

1. 一种监控软件应用的性能水平的方法,所述软件应用运行在连接至计算机网络的计算设备上,并与一个或多个连接至网络的设备通信,该方法包括:

监控在所述网络上的至少一个站上的信息交换;

测量与所述信息交换相关的至少两个性能指示符度量;以及

从所述指示符度量的非线性组合中推导出一个指示符参数。

2. 根据权利要求1所述的方法,其中所述至少两个性能指示符度量包括延迟、抖动、丢失、响应时间、吞吐量、有效吞吐量和对象大小中的至少两个。

3. 根据前述权利要求任一项所述的方法,其中所述信息交换是信息包的交换。

4. 根据前述权利要求任一项所述的方法,其中所述软件应用是万维网应用、协作应用、游戏应用、语音应用、视频应用、或者机器到机器应用。

5. 根据前述权利要求任一项所述的方法,还包括:

对每一个所述指示符度量运用变换,以获得相应的推导值,其中所述变换如此产生第一区域和第二区域,其中在所述第一区域内所述推导值相对弱地依赖于相应的度量,在所述第二区域内所述推导值相对强地依赖于相应的度量;以及

将至少两个所述推导值相加结合,以获得所述指示符参数。

6. 根据权利要求5所述的方法,其中一个第一所述度量与所述计算设备和一个连接至网络的设备之间的信息交换相关,且第二所述度量与所述计算设备和一个连接至网络的第二设备之间的信息交换相关,其中这两个度量都与同一软件应用相关。

7. 根据权利要求5所述的方法,其中一个第一所述推导值指示静态万维网内容,一个第二所述推导值指示动态万维网内容,其中所述静态万维网内容和所述动态万维网内容由分立的网络缓存或万维网服务器所传递。

8. 根据前述权利要求任一项所述的方法,包括其中所述计算设备包括用于帮助用户输入一个用户输入的分数值的工具,其中所述用户输入的分数值指示所述用户对所述软件应用的性能的感知,且其中所述系统包括用于将所述用户输入的分数与所述推导出的指示符参数进行比较的工具,且可选的,用于依赖于所述用户输入的分数来修正用于推导所述推导出的指示符参数的计算的工具。

9. 根据前述权利要求任一项所述的方法,包括产生关于计算出的指示符参数的可视化图形的步骤。

10. 根据前述权利要求任一项所述的方法,包括依赖于所述指示符参数的值来触发对所述网络的诊断追踪的步骤,以刺激来自所述网络或所述计算基础结构的响应。

11. 根据权利要求10所述的方法,其中通过所述诊断追踪刺激来自所述网络或所述计算基础结构的响应是路由数据、路由性能数据、系统性能数据、CPU数据、或与计算机网络节点或连接至网络的设备相关的其他管理数据或性能数据。

12. 根据前述权利要求任一项所述的方法,其中为运行在所述计算设备上的多个软件应用中的每一个推导一个单独的所述指示符参数。

13. 根据前述权利要求任一项所述的方法,其中通过获得关于应用产生的信息交换的度量,来被动地执行所述监控,或者,其中通过创建一个具体适应于测试所述计算基础结构或所述网络基础结构对一个具体刺激的响应的信息交换,来主动地执行所述监控。

14. 一种诊断和/或预测包括计算机网络的计算机系统中的问题的方法,所述方法包

括从连接至计算机网络的多个计算设备的每一个中,将根据前述权利要求任一项要求的方法推导出的指示符参数收集在数据库中,并从所述收集的参数中校对数据,以诊断和 / 或预测所述计算机系统的问题。

15. 根据权利要求 14 所述的方法,其中所述系统包括用于触发对所述网络的诊断追踪的工具,以依赖于所述指示符参数的值来刺激来自所述网络或所述计算基础结构的响应,且其中收集在数据库中的步骤包括,收集从依赖于所述指示符参数的值而被触发的所述诊断追踪得出的数据的步骤。

监控软件应用的性能的方法

[0001] 本发明涉及对计算机软件应用的监控,更具体地,涉及对联网的软件应用的性能的测量、管理和感知优化。

[0002] 企业越来越依赖软件应用来完成和实现经济目标。企业进程可被分类为那些涉及人的直接参与的进程,例如万维网应用(web application)、语音应用和视频应用;以及那些没有人参与的进程,例如仅有机器的工厂环境。

[0003] 企业对软件应用的依赖意味着,应用性能不佳的影响会是巨大的,尤其是当用户意识到性能不佳,并因为应用妨碍了他们高效完成任务的能力而对该应用感到失望或恼火时。这会具有严重的负面经济影响,降低企业进程效率,打击企业内的员工士气,并降低了客户忠诚度和企业之外的那些品牌价值。因此,企业必须确保各个应用始终如一地执行,并处于符合其支持的内部或外部企业进程的需要的水平,即确保用户不会对其使用的应用感到失望。

[0004] 然而,这并不是简单的,因为用户或机器所体验到的应用性能取决于日益复杂的根本的 IT 基础结构。例如,IT 基础结构通常会涉及多种技术,例如固定和移动无线网络、局域网(LAN)、公共和专用广域网(WAN)(互联网和内联网)、数据中心、服务器、缓存、虚拟机、专有和开放通信协议。这些技术常常作为服务通过多个第三方——例如服务提供商(SP)——提供给企业,这会增加 IT 基础结构的管理复杂度。每个技术常要求部署数十、数百或者数千个单个系统。在运行中,每个系统和技术的行为不仅不同,而且根据系统上任意时刻的负荷是动态的。最终,用户所感知到的应用体验是许多交互的单个系统行为的结果。更为复杂的是,不同的应用以不同的方式受到 IT 基础结构系统的组合行为的影响,因此感知也不同。

[0005] 因此,代表企业利益的应用性能管理(APM)的实施需要专业的技术人员以及先进的 APM 工具。目前可提供一系列 APM 工具和技术。一些 APM 工具关注 IT 基础结构技术的子集的性能管理。尽管这对于在系统级别上管理这些技术是有用的,但是这些工具的一个缺陷是,它们常常不能管理支撑一个应用的所有技术,因此不能管理用户所感知到的应用性能。因此,一些 APM 工具关注应用的整体的、端到端的性能。这些工具的方法要么是主动的,要么是被动的。

[0006] 主动工具能够模拟用户应用体验,例如通过执行模拟用户如何使用一个应用的脚本。通常,脚本在连接至企业的 IT 基础结构的工作站上执行。在模拟期间执行测量,为 APM 专家提供数据,帮助他们了解应用的性能。例如,可以测量应用的响应时间。

[0007] 相反,被动工具使用硬件和软件仪器测量用户对应用的实际使用。仪器点可被放置在 IT 基础结构中的不同位置,例如在数据中心或其他主要的企业位置。可测量实际的用户通信量,且为 APM 专家提供数据,帮助他们了解应用的性能。同样,测量通常描述应用的响应时间。

[0008] 响应时间测量的一个优势是,对于工程师来说响应时间是一个用来理解的直观概念。然而,响应时间测量的一个明显缺陷是,它不是一个对用户所感知到的性能的可靠指示,且不能指示用户是否对应用失望或者其他。这个感知的差异起源于各个用户以不同的

方式使用同一应用,或者对同一应用有不同的性能预期。因此用户对性能具有独特的个人感知。类似地,由于用户使用不同的应用来实现不同的任务,因此用户可能不会因为一个应用的响应时间慢而失望,但是可能会因为类似的另一个应用的响应慢而失望,尤其是如果任务的紧急性或重要性发生变化时。

[0009] 简单的响应时间测量的另一缺陷是,它不能解释是什么 IT 基础结构行为或系统行为组合导致性能降低,或者不能解释什么行为或者行为的组合会在未来导致性能降低。这使得对于 APM 专家来说难于诊断和确定具有不利经济影响(即由于失望的用户)的 IT 基础结构行为。

[0010] 响应时间测量的另一缺陷是,它不能指示 IT 基础结构行为或系统行为的组合导致用户感知到的性能降低的时刻。

[0011] 被动 APM 工具的另一缺陷是,这些工具要求对单个应用使用的协议和编码有详细的理解。这常意味着因有限的 APM 供应商资源,被动 APM 工具不支持所有的应用或应用类型。

[0012] 主动 APM 工具在它们开始运行前通常必须学习脚本或者具有手动产生的脚本。这常意味着因有限的 IT 专家资源,主动 APM 工具不支持所有的应用或应用类型。

[0013] 许多被动 APM 工具的另一缺陷是,它们要求安装和支持物理测量仪器设备。这些设备的购买、安装和运行成本通常较高,限制了它们的部署范围。被动 APM 工具还可存储个人用户数据,这些数据常会产生其他安全或隐私问题,从而导致企业的管理成本更高。

[0014] 根据本发明,提供一种监控软件应用的性能水平的方法,所述软件应用运行在连接至计算机网络的计算设备上,并与一个或多个连接至网络的设备通信,该方法包括:

[0015] 监控在所述网络上的至少一个站上的信息交换,例如信息包;

[0016] 测量与所述信息交换相关的至少两个性能指示符度量;以及

[0017] 从所述指示符度量的非线性组合中推导出一个指示符参数。

[0018] 在本发明的又一方面中,提供一种诊断和/或预测包括计算机网络的计算机系统的问题的方法,所述方法包括从连接至计算机网络的多个计算设备的每一个中,将根据前述的监控方法推导出的指示符参数收集在数据库中,并从所收集的参数中校对数据,以诊断和/或预测所述计算机系统的问题。

[0019] 这样的系统可包括用于触发对所述网络的诊断追踪的工具,依赖于所述指示符参数的值,刺激来自所述网络或所述计算基础结构的响应,且其中收集在数据库中的步骤包括,收集从依赖于所述指示符参数的值而被触发的所述诊断追踪得出的数据的步骤。

[0020] 优选的度量是延迟、抖动、丢失、响应时间、吞吐量、有效吞吐量和对象大小中的一个或多个。

[0021] 本文中使用的术语“软件应用”还包括在连接至网络的设备之间通信的对等应用和机器到机器应用。

[0022] 本发明是基于这样一个认识,即用户对网络应用的性能的感知与容易被测量的性能指示符度量不是以线性方式相关。相反,它反应出对这些度量存在复杂的相关性,这些相关性的形式是非线性的,且通常显示至少第一区域和第二区域,其中在该第一区域所述用户感知相对不受度量变化的影响,而在该第二区域所述用户感知相对强地受到度量变化的影响。相应的,在本发明的优选实施方案中,非线性变换被应用到每个所述指示符度量,以

获得相应的推导值,且所述推导值中的至少两个被相加组合,以获得所述指示符参数。所述非线性变换如此产生第一区域和第二区域,其中在所述第一区域内所述相应的推导值相对弱地依赖于所述相应的度量,在所述第二区域所述相应的推导值相对强地依赖于所述相应的度量。所述变换可以多种方式执行,例如通过具有线性、或非线性分量段的分段数学函数,或通过多元有理函数,所述变换可近似相应的分段数学函数的值。或者,可利用简单的二维查找表。

[0023] 对于每个非线性推导值的一般原则是:存在一个范围的度量值,对于该范围的度量值,对非线性推导值(因此对用户感知)仅具有少量影响;以及另一个范围的值,对于该另一范围的值,用户感知的质量随着度量变化明显降低。还可以有另外一个范围的值,其中感知质量的下降已经是可注意到的,并继续下降,但是没有到严重的程度。在所述范围中还可有一些部分,其中相应的推导值在度量值的该范围的一个子集上对相应的度量非弱依赖或强依赖。这可发生在如果所述软件应用响应于所检测到的减损来调整其操作时,例如如果所述信息丢失超过阈值,启用一种前向纠错机制来保护免受信息丢失。这种调整可提供用户感知的改善,但是仅在有限范围的度量值上,在该范围以外,推导值仍对相应度量弱依赖或强依赖。

[0024] 指示符参数(或计算的用户感知分数),表明用户对所述软件应用的性能的感知,并提供了比单一度量的测量值明显更可靠的用户体验的指示符。现有方法通常仅测量单个参数(通常为响应时间,或吞吐量),测量值对该单个度量(依照用户感知)的重要性通常没有被量化。用户感知分数是一组用户的代理对于他们对所述软件应用的性能的感知的投票。在其一般形式中,代理是非特定的一组投票者。对于机器到机器应用,分数表示位于该设备上的代理用户的感知。

[0025] 形成指示符的其他分量的非线性推导值可具有不同的感知含义,例如与所述软件应用的交互性相关,且可对应于所述软件应用一部分的通信的不同分量,例如从分立的网络缓存或网站服务器传递的静态和动态万维网内容(web content)。

[0026] 计算的指示符参数可以按照定性或定量方式图形可视化。通常,图形指示符被简化为显示或总结用户感知的多个水平(例如“很差”、“差”、“可以”、“好”和“很好”)。用于计算用户感知分数的过程可以调整其参数值,以随着一组投票者的代理而改变分数的专一性。视哪些人正在调整参数以及如何调整,该专一性的范围可从一般的、非特定参数,到特定组,再变化到特定用户。例如,工作站可从用户处接收反馈,指示用户的个人体验是否与显示的图形指示符一致,且响应所述用户输入,自动改变用于计算指示符分量和整个指示符参数(被称作用户感知配置)的计算方法。或者,组织机构中的公司IT团队可负责调整用户感知配置,以将其调节至具体用户组的需要,以及他们对特定软件应用的感知要求。对于机器到机器应用,参数被调整以表示对要求与其他机器通信的机器的进程的影响。

[0027] 指示符参数可被用于触发(直接地或概率上地)对网络和计算基础结构的诊断追踪。这使得能够在用户感知变差的时候(或者根据用户感知正在变差的可能性的比例)检查这些基础结构的行为。现有方法通常在用户感知正在变差被报告和进行诊断调查之间存在相当长的时间延迟。此外,单一的度量测量不能够被用作针对诊断追踪的可靠的触发器,这是因为测量值的重要性没有按照用户感知被量化。

[0028] 相应地,在本发明的一个优选方面,该方法包括如下步骤,即,依赖于所述指示符

参数的值,触发对所述网络的诊断追踪,以刺激来自所述网络或所述计算基础结构的响应。

[0029] 在一个实施方案中,信息交换的监控可在所述计算设备上执行。替代地(或额外地),所述信息交换的监控除了在所述计算设备上执行以外,还可以在连接至网络的设备上执行。

[0030] 与一些软件应用相关的信息交换可在所述设备上被监控,且为每个应用(例如每个网站)提供分立的图形显示以及触发分立的诊断追踪。

[0031] 信息监控可以是被动的(例如,通过测量应用产生的包的往返时间)或主动的(例如,向网络和计算基础结构提供定制要求,以测试对特定激励的响应)。

[0032] 关于许多用户对网络的用户感知体验的数据,可被收集在数据库中,在时间和空间(位置)上校对,并被用于总结、诊断和预测由网络和计算基础结构所导致的感知问题。

[0033] 从用户反馈得到的数据可被收集在数据库中,并被校对以提供用户感知分数的计算配置,这些配置对于不同的应用类型和用户种类而有所变化,从一般到特定。

[0034] 结合附图,下文将详细描述本发明的一些优选实施方案。

[0035] 图1是根据本发明的操作方法的示意图。

[0036] 图2到6示出了适用于将各种指示符度量变换成相应的推导值,以产生性能指示符的变换形式。

[0037] 图1示意性示出了根据本发明的一个优选方法,其中第一步是收集两个或更多个指示符度量(1)。然后将收集的指示符度量在信号处理设备2中以非线性方式(将在下文更详细描述)数学组合,以产生相应的推导值或“指示符分量”(3)。随后将若干个这样的推导值(3)在另一个计算元件(4)中相加组合,以产生指示计算出的用户感知分数的指示符参数(5)。该指示符参数(5)可以由计算元件(6)处理,并在显示元件7上图形显示。

[0038] 反馈(8)可由用户、或者由企业IT团队提供,并被用于经由控制机构(9)变换指示符参数的计算,得到输入参数(10)。数据可被收集在数据库整理(11)中,并被处理以产生输出(12),该输出(12)总结、诊断和预测由网络和计算基础结构导致的感知问题。

[0039] 图2到6图形示出了变换形式,其可被用于将指示符度量有效吞吐量(单位为比特/秒)、丢失(表示为丢失率)、(往返)延迟(秒)、响应时间(秒)和对象大小(比特)变换成相应的推导值,以产生性能指示符。为方便,并为了提供有意义的上下文情景,相应的非线性推导值分别被指定为快速性(celerity)、连续性(contiguity)、相互性(reciprocity)、响应性(responsiveness)和颗粒性(granularity)。概念上而言,这些推导值与下列性能相关的概念有关:

[0040] 快速性:对信息被传递并呈现给用户的速度的感知;

[0041] 连续性:对例如由信息的丢失或延迟传递导致的信息流的中断(或缺失)的感知;

[0042] 相互性:对可由信息的相互交换维持的会话互动的感知;

[0043] 响应性:对软件应用传递可感知的有意义的信息对象(或一组对象)的时间性的感知;

[0044] 颗粒性:对例如通过编解码速率或内容大小所测量的应用对象的详细程度的感知。

[0045] 从用户中心角度来说,这些推导值具有端到端意义,因此通常由在用户的计算设

备上测量的性能指示符度量推导出。然而,推导值还可由测量的具有局部意义的性能指示符度量产生(由此相互联系而获得端到端意义),或者可在信息流的路径上被采样。

[0046] 应用可包括在两个或更多个网络连接设备(例如机器、工作站、服务器)之间的信息流的多个分量,且信息流的每个分量可具有其自身的测量得出的性能指示符度量和感知指示符分量。

[0047] 从图2到6可看出,每个推导值是通过对应度量的非线性变换而获得的。在每一种情形中,图形的形式示出了第一区域和第二区域,在第一区域内,相应的推导值相对弱地依赖于相应的度量,在第二区域内,相应的推导值相对强地依赖于相应的度量。因此,如图2示出的,例如对于在1,000,000到2,000,000bps之间的有效吞吐量值,获得的“快速性”值相对弱地依赖于所测量的有效吞吐量值,但是对于500,000到1,000,000bps之间的有效吞吐量值,获得的“快速性”值相对强地依赖于所测量的有效吞吐量值。同样,如图3示出的,对于0.25到0.5之间的丢失率值,“连续性”值相对弱地依赖于所测量的丢失率,但是对于0.05到0.15之间的丢失率值,“连续性”值相对强地依赖于所测量的丢失率。其他三个推导出的参数示出了对相应的基础度量的类似的非线性相关性。

[0048] 该相关性的性质可通过针对每个度量和相应的推导值进行实验来经验性地确定。随后可通过具有线性或非线性分量段的分段数学函数、或通过多元有理函数来近似所述变换。或者,可利用简单的二维查找表。

[0049] 推导出的指示符参数可以是两个或更多个非线性推导值分量(3)的相加组合。能够通过用户反馈而被修正的参数(10),可被用于改变计算所述推导出的指示符参数的方式,以使得代理更具专一性地适应和预测用户感知分数,例如,针对单独的用户或者用户组,以及针对单独的应用或应用类型。该适应可直接由用户发起,或由用户的代理例如公司IT团队发起。

[0050] 所述用户感知分数能够通过各种各样的方式被利用,通过视觉呈现、报告和诊断(12)。现在将分别基于TCP和UDP通信建立的联网软件应用,来详细描述两个具体实施例。

[0051] 第一实施例是一个网站的例子,其集中了一系列的内容类型和对象大小,并经由TCP与用户的web浏览器通信。用户对网站的感知是由如下因素综合得出的:内容结构和大小、web服务器提供内容的能力、网络基础结构向用户的web浏览器传递内容的能力、以及用户的设备或工作站将内容呈现给用户的能力。

[0052] 给出了所有五个感知贡献类型(3),并利用下列测量的性能指示符度量:有效吞吐量(x_1)、有效丢失(x_2)、往返延迟(x_3)、响应时间(x_4)和对象大小(x_5)。

[0053] 该实施例的数学组合可以如下表示:

[0054]

$$\text{快速性}(x_1) = \sum_{i=0}^n [a_i(x_1 - b_i) + c_i] \cdot \Phi(x_1 - b_i)$$

[0055] 其中

[0056] $a = (04.444 \cdot 10^{-5} \ -3.444 \cdot 10^{-5} \ -10^{-5})$

[0057] $b = (0 \ 100000 \ 1000000 \ 2000000)$

[0058] $c = (-50 \ 0 \ 0 \ 0)$

[0059] $\Phi(\cdot)$ 是海维赛德(Heaviside)阶跃函数

[0060]

$$\text{连续性}(x_2) = \sum_{i=0}^n [a_i \cdot (x_2)^i]$$

[0061] 其中

[0062] $a = (0.4 \ -95 \ -4370 \ 46806 \ -229983 \ 665999 \ -1250167 \ 1592700 \ -1405434 \ 860190-358514 \ 97096 \ -15408 \ 1087)$

[0063]

$$\text{相互性}(x_3) = \sum_{i=0}^n [a_i \cdot (x_3)^i]$$

[0064] 其中

[0065] $a = (-0.1 \ 14.2 \ -1825.6 \ 32304 \ -264056 \ 1159425 \ -3065700 \ 5137225 \ -5508120 \ 3667760-1382893 \ 225760)$

[0066]

$$\text{响应性}(x_4) = \sum_{i=0}^n [[a_i \cdot (x_4 - b_i) + c_i] \cdot \Phi(x_4 - b_i)]$$

[0067] 其中

[0068] $a = (-2.8 \ -11 \ 13.8)$

[0069] $b = (0 \ 1.5 \ 8.45)$

[0070] $c = (0 \ 0 \ 0)$

[0071]

$$\text{颗粒性}(x_5) = \sum_{i=0}^n [[a_i \cdot (x_5 - b_i) + c_i] \cdot \Phi(x_5 - b_i)]$$

[0072] 其中

[0073] $a := (-4 \cdot 10^{-4} \ 3.98 \cdot 10^{-4} \ 2 \cdot 10^{-6})$

[0074] $b := (0 \ 10000 \ 1000000)$

[0075] $c := (0 \ 0 \ 0)$

[0076] 指示符参数——用户感知分数(5)由下式给出：

[0077] $UPS = 100 + \text{快速性}(x_1) + \text{连续性}(x_2) + \text{相互性}(x_3) + \text{响应性}(x_4) + \text{颗粒性}(x_5)$ ，

[0078] 且其被硬限制为 0 到 100 的范围（即如果总和小于 0 则被设置为 0，如果总和超过 100 则被设置为 100）。

[0079] 第二实施例是一个交互的对等应用的例子（在该具体实施例中是一个语音应用，但是相同方法可适用于其他交互的对等应用）。这个应用对消息流或消息信号（可以是连续的或间断的）进行编码，例如使用编解码器对语音消息流进行编码，并将产生的比特流放在一个或多个基于 UDP 的包流（在各方与该交互的应用之间的每个方向上至少一个）中。用户对该交互的应用的感知是由如下因素综合得出的：各方之间实现的比特速率（或者包速率）、以及网络基础结构在该速率下以最小丢失和抖动以及以足够低的往返延迟支持通话交互的传递包流的能力。

[0080] 给出了五个感知贡献类型中的四个（即，使用不可忽略的分子因数），并利用下列测量的性能指示符度量：吞吐量（ x_1 ）、有效丢失（ x_2 ）、往返延迟（ x_3 ）、和对象大小（ x_5 ，应用

每秒钟为每个消息流或消息信号产生的比特数,即比特率)。

[0081] 该实施例的数学组合可以如下表示:

[0082]

$$\text{快速性}(x_1) = \sum_{i=0}^n [[a_i(x_1 - b_i) + c_i] \cdot \Phi(x_1 - b_i)]$$

[0083] 其中

[0084] $a = (0 \ 1.25 \cdot 10^{-3} \ -0.75 \cdot 10^{-3} \ -0.5 \cdot 10^{-3})$

[0085] $b = (0 \ 20000 \ 40000 \ 70000)$

[0086] $c = (-40 \ 0 \ 0 \ 0)$

[0087]

$$\text{连续性}(x_2) = \sum_{i=0}^n [a_i(x_2)^i]$$

[0088] 其中

[0089] $a = (0.4 \ -200 \ -19370 \ 436740 \ -4517742 \ 27542621 \ -108844504 \ 291930000$
 $-542327435698798345 \ -613155445 \ 349600093 \ -116794010 \ 17350734)$

[0090] 其中有效丢失 x_2 是丢失或迟到的包与总计发送的包的比值。

[0091]

$$\text{相互性}(x_3) = \sum_{i=0}^n [a_i(x_3)^i]$$

[0092] 其中

[0093] $a = (-0.29 \ 20.56 \ -766.75 \ 6855.59 \ -29341.33 \ 68709.83 \ -97409.63 \ 87653.78$
 $-50493.4718068.78 \ -3661.87 \ 321.4)$

[0094]

$$\text{颗粒性}(x_5) = \sum_{i=0}^n [[a_i(x_5 - b_i) + c_i] \cdot \Phi(x_5 - b_i)]$$

[0095] 其中

[0096] $a = (0 \ 2.2410^{-4} \ -1.30810^{-4} \ -9.37510^{-5})$

[0097] $b = (0 \ 15000 \ 64000 \ 128000)$

[0098] $c = (-17 \ 0 \ 0 \ 0)$

[0099] 指示符参数——用户感知分数由下式给出:

[0100] $UPS = 100 + \text{快速性}(x_1) + \text{连续性}(x_2) + \text{相互性}(x_3) + \text{颗粒性}(x_5)$,

[0101] 且其被硬限制为 0 到 100 的范围。

[0102] 通过具体示例的方式,使用图 2 到 6 的示例性曲线,根据本发明的一个指示符参数可如下确定:

[0103] 有效吞吐量(测量的)是 750000bps,因此快速性是 -21。

[0104] 经历的丢失率是 2%,因此连续性是 -3。

[0105] 经历的往返延迟是 110ms,因此相互性是 -1.5。

[0106] 响应时间是 0.67s,因此响应性是 -2。

[0107] 下载的是 500000 比特的对象,因此颗粒性是 -5。

[0108] 因此,指示符参数的计算值是: $100-21-3-1.5-2-5 = 67.5$

[0109] 假设现在延迟和丢失增加(例如由于网络堵塞),则这对可获得的有效吞吐量有影响(有效吞吐量减少),因此增加了下载对象的响应时间。

[0110] 新的值是:

[0111] 有效吞吐量减少到 325000bps,因此快速性是 -40。

[0112] 经历的丢失率是 3%,因此连续性是 -5。

[0113] 经历的往返延迟是 220ms,因此相互性是 -2。

[0114] 响应时间是 1.5s,因此响应性是 -4。

[0115] 下载的是 500000 比特的对象,因此颗粒性是 -5。

[0116] 总的感知质量是: $100-40-5-2-4-5 = 44$ 。

[0117] 可注意到,一个通式可被表示为覆盖基于 TCP 和 UDP 的实施例。连续性被表示为丢失和延迟到达(即因为抖动而延迟)的函数,但是在 TCP 的情形中,抖动的贡献可忽略;在基于 UDP 的公式中,响应性被包括在内,但是其他因数可忽略。

[0118] 计算出的用户感知分数可以依据时间绘制,且在合适的时间标度上累积成柱状图的柱,柱的大小对应于用户感知的定性的不同的水平(例如“很差”、“差”、“可以”、“好”和“很好”)。

[0119] 在短的时间标度上(数量级为分钟),可视化使得用户可以提供关于用户感知分数与用户的(或他们的代理的)软件应用的实际体验的一致性的反馈。该反馈可被用于调整用于产生指示符分量的数学组合的因数,从而调整所述分数。根据用户的技术理解水平,可向用户提供此反馈的不同水平的详情。一种选择是为五个类型的指示符分量中的每一个提供独立的比例控制。在最简单的形式中,这通过控制度量轴的比例来实现。例如如果 s_i 是度量 x_i 的比例参数,则该万维网应用的用户感知分数可由下式给出:

[0120] $UPS = 100 + \text{快速性}(x_1, x_1) + \text{连续性}(s_2, x_2) + \text{相互性}(s_3, x_3) + \text{响应性}(s_4, x_4) + \text{颗粒性}(s_5, x_5)$

[0121] 在另一种形式中,每一个指示符分量 y_i 是指示符度量 x_i 的函数 $f_i()$,即

[0122] $y_i = f_i(x_i)$

[0123] 可以通过调整反馈参数—— a_i 和 s_i 来改变,如给出的:

[0124] $y_i = a_i f_i(s_i x_i)$,

[0125] 其中 a_i 是指示符分量 y_i 的附加因数, s_i 是度量 x_i 的比例参数。

[0126] 一个替代方案是,以协调方式从单一“交互性”控制中驱动这些比例或反馈参数控制。这后一种方法使得能够将软件应用在如下范围上进行分类:从高度主动(例如会话式,像语音或游戏应用),到交易式(例如需要充分响应的交互式数据应用),再到被动的(例如流,或者文件传输,其中重点是有足够高的下载速度)。

[0127] 例如,如果 a_i 和 s_i 是指示符分量 y_i 和指示符度量 x_i 的反馈参数, I 是交互性控制,范围从主动($I > 1$)到被动($0 < I < 1$),则反馈参数可被以如下方式表示为交互性控制的函数。对于连续性、相互性和响应性(即 i 分别为 2、3、4),这些可被表示为:

[0128] $a_i = 1 + 0.4 \log(I)$

[0129] $s_i = 0.9I$

[0130] 对于快速性和颗粒性(即 i 分别为 1、5),反馈控制 a_i 和 s_i 可以被如下表示:

[0131] $a_i = 1 + 0.5 \log(I)$

[0132] $s_i = 0.2I$

[0133] 反馈可以从现场或实验室环境中的用户获得。在现场中,减损可以在其发生时观察和被体验到。在实验室中,减损以受控方式引入系统,以降低软件应用的质量。用户组以投票形式反馈他们各自的分数,表明他们在所体验到的减损条件下的软件应用的性能的感知。将用户输入的分数与推导出的指示符分数进行比较,例如通过在一维回归分析中调整交互性控制,或者通过多维回归分析中调整两个或更多个反馈参数控制。

[0134] 同样在短的时间标度上,当前分数值(或者对分数的统计测量)可被用于触发对网络和计算基础结构的诊断追踪。实现诊断追踪的一种方式是具有此分数的阈值,在该阈值以下则启动诊断追踪,在该阈值之上则停止诊断追踪。一种替代方法是将指示符参数反映作为来自众多用户的平均分数的代理,以概率方式触发诊断追踪。例如,该分数被用作分布参数,且可以计算在阈值以下的概率。该概率随着分数值的减小而增大,从而更频繁地触发诊断追踪。

[0135] 在较长的时间标度上(数量级为小时或天),用户感知分数、他们的感知分量、基础性能指示符度量以及诊断追踪的结果可以被累积、校对和关联。这些作为规划进程的信息有许多可能的用途。例如,可累积网络和计算基础结构元件上的柱状图总况(由诊断追踪结果所识别),并使用趋势技术预测哪些基础结构元件最可能导致感知问题,从而要求容量升级。

[0136] 除了上文具体描述的内容,在随附权利要求的范围内,许多修改都是可能的。

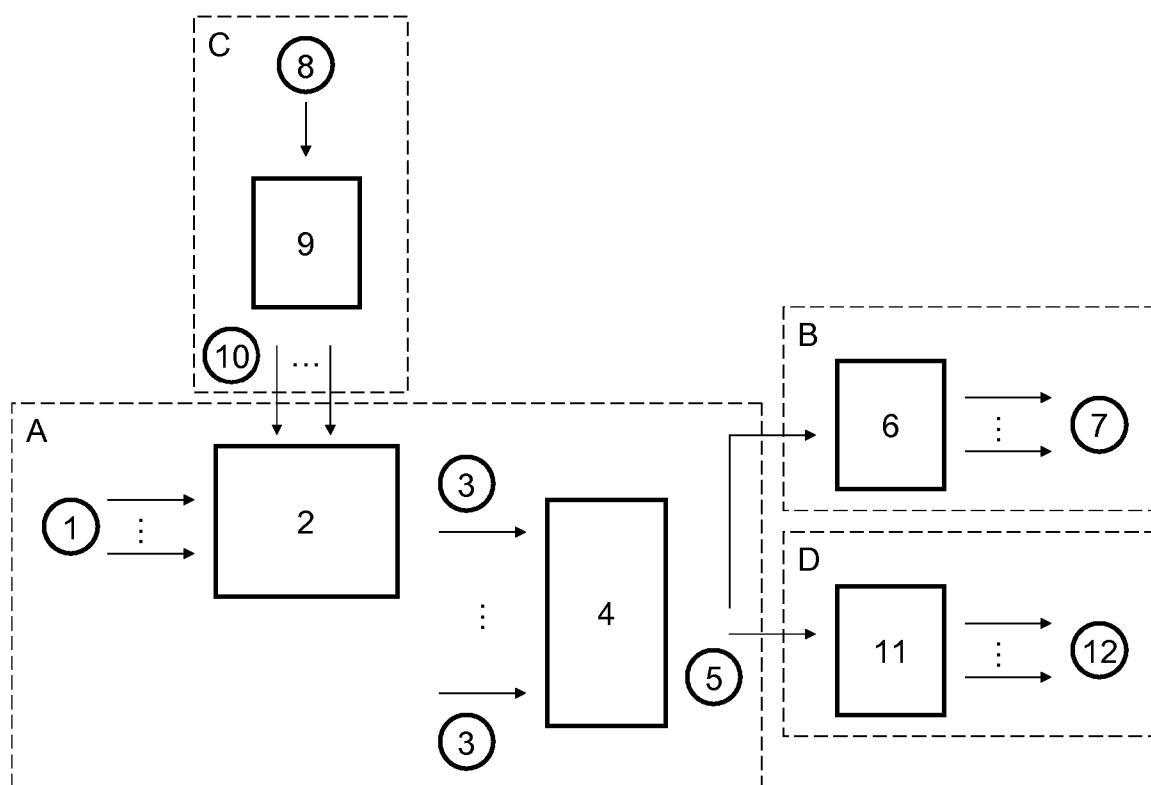


图 1

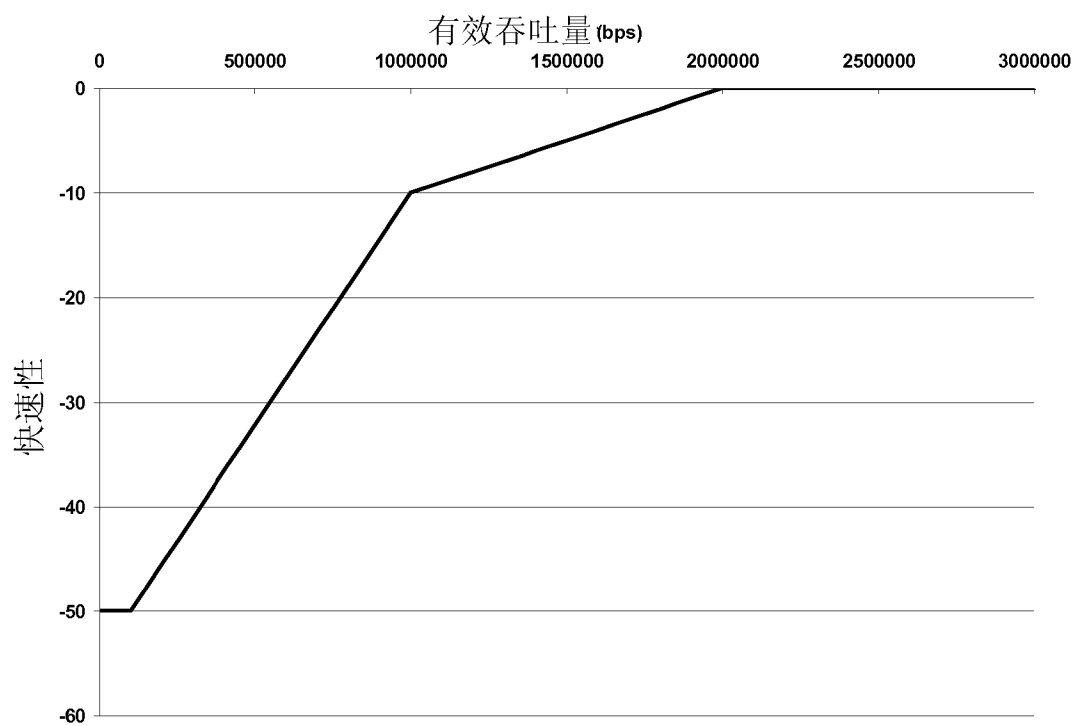


图 2

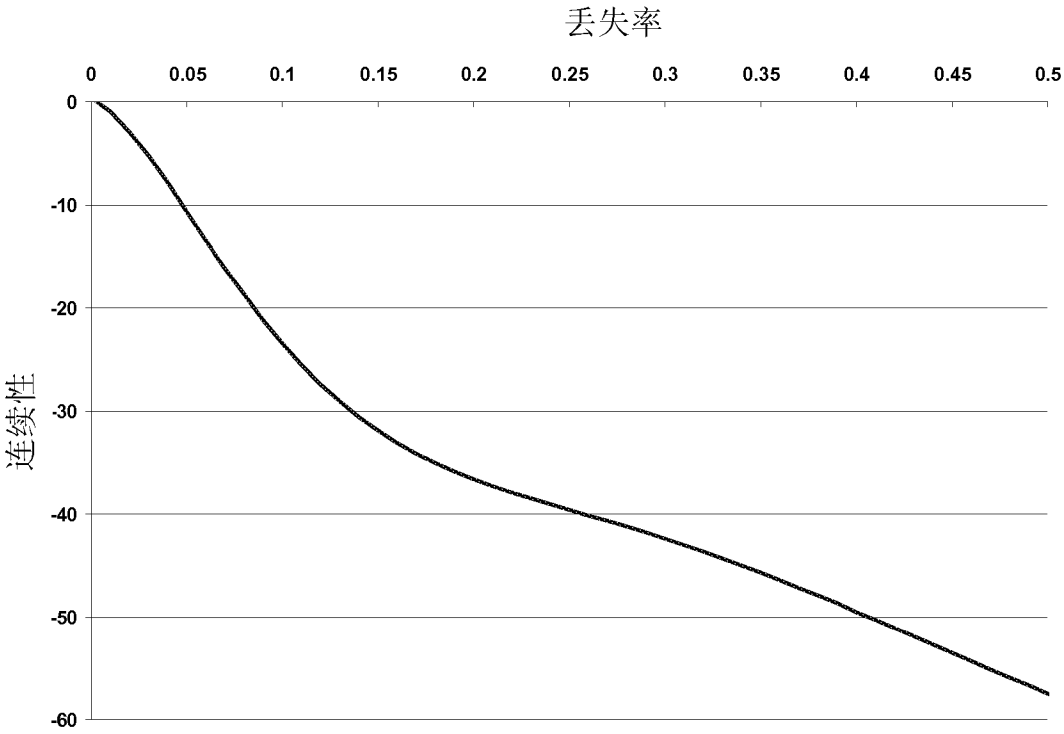


图 3

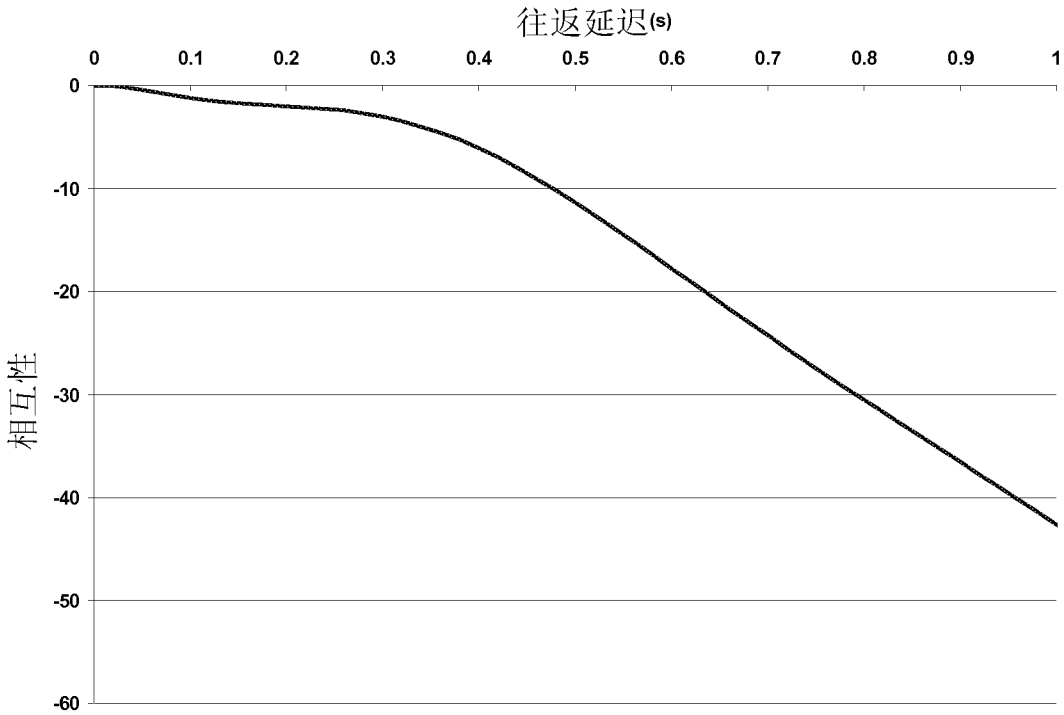


图 4

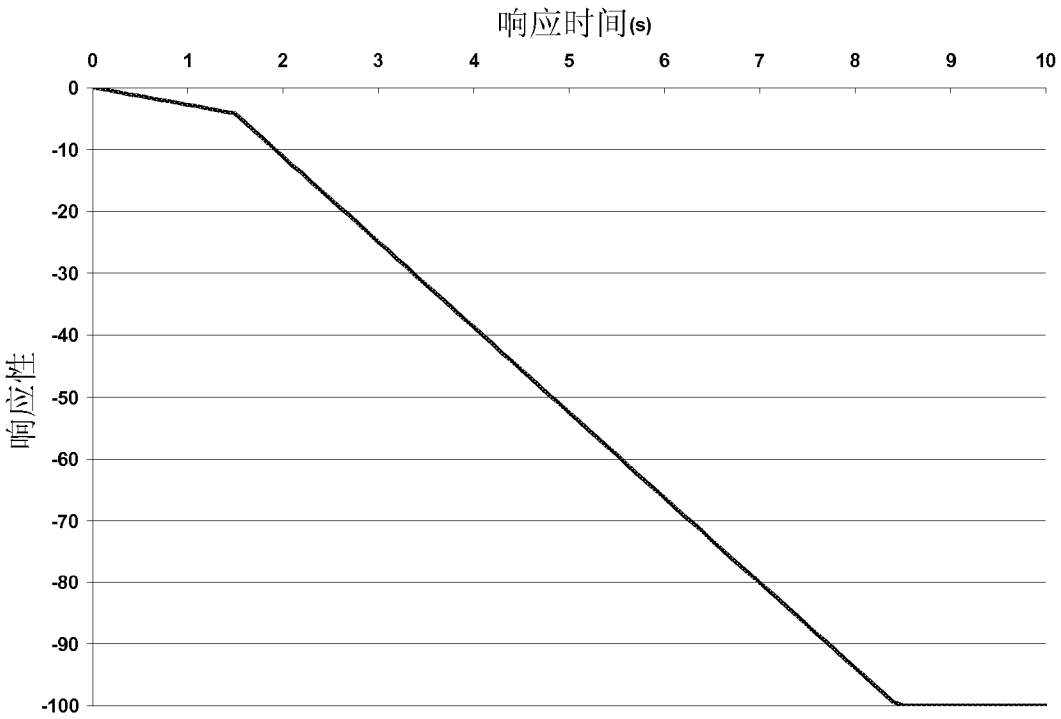


图 5

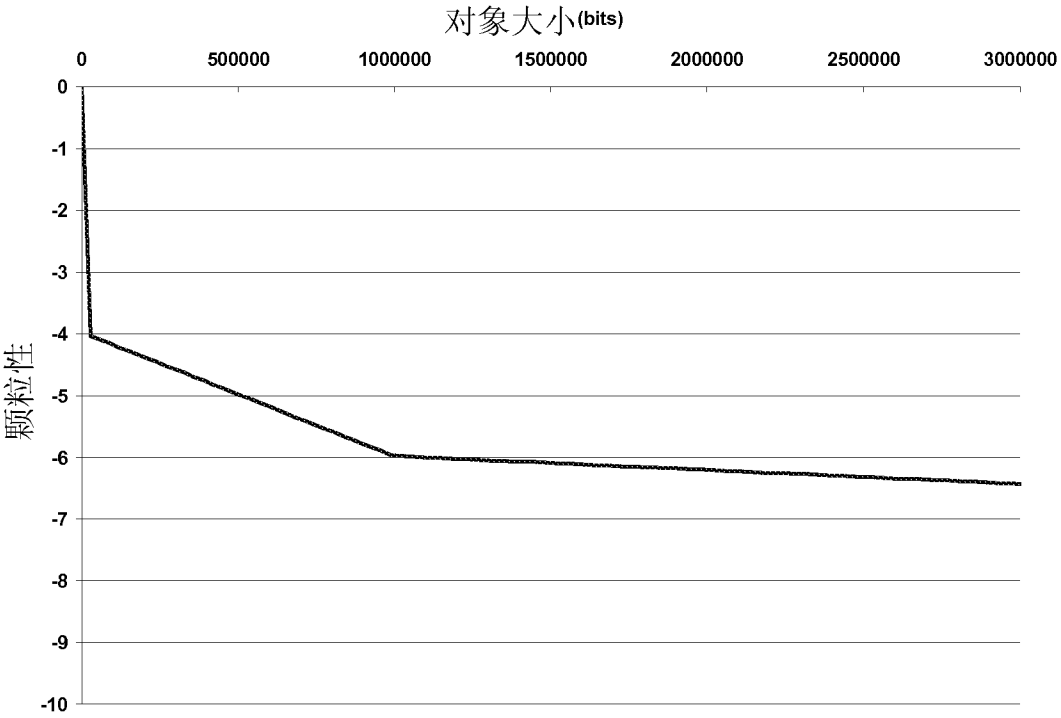


图 6