

知識天地

行動系統之喚醒管理：延長你的待機時間

修丕承副研究員 (資訊科技創新研究中心)

前言

我們都有類似經驗，手機擺著就算不用常常一下就沒電了，且應用軟體安裝越多，耗電越快。但這些耗掉的電對使用者體驗並沒有幫助，某種程度來說，這些電是浪費掉的。而浪費掉的電會不會比真正用掉的還多？應該如何解決電量浪費的問題？

手機擺著為何會耗電呢？

時下流行的行動軟體多屬於常駐程式，例如即時訊息 (LINE)、社交網絡 (Facebook) 和電子郵件 (Gmail) 等。這些應用軟體為了能即時收到訊息，手機需連網且處於待機狀態，持續地向遠端伺服器更新其狀態，如圖一所示。連網待機狀態(Connected Standby)是當螢幕關閉時網絡仍需開啟，電量於不知不覺中被消耗殆盡。一項針對行動用戶為期六個月的日常使用行為分析中發現，手機平均有89%時間是處於待機模式，待機消耗的電量占了總耗電量46%左右。最近一項測量報告顯示，若在手機上安裝三個常用的應用軟體，手機待機時間就明顯減少了一半，這是因為使用者察覺不到的手機背景活動頻繁地喚醒手機所致。可知，若手機安裝更多常駐應用軟體將加速電量消耗。



圖一、行動軟體分別週期性地喚醒手機與遠端伺服器同步狀態資訊

現有解決方案不足

為延長待機時間，行動系統採「積極睡眠」的電源管理思維，如非必要硬體元件會各自保持關閉或處於休眠模式，除非有應用軟體明確地向行動系統要求使用某些元件。這種管理思維將硬體元件何時獲取和釋放的工作交給了應用程式開發人員，若有缺乏經驗的程式設計人員不察，導致所開發的軟體存在「不睡眠臭蟲」，造成不必要的硬體持續開啟耗電。為了減少這種程式開發的錯誤，目前已開發出各種診斷工具來自動檢測不成熟應用軟體中的「不睡眠臭蟲」。還有，各種應用軟體正常的背景活動也可能導致行動系統頻繁被喚醒，例如，電子郵件同步，雖然很短，仍會顯著減少手機待機時間。因網路介面一旦開啟，將維持一段比真正需求還要長的時間。可以根據個別郵件收送的型態，動態調整網路介面開啟時間的長短來減少不必要的耗電。相反的，文件上傳通常是要求一段長時間的網路傳輸，傳輸耗電取決於網路速率，而網路速率與所在環境訊號，及其他使用者的網路用量高度相關。對於這樣的應用，可透過延遲數據傳輸，直到使用者進入較為省電的網路環境。

雖然已有一些方案利用各別應用服務的特性，試圖解決不必要的耗電，但是這種零星方案仍會造成行動系統受到不同應用軟體頻繁的異步請求而連續地被喚醒。從一個簡單實驗中可證實，若僅允許手機在固定時間間隔被喚醒，即能相當程度地減輕電量耗盡。這表示應該由行動系統集中管理喚醒請求，而非讓各別應用開發者提出單一解決方案，才是合理的設計思維。

手機的喚醒機制

行動系統使用「積極睡眠」策略來節能，因此行動裝置連同其硬體元件通常保持在低功率的休眠模式，除非被動地被某些應用或系統程式喚醒。此思維下，行動系統必須提供了一個喚醒機制，讓所有應用程式皆可來註冊它們的任務，以便在將來的某些時間點執行。例如，Android設計了稱為「AlarmManager」的喚醒機制來管理喚醒要求的註冊和運行。每當有任務想在特定時間點運行時，應用程式便向行動系統註冊一個喚醒要求，指定預計運行的時間點。行動系統依所有已註冊的喚醒需求，按預定時間先後來運行。在每個喚醒需求的時間

點，原本處於休眠狀態的行動系統就會醒來，等取得應用服務所需要的硬體元件，如Wi-Fi或GPS，即能運行計畫任務。

因背景活動不一定得在特定時間點運行，Android系統從4.4版本之後，引進了「不準時喚醒」的概念，允許在原預定的喚醒時間點前後一小段時間內運行任務，而不一定得在精準的時間點喚醒手機。這個概念增加了喚醒時間的靈活性，允許行動系統能根據不同喚醒要求的時間點，聰明地進行對齊 (Alarm Alignment)，避免手機被各種應用程式頻繁喚醒而導致電量快速耗盡。Android採用簡單的對齊策略，當某個應用軟體向系統註冊了一個新的喚醒需求，系統便按順序比對目前已註冊但仍未運行的其他喚醒需求。若這些喚醒需求，其運行時間區間與新增的喚醒要求的可運行時間區間重疊，那就將它們對齊，到時候手機醒來一起運行這些任務。反之，若沒有可對齊的喚醒需求，系統就將此新喚醒需求排入待運行任務中。然而，這引起了我們的好奇心，想深入了解一個技術性問題—究竟哪些喚醒需求應該對齊在一起，才能最小化手機在連網待機狀態下所浪費的電量？

實驗觀察的見解

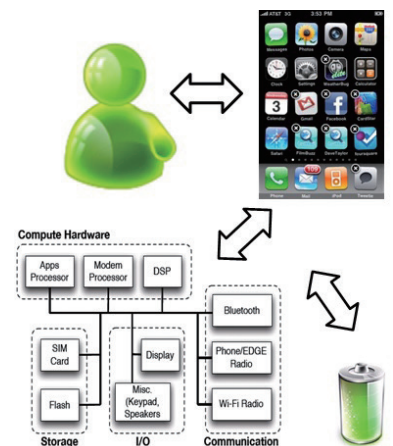
為了調查Android對齊策略的效能，我們在LG Nexus 5手機上進行了一項簡單實驗，安裝了兩個適地性應用軟體和一個行事曆軟體。兩個適地性應用各向系統註冊了一個週期性的喚醒要求，以便根據附近的無線網路和電信基地台信號，定期更新手機位置。同時，行事曆軟體也註冊了一個一次性的喚醒需求，透過手機振動來通知使用者預約時間。我們使用功率量測工具來測量運行這些喚醒任務所需消耗的電量，每次位置定位消耗電量3,650 mJ，日曆通知消耗400 mJ，而簡單地喚醒手機僅消耗180 mJ。若將其中一次位置定位與日曆通知對齊運行，三次的喚醒將總共消耗 $400 + 3650 \times 2 - 180 = 7,520$ mJ的電量。更省電的對齊方式是將兩次位置定位一起執行，總消耗電量只要 $400 + 3650 = 4,050$ mJ。

從實驗中，我們得出兩個有用的觀察，給了我們靈感設計出更省電的對齊策略。首先，喚醒對齊有時只能節省少量電量，若只節省了純粹使用在喚醒行動系統的耗電。但若被對齊的兩個喚醒要求使用相同硬體原件來運行任務，則能達到顯著的省電效果。第二，當運行時，有些喚醒需求得開啟螢幕或使用振動器，將引起使用者注意，而有些喚醒使用Wi-Fi或GPS，那使用者就不易察覺。這些用戶不可察覺的喚醒需求，占了絕大多數，能延遲運行而不會顯著地影響使用者體驗。簡而言之，若要將兩個喚醒需求對齊，它們之間是否使用相同的硬體元件，會直接影響可節省電量的多寡，而它們是否可在接近的時間點進行運行則影響了使用者體驗。故在喚醒管理中，我們必須解決兩個設計挑戰才能實現這個概念，達到節省電量且維持良好的使用者體驗。第一個挑戰是如何定義兩個喚醒要求間的「硬體相似性」(Hardware Similarity) 和「時間相似性」(Time Similarity)，讓這兩個度量指標可以適當地反映省電的多寡及對使用者體驗的影響程度。第二個挑戰是需要設計一個對齊策略，根據這兩個指標來達到顯著的省電效果，且同時維持良好的使用者體驗。

SIMTY - 基於相似性的喚醒管理策略

如圖二所示，在瞭解了使用者對於應用軟體的使用行為、個別應用軟體對於硬體資源的需求、以及不同硬體元件對於電池耗電的模型之後，我們將「喚醒相似性」的概念引入行動系統的喚醒管理思維中。基於手機背景活動的一些特點，我們將相似性分為三種等級：高、中和低。根據兩個喚醒需求間如何共享硬體元件，我們區分出三個等級的硬體相似度，完全共享、部分共享及不共享。對時間相似性，我們先區分出使用者可查覺與不可查覺的喚醒需求，然後依據它們可運行區間的重疊程度區分出有重疊、可重疊、不可重疊三個等級。然後，利用硬體和時間相似性，我們設計了一套對齊策略，主要原則是在時間相似性夠高之下，盡可能對齊硬體相似度越高的喚醒需求。

為了驗證此概念的實用性，我們將此設計整合進Android中，並採用18個



圖二、使用者、行動軟體、硬體元件、電池電量間之互動關係

Google Play上流行的應用軟體在LG Nexus 5手機上進行了一系列實驗。與原始的對齊策略相比，我們的設計能將手機的待機時間再延長四分之一到三分之一，同時維持良好的使用體驗。對研究細節感興趣的讀者，可以閱讀我們最近發表的研究論文：Chun-Hao Kao, Sheng-Wei Cheng, and Pi-Cheng Hsiu, "Similarity-Based Wakeup Management for Mobile Systems in Connected Standby," IEEE/ACM Design Automation Conference (DAC), pages 50:1-50:6, June 2016。

結語

我們主張應該由行動系統集中管理所有應用軟體註冊的喚醒要求，而不是將責任交給個別軟體的開發人員。本文介紹了我們設計的「SIMTY」喚醒管理策略，它利用喚醒之間的硬體和時間相似性，大幅延長待機時間。實驗結果顯示SIMTY可避免不必要的手機喚醒；甚至，能將喚醒次數減少至最低量。若要進一步降低待機耗電，可朝擴展SIMTY的概念，多考慮硬體使用時間長度的相似性 (Duration Similarity)，將使用時間長度接近的喚醒要求對齊在一起。但這仍需要未來的Android或其他行動系統提供一個機制，讓軟體開發者表明對於某硬體元件需要使用多久。希望本研究對於行動系統的喚醒管理，提供了一些有價值的見解，以因應未來手機需要安裝更多常駐應用軟體，造成待機時間過短的問題。