2009
Name and mailing address of the ISA/US Mail Stop PCT, Attn: ISA/US, Commissioner for Patents P.O. Box 1450, Alexandria, Virginia 22313-1450 Facsimile No. 571-273-3201		Authorized officer: Lee W. Young PCT Helpdesk: 571-272-4300 PCT OSP: 571-272-7774