行政院國家科學委員會專題研究計畫　成果報告

子計畫三：多階隨意網路上位置衍生的服務與應用(2/2)

計畫類別：整合型計畫
計畫編號：NSC93-2219-E-009-002-
執行期間：93 年 08 月 01 日至 94 年 07 月 31 日
執行單位：國立交通大學資訊科學學系(所)

計畫主持人：簡榮宏

計畫參與人員：蔡嘉泰、李奇育、張瑋倫、盧牧英

報告類型：完整報告

處理方式：本計畫可公開查詢

中　華　民　國 94 年 9 月 23 日

行政院國家科學委員會補助專題研究計畫期末報告

※※※※※※※※※※※※※※※※※※※※※※※※※
※　　　　　　　　　　　　　　　　　　　　　　　　　※
※　多階層行動隨意網路之設計及實作—子計劃三：　※
※　　多階隨意網路上位置衍生的服務與應用（2/2）　※
※　　　　　　　　　　　　　　　　　　　　　　　　　※
※※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：□個別型計畫　　☑整合型計畫
計畫編號：　NSC　93－2219－E－009－002
執行期間：　93 年 8 月 1 日至　94 年 7 月 31 日

計畫主持人：　簡　榮　宏

本成果報告包括以下應繳交之附件：
- □赴國外出差或研習心得報告一份
- □赴大陸地區出差或研習心得報告一份
- □出席國際學術會議心得報告及發表之論文各一份
- □國際合作研究計畫國外研究報告書一份

執行單位：　國立交通大學　資訊科學系

中　華　民　國　94 年 7 月 27 日

行政院國家科學委員會專題研究計畫期末報告

多階層行動隨意網路之設計及實作—子計劃三：多階隨意網路上位置衍生的服務與應（2/2）

Location-Base Services and Applications for Multi-tier Ad Hoc Networks

計畫編號：NSC 93-2219-E-009-002
執行期限：93 年 8 月 1 日至 94 年 7 月 31 日
主持人：簡榮宏　　國立交通大學資訊科學系
計畫參與人員：蔡嘉泰、李奇育、張瑋倫、盧牧英
國立交通大學資訊科學系

中文摘要

隨著無線區域網路的普及，影響了人們對無線網路服務與應用的需求，其中位置衍生(location-based)的服務與應用是最為重要的網路服務之一。在本計畫中，我們分別設計了(1)定位技術(2)位置閘道器(3)地理位置資訊 (4)位置追踪服務等四個模組來提供位置追踪服務應用系統。所開發的系統具有相容性、擴充性、模組化及不限使用環境的特性。

關鍵詞：無線區域網路、位置閘道器、位置追踪服務

Abstract

With the development of Wireless Local Area Networks (WLANs), people are interested in developing the location-based services for WLAN users. In this project, we design and integrate a location-based service system with four modules: location determination technologies, location gateway, geographic location provisioning and location tracking service, to provide a location tracking service and application. This system has the ability of compatibility, extensibility, module, and unlimited circumstance.

Keywords: Wireless local area networks, Location gateway, Location tracking service

目錄

一、前言

隨著無線網路的發展及手持式設備成本的降低，不論是機場、學校、咖啡店、速食店等公共場所亦或是個人的家中都有採用無線網路，如此普及的無線區域網路也促使了中小型的設備，如筆記型電腦、個人數位助理器、手機，甚至是資訊家電等都具備無線網路傳輸功能。尤其是小型的手持式設備，可以讓使用者攜帶在身邊隨時使用而不受限制的在任何場所、地點及時間。正由於其可移動、方便攜帶及無線上網的特性，衍生出許多的應用，如旅遊資訊導覽[1]、行車導航[2]及路況資訊的提供、緊急救援服務[3]等，而這麼多應用服務(application services)的開發其背後均需要位置資訊的提供才能完成該服務，然而所提供的位置資訊是否正確，精確度是否適合，都會影響著應用程式的服務品質及正確性。因此，位置資訊的研究便成了開發以位置資訊為導向(Location-based Services)應用程式的重要技術。

在提供位置資訊技術的研究方面，已有不少的研究結果可供參考，然而這些技術都有其適用的範及使用限制，例如全球衛星定位技術(Global Positioning System) [4, 5]，是在行動裝置上加裝接收器來達成高定位精確度，而其最大的缺點在於室內的環境中無法接收到訊號，而喪失定位功能；反之，也有只適用於室內的定位技術，如訊號特徵(signal fingerprinting)[6]定位方法，須事先在室內環境內量測訊號特徵值，並儲存於資料庫中，但在室外環境下，由於量測區域龐大，此一定位方法就顯得不適用。除了上述兩種定位系統外，尚有 DV-Hop[7]、質心定位法[8]、可移動參考點[9]、訊號到達基地台的夾角(Angle of Arrival, AOA))[10,11]、訊號到達基地台的時間差(Time Difference of Arrival, TDOA)[10,11]、輔助全球衛星定位系統(Assisted-GPS) [12, 13]、細胞識別定位系統等，這些常見的方法將在文獻探討中加以詳細說明。因此，由上述可知發展一個可用於室內及室外的定位技術是一個值得深入探討的研究課題。

另外，要發展一個適合室內及室外的定位技術要考慮的因素很多，影響定位系統準確度的變因也很多，想要將這樣的系統設計得完善是很不容易，以目前所有已經提出的定位技術中並沒有這樣的定位系統。所以，本計畫將考量各種定位技術、優缺點及其限制條件後，開發新的定位技術，來達成能在室內及室內環境下都具有定位功能的系統，並以此定位系統為基礎實作一個位置資訊為導向的應用程式，以驗證理論的可行性，及分析實際定位系統的準確度。

二、研究目的

定位技術的重要性已成為無線網路研究的重要項目之一，以目前定位技術的發展狀況，使用者在不同的環境下，所能使用的定位技術也有所不同，這是由於定位技術的適用範圍及其精確度的影響所致。然而這樣的差異對於開發位置資訊為導向應用程式的人員會造成很大的困擾，需要額外的花費來偵測目前使用者環境的是否變化，程式開發的複雜度也因此大大的升高。若程式開發者要求使用者自行設定其所在的環境，則會造成使用者的不便，使用者必須明確知道其所在的環境，當其變換了所處的環境就要更動設定。因此，為了降低開發位置資訊為導向應用程式的複雜度及提升使用者的便利性，發展出適用於室內及室外的定位系統是必要的。本計畫的目標在發展一個適用於室內及室外的定位系統。

然而直接發展一個適用於室內及室外的定位系統困難度及系統複雜度非常的高，因此，利用適當的決策模組來整合現有的室內及室外的定位技術是一個有彈性且具模組化的可行方法。具有彈性的優點是在於可以整合各種定位系統，或開發出新的定位技術後可以加以替換而不需更動已開發完成的應用程式及定位系統。具有模組化的優點是能讓

程式開發者專注於應用程式的領域來發展所需的位置資訊為導向應用程式，而不必再額外考慮使用者所在的環境，有助於降低位置資訊為導向應用程式發展的複雜度及困難度，也可讓專門研究定位技術的人員，針對其專長設計或改良出較好的室內或室外的定位方法後，就可輕易的合併、替換或整合到此定位系統內。

然而在整合室內及室外的定位技術中，對於定位技術的切換時機及無接縫切換的問題必需加以探討，因為若是切換時機不對或切換而造成長時間的延遲，都會造成位置資訊的誤差、延遲(delay)或中斷。因此，定位技術的切換時機與無接縫切換的問題也是我們在發展整合室內及室外的定位系統必需加以考量的重要因素。

綜合上述的各項因素，我們在本計畫中發展出一個整合室內及室外的無接縫定位系統，並以此系統為基礎，再建立所需的地理資訊系統，如此便可發展成一個位置追蹤服務的應用系統，實作此位置追蹤服務的應用系統所產生雛型系統可提供將來研究位置資訊為導向的服務來參考。

三、文獻探討

目前已有多種的定位技術及其應用被提出來，我們將其簡單分成三大類，分述如下：

1. 自行定位系統(Self-positioning system)：行動裝置能透過接收來自特定裝置所發送出來的位置相關訊號，自行收集、處理後可獲得自身的位置資訊。
   - DV-Hop[7]：利用計算從開始的節點至目的節點的 Hop 數目，並收集參考點的位置資訊，來估計節點的位置。
   - 質心定位法[8]：利用參考點的涵蓋區域重疊的特性，若使用者在某一涵蓋的重疊區域內，則計算此區域的質心位置來當成使用者的位置。
   - 全球衛星定位系統(Global Positioning System, GPS)[4, 5]：利用環繞地球的 24 顆衛星，將衛星精確的速度、高度、經度、緯度傳送到使用者的全球衛星定位系統接收器，然後再由手機自行計算自己的位置。
   - 可移動參考點[9]：利用數個可移動且週期性散佈其位置資訊的參考點，使用者可以收集參考點的位置並加以分析，以獲得使用者的位置資訊。
2. 遠端定位系統(Remote positioning system)：在網路設備中加裝特別的無線電頻率功能的裝置，經由測量訊號來源的方向或訊號來源的時間差。這此的量測資訊由中央伺服器負責收集，並計算出位置資訊。
   - 訊號到達基地台的夾角(Angle of Arrival, AOA)[10, 11]：基地台需額外建置一個能辨別訊號送至基地台時的角度的天線，利用使用者與其所有相臨的基地台的訊號夾角，再利用三角測量來獲得使用者的位置資訊。
   - 訊號到達基地台的時間差(Time Difference of Arrival, TDOA)[10,11]：基地台需額外建置一個設備，它能辨別訊號送至基地台時的時間差，利用使用者與其所有相臨的基地台的訊號時間差，來獲得使用者的位置資訊。
3. 間接定位系統(Indirect positioning system)：此定位系統是整合自行定位系統與遠端定位系統。欲定位的裝置先自行量測訊號資料，然後再將訊號資料送至遠端定位系統。遠端定位系統收集這些量測的訊號資料及處理位置資訊的偏差值，便能計算出位置資訊。
   - 輔助全球衛星定位系統(Assisted-GPS) [12, 13]：方法類似全球衛星定位系統，但在網路端加入一個位置修正伺服器。因為衛星傳送的資訊會因為地表空氣的折射干擾而產生誤差，故透過此位置修正伺服器將所計算出的位置資訊加以修正，以獲得較精確的位置資訊，並節省手機的電力消耗。
   - 訊號特徵(signal fingerprinting)[6]定位技術，利用已存在的網路架構，事先量測訊

號強度的特徵值，並分析儲存至資料庫中，啟動此定位後，使用者只需傳回其目前所接收到的訊號特徵給位置伺服器，位置伺服器便能依據事先定義的規則從資料庫中取出其所對映的位置資訊。最著名的應用即為利用無線網路來定位的RARDAR[14]系統。

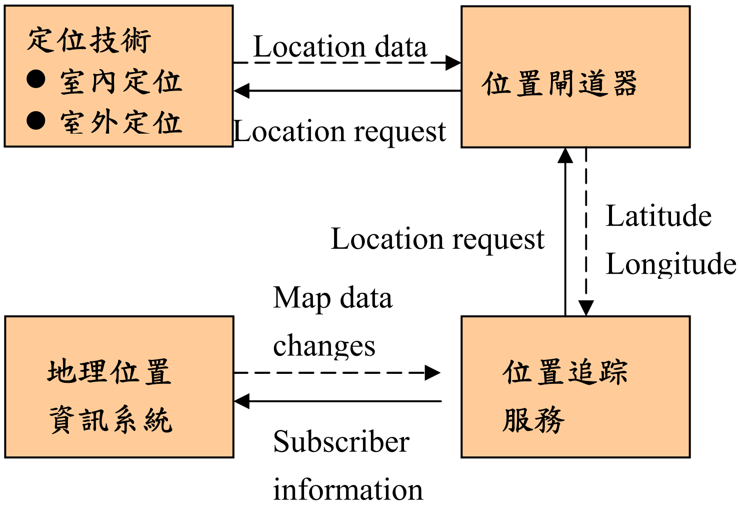
- 細胞識別定位系統：利用基地台發出個別的識別記號(Cell ID)，根據使用者接收到不同的識別記號群組，並透過網路的中央位置伺服器來決定其所在的位置，例如以細胞格為主的定位方法(cell-based position method)[15, 16]。

在愈來愈多的研究定位的方法被提出來，提升了定位系統的準確度，有了適當的定位系統，便可以發展位置衍生的服務與應用： 如 Active Badge[17]的辦公室人員電話轉接系統，每個人員配帶一個小型的發射器，每個發射器能發出一個唯一的識別碼，在辦公室內裝設適當的接收器，來判別使用者目前所在的地點，當有電話進來時，若人不在坐位上，便可以此系統找到人員所在位置，並將電話轉接到距其最近的話機，以方便人員接聽；利用無線電訊號強度的特性所發展的定位系統，如：RADAR[14]、SpotON[18]，其中 RADAR 是應用在二維平面的室內定位，而 SpotON[18]則可用在三維平面的室內定位；Cricket[19]定位系統是在室內的適當位置放置超音波發射器，使用者端安裝接收器，使用者就能根據所接收到不同的超音波發射器所發出的資料來決定位置資訊；同樣採用超音波發射器及接收器的原理，也發展出智慧家庭的居家老人追踪系統[20]，以監控其是否需要即時的照顧或是協助；旅遊導覽系統[1]的應用，則是讓遊客手持一個具有網路存取功能的數位助理器，將遊客目前所在位置周圍的特殊景點顯示在數位助理器上供遊客參考，可以再點選選項以提供詳細解說。

四、研究方法

本計畫將整合已發展的定位系統、位置閘道器及地理資訊系統來發展位置追踪服務，基本架構如圖一所示，共可分為四個主要的模組，(1)定位技術(Location Determination Technologies) (2)位置閘道器(Location Gateway) (3)地理位置資訊(Geographic Location Provisioning)(4)位置追踪服務 (Location tracking service)，各區塊內基本功能概述如下：

1. 定位技術(Location Determination Technologies)模組：負責接收位置資訊的需求，並根據位置閘道器的協助來決定所需啟動的定位技術，並參考資料庫內所存之無線網路存取器(Access point)的位置資訊及網路環境來計算位置資訊，並回傳其結果。而本模組內包含室內定位技術子模組與室外定位技術子模組，透過此二子模組的運作可以使此定位技術模組能應用在各種不同的環境。
2. 位置閘道器(Location Gateway)模組：負責接收位置追踪服務的位置要求(location request)並轉送給定位技術模組。在轉送位置要求前會先啟動一個決策子模組，它會根據網路系統的狀況及使用者的裝置來通知定位技術模組所需的適當定位技術，當定位技術模組回傳位置資訊後，本模組會將此結果回傳給位置追踪服務模組。
3. 地理位置資訊(Geographic Location Provisioning)模組：提供位置追踪模組所須之地理位置資訊，如地圖、座標、相對位置或經緯度等資訊。
4. 位置追踪服務 (Location tracking service)模組：在行動裝置上開發位置追踪服務應用程式，向位置閘道器提出位置資訊之需求，取得後再依位置資訊向地理位置資訊模組取得對應的地理位置資訊。



圖一：本計畫基本架構圖

本計畫根據所提之計畫書內容逐步完成上述各模組，並加以整合各模組之功能，完成位置追踪服務的應用系統，本系統架構的優點如下：

1. 本架構的模組化設計，讓各種不同功能的模組能獨立開發互不干擾，使得研發人員可以專心針對特定的模組研究，而所開發的模組之間又能相互溝通傳遞訊息，協力完成定位目標。
2. 位置閘道器內的決策子模組能根據位置追踪服務所提供的有限資訊來決定使用者目前所在的環境，讓使用者不必理會其所在環境都能獲得其位置資訊。
3. 定位技術模組包含有室內及室外的定位技術，使得本系統能在大部分的環境下正常運作。另外，藉由位置閘道器的協助，本模組可以輕易加入各種已開發或新開發的定位技術而不會影響整個系統的運作，相容及擴充性頗高。

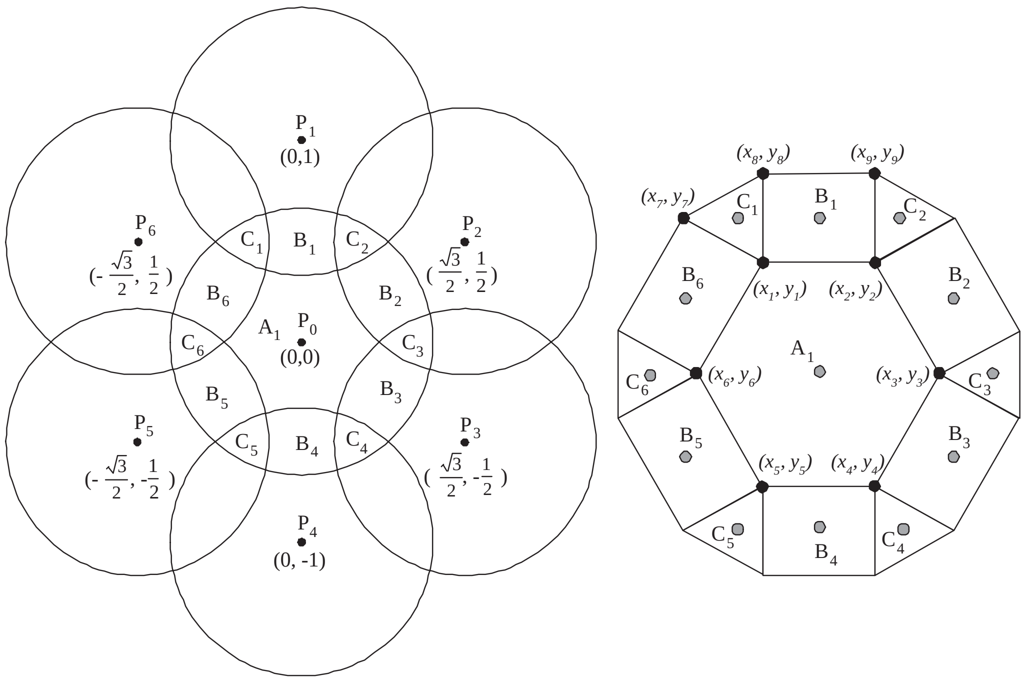
本系統整合地理位置資訊，所呈現的位置資訊可讀性高，配合使用者的程式介面，讓位置追踪的操作更便利。

本計畫之系統開發將針對(1)定位技術(Location Determination Technologies) (2)位置閘道器(Location Gateway) (3)地理位置資訊(Geographic Location Provisioning)(4)位置追踪服務等四個模組的開發分別說明如下：

1. 定位技術模組：本模組包含四種定位技術
   (1) 我們針對以細胞格為主的定位方法(cell-based positioning method)[15,16]研究、分析及改進，將訊號重疊的區域與面積的質心理論相結合，原本的定位精確度以訊號重疊區域的表示方式，改由該區域的質心來表示(如圖二所示)，並探討精確度的變化及系統的效能。另外，將原本集中式細胞格為主定位方法（位置資訊傳至位置伺服器處理）經由我們設計的訊框格式(beacon frame)夾帶位置訊息的方式，使得每個使用者可以利用本身的裝置，藉由簡單的數學運算即可自行定位，完成分散式的定位技術。然而，由於多階訊號強度的特性在細胞格為主的定位方法下可以提升定位的準確度(如圖三所示)，因此我們也在本模組內加以考量；
   (2) 權重式多角定位(multilateration) [21]的方法，分成兩個階段：第一階段是以節點的DV-hop的方式計算出初始的位置;第二階段則利用節點的鄰居位置與距離，反覆地更新與交換資訊，並根據資訊的可靠程度給予不同權重來進行多角定位；

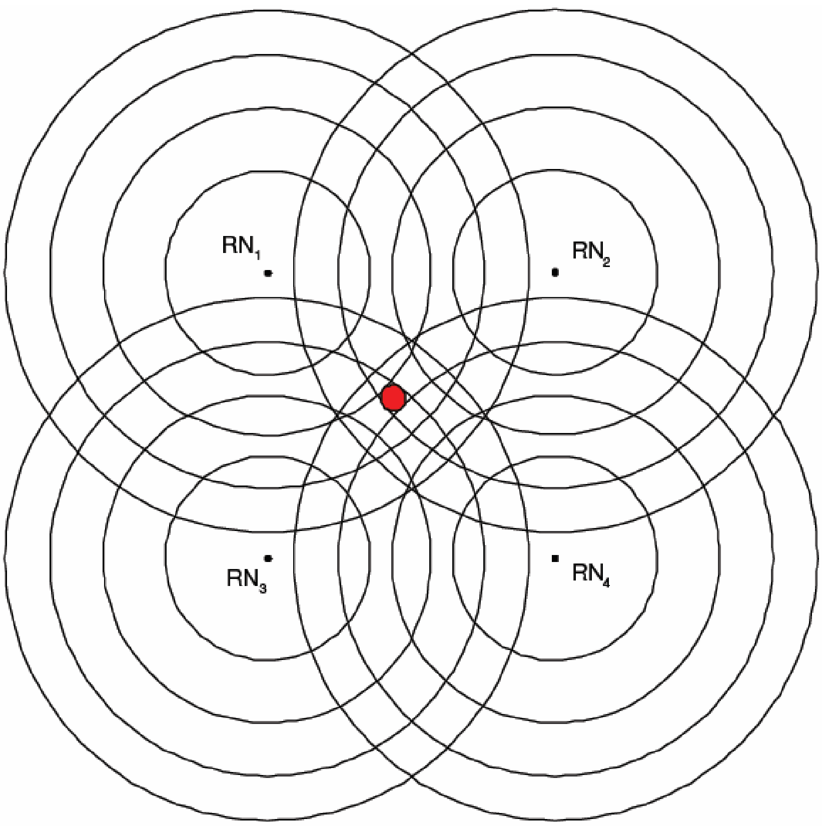
(3) 訊號特徵(signal fingerprinting)定位方法，是根據訊號在不同的位置會有不同的訊號強度，因此藉由事先量測訊號強度並記錄在資料庫中，便可利用訊號比對的方式來完成定位；

(4) 全球衛星定位系統(Global Positioning System)是利用環繞地球的24顆人造衛發射其位置訊號，使用者接收此訊號並加以處理便能獲得其位置資訊。



(a) (b)

圖二：(a)重疊區域(b)轉換成質心的表示方式

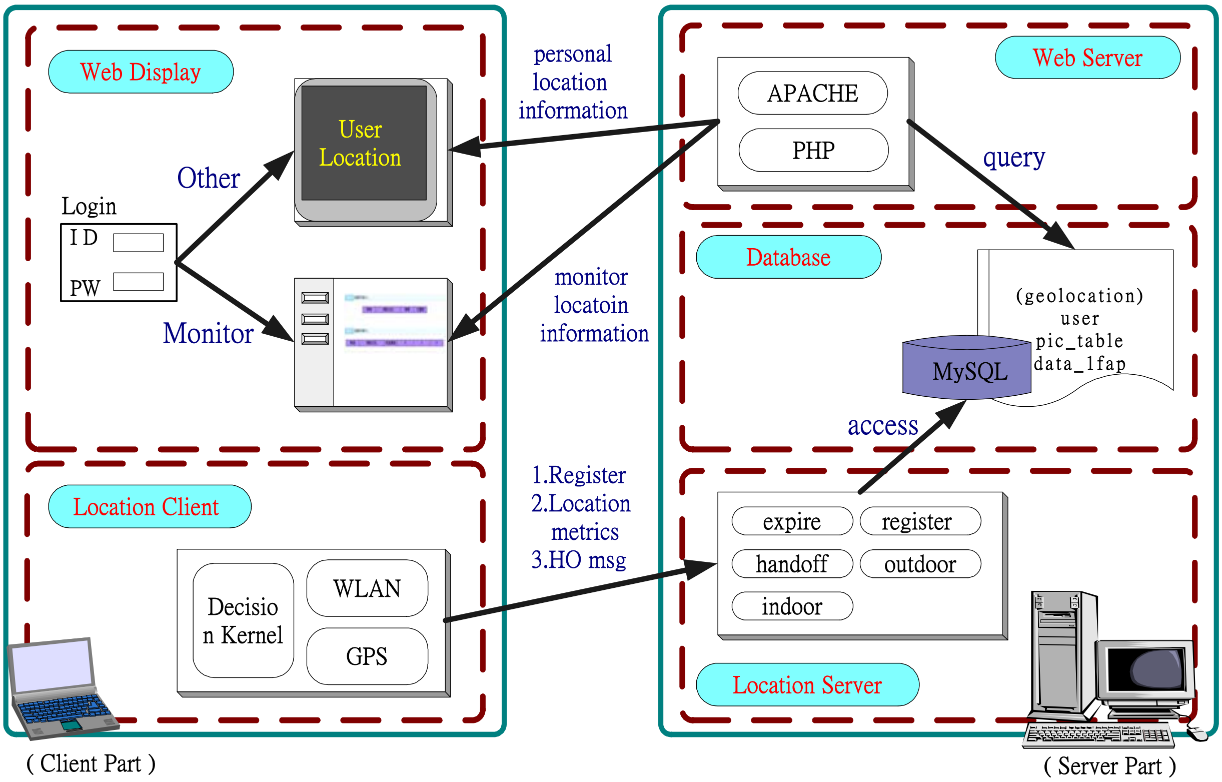


圖三：多階訊號強度所形成之區域

2. 位置閘道器模組：本模組主要的功能是整合不同性質定位方法，利用我們所設計的一個決策子模組，它會根據現行網路的環境及使用者的裝置來通知定位技術模組所需的適當定位技術，當定位技術模組回傳位置資訊後，本模組會將此結果回傳給位置追踪服務模組。

   在此我們先討論兩種常見的定位方式來實做整合，一個是室外的全球衛星定位(GPS)技術；另一個則是室內的訊號特徵(signal fingerprinting)定位技術，這是因為室外的全球衛星定位(GPS)技術，若使用在室內的環境下會完全失效，反之，室內的訊號特徵定位技術若使用在室外會因為沒有事先建立資料庫來比對而失效，藉由這兩個較特殊的定位技術來驗證系統的靈敏度及準確度。也因此，我們將提供一個無接縫的位置換手模式，使得當使用者所在的位置(室內或室外)改變時，系統能作適當的轉換並讓使用者不會感覺有任何異狀，即系統不因環境的改變而使其定位應用程式中斷服務。無接縫換手(seamless handoff)定位系統整體架構如圖四所示。

   此系統是一個主從式(client-server)的架構，每個部分由數個模組所組成，在客戶端(client)的主要工作：(1)進行環境的偵測及收集有關的位置訊息，並將此位置訊息送至伺服端(server)；(2)將使用者的位置呈現在網頁上。而伺服端(server)的主要工作也有兩項：(1)將客戶端送來的位置訊息配合定位的技術來估計使用者目前所在的位置，並將此位置資訊儲存在資料庫中；(2)查詢資料庫來取得使用者的位置資料，並將其回傳給使用者，讓使用者端能以網頁的方式呈現位置資訊。
3. 地理位置資訊模組：本模組內容為地理位置的資訊，我們將實作環境下的地圖資訊（交通大學校園地形圖）以經緯度均分的方式將其切割成數個小區域圖，並分別儲存在資料庫中，當位置追踪服務模獲得其位置資訊（室內參考位置或室外的經緯度資料）後，便以此向地理位置資訊模組要求適當的地圖資訊，並以網頁的方式呈現地圖結合使用者位置。
4. 位置追踪服務模組：使用者透過網路存取的方式與位置追踪服務模組溝通，以取得位置追踪服務。使用者依其權限成功的登入本模組後，管理者權限的使用者可以進入監控畫面，監控(追踪)本系統上所有使用者的位置資訊，而一般權限的使用者僅能看到個人位置資訊的畫面。所有的畫面呈現都會根據使用者的位置改變來動態的調整使用者目前所在的位置。



圖四：整合室內及室外的無接縫換手(seamless handoff)定位系統之架構圖。

五、結果與討論(含結論與建議)

本年度計畫所完成的工作項目除了根據所提之計畫書內容逐步整合各模組之功能，完成位置追蹤服務的應用系統外，也持續針對定位的方法加以研究、改進，我們觀察出多階訊號強度的特性應用在細胞格為主的定位方法下可以提升定位的準確度，因此我們作了相關的研究（如表一及表二）。根據模擬的結果顯示（表一），多階訊號強度在 10% (20%)的參考點損壞情況下, 平均位置估計誤差只有 11.92%R (18.48%R)，其中 R 為參考點訊號涵蓋範圍的半徑。另外，多階訊號強度在 20%的訊框遺失率情況下, 平均位置估計誤差只有 10.65%R（請參考表二）。

表一：在不同比率的參考點損壞情況下，平均位置誤差

參考點損壞率	無法定位	平均位置誤差
0% (Optimal)	0	5.869%R
1%	2	6.425%R
5%	66	8.769%R
10%	280	11.917%R
20%	1310	18.482%R

*R 是參考點的訊號涵蓋範圍的半徑

表二：在不同的訊框遺失率下，平均位置誤差

訊框遺失率	平均位置誤差
0% (Optimal)	5.869%R
1%	6.092%R
5%	6.99%R
10%	8.161%R
20%	10.65%R

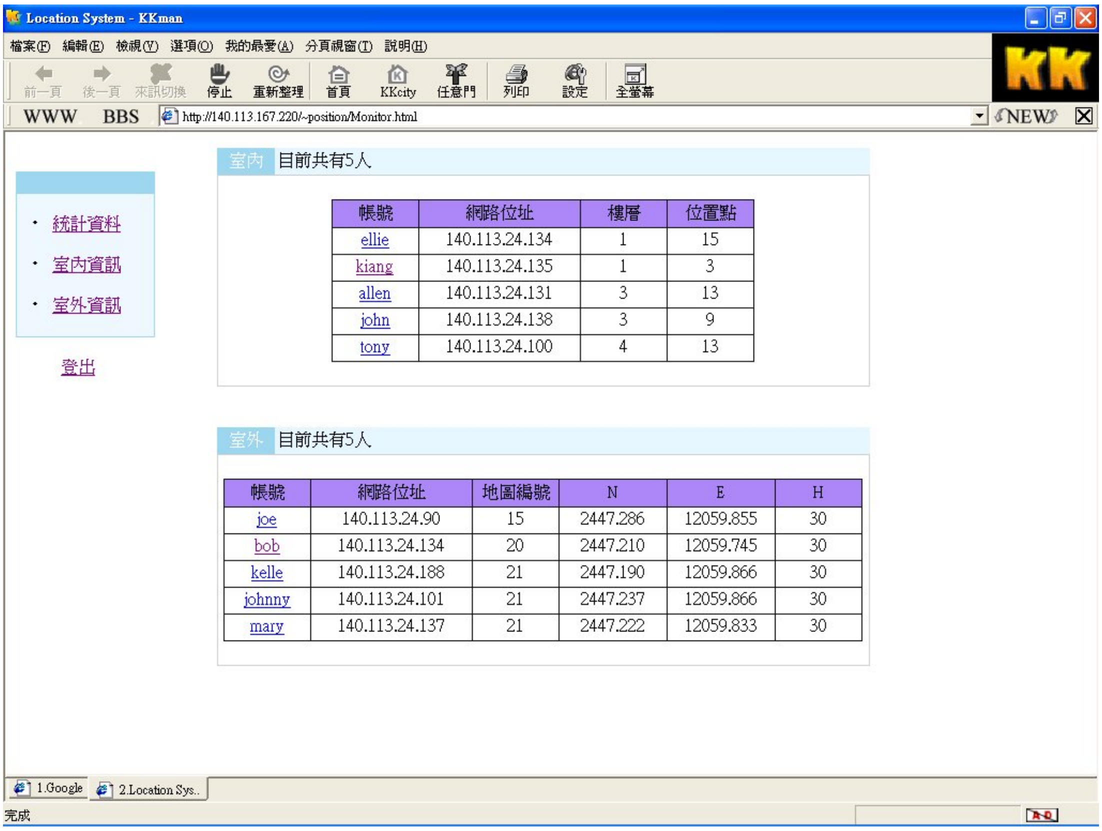
*R 是參考點的訊號涵蓋範圍的半徑

實作位置追蹤服務的應用系統其顯示介面如圖五～十所示。圖五是位置追蹤服務系統主畫面，當管理者登入本系統後，就可以看到此一畫面（系統設定登入後顯示室內的人員追蹤），可由左方的選項選擇位置資訊的呈現（如圖六所示）亦或是室外位置資訊的呈現（如圖七所示）。其中室外的定位呈現方式是將大地圖按其經緯度分割，不會因為大地圖的關係而造成人員追蹤時無法適時的反應其位置變化的結果。

另外，我們將此實作的系統搭配特殊設計的自走車及系館所提供的網路監控影像，除了可以達成位置追蹤的功能外，也可以利用自走車移動至指定的地點並傳回影像。此特殊設計的自走車是結合無線網路模組[22]、無線攝影機模組、RS232/UART 轉換介面模組以及 8051 單晶片自走車。我們利用 C 語言和 C++寫作 Socket 網路程式，在無線網路模組與監控者間建立伺服器端(Server)與使用者端(Client)的連線。而 8051 單晶片是利用組合語言來直接控制自走車的行進。藉由整合這兩者的功能，使得監控者能夠遠端遙控自走車，並透過無線網路即時回傳現在環境影像給監控者。

圖八是利用交通大學工程三館四樓之網路攝影機所監控的畫面。而圖九是我們特殊設計的自走車實物，該自走車與使用者連線後，配合使用者介面的控制便可以直接利用無線網路加以控制其行進方向（前進、左轉及右轉），亦可同時傳回自走車前方影像，傳回之即時影像監控畫面如圖十所示。

圖五：系統主畫面



室內 目前共有5人

帳號	網路位址	樓層	位置點
ellie	140.113.24.134	1	15
kiang	140.113.24.135	1	3
allen	140.113.24.131	3	13
john	140.113.24.138	3	9
tony	140.113.24.100	4	13

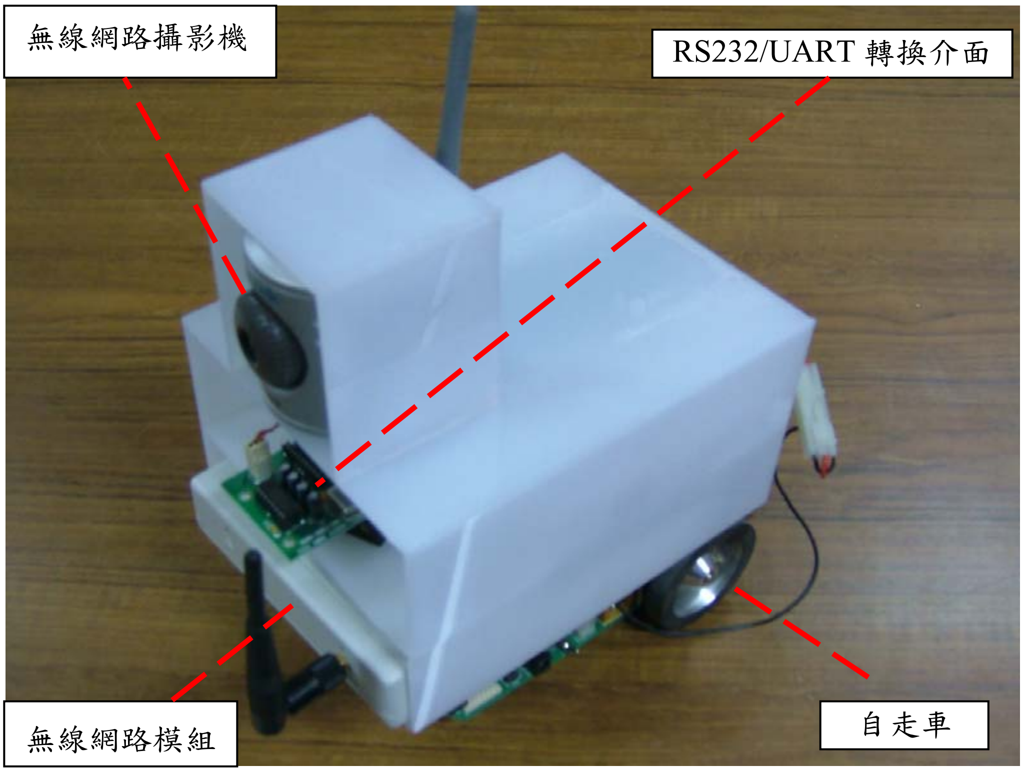
室外 目前共有5人

帳號	網路位址	地圖編號	N	E	H
joe	140.113.24.90	15	2447.286	12059.855	30
bob	140.113.24.134	20	2447.210	12059.745	30
kelle	140.113.24.188	21	2447.190	12059.866	30
johnny	140.113.24.101	21	2447.237	12059.866	30
mary	140.113.24.137	21	2447.222	12059.833	30

圖六：室內位置資訊的呈現

圖七：室外位置資訊的呈現

圖八：交通大學工程三館四樓網路攝影機之監控畫面



圖九　無線監控自走車實物


圖十：即時影像監控畫面

七、參考文獻

[1] N. Davies, K. Cheverst, K. Mitchell and A. Efrat, “Using and determining location in a context-sensitive tour guide”, Computer, vol. 34(8), Aug. 2001, pp. 35-41.

[2] T. S. Rappaport, J. H. Reed, and B. D. Woerner, “Position location using wireless communications on highways of the future,” IEEE Communications Magazine, pp. 33-41, Oct. 1996.

[3] J. M. Zagami, S. A. Parl, J. J. Bussgang, and K. D. Melillo, “Providing universal location services using a wireless E911 location network”, IEEE Communications Magazine, vol.36(4), Apr. 1998, pp. 66-71.

[4] E. G. Masters, C. Rizos, and B. Hirsch, “GPS...more than a real world digitizer”, IEEE Position Location and Navigation Symposium, 1994, pp. 381-387.

[5] K. Chadha, “The Global Positioning System: Challenges in Bringing GPS to Mainstream Consumers”, Proc. of IEEE International Conf. on Solid-State Circuits, 1998, pp. 26-28.

[6] Rong-Hong Jan and Yung Rong Lee, "An indoor geolocation system for wireless LANs," ICPP Workshops 2003, pp. 29-34.

[7] D. Niculescu and B. Nath, “DV based positioning in ad hoc networks,” Kluwer Journal Telecommun. Systems, vol. 22, no. 1, Jan. 2003, pp. 267–280.

[8] N. Bulusu, J. Heidemann, and D. Estrin, “GPS-less Low Cost Outdoor Localization For Very Small Devices,” IEEE Personal Communication, vol. 7, Oct. 2000, pp.28-34.

[9] K.-F. Ssu, C.-H. Ou and H. C. Jiau, “”Localization with mobile anchor points in wireless sensor networks,” IEEE Transactions on Vehicular Technology, vol. 54, no. 3, May 2005, pp. 1187-1197.

[10] C. Drane, M. Macnaughtan, and C. Scott, “Positioning GSM telephones,” IEEE Communications Magazine, vol. 36(4) , Apr. 1998, pp.46-54.

[11] J. Bensche, J. Cooke, E. Job, T. Luke, J. Kvaal, and N. Swatland, “Investing in The Wireless Location Services Market,” Lehman Brothers Report, Sep. 2000.

[12] G.M. Djuknic, and R.E. Richton, “Geolocation and assisted GPS”, Computer, vol. 34(2), Feb. 2001, pp. 123-125.

[13] E. Kotsakis, A. Caignault, W. Woehler, and M. Ketselidis “Integrating Differential GPS data into an Embedded GIS and its Application to Infomobility and Navigation”, 7th EC-GI & GIS WORKSHOP EGII -Managing the Mosaic Potsdam, Germany, June 13-15, 2001.

[14] P. Bahl, and V. Padmanabhan, ”RADAR: An In-Building RF Based User Location and Tracking System,” Proc. of IEEE INFOCOM, vol. 2, Mar. 2000, p.775-784.

[15] H.-C. Chu, and R.-H. Jan, "A cell-based location- sensing method for wireless networks," Wireless Communications and Mobile Computing, vol. 3, no. 4, pp. 455-463, June 2003.(SCI)

[16] R.-H. Jan, H.-C. Chu, and Y.-F. Lee, "Improving the accuracy of cell-based positioning for wireless networks", Computer network, vol. 46, pp. 817-827, Dec. 20, 2004.

[17]R. Want, A. Hopper, V. Falcao, and J. Gibbons, ”The Active Badge Location System,”

ACM Transactions on Information Systems, vol. 40, January 1992, pp.91-102.

[18] J. Hightower,G. Boriello, and R.Want, “SpotON: An indoor 3D location sensing technology based on RF signal strength,” Univ. of Washington, Tech. Rep. UW CSE 00-02-02, Feb. 2000.

[19] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, “The cricket location-support system,” in Proc. ACMInt. Conf. Mobile Computing Networking (MOBICOM), Boston, MA, Aug. 2000, pp. 32–43

[20] S. Helal, B. Winkler, C. Lee, Y. Kaddoura, L. Ran, C. Giraldo, S. Kuchibhotla and W. Mann, “Enabling location-aware pervasive computing applications for the elderly”, Proc. of the First IEEE International Conference on Pervasive Computing and Communications, 2003, pp. 531 - 536.

[21] Y.-H. Gau, H.-C. Chu, and Rong-Hong Jan, "A Weighted Multilateration Positioning Method for Wireless Sensor Networks," Workshop on Wireless, Ad Hoc, and Sensor Networks(WASN), National Central University, Session A1, pp.3-8, Taiwan, August 1-2, 2005.

[22] Host AP driver for Intersil Prism2/2.5/3, hostapd, and WPA Supplicant. Available: http://hostap.epitest.fi

本年度已發表之論文如下：

[A1] Rong-Hong Jan and Wen-Yueh Chiu, "An approach for seamless handoff among mobile WLAN/GPRS integrated networks," accepted and to appear in Computer Communications. (附件一)

[A2] Rong-Hong Jan, Ching-Peng Lin, and Maw-Sheng Chern, “An optimization model for Web content adaptation,” accepted and to appear in Computer Networks. (附件二)

[A3] Hung-Chi Chu and Rong-Hong Jan, "A GPS-less positioning method for sensor networks," Proc. of the 1st International Workshops on Distributed, Parallel and Network Applications(DPNA) 2005. (附件三)

[A4] Yu-He Gau, Hung-Chi Chu, and Rong-Hong Jan, "A Weighted Multilateration Positioning Method for Wireless Sensor Networks," Workshops on Wireless, Ad Hoc, and Sensor Networks(WASN), National Central University, Taiwan, 2005. (附件四)

ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computer Communications xx (xxxx) 1–10

computer communications

www.elsevier.com/locate/comcom

An approach for seamless handoff among mobile WLAN/GPRS integrated networks☆

Rong-Hong Jan*, Wen-Yueh Chiu

Department of Computer and Information Science, National Chiao Tung University, Hsinchu, 30050 Taiwan, ROC

Received 8 October 2003; revised 4 March 2005; accepted 16 March 2005

Abstract

Wireless local area networks (WLAN) and General Packet Ratio Service (GPRS) networks are two of the most widely used wireless network systems. In this paper, we propose a mobility support for mobile host roaming between WLAN and GPRS networks. In addition, a handoff decision model is presented to reduce the latency of the handoff procedure. The experimental results, including throughput, packet delay and handoff latency are given to show the performance of our approach.

Keywords: Wireless LAN; GPRS; Seamless hand-off

1. Introduction

Over the past decade, Internet use has exploded with people gaining rich information from the World Wide Web. Meanwhile, technology has made wireless devices smaller, less expensive and more powerful. Wireless networks have become increasingly popular for accessing the Internet because they enhance mobility. People can connect to the Internet and remain on-line while roaming.

Wireless local area networks (WLAN) are the most widely-used local wireless network system in schools, offices, airports, etc. Some organizations provide free WLAN service for their members. Although the data rate for WLAN can run up to 54 Mbit/s based on the IEEE 802.11a standard, its coverage area, known as its *hot spot*, is too small. There is no WLAN service outside of the hot spot. Thus, users cannot leave the hot spot until all transmissions are complete. On the other hand, General Packet Ratio Service (GPRS), based on the GSM system, provides high mobility and 'always on' connectivity for mobile users. However, the GPRS data rate is only up to 115 Kbit/s, and the cost of data transmission is much greater than that for WLANs. Therefore, mobile users like to have the two systems in their mobile hosts (MH). This lets people use the WLAN to access the Internet wherever it is available, yet switch to a GPRS network when they leave the hot spot.

A procedure that enables roaming between GPRS network and WLAN is known as *vertical handoff*. In order to achieve vertical handoff, several issues, such as handoff decision making, authentication, and mobility management, have to be addressed.

Some work on handoff decision making for WLAN/GPRS integrated networks has been reported in the literature. In [1], the physical layer parameters, such as received signal strength and signal decay, are used as decision criteria to trigger the handoff. In [2], they present a roaming scheme based on the relative bandwidth of WLAN and GPRS. In addition to physical layer parameters and network bandwidth, the network conditions, such as user preference, packet delay and packet loss, may also be the criteria for handoff decision.

After vertical handoff decision is made, we face the authentication issue. The authentication for WLAN/GPRS integrated networks can be divided into two approaches, SIM-based and WLAN-based. In SIM-based authentication, the roaming users are authenticated and charged using GSM Subscriber Identity Module (SIM) [3]. The SIM-based

☆This research was supported in part by the Communications Software Technology Project of Institute for Information Industry and in part by the National Science Council, Taiwan, ROC, under grant NSC 93-2219-E-009-002 and NSC93-2752-E-009-005-PAE.

* Corresponding author. Tel.: +886 3 5731637; fax: +886 3 5721490.
E-mail address: rhjan@cis.nctu.edu.tw (R.-H. Jan).

0140-3664/$ - see front matter © 2005 Published by Elsevier B.V.
doi:10.1016/j.comcom.2005.03.004

authentication works well if the WLAN system is owned by the GPRS operator. In [4], a WLAN-based authentication is proposed for the WLAN/GPRS integrated networks in which WLAN and GPRS networks can be owned by different service providers or operators. In WLAN-based authentication approach, an Authentication, Authorization, and Accounting (AAA) server, which is installed in both WLAN and GPRS networks, is required.

Next, the mobility management scheme that maintains a connection's continuity should be considered during vertical handoff. Two major theories for mobility support have been proposed: routing-based approach and pure-end system approach.

In the routing-based approach, the home network has an agent responsible for transferring packets between the correspondent node (CN) and the MH. Mobile IP [5], a standard of mobility support for IPv4 [6] that was drawn up by the Internet Engineer Task Force (IETF), is the most common solution for offering roaming in IP networks. Many studies [7–9] are based on mobile IP to support the host's mobility. This approach does not require modifying the correspondent nodes or its applications. However, mobile IP does have some problems. First, an inefficient data flow, called a triangular routing problem, exists in this kind of approach. Second, mobile IP may not cooperate with network address translation (NAT) protocol because the IP address information was encapsulated in its registration packet. In [10], a number of network-layer (IP-layer) handoff optimization techniques, such as fast router advertisement, fast router caching, and soft handoff, that can improve handoff performance WLAN/GPRS integrated networks are proposed. In [11], a UDP tunneling method is presented to solve the NAT problem for Mobile IP.

In the pure-end system approach, the current TCP/IP structure should be modified to support host mobility. The Migrate Internet Mobility Project [12] is one of this studies proposed by the Laboratory for Computer Science at MIT. By modifying the transport layer and its applications at the end users, no agent to transfer the data is needed and better performance is achieved, compared to the routing-based approach. Clearly, the pure-end system approach is not compatible with the current network environment and is difficult to promote.

The paper applies a routing-based approach to integrate WLAN and GPRS networks to provide roaming service. The proposed method can be simply applied to the current network system without updating hardware or software. Thus, the method can be promoted easily. The architecture of our proposed system differs from the standard mobile IP. We use a virtual network interface card (virtual NIC) instead of a foreign agent. Since IP address starvation is a serious problem, NAT protocol [13] is used widely. In order to inter-operate with NAT, the UDP tunneling method is applied in this system.

A handoff decision model which is designed to reduce the packet loss rate and increasing throughput is also proposed in this paper. By properly setting, both the threshold and hysteresis, the MH can handoff before leaving the hot spot, thus avoiding the so-called ping-pong effect. A pre-handoff mechanism is proposed to place the GPRS in a ready state before handoff occurs.

The remainder of this paper is organized as follows. In Section 2, we describe our system architecture, the message flow chart, and handoff decision model. Section 3 discusses the implementation and performance of the proposed method, and a conclusion is given in Section 4.

2. System architecture

In this section, we describe the proposed seamless service framework including system architecture, the functionality of each component, and the message sequence chart of the process.

2.1. Seamless handoff agent and proxy

Note that a host can be reached by other hosts on the Internet, depending on its IP address. If all hosts are fixed on the Internet, the packets can be routed to their destinations by examining the prefix of the destination IP address at each router. However, this mechanism limits the host's mobility. For example, once an MH with a home IP address visits a foreign network, two problems will occur. One, there is no node in the home network to deal with packets destined to the MH. And two, no node in the home network knows where the MH is. One way to make the MH reachable is to give the MH a new IP address with the foreign network's prefix. However, all previous connections are broken.

In order to provide mobility support for the MH, as shown Fig. 1, a seamless handoff proxy (SH proxy) server is added to the home network for forwarding the packets destined to the MH. In addition, a software agent, called as seamless handoff agent (SH agent), is installed in the MH for tunneling mechanism and handoff management.

Now, follows a brief description of the system's operation. As shown in Fig. 2, let H address denote the home IP address of the MH. SH agent binds H address to a virtual NIC. Note that the MH has two physical NICs, the WLAN NIC and the GPRS NIC. When MH moves to WLAN, it receives a new IP address, denoted as W address, from the foreign network. SH agent binds W address to the WLAN NIC and registers address pair (H,W) to the SH proxy. Similarly, when the mobile host moves to the GPRS network, it registers address pair (H,G) to the SH proxy where address G represents the new IP address received from GPRS.

The applications in the MH use H address as the source address to send the packet. The virtual NIC in the SH agent accepts the packet and then the SH agent tunnels the packet to the SH proxy. The SH proxy decapsulates the packet and transfers it to the CN. If CN has a packet ready to send to

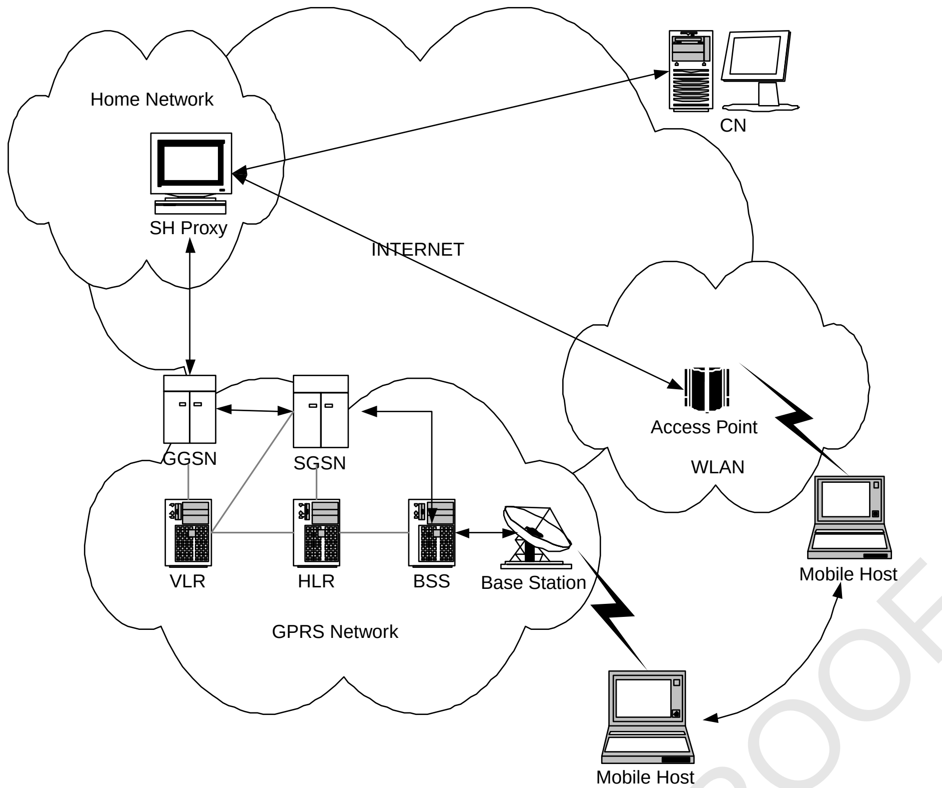


Fig. 1. System architecture.

MH, it uses H address as the destination address. The packet will be delivered to the home network. The SH proxy intercepts the packet destined to the H address and checks the address binding table to see if the MH is connecting to the WLAN or the GPRS. If the MH is connecting to WLAN, then the SH proxy tunnels the packet with the W address as the destination address to MH. After the MH receives the packet, the SH agent decapsulates the packet and forwards to the application via the virtual NIC.

In the following, we give a detailed description of the SH agent and SH proxy.

2.1.1. SH agent

The functions of the SH agent can be divided into five modules: the virtual NIC, sending module, receiving module, control module, and user interface. Each module is a thread independently operating in the system. Fig. 3 depicts the relationships of modules in the SH agent. Descriptions of the modules are given as follows:

1. The virtual NIC is software that simulates a physical network adapter. The MH's home IP address is bound with this virtual NIC. The connections established by applications in MH use this virtual adapter to communicate with other nodes. The existing connections will not be broken while roaming because the virtual NIC would never change its IP address.
2. The sending module is responsible for transmitting packets to the SH proxy. It catches packets from the virtual NIC sent by applications, and then encapsulates the packets as payload for new UDP packets (known as UDP tunneling). Finally, it sends those tunneling packets to the SH proxy with port number 5150 via physical network adapters using existing routing rules set by the control module.

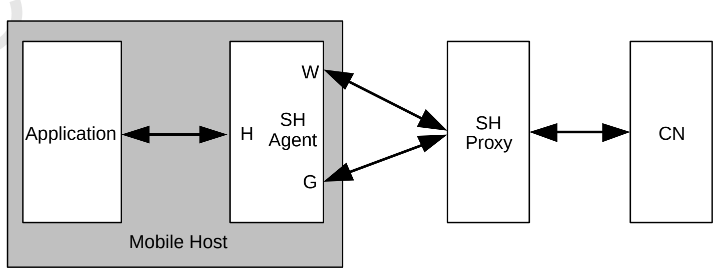


Fig. 2. Brief account of the system operation.

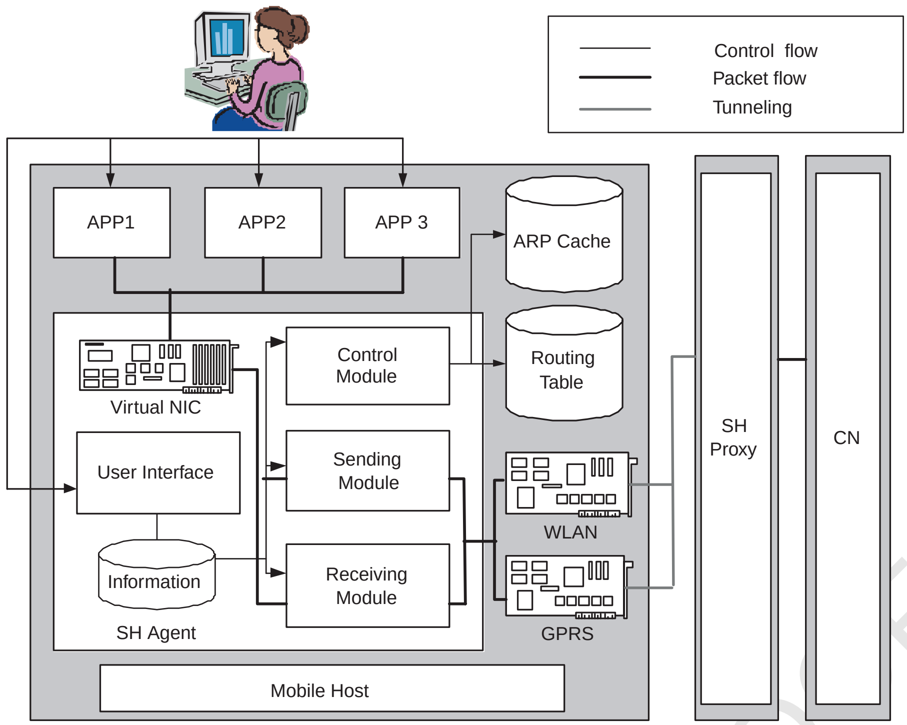


Fig. 3. Modules in SH agent.

3. The receiving module is responsible for receiving packets from the SH proxy. It receives tunneling packets with port number 5051 from the SH proxy, and then performs UDP decapsulation to get original packets. Finally, it forwards the packets to the virtual NIC; after that, the packets can be received by the corresponding application.
4. The control module is responsible to monitor all states of NICs, to select the working NIC and to execute corresponding procedures such as registration, handoff, and pre-handoff procedures. After the SH agent starts operation, the control module initializes the routing table and ARP cache. The routing table is modified according to the NICs' priority provided by the user. The WLAN or GPRS NIC is assigned as the working NIC for the tunneling packet. The control module also keeps an eye on each NIC's connection state until the SH agent stops the service. If the working NIC is disconnected, the control module should find another connectable NIC to communicate with the SH proxy. If a NIC with a higher priority than the current working NIC becomes connectable, the control module should switch to the NIC with the higher priority. The control module sends control messages using the UDP channel established by the receiving module. Thus, the SH proxy can collect both the IP address and port number from the receiving channel when the registration request or handoff request arrives. Even if the IP address or port number is changed by the NAT gateway, the SH proxy can still get a routable address to tunnel packets back to the SH agent.
5. Finally, a user interface lets the user assign some system parameters, such as the SH proxy IP address, home IP address, priority of each NIC, and so on. The user interface also shows the states of the system.

To manage all available NICs in MH, we built a table to record necessary information for each NIC. Each entry in the table includes the following fields: the NIC's name, description, current IP address of the NIC, gateway information, connection type, priority of the NIC, current state, and its index in the operating system. Note that the connection type and priority of the NIC should be provided by the user; this helps one to adopt a proper handoff policy, and other information can be collected by the SH agent itself.

2.1.2. SH proxy

Similarly, functions of the SH proxy can be divided into four modules as given below and each module is a thread operating independently in the system (see Fig. 4).

1. The 'FromSHAgenttoCN' module receives tunneling packets from the SH agent, decapsulates those packets, and sends the original packets to their target CNs.
2. The 'FromCNtoSHAgent' module catches packets destined to the MH that has registered to the SH proxy, and it also encapsulates these packets and tunnels them back to the corresponding SH agent.
3. The control module deals with control messages like registration or handoff requests. It collects necessary

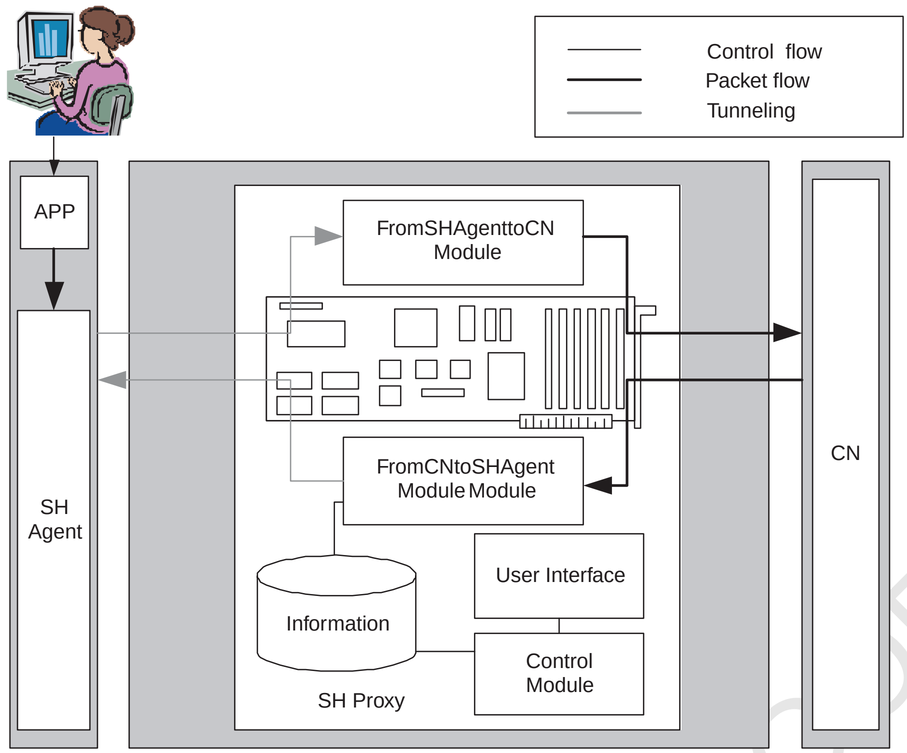


Fig. 4. Modules in SH proxy.

information to update SH agent information, and sends corresponding messages back to SH agent. The SH proxy stores information for registered SH agents in a table called 'SH agent information'. Each entry in the table contains the registered SH agent's Home IP address, current IP address, and port number. Note that the current IP address and port number are obtained from the header of registration request or handoff request by the SH proxy itself. The packet does not carry this information in its payload.

4. The system administrator can configure the SH proxy by user interface, and it also shows internal information like the states of currently registered SH agents.

2.2. Packet structure

The packet format for the proposed method is introduced as follows. As shown in Fig. 5, four fields, the type, ID, code, and flag are defined. The type field gives the indication of packet types. The packet types and related information are summarized in Table 1. The ID field contains the SH agent's home IP address. We use this field to identify the source node for the packet. The code field records the SH proxy's reply. The default value is zero.

IP header	UDP header	Type	ID	Code	Flags	Extension

Fig. 5. Packet fields.

The fourth field is reserved now. Depending on different packet types, some extension is attached to the packet.

2.3. Handoff decision model

The Handoff Decision Model is designed in the control modular of the SH agent. The goal of this model is to decide when to handoff so as to improve system performance. We assume the following: One, that GPRS is always available anywhere. And two, if WLAN is available, WLAN is chosen as the access network, because it has both a higher data rate and lower cost.

The received signal strength (RSS) of WLAN varies with many factors, including landforms, obstacles, power strength of access point, etc. In general, RSS is related to the distance between the transmitter and the receiver.

Table 1
Packet types

Packet name	Type	Code	Extension
Tunneling packet	0	0	Original packet
Registration request	1	0	
Registration reply	2	1 = accept 2 = reject	MAC address of gateway
Pre-handoff request	3	0	
Pre-handoff reply	4	1	
Handoff request	5	0	
Handoff reply	6	1	
SH agent deregistration	7	0	
SH proxy stop	8	1	

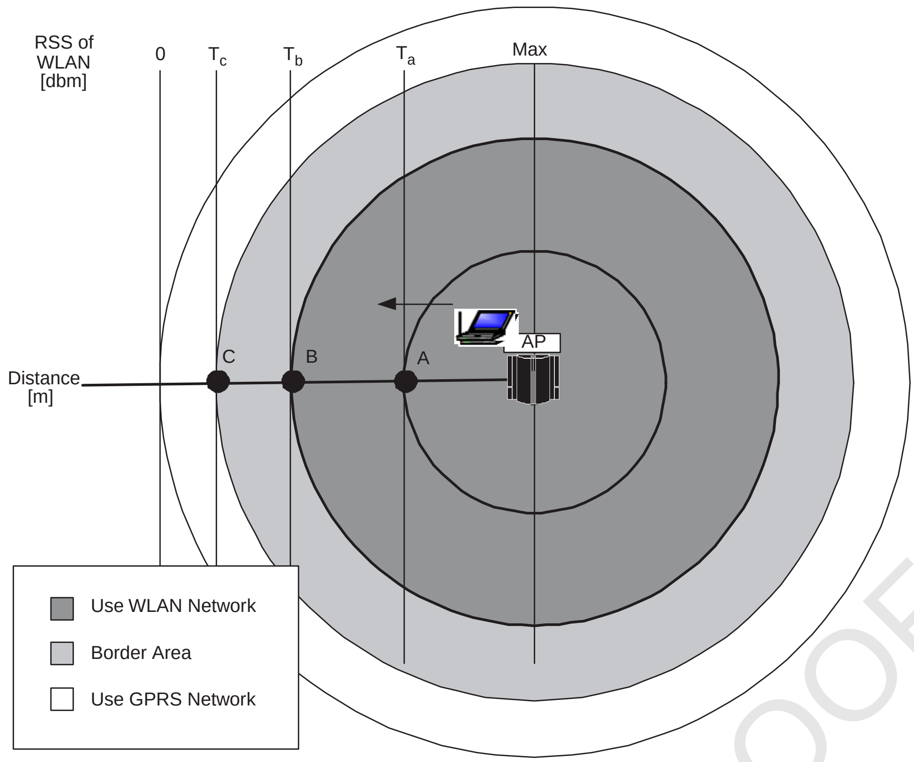


Fig. 6. Variables in decision model.

According to [20], the following signal propagation model may hold:

$$P(r) = P(r_0) - 10\alpha \log(r/r_0) \quad (1)$$

where $P(r)$ is signal power, in *dbm*, received by a given MH whose distance to the transmitter is r meters; $P(r_0)$ is the signal power at a reference point whose distance to transmitter is r_0; the parameter α called the exponent value, indicates the rate of path loss.

As shown in Fig. 6, suppose that an MH moves out from the access point, and if the RSS is lower than threshold T_c, the MH should perform handoff from WLAN to GPRS. In a GPRS network, if MH requests to be active, then SGSN moves the MH from standby to ready state. In ready state, the MH is attached to GPRS mobility management (GMM) and thus MH can receive and send data for all relevant service types. Thus, before the RSS decreases to T_c in the WLAN, MH should send a request to GPRS for activating. Otherwise, the MH must wait until SGSN moves it from standby state to ready state. This causes both longer packet delay and packet loss.

In our decision model, prior to data transmission handoff from the WLAN to the GPRS networks, the SH agent sends a pre-handoff request to the SH proxy via the GPRS network. This causes the SGSN to move the MH from standby to ready state. We call this procedure 'pre-handoff.' Suppose that (1) when MH reaches point B (see Fig. 6), it sends a pre-handoff packet, and (2) when MH reaches point C, a handoff occurs (i.e. the RSS is T_c). The distance D_{bc} between points B and C can by determined by:

$$D_{bc} = t_h \times v_h$$

where v_h is the moving speed of MH and t_h is the time for performing the pre-handoff mechanism. A reference point A with distance D_a to the access point is selected to measure the RSS in WLAN, say, T_a. One can then estimate the distance D_c between AP and the point C by applying (1) as follows.

$$D_c = D_a 10^{(1/10\alpha)(T_\alpha - T_c)}.$$

Similarly, one can apply (1) to find RSS T_b for the point B by:

$$T_b = T_c + 10\alpha \log(D_c/(D_b - D_{bc}))$$

Therefore, when MH is moving and measures the RSS of AP, if the RSS decreases to T_b, the pre-handoff request should be sent periodically. This keeps GPRS NIC and the state of HM in SGSN in a ready state. When the RSS decreases to T_c, every thing is ready for handoff. This way, one can reduce both the packet delay and packet loss.

2.4. Message sequence chart

Below is an example to illustrate all of the message sequences in this current approach. As shown in Fig. 7, the IP address of the home network is 140.113.167.0. The two visited networks are the WLAN network with an address of

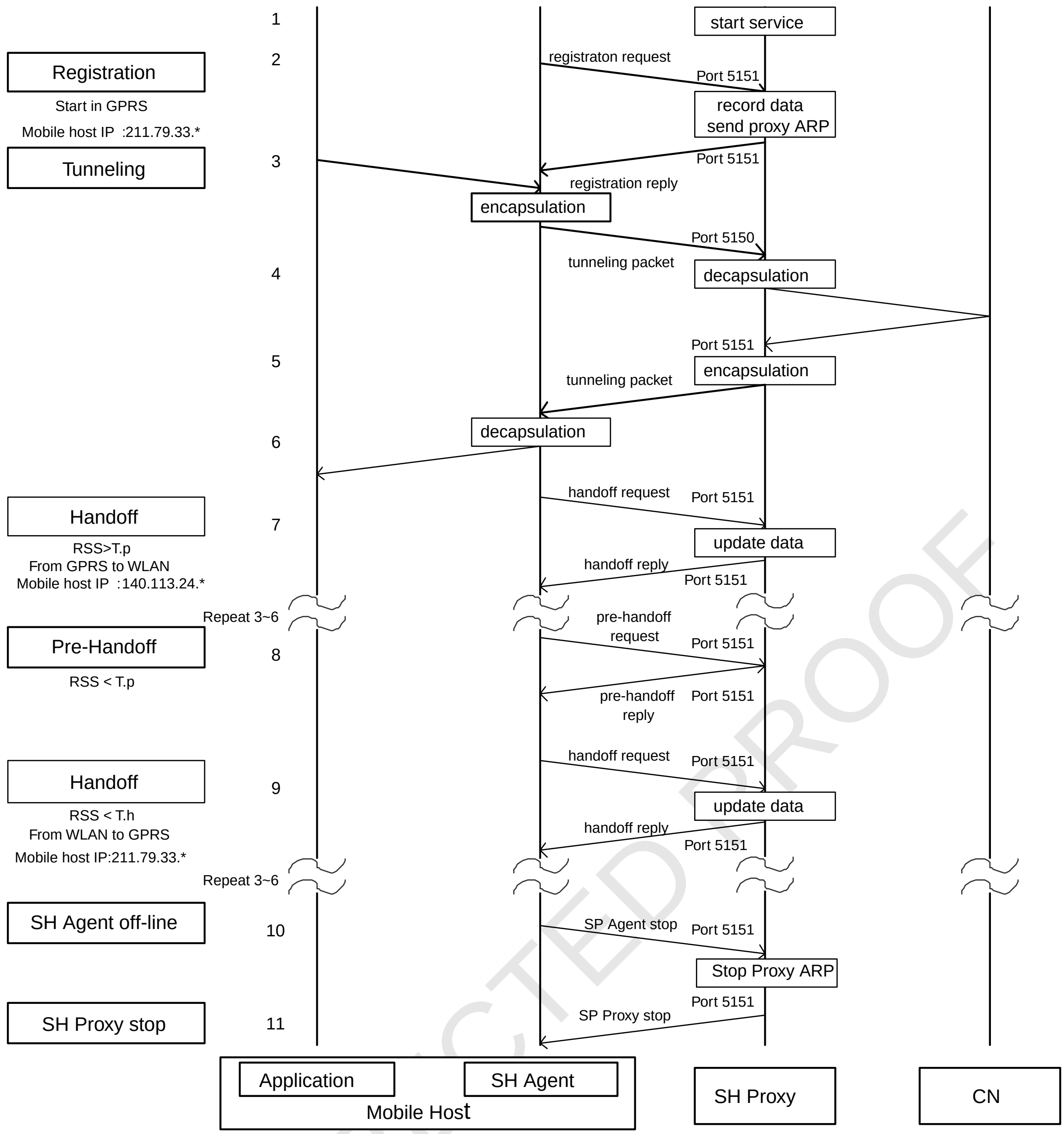


Fig. 7. Message sequence chart.

140.113.27.0 and the GPRS network with 211.79.33.0. All IP addresses for MH are given by the dynamic host configuration protocol (DHCP) servers. Assume that MH starts from the GPRS network, moves to the WLAN network and finally returns to the GPRS network. All messages in this approach are listed below:

1. When the SH proxy starts service in the home network, it sends an ARP request to get the gateway's MAC address. Then, two UDP channels are opened for the SH agent. Ports 5150 and 5151 are assigned to receive tunneling packets and control packets, respectively. Note that the SH proxy also uses port 5151 to send tunneling packets back to the SH agent.
2. The MH starts at the GPRS network and its SH agent sends registration request to the SH proxy. If the SH proxy accepts the request, it creates a new entry in the 'SH agent information' table to record related information. The SH proxy then sends proxy ARP to inform other nodes of the substitution for this MH, and sends the registration reply to the SH agent.
3. After a successful registration, the application in MH can communicate to CN. The application uses virtual NIC to send packets. Thus, the source IP address of the packets is 140.113.167.*. Next, the sending module of the SH agent encapsulates the packets into a UDP tunneling packet with a source IP address of 211.79.33.* and destination port 5150, and sends

the tunneling packets to SH proxy via GPRS NIC. After tunneling packets are received by the SH proxy, the 'FromSHAgenttoCN' module decapsulates the packets. Then, the original packets are transmitted to their destination using normal routing method.

4. CN receives the packets and transmits them back to the MH.
5. The SH proxy collects the packets that destined to the MH. By checking the IP address, the SH proxy finds the corresponding entry in the 'SH agent information' table and retrieves the current IP address for the MH. The 'FromCNtoSHAgent' module encapsulates these packets with a destination IP address of 211.79.33.* and sends the tunneling packets to the SH agent via the established UDP channel.
6. When the tunneling packets arrive at the SH agent, the receiving module decapsulates these packets and then delivers them to applications via virtual NIC.
7. If the control module decides to perform a handoff from GPRS to WLAN, it changes the routing table and sends a handoff request via WLAN NIC to the SH proxy. This way, the SH proxy can update the IP address and port number. After the handoff is complete, the SH agent uses WLAN network for transmission.
8. When the RSS of WLAN is lower than T_b, the control module decides to start the pre-handoff mechanism. Thus, the SH agent sends the per-handoff message to the SH proxy via GPRS NIC to prepare handoff. The SH proxy simply returns this message to the sender. Note that the pre-handoff mechanism is done to change the MH state in the GPRS network from standby to ready.
9. The control module performs a handoff from WLAN to GPRS if RSS of WLAN is lower than T_c. After a successful handoff, the SH agent uses GPRS NIC to send packets.
10. If the SH agent decides to go off-line, it sends an 'SH agent stop' message to the SH proxy. Then, the SH proxy removes the entry of the SH agent from the 'SH agent information' table.
11. If the SH proxy wants to stop the service, it sends an 'SH proxy stop' message to all registered SH agents.

3. Implementation and evaluation

The following section describes the developing environment, test bed and performance evaluation for the proposed system.

3.1. Developing environment

A laptop PC with two physical NICs, an Audiovox RTM 8000 GPRS card and a Lucent Orinoco volt 3.3 802.11b WLAN card, were used as the MH. An SH agent was developed in this PC. The operating system in the MH was Microsoft windows XP professional. We used a desktop PC as the SH proxy. The operating system in the SH proxy was NT4.0. The CN was a desktop PC with some endpoint testing software.

The IP address of the home network was 140.113.167.0. The SH proxy's IP address was 140.113.167.205 and the mobile host's home address was 140.113.167.242. Two networks visited were the WLAN network with an address of 140.113.27.0 and the GPRS network with 211.79.33.0, respectively. The MH's IP address in the visiting network was automatically assigned by the DHCP server. The operator of the GPRS network was Chunghwa Telecom, is the largest telecommunications company in Taiwan. Note that the ready timer of the GPRS mobility management system expired after the MH idled more than 44 s, and the transit time from standby to ready was 1–2 s. The CN was at National Central University, ChungLi City, Taiwan, with an IP address of 140.115.83.240. The distance between the MH and CN was 40 km.

3.2. Performance analysis

The following experiment shows the packet delay, the amount of packet loss caused by handoff latency, and the benefit of the handoff decision model. In addition, the compatibility of this approach with existing applications is examined.

3.2.1. Packet delay

The packet delay of this method is caused mainly by the tunneling mechanism. Since our method is based on the routing-based approach, the triangular routing problem exists. The round trip time (RTT) between the MH and CN was measured in the experiment. The RTT was composed of the packet transmitting time between the NH and SH proxy, the time of performing encapsulation/decapsulation mechanism, and the RTT of the connection between the SH proxy and CN. Both the quality of access network and the CPU load significantly influences the RTT.

Figs. 8 and 9 show the RTTs of our method from the MH to CN using the WLAN and GPRS networks, respectively. When the MH was in the WLAN, the RTT varied from 109 to 1999 ms, with an average RTT of 595 ms. (The one-way average was 297.5 ms.) When the MH was in the GPRS, the RTT varied from 992 to 11571 ms, with an average RTT of 2330.7 ms and a one-way average of 1165.4 ms. Compared to the system without mobility support, the one-way delays caused by the mobility support mechanism were 280.5 ms for WLAN and 615.44 ms for GPRS. Table 2 lists the one-way packet delays for the systems with/without mobility support.

3.2.2. Handoff latency

Handoff latency is the latency caused by the handoff procedure. Comparing the handoff latencies between GPRS and WLAN, the latency for a handoff from a WLAN to

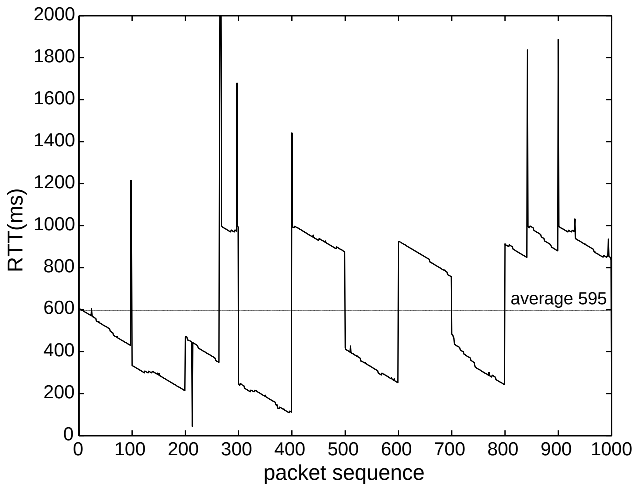


Fig. 8. RTT in WLAN.

a GPRS network is longer. This is because (1) the transition time from standby to ready in GPRS should be considered; (2) the bandwidth of WLAN is much higher than that of GPRS, and (3) the handoff procedure via WLAN is faster than that of GPRS. In the experiment, the MH sent ICMP echo request messages (ping packets) to the CN every 4096 ms.

Three methods for handoffs from WLAN to GPRS network are compared below. Method 1 has a handoff procedure that does not start until the WLAN is unavailable. In Method 2, if the RSS of the WLAN is lower than a given

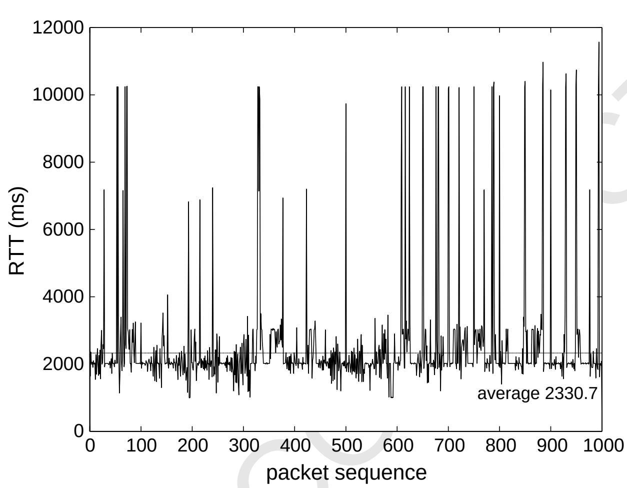


Fig. 9. RTT in GPRS.

Table 2
Packet delays for the systems with and without mobility support

Type	System without mobility support	System with mobility support
WLAN	10–20 ms	297.5 ms
GPRS	550–600 ms	1165.4 ms

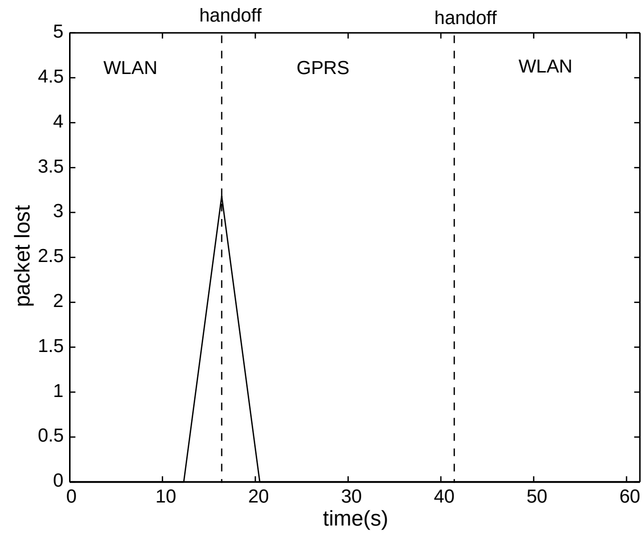


Fig. 10. Average packet lost with no decision model.

threshold, the handoff procedure starts. Method 3, our proposed method uses a handoff-decision model. Figs. 10–12 show the average numbers of packets lost for methods 1, 2 and 3, respectively, during the handoff procedures. Table 3 summarizes the average number of packets lost and handoff latencies for three methods. Packet loss is serious in Method 1 (see Fig. 10), with an average of 3.1818. In stark contrast, however, the average packet loss for Method 3 was a scant 0.1391. The average handoff delays for Methods 1–3 were 13032.9, 4542.3 and 570.1 ms, respectively. Clearly, Method 3, the proposed pre-handoff method promises far better performance.

3.2.3. Throughput evaluation

The throughput of the system is affected by the network bandwidth, the performance of the NIC and the MH CPU. We used ‘Qcheck’ software to generate UDP streams with

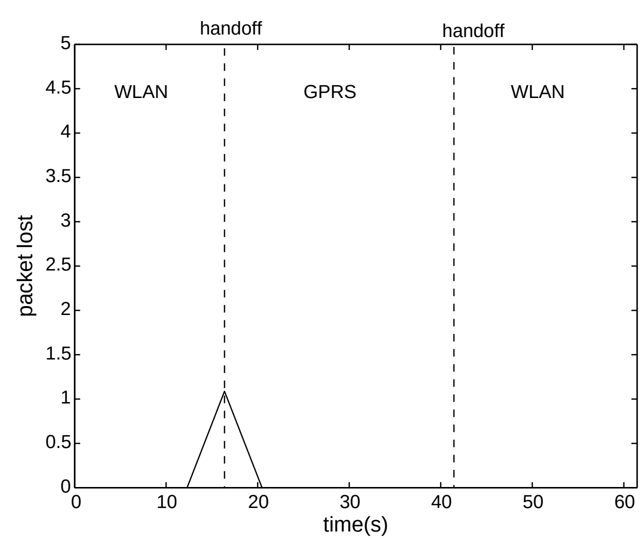


Fig. 11. Average packet lost with RSS decision model.

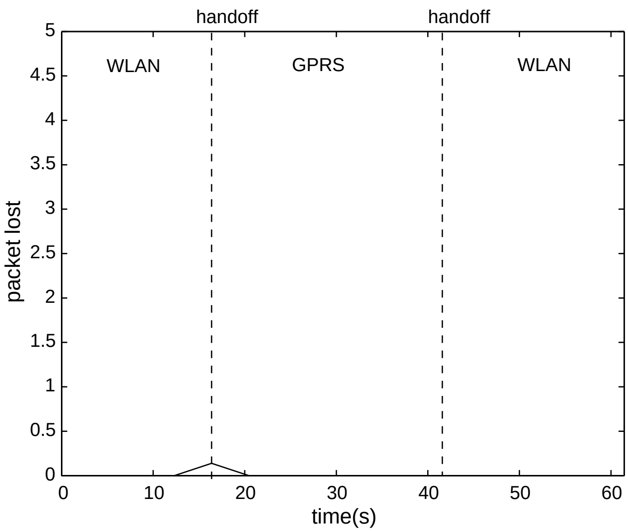


Fig. 12. Average packet lost with our decision model.

Table 3
Packet loss and handoff latency for methods 1—3

	Avg. number of packet loss	Handoff latency
Method 1	3.1818	13032.9 ms
Method 2	1.1089	4542.3 ms
Method 3	0.1391	570.1 ms

600 kbps in WLAN and 50 kbps in GPRS. The average throughput of WLAN was 481.14 kbps, and in GPRS networks, the average throughput was 16.82 kbps.

3.2.4. Compatibility test

Finally, we tested the compatibility of our system. Most of the applications worked normally in our system. Due to the high packet delays in GPRS networks, however, real-time application such as VoIP, did not work well. We hope such delays can be reduced in the future.

4. Conclusion

This paper presents a mobility support method to integrate WLAN and GPRS networks. This approach contains an SH proxy in a home network and an SH agent installed in the MH. A handoff decision model is designed to improve overall performance.

From the operator's point of view, our approach reduces the need to modify the existing environment, and the decision model also reduces the handoff latency from the WLAN to GPRS networks. Hence, operators can simply use this approach to provide roaming services for the integrated networks. From the user's point of view, only an SH agent is needed to install in his or her own mobile equipment, and all existing applications can still be run.

Looking ahead, developing both a security mechanism and billing system for integrated WLAN and GPRS networks might be interesting future work.

5. Uncited references

[14–19].

References

[1] Q. Zhang, C. Guo, Z. Guo, W. Zhu, Efficient mobility management for vertical handoff between WWAN and WLAN, IEEE Commun. Mag. 41 (2003) 102–108.
[2] K. Pahlavan, et al., Handoff in hybrid mobile data networks, IEEE Person. Commun. 7 (2000) 34–47.
[3] J. Ala-Laurila, J. Mikkonen, J. Rinnemaa, Wireless LAN access network architecture for mobile operators, IEEE Commun. Mag. 39 (2001) 82–89.
[4] M. Jiang, J. Chen, Y. Liu, WLAN-centric authentication in integrated GPRS-WLAN networks, Vehicular Technology Conference (VTC), vol. 3, 2003 pp. 2242–2246.
[5] C. Perkins, IP Mobility Support, IETF RFC 2002, 1996.
[6] J. Postel, Internet Protocol, STD 5, IETF RFC 791, 1981.
[7] First Steps Towards UMTS: Mobile IP Services. A European Testbed (FIT-MIP), [online]. Avaliable: http://www.eurescom.de/∼ftproot/web-deliverables/public/P1000-series/P1013/.
[8] M. Ylianttila, M. Pande, J. Makela, P. Mahonen, Optimization scheme for mobile users performing vertical handoffs between IEEE 802.11 and GPRS_EDGE networks, IEEE Global Telecommun. Confer. 6 (2001) 3439–3443.
[9] I. Wu, W. Chen, H. Liao, F. Young, A seamless handoff approach of mobile IP protocol for mobile wireless data networks, IEEE Trans. Consumer Electron. 48 (2002) 335–344.
[10] R. Chakravorty et al., Performance Issues with Vertical Handovers—Experiences from GPRS Cellular and WLAN Hot-spots Integration, Proceeding of the Second IEEE Annual Conference on Pervasive Computing and Communications, March 2004, pp. 155–164.
[11] H. Levkowetz, S. Vaarala, Mobile IP NAT/NAPT Traversal using UDP Tunneling, draft-ietf-mobileip-nat-traversal-07.txt, work in progress, IETF Internet-Draft, Nov. 2002.
[12] Migrate Internet Mobility Project [online]. Available: http://nms.lcs.mit.edu/projects/migrate/.
[13] K. Egevang, P. Francis, The IP network address translator (NAT), IETF RFC 1631, 1994.
[14] C. Perkins, IP in IP Tunneling, IETF RFC 2003, 1996.
[15] C. Perkins, Minimal Encapsulation within IP, IETF RFC 2004, 1996.
[16] S. Hanks, T. Li, D. Farinacci, P. Traina, Generic Routing Encapsulation (GRE), IETF RFC 1701, 1994.
[17] S. Deering, R. Hinden, Internet Protocol, Version 6 (IPv6) Specification, IETF RFC 1883, 1995.
[18] C. Pollini, Trends in handover design, IEEE Commun. Mag. 34 (1996) 82–90.
[19] Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, ISO/IEC 8801-11, ANSI/IEEE Std. 802.11, 1999 Edition.
[20] Y. Chen, H. Kobayashi, Signal strength based indoor geolocation, IEEE Int. Confer. Commun. 1 (2002) 436–439.

附件二

COMPNW 3174
3 August 2005 Disk Used
ARTICLE IN PRESS
No. of Pages 13, DTD = 5.0.1

ELSEVIER

Available online at www.sciencedirect.com
SCIENCE DIRECT®

Computer Networks xxx (2005) xxx–xxx

Computer Networks

www.elsevier.com/locate/comnet

An optimization model for Web content adaptation ☆

Rong-Hong Jan [a,*], Ching-Peng Lin [a], Maw-Sheng Chern [b]

[a] *Department of Computer and Information Science, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu 30050, Taiwan, ROC*

[b] *Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Hsinchu 30043, Taiwan, ROC*

Received 18 December 2003; received in revised form 7 June 2005; accepted 16 June 2005

Responsible Editor R. Boutaba

Abstract

This paper considers Web content adaptation with a bandwidth constraint for server-based adaptive Web systems. The problem can be stated as follows: Given a Web page P consisting of n component items $d_1, d_2, \ldots, d_n$ and each of the component items d_i having J_i versions $d_{i_1}, d_{i_2}, \ldots, d_{i_{J_i}}$, for each component item d_i select one of its versions to compose the Web page such that the fidelity function is maximized subject to the bandwidth constraint. We formulate this problem as a linear multi-choice knapsack problem (LMCKP). This paper transforms the LMCKP into a knapsack problem (KP) and then presents a dynamic programming method to solve the KP. A numerical example illustrates this method and shows its effectiveness.

Keywords: Content transcoding; Knapsack problem; Dynamic programming

1. Introduction

Over the past decade, Internet use has exploded with people gaining rich information from the World Wide Web (WWW). With traditional wired-line Internet, users can only access the Internet in fixed places. Recently, however, due to the technology explosion in wireless communication and portable communication devices, e.g., cellular phones, personal digital assistants, and pagers, it

☆ This research was supported in part by the Communications Software Technology Project of Institute for Information Industry and in part by the National Science Council, Taiwan, ROC, under grant NSC 93-2219-E-009-002 and NSC 93-2752-E-009-005-PAE.

* Corresponding author. Tel.: +886 3 573 1637; fax: +886 3 572 1490.
E-mail address: rhjan@cis.nctu.edu.tw (R.-H. Jan).

1389-1286/$ - see front matter © 2005 Published by Elsevier B.V.
doi:10.1016/j.comnet.2005.06.006

has become possible for people to connect to the Internet and remain on-line while roaming.

However, these portable communication devices are very different from the typical personal computers (PC). They vary widely in their screen size, resolution, color depth, computing power, and memory. From notebook PCs to cellular phones, the diversity of these devices makes it difficult and expensive to offer contents separately for each type of device. Many generic WWW servers lack the ability to adapt to the greatly varying bandwidths or to the heterogeneity of client devices. Therefore, the technologies that adapt the Web content to diverse portable communication devices will become very important in the future.

Many content adaptation technologies have been proposed for the WWW [1–10]. These adaptation methods can be divided into three categories: client-based, proxy-based and server-based adaptations. In client-based adaptations [7], the client transforms the original Web pages to the proper presentation according to its capability. However, this method does not work well for mobile devices because mobile devices have lower computing power. In proxy-based adaptations [4,8,9], the proxy intercepts the requested Web pages, performs the adaptation, and then sends the transformed content to the client. But this method requires huge calculations when transforming multi-media data. In contract, server-based adaptations [1,5,10] offer key advantages. Specifically, the server constructs Web pages in accordance to the users' device capabilities and network bandwidths. Repositing multi-versions of Web pages on Web servers in advance not only accelerates response time but also reduces network traffic.

In this paper, we consider a server-based adaptive Web system as shown Fig. 1. Clients can access the Internet via local area networks (LAN), wireless LAN, dial up, or GPRS networks. The Web server contains a set of multi-media Web pages. A multi-media Web page is composed of a number of component items. The clients browse Web pages by sending http requests with capability and preference information [11–13] to the Web server. The Web server parses the requests to learn the capabilities of the clients and probes the network to determine the bandwidth of the connection. Based on clients' capabilities and the bandwidth of the connection, the Web server generate an optimal version of the requested Web page and returns it to the clients.

This paper studies how to generate an optimal version of a Web page with a bandwidth constraint for the server-based adaptive Web system. Formally, the problem, denoted as a Web content selection problem, can be stated as follows: Given a Web page P consisting of n component items

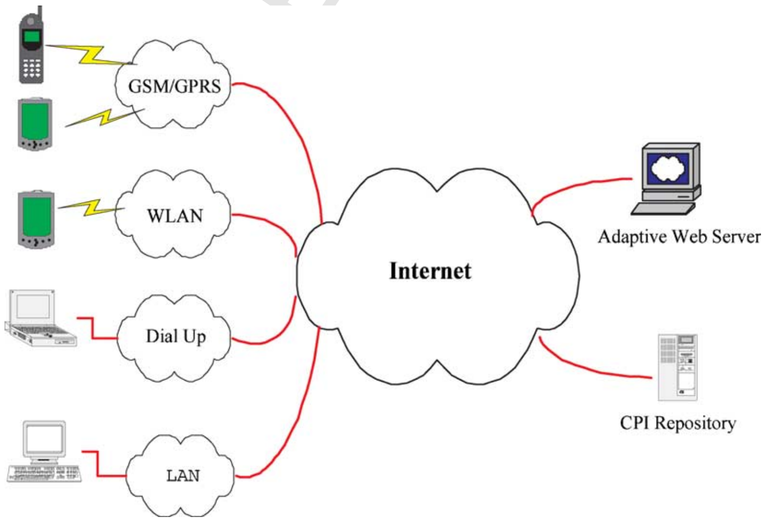


Fig. 1. An adaptive Web system architecture.

$d_1, d_2, \ldots, d_n$ and each of the component items d_i having J_i versions $d_{i_1}, d_{i_2}, \ldots, d_{i_{J_i}}$, for each component item d_i select one of the versions to compose the Web page such that the fidelity function is maximized subject to the bandwidth constraint. We formulate the Web content selection problem as a linear multi-choice knapsack problem (LMCKP) [14]. This paper transforms the LMCKP into a 0/1 knapsack problem (KP)[15,16]. The 0/1 KP problem is a well-known problem in combinatorial optimization. The problem has a large range of applications: capital budgeting, cargo loading, cutting stock, and so on. It can be solved by dynamic programming [17,18], branch and bound [19–21], and greedy methods. This paper presents a dynamic programming method for solving the 0/1 KP because dynamic programming can be easily extended to solve parametric LMCKP problem with different resources. This avoids having to solve the problem anew and slashes the computations needed.

The remainder of this paper is organized as follows. In Section 2, we formulate the Web content selection problem as an optimization problem. Section 3 discusses the solution method, and experimental results are given in Section 4.

2. Statement of the problem

Consider an adaptive Web server having three major modules: content analysis and transcoding, capability and preference information (CPI) filter, and content selection. The architecture of the adaptive server is based on [1]. Fig. 2 illustrates the content adapting process in the adaptive server. In the content analysis and transcoding module, the Web contents are analyzed and transformed into different versions. They are then organized into a content pyramid. The content is prepared in XML, which is converted to HTML prior to delivery. If the server receives an http request from a client, the CPI filter

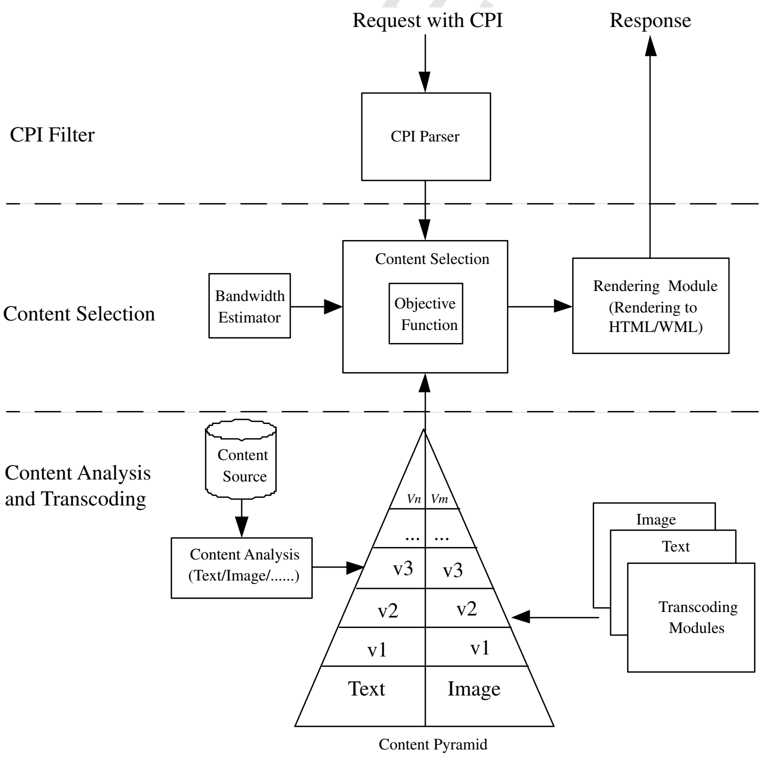


Fig. 2. Server-based content adaptation system architecture.

Fig. 3. An example of multi-media Web page.

module processes the capabilities of the request and forwards the results to the content selection module. The content selection module selects a set of feasible versions from the content pyramid and calls on the bandwidth probing engine [22,23] to find the bottleneck bandwidth between the client and server. With the client's capabilities and bandwidth information, the content selection module determines an appropriate version for each component item. Based on the appropriate versions, the rendering module tailors a style sheet represented by XML style-sheet language (SML), generates an adaptive content and replies to the client.

Note that a multi-media Web page is composed of a number of component items. For example, the document shown in Fig. 3 consists of five component items. These include four image component items and one text component item. Usually, the image component item can be described at multiple resolutions, called versions. The versions can be transformed from raw data at different resolutions. The different version of the component item has a different data size. Suppose a multi-media Web page, P, consists of a number of component items d_i where $P = d_1, d_2, \ldots, d_n$. A component item d_i can be computed by transcoding into versions, $d_{i1}, d_{i2}, \ldots, d_{iJ_i}$ with different resolutions and modalities. Let w_{ij} be the data size of version d_{ij}.

For each version d_{ij}, we can assign a measure of fidelity, called value v_{ij}. Value v_{ij} can be defined as follows:

$$v_{ij} = \frac{\text{perceived value of transcoded version } d_{ij}}{\text{perceived value of original } d_{i1}},$$

where $0 \leqslant v_{ij} \leqslant 1$.

With value v_{ij}, we can then compare different component items that are in different versions. The perceived value may either be assigned by the author for each version, or determined by a function of data size. In this paper, we assume $v_{ij} = f(w_{ij})$ that captures the general trend of fidelity in value. $f(w_{ij})$ may be a concave, convex/non-concave or discrete function of w_{ij}. In this paper, we define[1]

$$f(w_{ij}) = \sqrt{\frac{w_{ij}}{w_{i1}}},$$

where w_{i1} is the data size of item d_i with the original version (see Fig. 4). However, the Web content

[1] This paper is not to suggest that there actually exists a simple function for assigning values to v_{ij}. This is because measuring perceived quality of an image is not easy. Our optimization model allows one to assign arbitrary value to v_{ij} for Web content adaptation problem, by assuming $f(w_{ij})$.

Image							
Version	V1	V2	V3	V4	V5	V6	V7
Data Size (Kbits)	21.3	14.1	11.4	9.0	6.9	4.7	3.1
Value Sqrt(Vn/V1)	1.00	0.81	0.73	0.65	0.57	0.47	0.38

Fig. 4. An example of versions for an image item.

creator can define his own $f(w_{ij})$, say $f(w_{ij}) = w_{ij}/w_{i1}$ or $f(w_{ij}) = \ln w_{ij}/\ln w_{i1}$.

Thus, the Web server can be designed to select the best versions of content items from the Web document sets to meet the client resources while delivering the largest total value of fidelity. Usually, clients do not have the patience to wait for a long time for a Web page. One may expect to receive a Web page in a reasonable waiting time T_{total}, *say 15 s. The next problem for the Web server is to determine the data size W* (maximum) for transmission so as to fall within the expected waiting time.

Fig. 5 illustrates the browsing procedure. The total waiting time for the user is

$$T_{total} = T_{prop} + T_{probe} + T_{proc} + T_{trans} + T_{prop},$$

where T_{total} = total time to wait for each Web page; T_{prop} = propagation time $= T_1 - T_0 = T_5 - T_4$; T_{probe} = time to probe bandwidth $= T_3 - T_1$; T_{proc} = time to process Web content selection $= T_4 - T_3$; T_{trans} = time to transmit Web content $= T_6 - T_5$.

Here we assume for simplicity that T_{prop}, T_{probe}, and T_{proc} are constants. Then data size $W = b \times t = b \times (T_{total} - 2T_{prop} - T_{probe} - T_{proc})$ where b is the bottleneck bandwidth and $t = T_{trans}$. For example, if $T_{total} = 15$, $2T_{prop} + T_{probe} + T_{proc} = 4$, and $b = 10$ Kbps, then the Web server will send a Web page with size not greater than $W = (15 - 4) \times 10 = 110$ KB.

Therefore, the Web content adaptation can be mathematically stated as follows.

Problem LMCKP:

$$\text{Maximize} \quad \sum_{i=1}^{n} \sum_{j=1}^{Ji} v_{ij} x_{ij} \tag{1}$$

$$\text{Subject to} \quad \sum_{i=1}^{n} \sum_{j=1}^{Ji} w_{ij} x_{ij} \leqslant W, \tag{2}$$

$$\sum_{j=1}^{Ji} x_{ij} = 1, \quad 1 \leqslant i \leqslant n, \tag{3}$$

$$x_{ij} = 0 \text{ or } 1, \quad \text{for all } i, j,$$

where v_{ij} and w_{ij} are the measures of fidelity and data size of version d_{ij}, respectively. W is the maximum payload. x_{ij} is the decision variable where $x_{ij} = 1$ indicates version j is selected for item i; otherwise, $x_{ij} = 0$. Constraint (2) ensures that the size of the Web page is not greater than Web page $W = b \times t$. Constraint (3) limits our choice for each item to be one of its versions.

Note that this problem is known as the linear multiple choice knapsack problem (LMCKP). We can apply the dynamic programming method to find the optimal solution for problem LMCKP. An appropriate content can be determined by solving the LMCKP problem.

3. The solution method

The LMCKP is a well-known problem. Many solution methods that have been presented for solving it. This section transforms the LMCKP into a 0/1 KP and apply the dynamic programming method to solve the 0/1 KP.

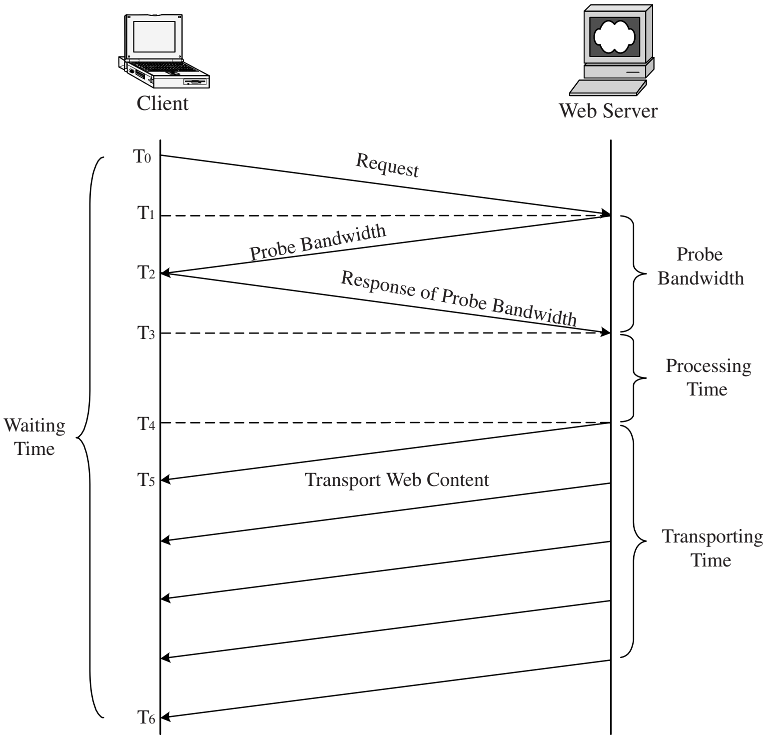


Fig. 5. Event timing for browsing an adaptive Web page.

3.1. Transformation of the problem

At first, we define a knapsack problem, which is equivalent to LMCKP, as follows.

For each i, let

$$\begin{aligned} y_{i1} &= x_{i1}, \\ y_{i2} &= x_{i1} + x_{i2}, \\ &\cdots \\ y_{iJ_i-1} &= x_{i1} + x_{i2} + \cdots + x_{iJ_i-1}, \\ y_{iJ_i} &= x_{i1} + x_{i2} + \cdots + x_{iJ_i} = 1. \end{aligned}$$

Then, we can rewrite the objective function (1) as in the following:

$$\begin{aligned} \sum_{i=1}^{n}\sum_{j=1}^{Ji} v_{ij}x_{ij} &= \sum_{i=1}^{n} v_{i1}y_{i1} + v_{i2}(y_{i2} - y_{i1}) + \cdots \\ &\quad + v_{iJ_i-1}(y_{iJ_i-1} - y_{iJ_i-2}) + v_{iJ_i}(y_{iJ_i} - y_{iJ_i-1}) \\ &= \sum_{i=1}^{n} (v_{i1} - v_{i2})y_{i1} + (v_{i2} - v_{i3})y_{i2} + \cdots \\ &\quad + (v_{iJ_i-1} - v_{iJ_i})y_{iJ_i-1} + v_{iJ_i}y_{iJ_i} \\ &= \sum_{i=1}^{n}\left[\sum_{j=1}^{J_i-1}(v_{ij} - v_{ij+1})y_{ij} + v_{iJ_i}y_{iJ_i}\right] \\ &= \sum_{i=1}^{n}\sum_{j=1}^{J_i-1}(v_{ij} - v_{ij+1})y_{ij} + \sum_{i=1}^{n} v_{iJ_i}. \end{aligned}$$

Similarly, the constraint (2) can be rewritten as

$$\sum_{i=1}^{n}\sum_{j=1}^{J_i-1}(w_{ij} - w_{ij+1})y_{ij} + \sum_{i=1}^{n} w_{iJ_i} \leqslant W.$$

Note that $\sum_{i=1}^{n} v_{iJ_i}$ and $\sum_{i=1}^{n} w_{iJ_i}$ are constants. Let $e_{ij} = v_{ij} - v_{ij+1}$, $d_{ij} = w_{ij} - w_{ij+1}$ and $W' = W - \sum_{i=1}^{n} w_{iJ_i}$. Then, the problem LMCKP (Eqs. (1)–(3)) is equivalent to the following KP:

$$\text{Maximize} \quad \sum_{i=1}^{n}\sum_{j=1}^{J_i-1} e_{ij}y_{ij} \tag{4}$$

$$\text{Subject to} \quad \sum_{i=1}^{n}\sum_{j=1}^{J_i-1} d_{ij}y_{ij} \leqslant W', \tag{5}$$

$$y_{ij} = 0 \text{ or } 1, \quad y_{i1} \leqslant \cdots \leqslant y_{iJ_i}, \ 1 \leqslant i \leqslant n, \quad 1 \leqslant j \leqslant J_i - 1. \tag{6}$$

Note that the above problem can be rewritten as a precedence constraint 0/1 KP [24,25] as follows.

Problem KP:

$$\text{Maximize} \quad \sum_{i=1}^{m} p_i z_i \tag{7}$$

$$\text{Subject to} \quad \sum_{i=1}^{m} d_i z_i \leqslant M, \tag{8}$$

$$z_i = 0 \text{ or } 1, \quad z_h \leqslant z_k, \ (h,k) \in A_i, \quad 1 \leqslant i \leqslant m,$$

where $m = \sum_{i=1}^{n}\sum_{j=1}^{J_i-1} 1; M = W'$ and A_i is the precedence constraint as described in (6).

3.2. Dynamic programming method

The precedence constraint 0/1 knapsack problem can be solved by dynamic programming method as that for the ordinary 0/1 knapsack problem with slight modification. We may make a decision on z_1 first, then on z_2, then on z_3, etc. The solution to the 0/1 KP problem can be viewed as the result of a sequence of decisions. An optimal sequence of $z_1, z_2, \ldots, z_k$ will maximize the objective function and satisfy the constraint. Moreover, we can apply dynamic programming to solve the *parametric precedence constraint* 0/1 KP problem with right-hand side $M \in [a, b]$.

Let KP(j, S) denote the problem

$$\text{Maximize} \quad \sum_{i=1}^{j} p_i z_i$$

$$\text{Subject to} \quad \sum_{i=1}^{j} d_i z_i \leqslant S,$$

$$z_i = 0 \text{ or } 1, \quad z_h \leqslant z_k, \ (h,k) \in A_i, \quad 1 \leqslant i \leqslant j,$$

where $1 \leqslant j \leqslant n$ and $0 \leqslant S \leqslant M$. Note that KP($j$, S) is a sub-problem of Problem KP with variables $z_1, z_2, \ldots, z_j$ and right-hand side S. Problem KP is KP(n, M). Let $f_k(s)$ be the value of an optimal solution to KP(k, s). From the principle of optimality it follows that:

$$f_k(s) = \max\{f_{k-1}(s), f_{k-1}(s - d_k) + p_k, \text{ subject to precedence constraints}\}. \tag{9}$$

Clearly, $f_n(M)$ is the value of an optimal solution to KP(n, M). $f_n(M)$ can be solved by beginning with $f_0(s) = 0$ for all $s > 0$ and $f_0(s) = -\infty$, $s < 0$. Then $f_1, f_2, \ldots, f_n$ can be successively computed using Eq. (9). Notice that $f_k(s)$ is an ascending step function; i.e., there are a finite number of s, $s_1 < s_2 < \cdots < s_t$, such that $f_k(s_1) \leqslant f_k(s_2) \leqslant \cdots \leqslant f_k(s_t)$. For the parametric precedence constraint 0/1 KP problem, we solved the problem KP(n, M) for each $M \in [a, b]$ at the last stage.

3.3. A numerical example

Consider an example of a Web page with three image items (i.e., $P = d_1, d_2, d_3$). Each item has three versions. The right-hand side $W \in [6.7, 30.8]$. The data sizes w_{ij} and the values v_{ij}, $i, j = 1, 2, 3$, are

$$[w_{ij}] = \begin{bmatrix} 9.0 & 4.4 & 1.3 \\ 10.8 & 4.6 & 1.0 \\ 11.0 & 5.4 & 1.3 \end{bmatrix},$$

$$[v_{ij}] = \left[\sqrt{\frac{w_{ij}}{w_{i1}}}\right] = \begin{bmatrix} 1.0 & 0.7 & 0.4 \\ 1.0 & 0.7 & 0.3 \\ 1.0 & 0.7 & 0.3 \end{bmatrix}.$$

The content selection problem can be formulated as follows:

$$\text{Maximize} \quad v_0 = \sum_{i=1}^{3}\sum_{j=1}^{3} v_{ij} x_{ij}$$

$$\text{Subject to} \quad \sum_{i=1}^{3}\sum_{j=1}^{3} w_{ij} x_{ij} \leqslant W, \quad W \in [6.7, 30.8],$$

$$\sum_{j=1}^{3} x_{ij} = 1, \quad 1 \leqslant i \leqslant 3,$$

$$x_{ij} = 0 \text{ or } 1, \quad 1 \leqslant i \leqslant 3, \ 1 \leqslant j \leqslant 3.$$

3.3.1. Transformation of the problem

For $i = 1, 2, 3$, let $y_{i1} = x_{i1}$, $y_{i2} = x_{i1} + x_{i2}$, and $y_{i3} = x_{i1} + x_{i2} + x_{i3} = 1$. Then, the problem can be transformed as follows:

$$\text{Maximize} \quad 0.3y_{11} + 0.3y_{12} + 0.3y_{21} + 0.4y_{22} + 0.3y_{31} + 0.4y_{32} + 1$$

$$\text{Subject to } 4.6y_{11}+3.1y_{12}+6.2y_{21}+3.6y_{22} +5.6y_{31}+4.1y_{32}+3.6 \leqslant W,\quad W\in[6.7,30.8],$$
$$y_{11}\leqslant y_{12},\ y_{21}\leqslant y_{22},\ y_{31}\leqslant y_{32};$$
$$y_{11},y_{12},y_{21},y_{22},y_{31},y_{32}=0 \text{ or } 1.$$

Let $z_1=y_{11}$, $z_2=y_{12}$, $z_3=y_{21}$, $z_4=y_{22}$, $z_5=y_{31}$, $z_6=y_{32}$. Clearly, the above problem is equivalent to the following problem, KP(6, *M*), $M\in[3.1,27.2]$:

$$\text{Maximize } 0.3z_1+0.3z_2+0.3z_3+0.4z_4+0.3z_5+0.4z_6$$
$$\text{Subject to } 4.6z_1+3.1z_2+6.2z_3+3.6z_4+5.6z_5+4.1z_6\leqslant M,\quad M\in[3.1,27.2],$$
$$z_1\leqslant z_2,\ z_3\leqslant z_4,\ z_5\leqslant z_6;\ z_i=0 \text{ or } 1,\ i=1,\ldots,6.$$

3.3.2. Dynamic programming method

Let $f_k(s)$ be the value of an optimal solution to KP(k, s) where $s\in[0,27.2]$ and $i=1,\ldots,6$. Clearly, $f_6(M)$ is the value of an optimal solution to KP(6, *M*). Applying Eq. (9), $f_6(M)$, $M\in[a,b]$ can be solved by starting with $f_0(s)=0$ for all $s>0$. Then $f_1(s),f_2(s),\ldots,f_6(s)$, $s\in[0,27.2]$ can be successively found. For example

$$f_4(s)=\max_{z_4=0,1}\begin{cases}0.4+f_3(s-3.6), & z_4=1,\ s\geqslant 3.6,\\ f_2(s-3.6), & z_4=0.\end{cases}$$

Fig. 6 graphically shows $f_1(s),f_2(s),\ldots,f_5(s)$, $s\in[0,27.2]$, and $f_6(s)$, $s\in[3.1,27.2]$. Thus, the optimal values and the optimal solutions for $M\in[3.1,27.2]$ is summarized as follows.

For example, the optimal solution to KP(6,27.2) is $f_6(27.2)=2.0$. The optimal solution of KP(6,27.2) is: $(z_1,z_2,z_3,z_4,z_5,z_6)=(y_{11},y_{12},$

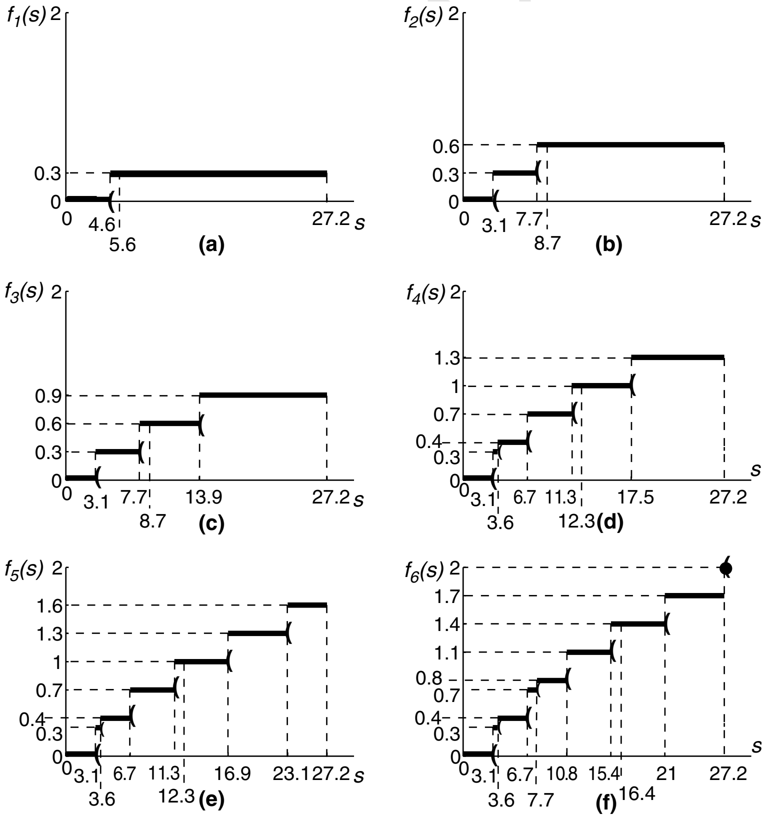


Fig. 6. Functions of $f_1(s),f_2(s),f_3(s),f_4(s),f_5(s)$, and $f_6(s)$.

$y_{21}, y_{22}, y_{31}, y_{32}) = (1,1,1,1,1,1)$. Thus, the optimal solution for the content selection problem is

$$(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}) = (1,0,0,1,0,0,1,0,0)$$

and the optimal value is 3.0. That is, version 1 is selected for each item.

If another request for this page arrives, the adaptive server finds $b = 16$ Kbps and $t = 5$ s for this connection. Then, the total data size W that the adaptive server may return to the client is

$$W = \frac{16 \times 10}{8} = 20 \text{ Kbps}.$$

Note that the adaptive server does not need to solve the problem KP(6, 20 − 3.6) anew. The optimal solution for KP(6, 16.4) can be found in Table 1. Since $M = 16.4$, we look in Table 1 down the $M \in [15.4, 21.0)$ row. We find that the optimal solution for KP(6, 16.4) is $(y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32}) = (1,1,0,1,0,1)$, and the optimal solution for the content selection problem is

$$(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}) = (1,0,0,0,1,0,0,1,0).$$

That is, the returned Web page is compose of version 1 for item 1 and version 2 for items 2 and 3.

4. Experimental results

In order to test our optimization model for Web content adaptation, we built three Web servers: two adaptive and one non-adaptive servers. Both adaptive Web servers consisted of three major modules (content analysis and transcoding, CPI filter, and content selection) as shown in Fig. 2. The difference between them is in the content selection module. One, denoted as Sever 1, found optimal solutions of parametric LMCKP by dynamic programming method in advance after the Web page was created. When the request arrives, it just looks up the optimal solutions table. The other, denoted as Sever 2, selects contents by using greedy algorithm [14] to solve LMCKP whenever the request arrives. The non-adaptive server is denoted as Sever 3. A Linux operating system and an Apache server were selected as developing platform for three servers. Apache is a well-known, open source Web server that performs well. The machines for the three servers are the desktop computers with AMD K7-850 and 256 MB memory. The test Web page, a sub-page of the University's Web pages, consists of seven component items with 140 KB data size. These include six image component items and one text item. Each image item has six versions.

The Servers 1, 2 and 3 were tested by two clients. The clients' browser was modified from Internet Explorer (IE) so that the users can specify the expected time, CPI data, and where the CPI profiles are by URLs (see Fig. 7). Client 1 was a notebook PC with Intel P3-650 and 256MB memory, using PPP-dialup 56 Kbps (campus dialup service) connecting to the campus Internet. The expected waiting time was set to 15 s when browsing the Servers 1 and 2. For the measurement, the system clock of the three servers and two clients was synchronized using the Network Time Protocol (NTP). Client 1 browsed the Web page 10 times. We use the t distribution with 9 degrees of freedom

Table 1
Summary of the optimal solutions

Right-hand side	$f_6(M)$	$(y_{11}, y_{12}, y_{21}, y_{22}, y_{31}, y_{32})$	v_0	$(x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33})$
$M \in [3.1, 3.6)$	0.3	(0,1,0,0,0,0)	1.3	(0,1,0,0,0,1,0,0,1)
$M \in [3.6, 6.7)$	0.4	(0,0,0,1,0,0)	1.4	(0,0,1,0,1,0,0,0,1)
$M \in [6.7, 7.7)$	0.7	(0,1,0,1,0,0)	1.7	(0,1,0,0,1,0,0,0,1)
$M \in [7.7, 10.8)$	0.8	(0,0,0,1,0,1)	1.8	(0,0,1,0,1,0,0,0,1)
$M \in [10.8, 15.4)$	1.1	(0,1,0,1,0,1)	2.1	(0,1,0,0,1,0,0,1,0)
$M \in [15.4, 21.0)$	1.4	(1,1,0,1,0,1)	2.4	(1,0,0,0,1,0,0,1,0)
$M \in [21.0, 27.2)$	1.7	(1,1,0,1,1,1)	2.7	(1,0,0,0,1,0,1,0,0)
$M = 27.2$	2	(1,1,1,1,1,1)	3	(1,0,0,1,0,0,1,0,0)

Fig. 7. An example of CC/PP browser.

and a 95% confidence interval to estimate the delays. The average delays for Client 1 are shown in Table 2a. Table 2b shows the percentage of measured delays out of the total delay. The other client, Client 2 was a Compaq pocket PC using IEEE 802.11b wireless LAN connecting to campus Internet. The results are summarized in Tables 3a and 3b. Our experiments use campus Internet as network testbed. There are many factors, such as irrelevant traffic in the network, buffer sizes, etc., that may influence (or pollute) the results. Tables 2 and 3 are only intended to offer the reader some realistic feeling about the respond time and how the system works. From the theoretical point of view, the time complexity for Server 1 to pick up an optimal solution from the optimal solutions table is O(1) while the time complexity of the greedy method for Server 2 is $O(n \log n)$ where n is the number of variables.

From Table 2a, note that the total delays for browsing the Servers 1 and 2 were $15\,794 \pm 20$ ms and $16\,443 \pm 22$ ms, while for browsing the Server 3 it was $46\,386 \pm 117$ ms. The adaptive servers show their benefits. The total delays of Servers 1 and 2 are controlled and close to the expected waiting time 15 s. For processing time T_{proc}, Server 1 performs better than Server 2 because Server 1 finds optimal solutions by dynamic programming method in advance and looks up the optimal solutions table when the request arrives. The advantage of dynamic programming method is that it

Table 2a
Results for notebook PC with 56K dialup

	$2 \times T_{prop}$ (ms)	T_{probe} (ms)	T_{proc} (ms)	T_{trans} (ms)	T_{total} (ms)	W (KB)
Server 1	179.5 ± 2.0	6000.1 ± 2.0	<0.001	9614.3 ± 20.2	15794 ± 20	28.9
Server 2	173.5 ± 2.8	6123.3 ± 1.8	4.8 ± 0.3	10141.5 ± 23.4	16443 ± 22	30.0
Server 3	180.2 ± 2.0	–	–	46205.4 ± 116.9	46386 ± 117	140

COMPNW 3174
3 August 2005 Disk Used
ARTICLE IN PRESS
No. of Pages 13, DTD = 5.0.1

Table 2b
Percentage of the measured time out of T_{total} for Table 2a

	$2 \times T_{prop}/T_{total}$ (%)	T_{probe}/T_{total} (%)	T_{proc}/T_{total} (%)	T_{trans}/T_{total} (%)
Server 1	1.14	37.99	0.00	60.87
Server 2	1.05	37.24	0.03	61.68
Server 3	0.39	0.00	0.00	99.61

Table 3a
Results for pocket PC with 802.11b

	$2 \times T_{prop}$ (ms)	T_{probe} (ms)	T_{proc} (ms)	T_{trans} (ms)	T_{total} (ms)	W (KB)
Server 1	5.8 ± 0.5	2028.3 ± 23.6	<0.001	118.8 ± 1.9	2153 ± 24	28.9
Server 2	6.5 ± 0.4	2212.6 ± 16.7	4.4 ± 0.4	109.7 ± 1.8	2333 ± 17	28.9
Server 3	6.4 ± 0.4	–	–	641.9 ± 6.8	648 ± 7	140

Table 3b
Percentage of the measured time out of T_{total} for Table 3a

	$2 \times T_{prop}/T_{total}$ (%)	T_{probe}/T_{total} (%)	T_{proc}/T_{total} (%)	T_{trans}/T_{total} (%)
Server 1	0.27	94.21	0.00	5.52
Server 2	0.28	94.83	0.19	4.70
Server 3	0.99	0.00	0.00	99.06

can be easily applied to solve the parametric LMCKP with a set of right-hand sides.

In Tables 2a and 3a, the values of probing delay T_{probe} are very large. This is because the probing method ("pathchar" algorithm [23]) sends a few dozen packets with varying sizes, measures their round trip times (RTTs), and then finds the available bandwidth by correlating the RTTs with packet sizes. The RTT depends on traffic load. Thus, the probing delay T_{probe} varies as traffic load varies. The traffic load in our campus Internet varies dynamically from one instant to another. Consequently, the mean probing time may be different for the same mobile device in the case of Server 1 and Server 2.

Note that if the access network has a higher data rate, the value of T_{probe} dominates the value of T_{trans} (see Table 3b). That is, the overhead of bandwidth estimation is too large, thus negating the advantage of shorter transmission times. In fact, instead of measuring bandwidth, we can just use predefined classes of data rate r, (say dialup 54 Kbps, T1 1.544 Mbps, or WiFi 11 Mbps) and set available bandwidth $b = \alpha \times r$, $0 < \alpha < 1$. Note that data rate r can be obtained from the CPI profile. By this way, the overhead of bandwidth estimation is eliminated and thus the adaptive Web server can give a smaller latency.

5. Conclusion

This paper formulates a Web content adaptation problem as a linear multi-choice knapsack problem and presents a dynamic programming method to solve it. We think that the dynamic programming is very suitable for solving this kind of problems because dynamic programming can be easily extended to solve parametric LMCKP problem with different resources. This avoids having to solve the problem anew and slashes the computations needed.

In practical implementation, we can analyze and transform the component items of Web page into different versions when the Web page is created. Then, dynamic programming is applied to solve a parametric LMCKP problem and a binding table which binds the optimal solutions to different resources can be created. If a request for this page arrives, we just look up the binding table to find the optimal versions of the component items for the request.

In this paper, we assumed that the content items are independent of each other. However, this assumption may not hold in some cases. Consider a news story page. If the story has to be discarded due to space limitations, then the pictures for the story has also to be discarded. For such cases, the content creator has to define the dependencies among the items of the Web page. Then, our optimization model can be extended by adding the constraint $\sum_{j=1}^{Ji} x_{ij} \leqslant \sum_{j=1}^{Jk} x_{kj}$ to Problem LMCKP if item d_i is dependent on item d_k.

There also exists coarse-grained approaches for content adaptation. The coarse-grained approaches format the Web content for several well-known kinds of clients to suit everyone. However, we think delivering a customized content is worth for content providers. This paper presents a fine-grained adaptation that selects the best content representation to match the resources and capabilities of individual clients.

Looking ahead, integrating both adaptive Web server and transcoding proxy server for wireless Internet access might be interesting future work.

6. Uncited reference

[26].

References

[1] R. Mohan, J.R. Smith, C.S. Li, Adapting multimedia Internet content for universal access, IEEE Transactions on Multimedia (Mar) (1999) 104–114.

[2] V. Cardellini, P.S. Yu, Y.W. Huang, Collaborative proxy system for distributed Web content transcoding, in: Proceedings of the 9th International ACM Conference on Information and Knowledge Management, November 2000, pp. 520–527.

[3] R. Han, V. Perret, M. Naghshineh, WebSplitter: a unified XML framework for multi-device collaborative Web browsing, in: ACM Conference on Computer Supported Cooperative Work (CSCW), December 2000.

[4] P.A. Singh, A. Trivedi, K. Ramamritham, PTC: proxies that transcode and cache in heterogeneous Web client environments, Web Information Systems Engineering (2002) 11–20.

[5] B. Knutsson, H. Lu, J. Mogul, Architecture and pragmatics of server-directed transcoding, in: Proceedings of the 7th International Web Content Caching and Distribution Workshop, August 2002, pp. 229–242.

[6] A. Fox, S.D. Grebble, Y. Chwathe, E.A. Brewer, Adapting to network and client variation using infrastructural proxies: lessons and perspectives, IEEE Personal Communications (August) (1998) 10–19.

[7] B. Noble, System support for mobile adaptive applications, IEEE Personal Communications (Feb) (2000) 44–49.

[8] H. Bharadvaj, A. Joshi, S. Auephanwiriyakul, An active transcoding proxy to support mobile Web access, in: Proceedings of IEEE Symposium on Reliable Distributed System, October 1998, pp. 118–123.

[9] S. Acharya, H.F. Korth, V. Poosala, Systematic multiresolution and its application to the World Wide Web, IEEE Data Engineering (1999) 40–49.

[10] F. Kitayama, S. Hirose, G. Kondoh, Design of a framework for dynamic content adaptation to Web-enabled terminals and enterprise applications, in: IEEE Software Engineering Conference, 1999, pp. 72–79.

[11] F. Reynolds, J. Hjelm, S. Dawkins, S. Singhal, Composite Capability/Preference Profiles (CC/PP): A User Side Framework for Content Negotiation, W3C note, 27 July 1999.

[12] H. Ohto, J. Hjelm, CC/PP Exchange Protocol Based on HTTP Extension Framework, W3C note, 24 June 1999.

[13] W3C, The Resource Description Framework. Available from: <http://www.w3.org/RDF/>.

[14] E. Zemel, The linear multiple choice knapsack problem, Operations Research 28 (November) (1980) 1412–1423.

[15] E. Balas, E. Zemel, An algorithm for large zero–one knapsack problems, Operation Research 28 (1980) 1130–1154.

[16] S. Martello, P. Toth, Knapsack Problems: Algorithms and Computer Implementations, Wiley, Chichester, UK, 1990.

[17] R.E. Bellman, Dynamic Programming, Princeton University Press, Princeton, NJ, 1957.

[18] P. Toth, Dynamic programming algorithms for the zero–one knapsack problems, Computing 25 (1980) 29–45.

[19] P.J. Kolesar, A branch and bound algorithm for the knapsack problem, Management Science 13 (1967) 723–735.

[20] E. Horowitz, S. Sahni, Computing partitions with applications to the knapsack problem, Journal of ACM 21 (1974) 277–292.

[21] S. Martello, P. Toth, An upper bound for zero–one knapsack problem and a branch and bound algorithm, European Journal of Operation Research 1 (1977) 169–175.

[22] K. Lai, M. Baker, Measuring bandwidthProceedings of INFOCOM'99, vol. 1, IEEE Computer and Communications Societies, 1999, pp. 235–245.

[23] V. Jacobson, Pathchar, 1997. Available from: <ftp://ftp.ee.lbl.gov/pathchar/>.

[24] D.S. Johnson, K.A. Niemi, On knapsacks, partitions, and a new dynamic programming technique for tree, Mathematics of Operations Research 8 (1983) 1–14.
[25] H. Kellerer, U. Pferschy, D. Pisinger, Knapsack Problems, Springer-Verlag, Berlin, 2004.
[26] C.C. Hung, L.Y. Hong, Bandwidth sensitive content transformation in pervasive computing, in: Proceedings of ISCC 2000, IEEE Computers and Communications, 2000, pp. 14–19.

Rong-Hong Jan received the B.S. and M.S. degrees in Industrial Engineering, and the Ph.D. degree in Computer Science from National Tsing Hua University, Taiwan, in 1979, 1983, and 1987, respectively. He joined the Department of Computer and Information Science, National Chiao Tung University, in 1987, where he is currently a Professor. During 1991–1992, he was a Visiting Associate Professor in the Department of Computer Science, University of Maryland, College Park, MD. His research interests include wireless networks, mobile computing, distributed systems, network reliability, and operations research.

Ching-Peng Lin received the B.S. degree in Computer Science from Fu Jen Catholic University in 2000 and M.S. degree in Computer and Information Science from National Chiao Tung University, Taiwan, in 2002. His research interests include wireless networks, mobile computing and wireless Internet.

Maw-Sheng Chern received a B.S. degree and an M.S. degree in Mathematics from National Taiwan Normal University and National Tsing Hua University respectively, an M. Math. in Combinatorics and Optimization and a Pd.D. in Management Sciences from the University of Waterloo. He joined the Department of Industrial Engineering and Engineering Management, National Tsing Hua University in 1980 and was the department chair from 1991 to 1994. Dr. Chern served as the program director for Industrial Engineering and Management Division, National Science Council, ROC from 1995 to 1998. He also served on the editorial board of IIE Transactions on Logistics and Scheduling (1997–200), International Journal of Industrial Engineering—Theory, Applications and Practice (1994–1996), Chiao-Ta Management Review, and Journal of Management & Systems. His current research interests include combinatorial optimization, production scheduling, and network programming.

附件三

A GPS-less self-positioning method for sensor networks*

Hung-Chi Chu and Rong-Hong Jan†
Department of Computer and Information Science
National Chiao Tung University
Hsinchu, 30050, Taiwan

Abstract

One challenging issue in sensor networks is to determine where a given sensor node(SN) is physical located. This problem is especially crucial for very small SNs. In this paper, we present a GPS-less self-positioning method for sensor networks. In our method, a set of nodes, called as reference points (RPs), are deployed in the sensor network with overlapping regions of coverage. The RP periodically broadcasts beacon frames which contain localization data. The SN collects the beacon frames from RPs and processes the data in the frame, and then it can localize itself simply. The analysis of positioning accuracy are also given to show how well a SN can correctly localize itself.

1. Introduction

The fast progress of micro-electro-mechanical systems (MEMS) technology and wireless communications has enabled us to deploy a large number of low-cost, low-power and networked sensors in wide area. The sensor nodes(SNs) can collect, store, and process the sensed data, and communicate with neighboring nodes to provide observation of environmental system. This makes us to monitor and control the physical world more convenient and efficient. In such sensor network systems, we need SNs to be able to locate themselves in various environments. The location data of SNs is useful for the centralized server or the managing node to analyze their sensing information. Not only SNs but also other objects which are in the network need to be located. For example, the forest fire detection system should detect exactly where is the scene of a fire. In location-aware applications, localization enables the intelligent context selection includes tour guide [1], points of interest and so on. In ad hoc networks, localization helps the transmitting node to recognize where the communicating node is and thus reduces the power consumption. Hence the localization is important for many applications of sensor network.

For localization system, Global Positioning System (GPS) [2] is a good solution in outdoor environment. However, it is not suitable to use GPS on all SNs in sensor networks. This is because SNs have size, cost and power constraints. This paper focus on the problem of GPS-less low-cost localization for wireless sensor networks.

Generally speaking, the localization can be divided into three major classes, self-positioning, remote positioning, and indirect positioning. The basic operations of these classes are summarized in the following:

A. *Self-positioning system (SPS)*
The positioning receiver receives the appropriate signal measurements form geographically distributed transmitters and then uses these measurements to localize itself. GPS is a typical SPS and several SPSs [3, 4] for sensor networks are presented recently. In [3], they measure the received signal strength and apply a triangulation method to localize moving sensors and handle dynamically changing sensor topologies. In [4], some fixed reference points (RPs) with overlapping regions of signal coverage are configured. These RPs transmit periodic beacon signals, and then SNs can localize themselves based on the received beacons.

B. *Remote positioning system (RPS)*
A set of nodes with special RF functions are deployed in some fixed place and measure the direction or the time delay of a signal which is originating from, or reflecting off, the transmitter nodes. After that, these measurements are collected by a centralized location server to determine the location of transmitter node. Typical RPSs are Angle of Arrival (AOA) [5] and Time Difference of Arrival (TDOA) [5]. The AOA collects the direction of transmitter's signals and the TDOA collects the time delay of transmitter's signals. Such solutions do not require any modification to the objects

[1]This work was supported in part by the National Science Council, Taiwan, ROC, under grants NSC 93-2752-E-009-005-PAE and NSC 93-2219-E-009-002, respectively, and in part by Intel.

[2]Corresponding Author. Fax: 886-3-5721490; e-mail: rhjan@cis.nctu.edu.tw

but they have low accuracy and high network costs.

C. *Indirect positioning system (IPS)*
The IPS combines SPS and RPS. First, the node measures signal data and transfers to the RPS. Next, the RPS collects these measurements, processes position bias and then determines node's position. Typical IPSs are Assisted GPS(AGPS) [6], Differential GPS(DGPS) [7], and cell-based positioning [8] where AGPS and DGPS have the highest accuracy.

The cell-based positioning system simply utilizes the characteristic of cell overlapping in geometry. However, it determines the location in a centralized server. When a SN needs to localize itself, it sends location request to the location server. The location server determines the sensor's location, and then replies the location to the SN. Communications between the sensor and location server cost a lot of energy. It is not suitable for sensor networks. Based on the idea of cell overlapping, this paper presents a GPS-less self-positioning method for sensor networks. In the proposed method, a set of nodes, called as reference points (RPs), are deployed in the sensor network with overlapping regions of coverage. The RPs broadcast periodic beacon frames which contain localization data. The SN in the sensor network receives the beacon frames from RPs and processes the information in the frame, and then the localization can be determined by itself. The proposed method has the distributed and simple characteristics. The Distributed property means that the location can be determined by SN itself without GPS and centralized server. The simple property means that SNs only use a simple connectivity metric and localization data in the beacon frame to calculate their locations. That is, SNs only take little computation to localize themslves.

2. Cell overlapping model

Consider that a set of RPs are deployed in the sensor network with overlapping regions of coverage. They are located at known positions and form a regular structure. As shown in Fig. 1(a), these RPs form a hexagonal structure. In idealized radio model, we assume a perfect spherical radio propagation and identical transmission range for all RPs. The area covered by the RP is called a *cell* and each cell is circle-shaped. The SN can receive radio signal from the RP if it is within the signal coverage of that RP. Take Fig. 1(a) for example, a SN in region A_1 can listen to signals from RP P_0; in region B_1, from RPs P_0 and P_1; and in region C_1, from RPs P_0, P_1 and P_6. The localization region is defined as the region in which every SN can listen a unique set of RPs' signals. The coverage of RP P_0 has 13 localization regions, i.e., regions $A_1, B_1, \ldots, B_6, C_1, \ldots, C_5$ and C_6. The localization regions in the coverage of a RP can be di-

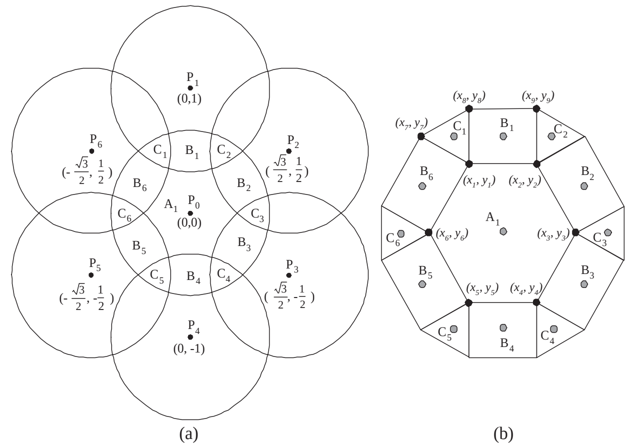


Figure 1. The physical layout of reference points with a hexagonal structure.

vided into three types according to the number of receiving signals as follows.

- Type 1 region: The region is covered by only one RP's signal, e.g., region A_1.
- Type 2 region: The region is covered by two RPs' signal coverage, e.g., regions B_i, where $1 \leq i \leq 6$.
- Type 3 region: The region is covered by three RPs' signal coverage, e.g., regions C_i, where $1 \leq i \leq 6$.

Note that the radio coverage of RP is represented as a circle. By using simple geometry, we can find all the intersections of the circles. For each localization region, we find the centroid (x_c, y_c) of the region by

$$(x_c, y_c) = (\frac{x_1 + x_2 + ... + x_n}{n}, \frac{y_1 + y_2 + ... + y_n}{n})$$

where (x_1, y_1), (x_2, y_2), ..., (x_n, y_n) are the vertices of the region. If a SN can localize itself in the region, we use (x_c, y_c) to estimate the location of the SN. For example, as shown in Fig. 1(b), if a SN localizes itself in region B_1, the estimated location of SN is $(\frac{x_1+x_2+x_8+x_9}{4}, \frac{y_1+y_2+y_8+y_9}{4})$.

Given a set of RPs deployed in a hexagonal structure in which the distance between two neighboring RPs is one unit and the transmission range of RP is $r = 0.78$, we find the centroids for all localization regions. The results are summarized in Table 1.

3 Self-positioning algorithm

As stated in the previous section, we can deploy RPs in a hexagonal structure and find the localization regions for each RP. The RP periodically broadcasts the beacon frame

Table 1. The centroids of all regions in the hexagonal network structure.

Region	centroid	Region	centroid
A_1	(0, 0)	C_1	(-$\frac{\sqrt{3}}{6}$, $\frac{1}{2}$)
B_1	(0, $\frac{1}{2}$)	C_2	($\frac{\sqrt{3}}{6}$, $\frac{1}{2}$)
B_2	($\frac{\sqrt{3}}{4}$, $\frac{1}{4}$)	C_3	($\frac{\sqrt{3}}{3}$, 0)
B_3	($\frac{\sqrt{3}}{4}$, -$\frac{1}{4}$)	C_4	($\frac{\sqrt{3}}{6}$, -$\frac{1}{2}$)
B_4	(0, -$\frac{1}{2}$)	C_5	(-$\frac{\sqrt{3}}{6}$, -$\frac{1}{2}$)
B_5	(-$\frac{\sqrt{3}}{4}$, -$\frac{1}{4}$)	C_6	(-$\frac{\sqrt{3}}{3}$, 0)
B_6	(-$\frac{\sqrt{3}}{4}$, $\frac{1}{4}$)	-	-

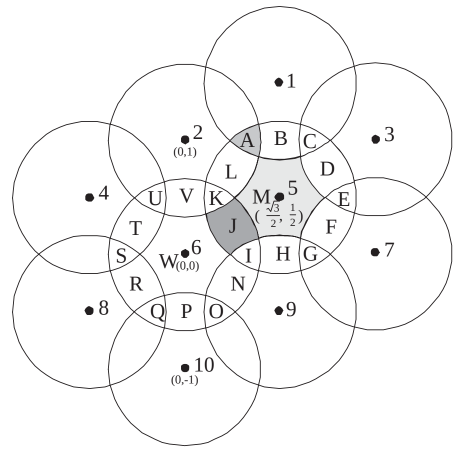


Figure 2. An example of localization regions for hexagonal structure.

to notify all of the SNs staying in its signal coverage area. We assume that each RP knows all centroids of its localization regions. For example, RP P_0 knows the centroids of 13 localization regions that was computed in the deployment stage. The beacon format contains the following data:

$$S = \{t_n, (t_{r_a}, \{(x_{c_1}, y_{c_1}), \ldots, (x_{c_a}, y_{c_a})\}), \ldots, (t_{r_k}, \{(x_{c_1}, y_{c_1}), \ldots, (x_{c_k}, y_{c_k})\})\}$$

where t_n represents the type of RP's structure, (e.g., $t_n = 1$ for hexagonal structure and $t_n = 2$ for meshed structure); t_{r_i} represents the type of localization region (e.g., $t_{r_i} \in \{1, 2, 3\}$ for hexagonal structure); and (x_{c_i}, y_{c_i}) represents the centroid of the region. Note that the type number of the region is equal to the number of signals can be received in that region. Take Fig. 2 for example, the beacon frames of RP 5 and RP 6 are

$$S_5 = \{1, (1, \{M\}), (2, \{B, D, F, H, J, L\}), (3, \{A, C, E, G, I, K\})\}$$
$$S_6 = \{1, (1, \{W\}), (2, \{J, N, P, R, T, V\}), (3, \{K, I, O, Q, S, U\})\}$$

where the symbols $A, B, \ldots, W$ represent the centroids of localization regions (e.g., $M = (\frac{\sqrt{3}}{2}, \frac{1}{2}), W = (0, 0)$).

Then, the SN collects the beacon signals from the RPs and determines its location. The operations of SN are given as follows.

1. Collect and store the beacon signal that it receives.

2. Determine the number of RP, denoted as m, that it can listen. Then, extract the centroid set with the type m from the beacon frames , denote as S^m. Note that we can find m different centroid sets. For example, if a SN can receive beacons from RP 5 and RP 6, it extracts the centroid set with type 2 from the received beacon frames as follows.

$$S_5^2 = \{B, D, F, H, J, L\}, S_6^2 = \{J, N, P, R, T, V\}.$$

3. The SN finds a centroid by intersecting the centroid sets as its location, i.e., find $\bigcap_i S_i^m$. For example,

$$S_5^2 \bigcap S_6^2 = \{B, D, F, H, J, L\} \bigcap \{J, N, P, R, T, V\} = \{J\} = \{(\frac{\sqrt{3}}{4}, \frac{1}{4})\}.$$

4 Positioning accuracy analysis

Let the coordinate of actual location of SN be (X, Y) where X and Y are random variables. In our proposed method, the SN localizes itself to the centroid of the localization region. Thus, the error distance D is

$$D = \sqrt{(X - x_c)^2 + (Y - y_c)^2}$$

where (x_c, y_c) is the centroid of the localization region (i.e., the estimated location of the SN). The *precision* $e(r)$ can be defined as the probability that the SN can localize itself within distance r. That is, $e(r) = P\{D < r\}$. Assume that the SN falls equally likely to any point in the location region R. Then, the probability density function $f(x, y)$ of (X, Y) can be written as follows:

$$f(x, y) = \begin{cases} c & \text{if } (x, y) \in R \\ 0 & \text{otherwise} \end{cases}$$

where

$$\int_R \int f(x, y) dx dy = \int_R \int c \, dx dy = 1.$$

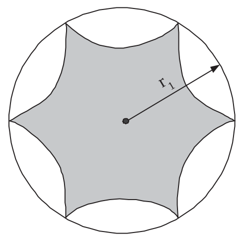


Figure 3. The shape of type 1 in hexagonal structure.

This gives

$$c = \frac{1}{\int_R \int \, dxdy} = \frac{1}{\text{area of R}}$$

Therefore, the precision

$$e(r) = P\{D < r\} = \int_{C_r} \int f(x,y)dxdy = \frac{\text{area } C_r}{\text{area of } R}.$$

where

$$C_r = \{(x,y) | \sqrt{(x-x_c)^2 + (y-y_c)^2} < r\} \cap R.$$

1) *The worst-case accuracy*
Let us consider the shape of type 1 in hexagonal structure as shown in Fig. 3. The precision $e(r)$ is the area of C_r over the area of localization region R, if r is less then r_1. If r is greater than r_1, the precision $e(r)$ is 1. This means that SN can localize itself within distance r_1 with probability 1. In other words, if SN localizes itself in type 1 region and the tolerance of error distance d is greater than r_1, the position of SN can be correctly determined. The radius r_1 is called as *critical radius*. Furthermore, let $r^* = \max\{r_1^{(1)}, r_1^{(2)}, r_1^{(3)}\}$ where $r_1^{(i)}$ is the critical radius for type i region. Thus, we can say that SN localizes itself correctly within distance r^* that is the *worst-case accuracy*.

For example, consider that a set of RPs are deployed in a hexagonal structure in which the distance between two neighboring RPs is one unit and the transmission range of RP is 0.78. We can compute the precision $e_i(r)$ for each type i that was shown in Fig. 4. Note that $r^* = \max\{r_1^{(1)}, r_1^{(2)}, r_1^{(3)}\} = \max\{0.2685, 0.2993, 0.3088\}$=0.3088. That is, for this hexagonal structure, SN localizes itself correctly within distance 0.3088.

Note that critical radius $r_1^{(i)}$ is a function of RP's transmission range d. Let $f_i(d)$ be the critical radius for type i, $i = 1, 2, 3$. Then, the worst-case accuracy r^* can be rewritten as $r^*(d) = \max\{f_1(d), f_2(d), f_3(d)\}$. If the transmitting power of RP can be adjusted, the transmission range of RP will vary. We assume that the radius d is bounded within

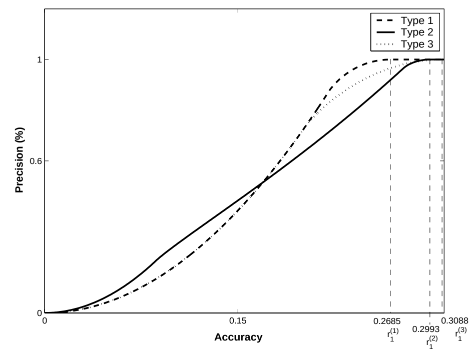


Figure 4. The precision $e_i(r)$ of SN in the type 1, 2, and 3 areas.

$[\frac{1}{\sqrt{3}}, \frac{\sqrt{3}}{2}]$[1]. Let us consider how to arrange the transmission range of RP such that the worst-case accuracy is optimized. This problem is equivalent to finding a radius d such that $r^*(d) = \max\{f_1(d), f_2(d), f_3(d)\}$ is minimized. That is,

$$\begin{aligned} z &= \min_{\frac{1}{\sqrt{3}} \le d \le \frac{\sqrt{3}}{2}} r^*(d) \\ &= \min_{\frac{1}{\sqrt{3}} \le d \le \frac{\sqrt{3}}{2}} \max\{f_1(d), f_2(d), f_3(d)\} \end{aligned} \quad (1)$$

Figure 5 shows the functions $f_1(d), f_2(d)$, and $f_3(d)$, for $\frac{1}{\sqrt{3}} \le d \le \frac{\sqrt{3}}{2}$. The function $f_1(d)$ is an decreasing function and the function $f_3(d)$ is a increasing function where $\frac{1}{\sqrt{3}} \le d \le \frac{\sqrt{3}}{2}$. Let d^* be the radius such that $f_1(d^*) = f_3(d^*)$. Thus,

$$\max\{f_1(d), f_2(d), f_3(d)\} = \begin{cases} f_1(d) & \text{if } \frac{1}{\sqrt{3}} \le d \le d^* \\ f_3(d) & \text{if } d^* \le d \le \frac{\sqrt{3}}{2} \end{cases}$$

and the minimum of $\max\{f_1(d), f_2(d), f_3(d)\}$ occurs at $f_1(d) = f_3(d)$. By numerical method, we find $d^* = 0.7638$ such that $f_1(d^*) \approx f_3(d^*) = 0.2887$.

2) *The average-case accuracy*
Given that the location (x, y) of SN falls in the type i area, the expected accuracy D_i is

$$E[D_i] = \int_{(x,y)\in R_i} \sqrt{(x-x_{c_i})^2 + (y-y_{c_i})^2} \, f(x,y)dxdy$$

where R_i is the localization region of type i and (x_{c_i}, y_{c_i}) is the centroid of R_i. Thus, the expected accuracy of D for the network with hexagonal structure can be found by

$$E[D] = \sum_{i=1}^{3} p_i \, E[D_i]$$

[1]This is because 1) if $d < \frac{1}{\sqrt{3}}$, then there exists some areas that do not covered by RP's signal; 2) if $d > \frac{\sqrt{3}}{2}$, then the type 2 area will be separated into 2 sub-areas.

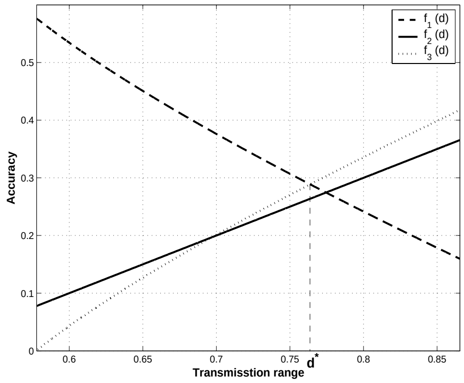


Figure 5. The worst-case accuracy for hexagonal structure.

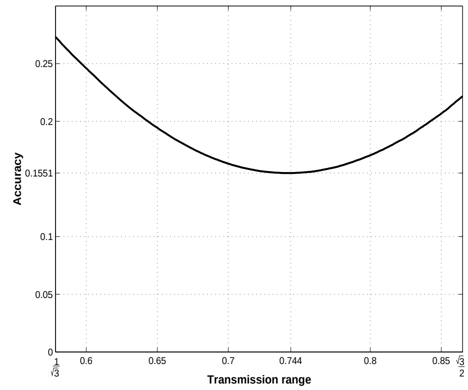


Figure 6. The average accuracy for hexagonal structure.

where p_i is the probability that SN falls in the type i area. By this way, the average accuracy of the proposed method can be evaluated.

Note that the average accuracy $E[D]$ is also a function of RP's transmission range d. Let $g(d)$ be the average accuracy $E[D]$ for the RPs with hexagonal structure having transmission range d. Let us consider how to arrange the transmission range of RP such that the average accuracy is minimized. The problem is to find a radius d such that $z = \min_{\frac{1}{\sqrt{3}} \leq d \leq \frac{\sqrt{3}}{2}} g(d)$.

We can evaluate the average accuracy $E[D]$ by simulation. In our simulation, 10,000 SNs were generated with uniform distribution in the working area of 100×100 unit square. We assume that all RPs are deployed in a hexagonal structure with transmission range $\hat{d}$ and their locations are known in advance. By proposed self-positioning method, each SN can localized itself at position (x_c, y_c). Thus, the positioning error can be found and the average accuracy $g(\hat{d})$ can be evaluated. Furthermore, we find $g(d)$, for $\frac{1}{\sqrt{3}} \leq d \leq \frac{\sqrt{3}}{2}$, as shown in Fig. 6 and the minimum of $g(d)$ is 0.1551 where $d = 0.744$.

5 Conclusions

We proposed a GPS-less self-positioning method for sensor networks. In our method, a set of RPs with overlapping regions of coverage are arranged in hexagonal structure in the sensor network and broadcast the beacon frames. SNs only collect the beacon frames from RPs and use the localization data in the beacon frame to calculate their locations. Note that SNs only take little computation to localize themselves. This kind of localization system with low cost and low computation is very suitable for sensor networks.

References

[1] N. Davies, K. Cheverst, K. Mitchell, and A. Efrat, ”Using and determining location in a context-sensitive tour guide”, *Computer Magazine*, vol. 34, Aug. 2001, pp. 35-41.

[2] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, ”Global positioning system: theory and practice”, fifth edition, *SpringerVerlag*, 2001.

[3] F. Mondinelli and Z.M. Kovacs Vajna, ”Self localizing sensor network architectures”, *Proc. of IMTC*, vol. 1, May 2002, pp. 823-828.

[4] N. Bulusu, J. Heidemann, and D. Estrin ”GPS-less Low Cost Outdoor Localization For Very Small Devices”, *IEEE Personal Communications Magazine*, vol. 7, no. 5, Oct. 2000, pp. 28-34.

[5] C. Drane, M. Macnaughtan, and C. Scott, ”Positioning GSM telephones,” *IEEE Communications Magazine*, vol. 36, Apr. 1998, pp. 46-54.

[6] G.M. Djuknic, and R.E. Richton, ”Geolocation and assisted GPS”, *Computer Magazine*, vol. 34, Feb. 2001, pp. 123-125.

[7] G.J. Morgan-Owen and G. T. Johnston, ”Differential GPS positioning”, *Electronics and Communication Engineering Journal*, vol. 7, Feb. 1995, pp. 11-21.

[8] H.C. Chu and R.H. Jan, ”A Cell-Based Location-Sensing Method for Wireless Networks”, *Wireless Communication and Mobile Computing*, vol. 3, no. 4, Jun. 2003, pp.455-463.

[9] S. Ghahramani, ”Fundamentals of probability”, second edition, *Prentice Hall*, 2000.

附件四

A Weighted Multilateration Positioning Method for Wireless Sensor Networks *

Yu-He Gau, Hung-Chi Chu, and Rong-Hong Jan†
Department of Computer and Information Science
National Chiao Tung University, Hsinchu, 30050, Taiwan
{gis92509, hcchu, rhjan }@cis.nctu.edu.tw

Abstract

Localization problem is one of the most important research issues for wireless sensor networks (WSNs). In this paper, we present a two-phase localization algorithm for WSNs. In the first phase, each sensor node obtains its initial position by DV-hop method. In the second phase, each sensor node gathers the locations and distances to its neighbors, updates and exchanges these location information periodically and then operates the multilateration with different weight values. The simulation result shows that the average position error of the proposed two-phase method is less than 20% of the radio range, and the number of sensor nodes which can be located is larger than 70% of total nodes for a network with low density.

1 Introduction

The topology of wireless sensor network (WSN) is similar to ad-hoc network in which each sensor node in the network should communicate and cooperate with its neighbors to achieve the goal of task. Each node collects, stores, and processes the sensed data, such as temperature, brightness, sound, and so on, and then communicates with neighboring nodes to provide the environmental observation.

In order to make the sensed data more useful, the location of sensor nodes should be determined. There are several location determination methods to be presented [1-17]. In general, the location methods can be divided into two main classes, one is centralized system and the other is distributed system. The centralized system has a central location server receiving the location query, calculating the location information, and replying it back to the query node. In contrast, the distributed system, each node utilizes the location algorithm to calculate its position by itself. The distributed system is more feasible than the centralized system because of the following reasons: First, the location server is a bottleneck of the centralized system and nodes near the location server consume more energy in forwarding location information. Second, system stability depends on the communication links of central location server. If these links are failure, sensor nodes fail to determine their locations.

One of the most popular distributed location methods is Global Positioning System (GPS)[1] that has been shown to be an integrated part of modern navigation. However, it is not suitable for all sensor nodes because of the limitation of sensor node in size, cost, electric power and computational power. Besides, GPS signal is degraded by the environment or jamming. Especially in indoor environments, GPS signal would be blocked and almost unavailable.

In this paper, we develop a distributed location method for sensor nodes with some *beacon nodes*. The *beacon node*, (or called as *beacon* in short) is a sensor node that knows its location. This paper presents a two-phase location method for WSN with a set of beacons. In the first phase, each node estimates its initial position by a rough DV-hop method[2]. In the second phase, each sensor node gathers the location information that includes the locations and distances of its neighbors. And then it updates and exchanges the location information periodically, and operates the multilateration with different weight values. The weight is defined as an error function of positions and distances to show the effect of data accuracy. The multilateration technique does not work well if the

*This research was supported in part by the National Science Council, Taiwan, ROC, under grant NSC 93-2219-E-009-002 and NSC93-2752-E-009-005-PAE and in part by the Intel.

†Corresponding Author. Fax: 886-3-5721490; e-mail: rhjan@cis.nctu.edu.tw

number of available neighbors' location is less then three. In this paper, we propose a method to deal with the multilateration with two neighbors. The simulation results show that the average position error is noticeable.

The paper is organized as follows. In section 2, the related work of existing location method will be introduced. The weighted multilateration algorithm and the simulation result are shown in sections 3 and 4, respectively. Finally, the conclusion and future work are given in section 5.

2 Related work

For WSN applications, location is one of the important issues. In [3], a survey paper notes that early classical location systems such as RADAR [4], Cricket [5], and Active Badge [6] may not suitable for sensor networks.

Based on the operation model, the positioning system can be divided into two main classes: anchor-based and anchor-free. In anchor-based location system, it utilizes some anchor nodes to achieve location determination. In [7] and [8], they use Multidimensional Scaling (MDS) to get global relative coordinate system, and then map this system into absolute coordinate system via anchor node by self-designed location algorithm. In contrast, the location system without anchors is called as anchor-free location system. In [9] and [10], nodes exchange local distance information with others to generate their relative coordinates. The transformation from relative coordinates to absolute coordinates was handed over to certain post-process methods.

3 The weighted multilateration algorithm

The weighted multilateration algorithm is a two-phase method. In the first phase, the DV-hop method in APS [2] is modified. Beacon nodes broadcast their position information, and then those unknown nodes can get their initial positions. In the second phase, unknown nodes only exchange information with their neighbors and then estimate their location by multilateration with weighted coefficient to refine initial node positions. This is because neighbors' position and distance to neighbors have different reliability. In the following, we will present the weighted multilateration algorithm in details.

3.1 The first phase

The goal of this phase is to estimate roughly the initial positions for unknown nodes. The location information will be used in the second phase. This phase is modified from DV-hop method. The stepwise description is given as follows.

1. Beacons broadcast their positions periodically.
2. Beacons maintain the shortest paths to other beacons, and unknown nodes maintain the shortest paths to beacons.
3. Beacons receive information from other beacons to compute their one-hop distances, and then broadcast these values to unknown nodes.
4. Nodes receive information from at least three beacons and estimate distances to beacons. This is done by average one-hop distance multiplied shortest path hop count.
5. Unknown node operates classical multilateration to estimate their positions.

For example, as shown in Figure 1, nodes 1, 2, and 3 are beacons and others are unknown nodes. The details of steps are described as follows.

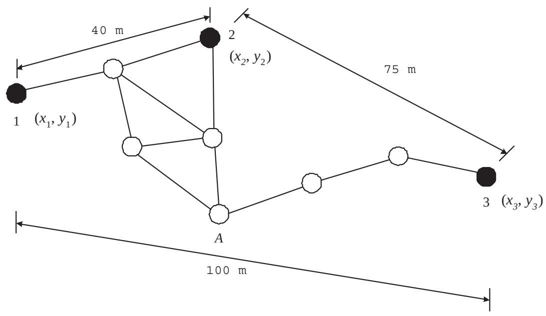


Figure 1: The example for the first phase.

Step 1: Beacons 1, 2, and 3 broadcast their position $(x_1, y_1), (x_2, y_2)$, and (x_3, y_3), respectively.

Step 2: Beacons 1, 2, and 3 estimate Euclidean distance between beacons, i.e. $d_{12} = d_{21} = 40m$, $d_{23} = d_{32} = 75m$, and $d_{13} = d_{31} = 100m$. They also maintain the hop count between beacons, i.e. $h_{12} = d_{21} = 2$, $h_{23} = d_{32} = 5$, and $h_{13} = d_{31} = 6$. Unknown node A can obtain hop count information, i.e., $h_{A1} = 3$, $h_{A2} = 2$, and $h_{A3} = 3$.

Step 3: Beacons compute their average one-hop distances:

1) Beacon1's one-hop distance= $\frac{d_{12}+d_{13}}{h_{12}+h_{13}} = \frac{40+100}{2+6} = 17.5$,
2) Beacon2's one-hop distance= $\frac{d_{21}+d_{23}}{h_{21}+h_{23}} = \frac{40+75}{2+5} = 16.43$,
3) Beacon3's one-hop distance= $\frac{d_{31}+d_{32}}{h_{31}+h_{32}} = \frac{100+75}{6+5} = 15.91$,

Step 4: Node A computes average one-hop distance and then estimates distances to beacons:

1) Average one-hop distance= $\frac{17.5+16.43+15.91}{3} = 16.61$;
2) Measured distance from A to 1, $d_{A1} = 16.61 \times 3 = 49.83$;
3) Measured distance from A to 2, $d_{A2} = 16.61 \times 2 = 33.22$;
4) Measured distance from A to 3, $d_{A3} = 16.61 \times 3 = 49.83$.

Step 5: Node A operates classical multilateration to estimate its initial position:

1) Error functions between estimated distances and measured distances to beacons:

$$\begin{cases} f_{A1} = \sqrt{(x_A - x_1)^2 + (y_A - y_1)^2} - d_{A1}. \\ f_{A2} = \sqrt{(x_A - x_2)^2 + (y_A - y_2)^2} - d_{A2}. \\ f_{A3} = \sqrt{(x_A - x_3)^2 + (y_A - y_3)^2} - d_{A3}. \end{cases}$$

2) Apply Minimum Square Estimation (MSE) to above equations , and then (x_A, y_A) can be estimated by min $\{F = f_{A1}^2 + f_{A2}^2 + f_{A3}^2\}$.

3.2 The second phase

Based on the initial positions obtained in the first phase, nodes exchange location information with neighbors to refine their positions in the second phase. When nodes receive information from at least three neighbors, they can apply certain ranging technology (e.g., RSSI) to measure distances to neighbors, and then use classical multilateration to estimate their positions. However, positions and distances to neighbors have different accuracy. It is obviously that the accuracy of position information between beacons and unknown nodes is different. Similarly, according to the signal degression properties for ranging measurement, the near and far nodes' position information also have different accuracy. Hence, we define a parameter, *weight*, that is the production of neighbor's position weight and distance weight. Position weight represents the accuracy of neighbor's position, and distance weight represents the accuracy of ranging measurement. The classical multilateration can be transformed into "weighted multilateration" with given separate weight for each neighbor. The stepwise description is given as follows.

1. Nodes receive neighbors' positions and their position weight w_p, where $w_p = 0.1$ for unknown node and $w_p = 1$ for beacon.
2. Nodes use ranging technology (e.g. RSSI) to measure distances to their neighbors, and assign the distance weight w_d according to receiving power, where $0.1 \leq w_d \leq 1$.
3. Nodes can obtain the weight w by the production of neighbors' position weight and distance weight (i.e. $w = w_p \times w_d$).
4. Nodes get information from at least three neighbors. The information contain their positions, distances, and weight. The position of nodes can be estimated by the weighted multilateration.
5. Nodes update their position weight to be average neighbors' weight, i.e. $w_p = \frac{\sum w}{N}$ (where N is the number of neighbors), and then flood their w_p and positions to neighbors.
6. Repeating steps $1 \sim 5$, until position error converged. In simulation, the convergent situation will happened less than ten times of repetition.

Take Figure 3.2 for example, node 3 is a beacon and others are unknown nodes.

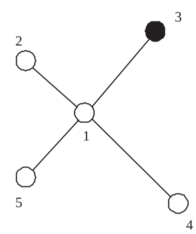


Figure 2: The example for the second phase.

Step 1: Node 1 receives the position of neighbors, $\{(x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), \}$ and the weight of neighbors, $\{w_{2p} = w_{4p} = w_{5p} = 0.1$ and $w_{3p} = 1\}$.

Step 2: Node 1 utilizes the ranging technology to get the distances to its neighbors, $\{d_{12}, d_{13}, d_{14}, d_{15}\}$, and looks up table 1 to obtain the distance weight, $\{w_{2d}, w_{3d}, w_{4d}, w_{5d}\}$.

Step 3: The weight w_j represents the accuracy of neighbors' positions and distances to neighbors. $w_2 = w_{2d} \cdot w_{2p}$, $w_3 = w_{3d} \cdot w_{3p}$, $w_4 = w_{4d} \cdot w_{4p}$, $w_5 = w_{5d} \cdot w_{5p}$.

Step 4: The classical multilateration is modified to "weighted multilateration" and applied the MSE to esitmate (x_1, y_1) by min $\{F = (w_2 \cdot f_{12})^2 + (w_3 \cdot f_{13})^2 + (w_4 \cdot f_{14})^2 + (w_5 \cdot f_{15})^2\}$, where $f_{12} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} - d_{12}$, $f_{13} = \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} - d_{13}$, $f_{14} = \sqrt{(x_1 - x_4)^2 + (y_1 - y_4)^2} - d_{14}$, and $f_{15} = \sqrt{(x_1 - x_5)^2 + (y_1 - y_5)^2} - d_{15}$.

Step 5: Node 1 updates its position weight to be average neighbors' weight:

$$w_{1p} = \frac{w_2 + w_3 + w_4 + w_5}{4}$$

Step 6: Repeat the steps 1 $\sim$ 5 for ten times.

4 Simulation result

All simulations were performed in ns-2[18] environment. We discussed about connectivity and number of nodes to show the performance of the weighted multilateration algorithm. Connectivity of node represents the average number of neighbors of nodes and it was affected by radio transmission range. In order to show the effect of connectivity, we simulated various connectivity for sensor nodes about the position error and the result was shown in Figure 3. In this case, the working area for 100 sensor nodes with random placement was $1000 \times 1000\, m^2$. The number of beacons were 20% of total sensor nodes (i.e. beacon ratio was 20%). When the connectivity was 12, the position error was less than 50%R in the first phase and was less than 10%R in the second phase. Note that the 50%R means that the position error is half of the radio range R. The improvement of position error from the first phase to the second phase is about one-fifth when connectivity was 12.

With the same working area, sensor nodes and beacon ratio, we considered about the number of

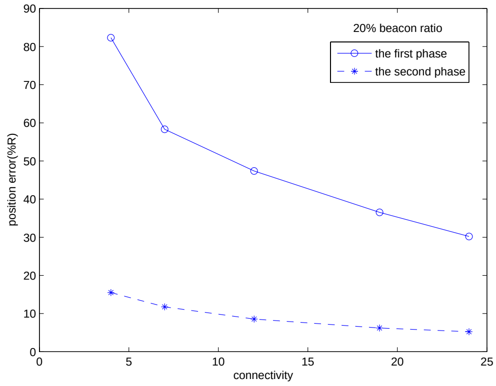


Figure 3: The position error for various connectivity in the two phases.

located nodes for various connectivity and the result was shown in Figure 4. When connectivity was 4, the number of located nodes was greater than 70% both in the first phase and the second phase. When connectivity is larger than 18, the number of located nodes is close to 90% in the two phases. It is obvious that the number of located nodes in the first phase is greater than in the second phase. This is because a node can not be located in the first phase(DV-hop mehtod) when it disconnected the network. In the second phase, the weighted multilateration method needed at least three neighbors to perform location determination.

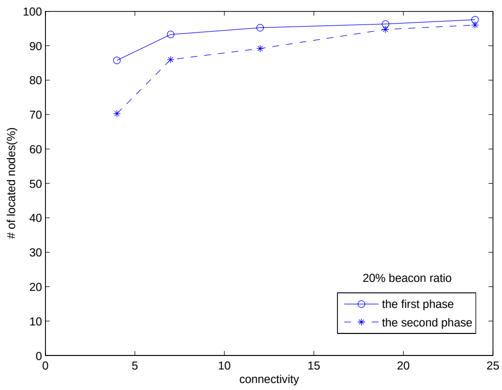


Figure 4: The number of located nodes for various connectivity in the two phases.

The relationship about connectivity and position error with different beacon ratio for the two phases was shown in Figure 5 and 6. The position error was decrease from 200%R (40%R) to 50%R

($10\%R$) in the first(second) phase with 10% beacon ratio. The second phase can truly refine the position error from the first phase.

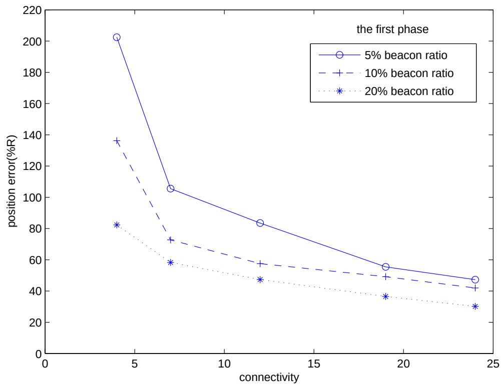


Figure 5: The relationship between connectivity and position error for 5%, 10% and 20% beacon ratio in the first phase.

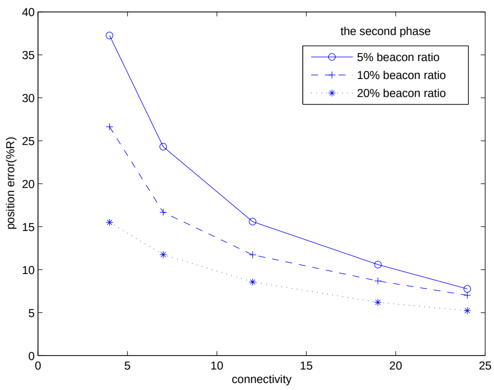


Figure 6: The relationship between connectivity and position error for 5%, 10% and 20% beacon ratio in the second phase.

The proposed algorithm was compared with the referenced algorithm [11] in Figure 7. When connectivity is less than 24, the position error of proposed algorithm is better than the referred algorithm especially for low connectivity. In contract, when the connectivity is larger than 24, the performance of the both algorithms were closed to each other. This is because that high connectivity will give more information from its neighbors to improve the both algorithms.

After the discussion of connectivity, we considered about the position error and the number of located nodes for the number of nodes in the network. The simulation results were shown in Figure 8 and 9. In this case, the working area is $1000 \times 1000\ m^2$. Each sensor nodes and beacons had the same radio range $R = 100\ m$ and they are put with random. In Figure 8, the position error for both of the first and second phase will be improved as the number of nodes increases. The refinement of the second phase will keep the improvement of $50\%R$ of the first phase for the number of nodes is less than 100.

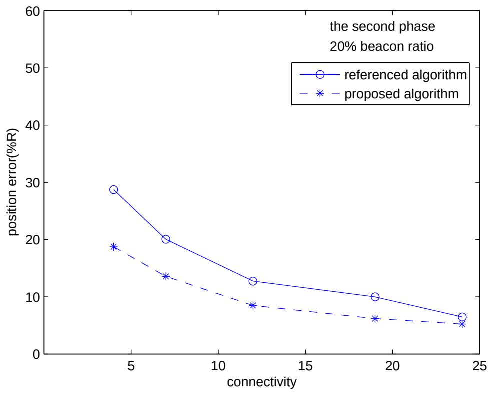


Figure 7: Comparison of referenced algorithm and proposed algorithm.

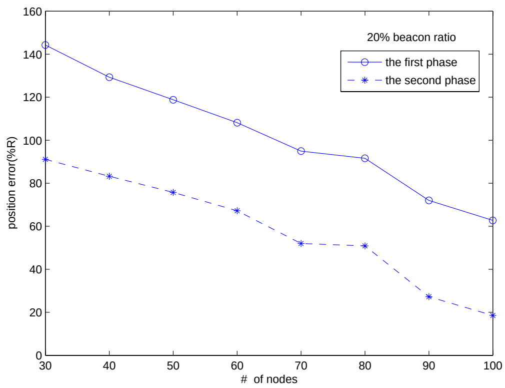


Figure 8: The position error in the two phases with various number of nodes.

In Figure 9, it showed the relationship between the number of nodes and located nodes. The number of located nodes in the first phase remain larger than in the second phase. However, as the number of nodes increase, the number of located nodes were increase in both phases. This is because as the number of nodes increase, the connectivity also increase.

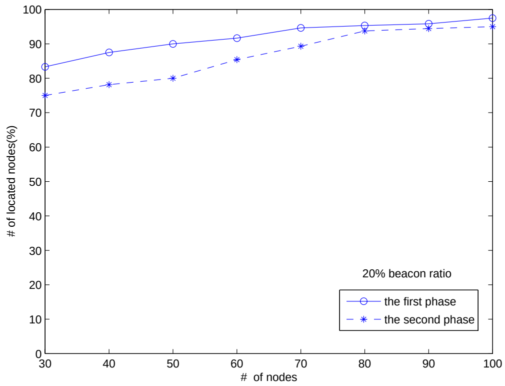


Figure 9: The number of located nodes in the two phases with various number of nodes.

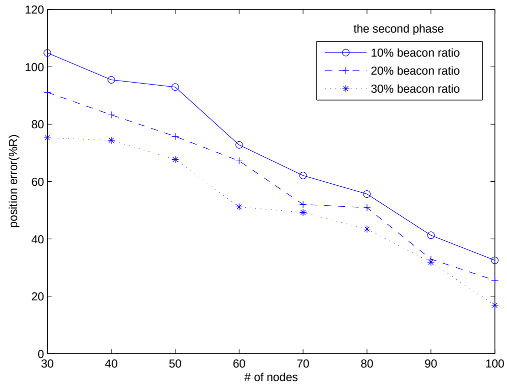


Figure 11: The relationship between number of nodes and position error for 10%, 20%, and 30% beacon ratio in the second phase.

The relationship between the number of nodes and the position error with different beacon ratio for these two phases was shown in Figure 10 and 11. The position error was decrease from 160%R(105%R) to 70%R(35%R) in the first(second) phase. The second phase can truly refine the position error from the first phase.

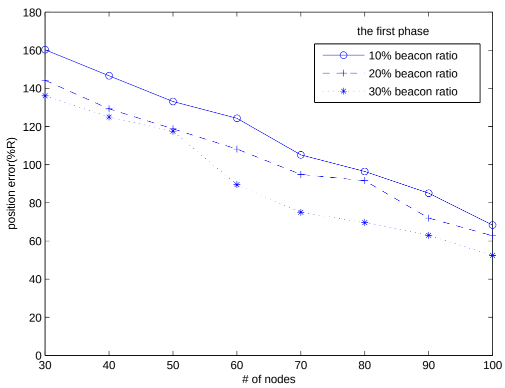


Figure 10: The relationship between number of nodes and position error for 10%, 20%, and 30% beacon ratio in the first phase.

According to the simulation, the performance of the proposed method is better than the referred method with reasonable position accuracy. Furthermore, the second phase can truly refine the position error from the first phase.

5 Conclusion

The proposed weighted multilateration positioning method only needs fewer beacons to perform the positioning function. It contains two phases. The first phase adopts the DV-hop method to compute the average one-hop distance and the shortest path of hop count. These location information can be applied to the classical multilateration to estimate initial positions. In the second phase, unknown nodes exchange information with neighbors and apply the modified weighted multilateration method to refine their positions repeatedly. In simulation, the position error of the proposed method is better than the referred algorithm when the connectivity is less than 24. The limitation of the proposed method is that if the unknown node has less than three neighbors, it can not be located. We will try to solve this for two ways in the future work. One may use the processing delay trick to wait for enough location information. If a unknown node can be prior processed and successfully turn into located node, it can provide its location information to its neighbors. The other way may use the n-hop message passing to obtain more location information that was came from n-hop nodes.

References

[1] B. Parkinson and J. Spilker. "Global Positioning System(GPS): theory and application". *American Institute of Astronautics and Aeronautics*, vol. 1 and 2, 1996.

[2] D. Niculescu and B. Nath. "Ad-hoc positioning system(APS)". *In IEEE GlobeCom*, vol. 5, pp. 2926-2931, November 2001.

[3] J. Hightower and G. Boriello. "Location systems for ubiquitous computing". *IEEE Computer*, vol. 8, pp. 57-66, August 2001.

[4] P. Bahl and V. Padmanabhan. "RADAR: an in-building RF-based user location and tracking system". *In Proceedings of IEEE Computer and Communications Societies*, pp. 775-84, vol. 2, March 2000.

[5] N. Priyantha, A. Chakraborthy, and H. Balakrishnan. "The cricket location support system". *In ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 32-43, August 2000.

[6] R. Want, A. Hopper, V. Falcao, and J. Gibbons. "The active badge location system". *ACM Transactions on Information Systems 10*, pp. 91-102, Janauary 1992.

[7] X. Ji and H. Zha. "Robust sensor localization algorithm in wireless ad-hoc sensor networks". *In IEEE International Conference on Computer Communications and Networks*, pp. 527-532, 2003.

[8] Y. Shang, W. Ruml, Y. Zhang, and P.J Fromherz. "Localization from mere connectivity". *In Proceedings of the 4th ACM international symposium on Mobile Ad-hoc Networking and Computing*, pp. 201-212, June 2003.

[9] S. Capkun, M.Hamdi, J.P. Hubaux. "GPS-free positioning in mobile ad-hoc networks". *In Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, pp. 3481-3490, January 2001.

[10] N.B. Priyantha, H. Balakrishnan, E. Demaine, and S. Teller. "Anchor-free distributed localization in sensor networks". *Technical Report MIT-LCS-TR-892, MIT Lab for Computer Science*, April 2003.

[11] C. Savarese, J. Rabaey, and K. Langendoen. "Robust positioning algorithm for distributed ad-hoc wireless sensor networks". *In USENIX Technical Annual Conference*, pp. 317-327, June 2002.

[12] A. Savvides, C.C. Han, and M. Srivastava. "Dynamic fine-grained localization in ad-hoc networks of sensors". *In Proceedings of the 7th ACM International Conference on Mobile Computing and Networking*, pp. 166-179, July 2001.

[13] A. Nasipuri and K. Li. "A directionality based location discovery scheme for wireless sensor networks". *In 1st ACM Int'l Workshop on Wireless Sensor Networks and Applications*, pp. 105-111, September 2002.

[14] D. Niculescu and B. Nath. "Ad-hoc positioning system(APS) using AOA". *In Proceedings of the 20st Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3, pp. 1734-1743, April 2003.

[15] A. Savvides, H. Park, and M. Strvastava. "The bits and flops of the n-hop multilateration primitive for node localization problems". *In 1st ACM Int'l Workshop on Wireless Sensor Networks and Application*, pp. 112-121, September 2002.

[16] C. Savarese, J. Rabaey, and J. Beutel. "Locationing in distributed ad-hoc wireless sensor networks". *In Proceedings of ICASSP*, pp. 2037-2040, May 2001.

[17] N. Bulusu, J. Heidemann, and D. Estrin. "GPS-less low cost outdoor localization for very small devices". *Technique Report 00-729, Computer Science Department*, April 2000.

[18] UCB/LBNL/VINT, Official ns manual [Online], Available: http://www.isi.edu/nsnam/ns/doc/index.html