

一种基于上下文的 Android 电源管理工具

张高翔

(北京大学 信息科学技术学院 计算机科学与技术系)

摘要: 随着智能设备的广泛使用,设备中的硬件组件不断地被开发和更新,以提供用户更丰富的使用体验。但是由于电池的容量和发展水平的限制,续航时间不足经常成为设备性能和用户体验下降的主要因素。用户应用对这些耗能组件的持续占用会导致电量迅速耗尽,所以终端用户的交互方式能够在很大程度上影响设备的续航时间。之前的工作和应用尝试解决这一问题,但是其方案不能针对所有用户,或者不能充分说明有效性。本文设计并实现了一款基于 Android 平台的电源管理应用 Power Supervisor,能够利用上下文信息判断使用模式,对设备中的一些硬件资源进行重新配置,以达到省电的目标。软件能够 1) 提供电源管理相关系统选项的快捷方式; 2) 定时设置任务修改硬件状态; 3) 利用用户的使用模式自动修改硬件状态。广泛的问卷调查和用户测试验证了应用功能的必要性和易用性。最重要的是,应用能够借助使用模式对硬件组件进行调度,来实现良好的电源管理,从而达到节约电量的目标。

关键词: 电源管理; 使用模式; 上下文信息; 移动应用

1 引言

近年来,以 Android 和 iOS 为代表的智能生态系统飞速发展。在这一趋势下,多样化的应用被开发以提供更加丰富的用户体验。为了适应软件的发展,越来越多的硬件组件被集成到设备中,现有组件也在不断更新,以达到更好的性能和交互效果。而电池作为能源的提供者,虽然容量也在不断增加,但是制造技术和设备体积的限制使其远远落后于其他硬件的发展速度,成为设备的性能瓶颈。同时,根据我们的用户调查(章节 2.1),82%的用户经常担心手机电量,1/3 的用户在更换手机时,对续航较差手机的品牌好感度下降。可见,如果能够改善设备的电源管理,达到较长的续航时间,无论对于设备性能,用户体验还是企业利益都有很大的作用。

Zhang^[1]等人通过实验得到结论,Wi-Fi、2G/3G、GPS 和 OLED 等组件是设备能耗的主要来源,并且根据所处状态的不同,耗电量也有较大的差异。由于应用提供多样化的功能,即使在用户不需要使用,设备闲置时也会不断地去触发这些组件的开启^[2],这样就会导致它们被长时间占用而处于高能耗的状态,电量被快速耗尽。针对这一问题,开发者们在 Google Play 等应用平台上发布了相关的应用,目标是对这些耗电组件进行启发式或是定时的管理,期望达到省电的目的。但是之前的研究表明,在用户使用习惯各异、电源管理知识水平参差不齐的时候,这些启发式或者定时设置的功能不能满足所有用户的需求,因此他们提出了基于上下文的电源管理^[3]。然而,在这一研究中,建模过程使用了很多无关参数,比如系统信息,CPU 负载等;同时在管理时仍然使用启发式信息进行判断,没有充分利用上下文;并且在实验时人为构造了测试样例,没有对真实性进行充分说明。

在之前工作的基础上,本文将用户使用习惯对硬件状态的控制以及硬件状态对电量的影响相结合,使用上下文信息进行电源管理,弥补了现有应用和研究不能满足所有真实用户需求的缺点。同时,本文在评测阶段从易用性、有效性和准确性三个角度对用户日常的使用进行评测,增加了结果的

可信度。

2 电源管理系统设计

2.1 用户调查

本文首先进行了广泛的用户调查,试图了解用户的使用习惯和电源管理相关知识的熟悉程度,目的是对本文论点提供支持,并作为系统的设计参考。调查以网络问卷的形式进行,共有 14 个问题,内容主要包括受访者的基本信息、手机使用习惯和 Android 电源管理知识。

我们在三天的时间内总共收集到 101 份问卷,其中 99 位用户是智能手机的使用者,作为有效样本。

2.2 设计目标

受到相关的研究成果和现有软件的启发,本文的设计目标是实现基于上下文的电源管理,即对用户使用数据进行收集,并对新的场景进行分析和自动设置的一种管理方式。为了实现这一目标,应用的设计应该具有以下特点:

1. 目标人群是每一位 Android 智能手机的用户。本文提出的基于上下文的自动管理方法消除了用户使用模式和知识水平的差异。同时为了提高实用性,应用也应该结合现有软件的实用功能,方便用户的使用。
2. 由于应用的目标是实现电源的有效管理,并提高设备的待机时间,因此需要牺牲部分准确度以换取更小的开销,为此应该 1) 使用能耗较低的上下文获取方式; 2) 降低操作或监控频率; 3) 使用简单有效的算法; 4) 对使用模式进行聚类或压缩; 5) 界面简洁易用。
3. 由于需要收集用户的使用模式,并进行自动设置,所以应减小收集过程中对用户产生的额外影响,以及减少在设置过程中错误判断对用户产生的额外操作。

2.3 电源管理应用总体设计

根据设计目标,电源管理软件 Power Supervisor 的主要设计如图 2-1:

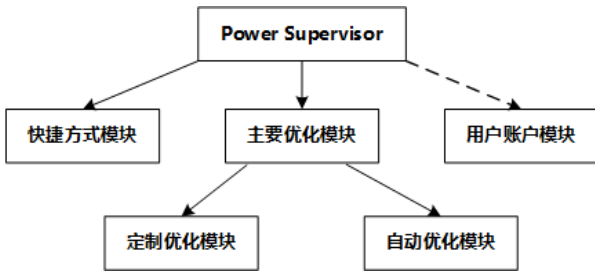


图 2-1 Power Supervisor 的总体设计

2.4 提供系统选项的快捷方式模块

快捷方式模块的主要功能是为用户提供一个和 Android 系统设置菜单类似的，但是排除不相关选项，专门为用户提供电源管理相关参数和信息的界面。根据 2.1 中的调查结果，很多用户对系统设置菜单中的选项，尤其是层次较深的选项，比如锁屏时间、触摸振动等的位置不甚了解。所以快捷方式模块设计的动机便是方便用户的使用。表 2-1 显示了本文经过调研之后预期实现的电源管理相关选项或信息的列表。

信息分类	具体项目
连接选项	Wi-Fi、GPRS、蓝牙、飞行模式、无线热点、GPS
交互选项	亮度、音量、锁屏时间、互动效果
其他选项	当前运行的程序、同步设置
系统信息	流量统计、系统信息、传感器列表

表 2-1 快捷方式界面提供的信息分类和具体项目

2.5 基于固定使用习惯的定制优化模块

根据 2.1 中的用户调查，75.76% 的受访者表示使用手机的时间比较固定，所以在 Power Supervisor 中，本文计划实现定制优化模块。定制优化的功能和现有软件类似，通过用户设置 Wi-Fi、GPRS、亮度等参数作为定制内容，并设置开启和结束时间，系统在这一时间之内设置对应的内容。和现有软件不同的是，本应用希望能够在定制内容到来时给出提示，避免系统参数突然改变，影响用户体验。

2.6 基于上下文的自动优化模块

为了进行自动优化，本文的设计方案是将实时收集到的上下文信息和已有使用模式中的上下文信息进行模式匹配，从而推断出现在的使用模式。这些模式中包含了硬件参数的合理设置，根据这些设置信息对当前硬件组件的状态进行设置，即可实现电源的有效管理，达到省电的目的。

2.6.1 上下文特征选择和生成

由于智能设备中具有丰富的传感器和监控指标，为了得到使用模式，首先需要进行上下文源域和目标域的特征选择。

对于源域信息，需要选择客观环境和用户信息，以推断出用户的使用模式。为了对特征空间进行降维，提高模式匹配的效率，应选择最具有代表性的特征，因此本文选择时间、地点、正在运行的应用、电量状况、网络流量和屏幕状态（是否处于交互）作为源域特征。而对于目标域，选择能够在未

root 的 Android 设备中被设置的硬件参数，因此本文选择 Wi-Fi、GPRS、蓝牙、亮度、音量和锁屏时间作为目标域特征。

由于所有的源域和目标域特征都能够在用户使用时直接被捕捉，所以在用户使用时进行定时监控获得这些上下文信息，进而产生模式。

2.6.2 特征聚类

系统运行的过程中，会不断收集上下文信息而产生模式，这就会导致模式增多，造成系统存储空间和运行效率的下降，所以需要相似的特征进行合并，以产生概括性特征。

这个问题可以看作一个无向图问题，每个特征为一个节点，节点之间存在边且仅当特征相似。则聚类的目标是找到尽量少的分割，使得每个子图中至少有一个节点和子图中每个节点都相似，这个节点就可以作为代表性的特征。

为解决这一问题，本文采用贪心算法，每次在图中找到度数最大的节点，将它和所有相似的点划分为子图，在原图中去掉子图中的点以及和这些点相连的边，直到所有的节点被划分，将每个子集中的中心点作为概括性特征存入数据库。

2.6.3 上下文匹配

和概率匹配中计算每条上下文信息的总相似度，最后找到总相似度最大的模式不同，本文选择的顺序匹配根据某个特征的相似度筛选出符合条件的上下文子集，进行下一个特征的筛选。可以看出，由于每次需要匹配的集合中的元素都会减少，所以顺序匹配具有较高的效率。

匹配过程的核心是相似度计算。对于时间和地点这一组标量，本文的方法是通过计算差值来选择时间和距离都低于阈值的上下文子集。对于正在运行的应用列表，我们使用 Jaccard 系数来进行相似度计算：

$$\text{Application Similarity} = \frac{|A \cap B|}{|A \cup B|}, A, B \neq \emptyset$$

当遇到时间地点都相似，应用相似度相等的上下文时，选择电量差值小的模式，如果仍相等，则选择距离现在最近，和用户现在的状态最接近的模式。

如果没有匹配到符合条件的模式，比如系统刚开始运行或者用户长时间没有使用记录时（比如凌晨），本文使用如下的启发式设置：1) 根据网络流量或连接状态选择开启或关闭 Wi-Fi 和 GPRS，默认关闭蓝牙。2) 亮度保持最低，声音类型保持不变，锁屏时间保持最短。这样的设置既减少了应用的预热过程，又保证了空闲时间设备处于低能耗的状态。

3 电源管理系统实现

本文实现了 Android 平台的客户端应用程序，包括之前所述的快捷方式、定制优化和基于上下文的自动优化功能。用户界面设计上采用了最新的 Android L Material Design 风格，以达到精简软件界面，适应 Android 生态环境的目标。

图 3-1 显示的是快捷方式模块的设计，图 3-2 显示的是定制优化模块的设计：

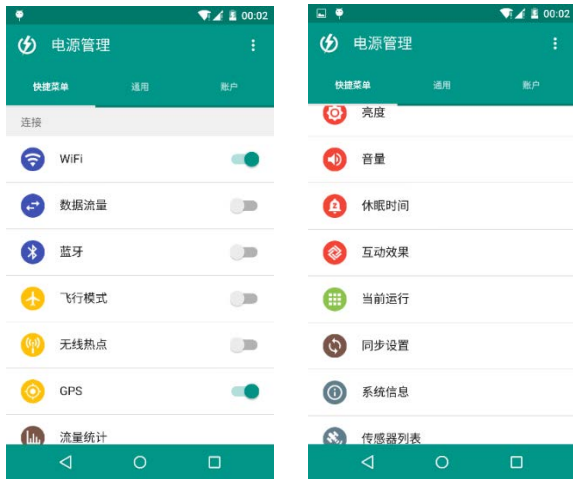


图 3-1 快捷方式模块界面



(a) 模式管理界面

(b) 模式设置界面

图 3-2 定制优化模块界面

4 电源管理系统评估

Power Supervisor 有别于其他基于使用模式的电源管理方法，其最重要的特点之一是实现了完整的 Android 客户端程序，能够根据用户日常使用情况进行硬件状态的调整，通过产生的结果对真实性进行度量。

4.1 测试环境

为了验证 Power Supervisor 的易用性和“自动优化”功能的有效性和准确性，我们邀请了 4 位 Android 智能手机的用户安装应用并进行使用。我们根据 RAM 容量和系统版本选择了编号 2 和 3 的设备进行优化功能的测试。实验要求这两位用户开启自动优化并正常使用三天，如果用户发现设置不能满足需求则手动进行修改。

4.2 易用性

首先，所有用户均表示应用界面友好，能够第一时间掌握软件使用方法。其次，经过用户的使用，应用在这些机型所处的不同分辨率、系统版本下都能正常地工作。

4.3 有效性

为了验证有效性，即保持硬件在低能耗状态下的能力，我们定义有效值为一段时间内硬件状态的平均值。

设备编号	2	3
初始 Wi-Fi 有效值	100	100
优化后 Wi-Fi 有效值	42.8	49.3
初始亮度有效值	84	101
优化后亮度有效值	25.03	33.33
优化后亮度有效值（亮屏时）	37.90	46.86

表 4-1 有效性实验结果

如表 4-1 所示，假设 Wi-Fi 的初始有效值为 100，即始终保持开启状态，经过优化，有效值为 42.8 和 49.3，两台设备均下降了 50% 以上，意味着有一半以上的时间 Wi-Fi 处于关闭状态。假设亮度的有效值为开启前的瞬时值，分别为 84 和 101，优化后降低为 25.03 和 33.33，为了排除锁屏干扰，在屏幕点亮时的有效值为 37.90 和 46.86，同样有比较明显的下降。这也就验证了应用在有效性上表现良好。

4.4 准确性

当预测出现错误时，用户根据需求会对硬件状态进行重新配置，我们希望能够尽量减少用户重新配置的操作，即对所处情景进行准确判断。本文定义准确率为：

$$Accuracy = 1 - \frac{\text{用户交互时系统参数改变的次数}}{\text{用户操作的周期数}} \times 100\%$$

指标	初始	1/3 时间	2/3 时间	总时间
设备 2 亮度	100%	100%	98.60%	98.30%
设备 3 亮度	100%	90.90%	92.00%	93.30%
设备 2 Wi-Fi	100%	26.70%	81.20%	84.00%
设备 3 Wi-Fi	100%	76.70%	84.50%	86.70%

表 4-2 准确性实验结果

如表 4-2 所示，我们将时间等分为 3 个部分，分别计算每个时间点上的累计准确度。可以看出，在第一个时间范围内，准确性表现一般，甚至降到了 26.70%。但是随着时间变化，除了设备 2 亮度准确率略有下降之外，其他情况的准确率逐渐上升。这也说明了模式匹配逐渐开始代替启发式设置，能够更好地对使用情况进行预测。在总时间到来时，所有情况的准确率都达到了 84% 以上。这验证了应用在准确性上表现良好。

参考文献：

- [1] Zhang L, Tiwana B, Qian Z, et al. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In Proc. of ACM CODES+ISSS, 2010
- [2] Metri G, Agrawal A, Peri R, et al. What is eating up battery life on my smartphone: A case study, Energy Aware Computing, 2012 International Conference on. IEEE, 2012
- [3] Datta S K, Bonnet C, Nikaiein N. Android power management: Current and future trends, ETSIoT, 2012