



(12) 发明专利申请

(10) 申请公布号 CN 103324777 A

(43) 申请公布日 2013.09.25

(21) 申请号 201310191613.3

(22) 申请日 2013.05.21

(71) 申请人 网视科技(杭州)有限公司

地址 310051 浙江省杭州市滨江区南环路  
3820号卓信大厦1406室

(72) 发明人 陈顺访 俞志君 康华

(51) Int. Cl.

G06F 17/50 (2006.01)

G06F 11/00 (2006.01)

权利要求书1页 说明书4页 附图2页

(54) 发明名称

一种基于时间的 Android 应用程序能耗建模方法

(57) 摘要

本发明提出一种基于时间的 Android 应用程序能耗建模方法,使用时间变量刻画移动终端的能耗。由于运行时间容易精确测量和获取,故而能够快速估算应用程序运行时移动终端产生的能耗。与 Android 操作系统提供的基于组件的能耗模型相比,在保证一定误差范围(0.54%~9.43%)的基础上,方便用户利用独立于硬件功耗特性的时间变量估算应用程序能耗。

1. 一种基于时间的 Android 应用程序能耗建模方法,其特征在于建立该方法的步骤如下:

1) 根据智能终端设备的硬件组成,应用程序能耗可分为处理器能耗、存储器能耗、I/O 设备能耗和其它硬件组件能耗,即:

$$E(t) = \sum_{i=1}^N E_i(t) = \sum_{i=1}^N \sum_{j=1}^K P_i^j \cdot t_i^j$$

其中, N 表示终端设备包含的组件个数, K 表示组件的功耗状态个数, E(t) 表示应用程序在执行时间 t 内的能耗,  $E_i(t)$  表示第 i 个组件在时间 t 内的能耗,  $P_i^j$  表示第 i 个组件在第 j 种功耗状态下的功率,  $t_i^j$  表示第 i 个组件在第 j 种功耗状态下工作的时间;

2) 步骤 1) 公式中的  $P_i^j$  可以根据 Android 应用框架提供的能耗分析器配置文件 (PowerProfile.xml) 获得;

3) 步骤 1) 公式中的  $t_i^j$  可以根据 Android 应用程序框架的 BatteryStats 和 BatteryStatsImpl 提供的接口函数获得;

4) 依据步骤 1) 公式对 Android 应用程序能耗进行统计测试与分析,发现应用程序与其运行时间之间存在近似线性关系,因此在时间 t 内 Android 应用程序能耗 E(t) 可通过下式计算

$$E(t) = \beta_1 t + \beta_0$$

式中  $\beta_0$ 、 $\beta_1$  为独立于设备特性的系数;

5) 步骤 4) 公式中的系数  $\beta_0$ 、 $\beta_1$  通过采集 N 组包含应用程序能耗 E(t) 和时间 t 的训练数据,使用线性回归拟合确定;

6) 获取步骤 4) 中的应用程序能耗 E(t) 的方法如下:

Android 系统提供了对应用程序能耗进行分析和测量的能耗分析器,利用基于能耗分析器设计的能耗监控器收集应用程序能耗是现有 Android 平台提供的软件能耗在线测量方式,后台运行能耗监控器,让其每 20s 收集一次应用程序、硬件组件以及整个系统的能耗,并记录在日志文件中,为提高应用程序能耗测量的准确性,应用程序实际的总能耗 E(t) 需排除只运行能耗监控器时的系统基准能耗,测量过程是关闭 Android 移动终端的通信接口,使 Android 系统的能耗监控器在后台运行,关闭显示屏,时间 t 后从能耗监控器日志文件获得一次系统基准能耗,迭代 K 次,对获取的 K 次系统基准能耗求均值,计算出系统基准能耗均值  $E_{base}(t)$ ,然后开启移动终端的通信接口,打开显示屏,让应用程序在能耗监控器开启的情况下运行 t 时间,从能耗监控器日志文件获得一次系统总能耗,迭代 K 次,对获取的 K 次系统总能耗求均值,得到应用程序执行过程中整个系统的能耗均值  $E_{total}(t)$ ,应用程序能耗 E(t) 通过式子  $E(t) = E_{total}(t) - E_{base}(t)$  计算得到。

## 一种基于时间的 Android 应用程序能耗建模方法

### 技术领域

[0001] 本发明涉及智能移动终端能耗优化技术领域,尤其是涉及一种基于时间的 Android 应用程序能耗建模方法。

### 背景技术

[0002] 智能移动终端上丰富的应用程序为用户提供视听享受的同时也消耗着大量的电池电能。2012 年移动终端应用程序下载量将超过 456 亿次。但在这样繁荣的应用程序市场中却存在大量应用程序并未采用有效的节能技术为用户节能。因此为终端用户(如开发人员和普通用户)提供测量与估算应用程序能耗的原型工具与方法就显得十分重要。应用程序的执行驱动了相关底层硬件的电路活动,而硬件的电路活动直接导致了终端设备能耗的产生。因此应用程序能耗是指其执行过程中驱动硬件(包括处理器、存储器和 I/O 设备)所产生的能耗总和。

[0003] 对智能移动终端上应用程序能耗的度量是开展移动终端应用程序能耗分析与优化的基础性工作。度量应用程序能耗最直接和最可靠的方法是使用功率测量仪器(如功率计)进行测量。但专业功率测量工具存在成本高,使用难度大等问题,对于终端用户不方便直接,同时,也无法为用户日常使用终端获得实时的应用程序能耗值。另一种方法是基于应用程序能耗模型设计能耗监控器测量应用程序能耗,在应用程序执行的同时通过统计相关硬件能耗估算软件能耗。

[0004] 近年来国际上已经对智能移动终端应用程序能耗模型进行了广泛研究。出现了基于组件利用率的功耗模型,基于函数、例程和服务的内核能耗估算模型,基于路径的体系结构级能耗分析评估模型,基于系统调用的能耗分析与建模方法等多种方法与模型。Kim M. 等人基于多核智能手机不同组件的功耗特性提出了针对硬件组件的功耗模型;Zhang L. 等人提出了一种在线功耗估算模型,同时为终端用户提供了 PowerTutor 能耗估算工具评价软件能耗;Jung W. 等人提出一种在更新频率和组件控制间的同步技术,克服了使用电池管理单元的在线功耗模型的不足,并且为五种不同组件处理器,显示屏, Wi-Fi 网络接口,蜂窝网络接口和 GPS 提出了一种自动化方法对组件功耗进行分析,获得组件功耗模型;Balasubramanian N. 等人详细介绍了一种功耗测量方法,并提出了 I/O 设备(如 3G 和 GSM 接口)存在明显的尾部功耗特性,针对网络接口提出了一种能耗测量模型;Chen H. 等人采用 pTopW 获取进程级实时能耗信息,并提出一种细粒度能耗感知策略(EnergyGuard),基于 EnergyGuard 可以帮助终端用户发现应用程序能耗异常;在上述能耗模型中,主要分析了终端设备各组件的功耗特性,分别为组件提出了功耗模型,统计了应用程序执行过程中各相关组件的能耗总和,不足之处在于通过能耗监控器实时统计组件能耗系统开销大,监控器自身能耗不可忽视。

[0005] 本发明提出一种基于时间的 Android 应用程序能耗建模方法。与 Android 操作系统提供的基于组件的能耗模型相比,在保证一定误差范围(0.54%~9.43%)的基础上,方便用户利用独立于硬件功耗特性的时间变量估算应用程序能耗。由于使用时间变量去包含和

刻画了硬件的复杂的功耗特性,加上时间变量容易获得,所以基于时间能耗模型的能耗监控器实现简单,系统开销小,自身的能耗明显降低。

### 发明内容

[0006] 本发明的目的在于提供了一种基于时间的 Android 应用程序能耗建模方法。

[0007] 本发明解决其技术难题所采用的技术方案步骤如下:

[0008] 1) 根据智能终端设备的硬件组成,应用程序能耗可分为处理器能耗、存储器能耗、I/O 设备能耗和其它硬件组件能耗,即:

$$[0009] \quad E(t) = \sum_{i=1}^N E_i(t) = \sum_{i=1}^N \sum_{j=1}^K P_i^j \cdot t_i^j$$

[0010] 其中, N 表示终端设备包含的组件个数, K 表示组件的功耗状态个数, E(t) 表示应用程序在执行时间 t 内的能耗,  $E_i(t)$  表示第 i 个组件在时间 t 内的能耗,  $P_i^j$  表示第 i 个组件在第 j 种功耗状态下的功率,  $t_i^j$  表示第 i 个组件在第 j 种功耗状态下工作的时间;

[0011] 2) 在步骤 1) 公式中的  $P_i^j$  可以根据 Android 应用框架提供的能耗分析器配置文件 (PowerProfile.xml) 获得;

[0012] 3) 在步骤 1) 公式中的  $t_i^j$  可以根据 Android 应用程序框架的 BatteryStats 和 BatteryStmsImpl 提供的接口函数获得;

[0013] 4) 依据步骤 1) 公式对 Android 应用程序能耗进行统计测试与分析,发现应用程序与其运行时间之间存在近似线性关系。因此,在时间 t 内 Android 应用程序能耗 E(t) 可通过下式计算

$$[0014] \quad E(t) = \beta_1 \cdot t + \beta_0$$

[0015] 式中  $\beta_1$ 、 $\beta_0$  为独立于设备特性的系数;

[0016] 5) 在步骤 4) 公式中的系数  $\beta_1$ 、 $\beta_0$  通过采集 N 组包含应用程序能耗 E(t) 和时间 t 的训练数据,使用线性回归拟合确定;

[0017] 6) 获取步骤 4) 中的应用程序能耗 E(t) 的方法如下:

[0018] Android 系统提供了对应用程序能耗进行分析和测量的能耗分析器,利用基于能耗分析器设计的能耗监控器收集应用程序能耗是现有 Android 平台提供的软件能耗在线测量方式。后台运行能耗监控器,让其每 20s 收集一次应用程序、硬件组件以及整个系统的能耗,并记录在日志文件中。为提高应用程序能耗测量的准确性,应用程序实际的总能耗 E(t) 需排除只运行能耗监控器时的系统基准能耗。测量过程是关闭移动终端的通信接口,使能耗监控器在后台运行,关闭显示屏,时间 t 后从日志文件获得一次系统基准能耗,迭代 k 次,对获取的 k 次系统基准能耗求均值计算出系统基准能耗均值  $E_{base}(t)$ 。然后开启移动终端的通信接口,打开显示屏,让应用程序在能耗监控器开启的情况下运行 t 时间,从日志文件获得一次系统总能耗,迭代 k 次,对获取的 k 次系统总能耗求均值,得到应用程序执行过程中整个系统的能耗均值  $E_{total}(t)$ 。应用程序能耗 E(t) 通过式子  $E(t) = E_{total}(t) - E_{base}(t)$  计算。

[0019] 本发明与背景技术相比,具有的有益效果是:

[0020] 1) 精确性:利用时间能耗模型估算应用程序在执行时间 t 内的能耗值,并与测量值进行比较,发现时间能耗模型具有较高的准确性,平均误差在 5% 以内。

[0021] 2) 实用性：实用时间刻画硬件组件能耗特性，可以快速估算应用程序在智能移动终端上运行时的能耗值，为开展应用程序能耗优化研究和开发工作提供帮助。

### 附图说明

[0022] 图 1 三种应用程序能耗与时间之间的关系比较

[0023] 图 2 浏览器能耗预测值与测量值比较

[0024] 图 3 视频播放器能耗预测值与测量值比较

[0025] 图 4 Angry Birds 能耗预测值与测量值比较

[0026] 图 5 能耗监控器的能耗

### 具体实施方式

[0027] 下面结合实例对本发明作进一步说明。

[0028] 选取 Android 浏览器、视频播放器、愤怒的小鸟游戏作为例子。

[0029] 1) 根据智能终端设备的硬件组成，应用程序能耗可分为处理器能耗、存储器能耗、I/O 设备能耗和其它硬件组件能耗，即：

$$[0030] \quad E(t) = \sum_{i=1}^N E_i(t) = \sum_{i=1}^N \sum_{j=1}^K P_i^j \cdot t_i^j$$

[0031] 其中，N 表示终端设备包含的组件个数，K 表示组件的功耗状态个数，E(t) 表示应用程序在执行时间 t 内的能耗， $E_i(t)$  表示第 i 个组件在时间 t 内的能耗， $P_i^j$  表示第 i 个组件在第 j 种功耗状态下的功率， $t_i^j$  表示第 i 个组件在第 j 种功耗状态下工作的时间；

[0032] 2) 设置移动终端设备为飞行模式，退出所有用户程序，关闭显示屏，根据步骤 1) 公式测量系统能耗，迭代 k 次，对获取的 k 次系统基准能耗求均值计算出系统基准能耗均值  $E_{base}(t)$ 。

[0033] 3) Android 浏览器：移动终端设备接入 Wi-Fi 无线网络，打开浏览器，访问新闻站点，15 分钟后获取能耗监控器记录的日志文件，获得 15 分钟内系统能耗值，迭代 k 次，对获取的 k 次系统能耗求均值计算出 Android 浏览器 15 分钟执行过程中的系统能耗均值  $E_{total}(t)$ ，使用  $E(t) = E_{total}(t) - E_{base}(t)$  计算 Android 应用程序在 15 分钟的执行过程中产生的能耗。迭代 k 次的能耗值与时间 t 进行线性拟合，得到时间能耗模型，即：

$$[0034] \quad E(t) = 0.49638 \cdot t + 3011.47886$$

[0035] 4) 视频播放器：移动终端设备打开视频文件列表，连续播放 4 段视频片段，播放时间总计 18 分钟，18 分钟后获取能耗监控器记录的日志文件，获得视频播放过程中的系统能耗值，迭代 k 次，对获取的 k 次系统能耗求均值计算出视频播放器执行过程中的系统能耗均值  $E_{total}(t)$ ，使用  $E(t) = E_{total}(t) - E_{base}(t)$  计算视频播放器产生的能耗。迭代 k 次的能耗值与时间 t 进行线性拟合，得到视频播放器的时间能耗模型，即：

$$[0036] \quad E(t) = 0.37610 \cdot t - 9.25696$$

[0037] 5) 愤怒的小鸟：移动终端设备接入 Wi-Fi 无线网络后，执行愤怒的小鸟 15 分钟，15 分钟后获取能耗监控器记录的日志文件，获得 15 分钟内系统能耗值，迭代 k 次，对获取的 k 次系统能耗求均值计算出 Angry Birds 执行过程中的系统能耗均值  $E_{total}(t)$ ，使用  $E(t) = E_{total}(t) - E_{base}(t)$  计算 Angry Birds 在 15 分钟的执行过程中产生的能耗。迭代 k 次的能耗

值与时间  $t$  进行线性拟合,得到时间能耗模型,即:

$$[0038] \quad E(t)=0.54660 \cdot t+7.59640$$

[0039] 时间能耗模型得到的应用程序能耗估算值,与使用能耗监控器获得的测量值相比,平均误差接近 5%。

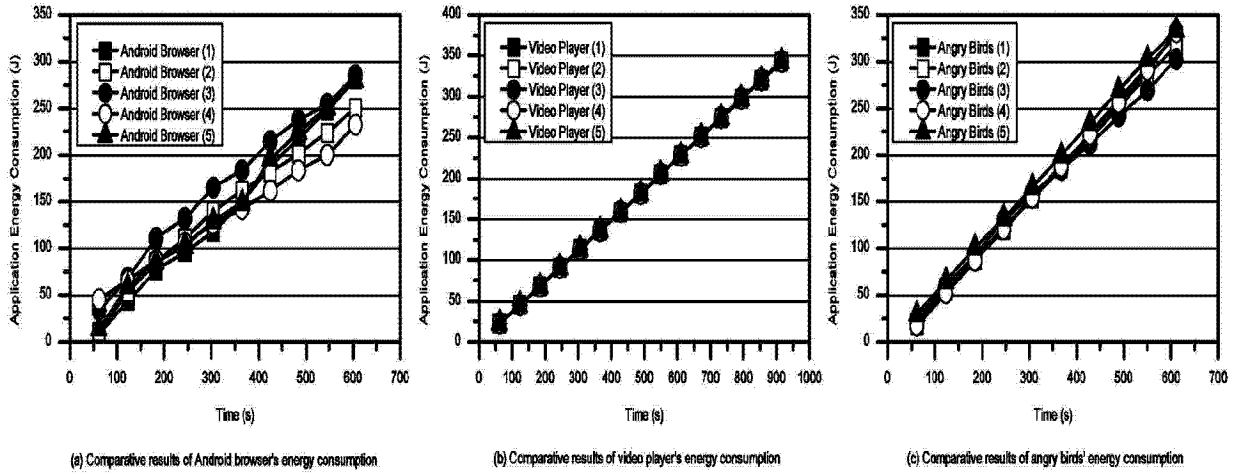


图 1

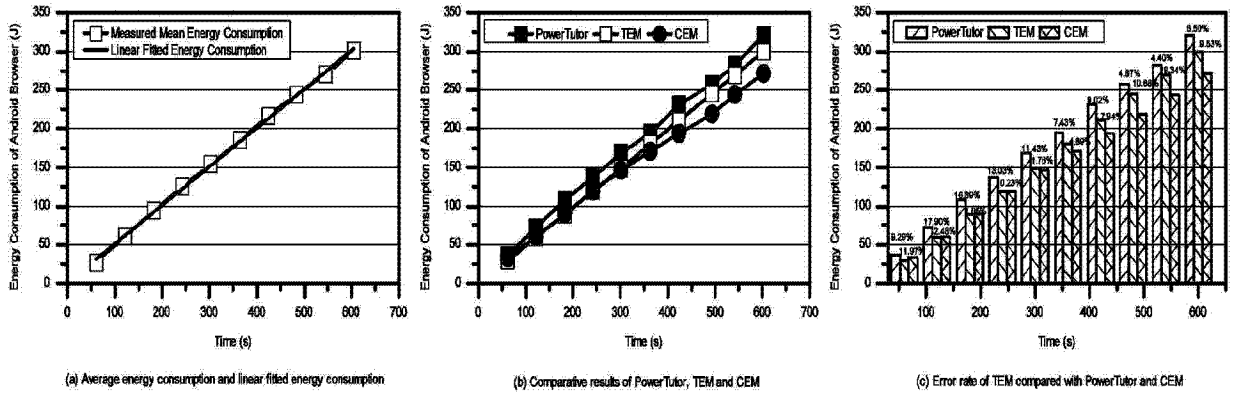


图 2

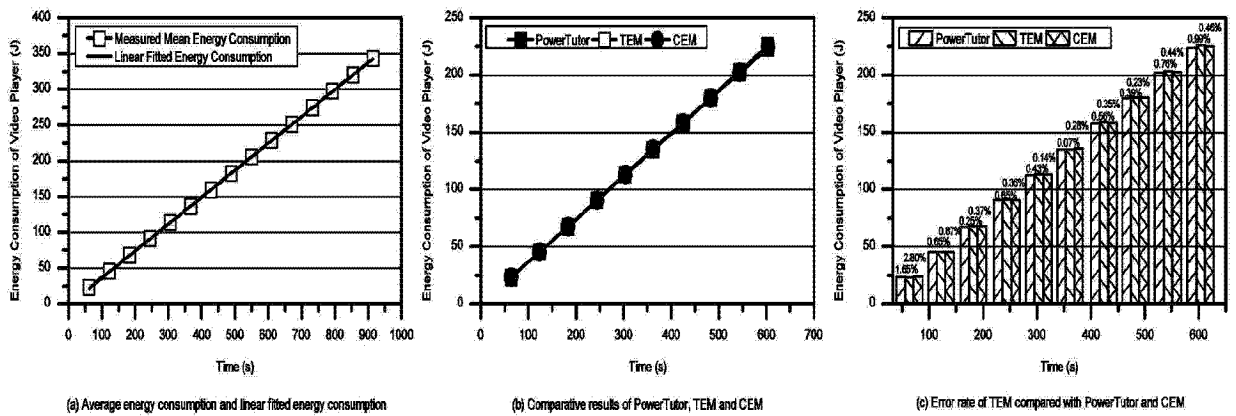


图 3

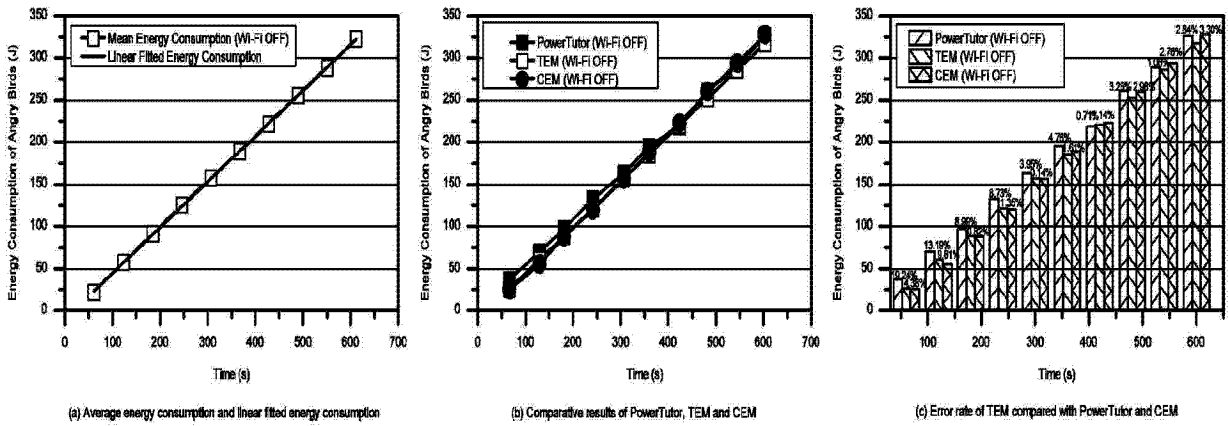


图 4

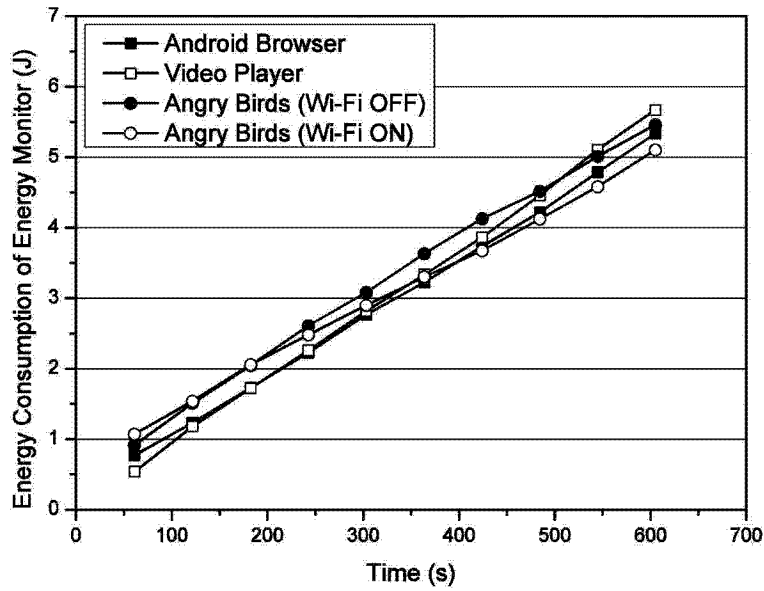


图 5