

策略與計費控制系統之深度封包檢測成本與效能分析

Cost Analysis of Deep Packet Inspection in PCC Architecture

林專勝

國立成功大學電腦與通信工程研究所

cslin.ncku@gmail.com

蘇淑茵

國立成功大學電腦與通信工程研究所

sisou@mail.ncku.edu.tw

摘要

隨著行動裝置的普及，有越來越多的行動應用程式被開發出來，然而這些應用程式大多屬於 Over-The-Top (OTT) 應用程式，即由應用程式開發商所提供，僅使用電信業者的網路頻寬，這造成電信業者有非常大量的頻寬支出卻僅帶來極少量的收入。因此有非常多的電信業者嘗試加入深度封包檢測(Deep Packet Inspection, DPI) 的機制，將封包依其特徵賦予不同的 Quality of Service (QoS) 管理機制，然而對於路由器及整體系統來說，這將會是一個非常大的負擔。因此，在本論文中，主要分析深度封包檢測的機制對於整體系統的影響，探討在不同的偵測頻率下，對於封包的辨識率的影響，以期能夠達到在較低的成本下得到較高的封包特徵辨識成功率，以供電信商在建立深度封包檢測機制的參考依據。

關鍵詞：深度封包檢測、長期演進技術、流量管理、QoS、策略與計費控制系統。

Abstract

Along with being more popular of the mobile device, there are more and more mobile applications have been developed nowadays. However among these applications are "Over-The-Top (OTT)" applications which have been provided by the vendor of applications holds the biggest amount of them. And OTT merely use bandwidth which is offered by operators and this has caused operator have to afford a big amount of money for bandwidth but with very less income in return. Hence, here are many operators trying their best to join the mechanism of Deep Packet Inspection (DPI), and then assign the packets with different Quality of Service (QoS) rules according to their different kinds of signature. However to the router and backbones this will be a heavy burden. Consequently, this

research mainly analyze how the mechanism of the DPI affects the Policy Control and Charging (PCC) System, and moves forward a step to investigate the rate of identification of DPI at different detecting rate in order to get the high successful detecting rate but with the lower payment. In conclusion, this research's main purpose is to build reference of DPI mechanism.

Keywords: DPI, LTE, PCC, QoS, Data Traffic Management.

一、前言

隨著行動裝置的普及，全球對於行動網路頻寬的需要日益增加，根據思科 2014 年的研究報告[1]，預計 2013 至 2018 年，每年的行動網路流量將以 61% 的幅度增加，如圖 1 所示，在 2013 年時每月的流量為 1.5EB 而預估 2014 年每月的行動網路流量會成長至 2.6EB，而到 2018 年每月將會高達 15.9EB。而智慧型裝置產生流量的在 2014 年預估占所有行動流量的 90%，而到 2018 預估將達到 95%，這意味著大多數的行動流量是由智慧型裝置所產生的。

也因為行動裝置的普及，有越來越多的行動應用程式問世，例如：Line、Facebook、Gmail、YouTube 等，而多數的應用程式是屬於 "Over-The-Top" (OTT) 的應用程式[2] [3]，即透過電信業者提供的網路頻寬來提供屬於自己的服務，例如：透過 Line、Skype 的應用程式，可以達到打電話或是簡訊的功能，但除了頻寬的使用費外，使用者並不需要支付傳統的電話費。這對於電信業者而言是一大負擔，因為當這些 OTT 應用程式提供相似於電信業者的服務，來取代原有電信業者所提供的服務，會造成電信業者的頻寬支出大幅增加，但營收卻大量降低。

而從圖 2 可以看到，在行動應用流量上，影音流量占了整體約八成，這對於網路頻寬管

理上更是一大問題，因為大多數的影音應用程式會消耗大量的網路頻寬，造成其他的使用者可能得不到足夠的網路頻寬，而這些影音流量並非全部都必須給予非常高的優先權，因為對於使用者而言，使用影像服務能夠忍受的等待時間是高於其他的應用程式的，例如：使用者可以忍受在 YouTube 上等待 10 秒鐘看一段影片，但卻未必能夠忍受等待 1 秒鐘來開啟一個網頁[4]。

因此，目前有非常多的研究是在策略控制與計費系統 (Policy Control and Charging, PCC)[11] 中加入分析封包特徵的機制，並根據封包的特徵而給予不同的 Quality of Service (QoS) 管理規則[4] [12]，例如：Gmail 的封包優先權或許會高於 YouTube 的封包等。而目前常見的深度封包檢測 (Deep Packet Inspection, DPI) 方法有[6] [7] [8]：

- 分析通訊埠：根據封包所使用的通訊埠來進行封包分類，例如：通訊埠 80 代表是 HTTP 協定，443 是 HTTPS，110 是 POP3。然而僅判斷通訊埠，以目前來說是不足夠的，因為有非常多的應用程式會改為使用 80 和 443 通訊埠。
- 封包字串比對：分析封包的標頭檔是否包含特定的字串，例如：Line 的標頭檔內將會包含字串“line”，利用這種特徵也能夠尋找出特定的封包。[9]
- 分析封包長度：有部分的應用程式的封包長度和長度的序列是固定的，例如：Skype 在建立通話時，應用程式會先發送一個長度為 18 bytes 的訊息給伺服器，而伺服器會回傳一個長度為 11 bytes 的訊息給應用程式，當應用程式接收到訊息後會再傳送一個 23 bytes 的訊息到伺服器，最後伺服器會回傳一個封包長度為 18、51 或 53 bytes 的封包給應用程式，以此建立連線，因此可以利用這樣的方式判斷目前使用者使用的應用程式為何。
- 分析封包傳送行為：部分應用程式傳送封包的行為是固定的，例如：P2P 應用程式的傳送封包的數量必定比一般 HTTP 的封包多，且封包的傳送和接收量通常大於 HTTP 且速度較不穩定，因此透過不同的封包行為，我們可以判斷不同可能是屬於不同的應用程式。[10]

然而，不論是使用哪一種方法，均會造成策略管理與計費系統非常大負擔，因為我們必須針對每一個使用者服務串流進行分析，而

這個分析可能必須要將多個封包進行多次的比對，然後再給予不同的 QoS 規則，這個步驟可能需要耗用非常多的計算，也可能因為封包量過多，而降低網路速度。

因此，在本篇論文中，我們將分析服務串流的行為，並且提出建立分析模型，以讓電信業者在建立 DPI 分析機制時，能考慮到服務串流的特性，以及各種不同的影響，以達到能夠在較低的成本下，盡可能地分析出大多數的串流。

在接下來第二章將會介紹我們所建立的分析模型，第三章將會建立模擬模型來分析在不同的因素下，對於 DPI 所需要的成本及結果，最後是本論文的結論。

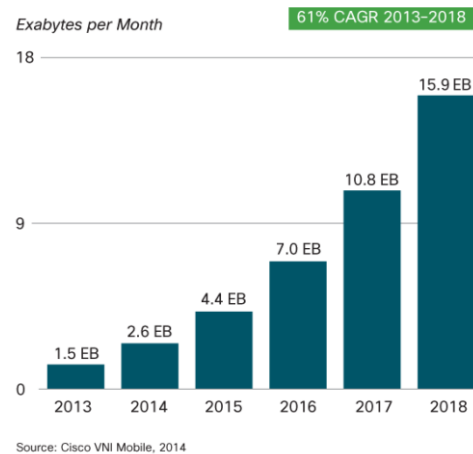


圖 1 全球型行動網路流量

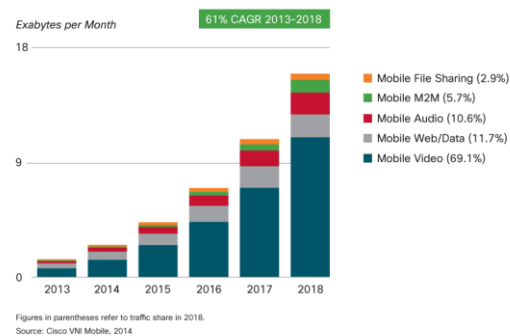


圖 2 行動應用流量比例

二、分析模型

一般而言，在 3GPP 核心網路中，當使用者要求建立一個 Session 時，策略與計費執行功能單元 (Policy and Charging Enforcement Function, PCEF) 會對使用者的封包進行偵測，以了解使用者正在使用何種應用程式，即

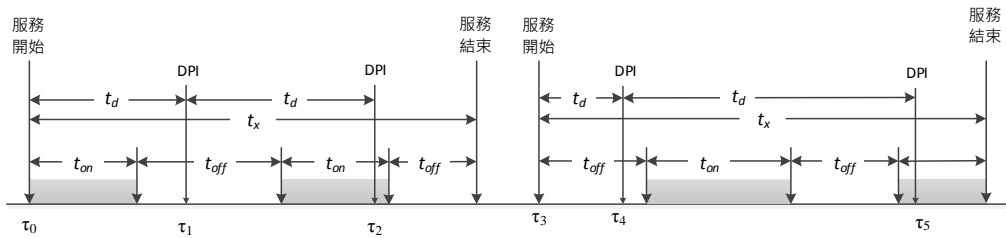


圖 3 深度封包檢測與 session 之時序圖

進行深度封包檢測[11][12]，如果被偵測的封包具有某些應用程式的特徵值，則策略和計費規則功能單元(Policy and Charging Rule Function, PCRF)可以再針對使用者目前所使用的應用程式提供不同的 QoS 規則，以達到較好的網路品質管理，然而可能不是所有的封包皆具有應用程式的特徵值，因此如果 PCEF 對每一個封包都進行檢測將會耗費大量的資源，所以比較可行的方法應該是隨機針對使用者的封包進行檢測。

圖 3 是我們所建立的分析模型[13]，圖 3 繪出從服務開始到結束，深度封包檢測與 session 的封包之間的關係的時序圖[12]，我們可以藉由此分析模型來對深度封包檢測的效能進行更深入的探討。

一個服務的服務時間長度，主要可以由兩個方式來決定，一個是系統所允許使用者使用的最長時間，即系統可能每次僅允許使用者使用該服務一段固定的時間，若時間到而使用者還需要使用的話，則使用者必須要重新提出申請，而另一個是使用者真實的使用時間，即使用者自己提出停止使用該項服務，如圖 3 中， t_x 代表從服務開始到結束所經過的時間， t_x 可以用 $\min\{t_m, t_l\}$ 來表示，即 t_m 和 t_l 的最小值，其中 t_m 是指系統所允許使用者使用服務的最長時間， t_l 是指使用者真實的使用時間。

而在服務時間 t_x 內，使用者的封包可能會包含應用程式的特徵值，在圖 3 中我們使用 t_{on} 來表示一段時間內的封包內擁有特徵值，而如果封包內不包含應用程式的特徵值，則使用 t_{off} 來表示之。

而圖 3 中，DPI 代表 PCEF 對封包進行深度封包檢測，而 t_d 代表兩次 DPI 的間隔時間，而從圖中可以觀察到，當 DPI 落在 t_{on} 的時間內(如圖 3 中 τ_2, τ_4)，則代表此次的 DPI 檢測是有效的，PCEF 確實了解此服務是屬於什麼應用程式，PCRF 能夠重新制定 QoS 規則，而 PCEF 也不需要再對此服務進行檢測；若 DPI 是落在 t_{off} (如圖 3 中 τ_1, τ_3)，則代表此次 DPI

檢測無法測出目前使用者正在使用何種應用程式，必須要再繼續進行檢測。

對於深度封包檢測來說，主要的目標是在最短的時間內能夠偵測出使用者正在使用何種的應用程式，以期能夠立即執行相對應的 QoS 規則，然而如果無法在服務時間內檢測出封包的特徵值，或是為了檢測出封包的特徵值而進行了大量的測試，都不是一個良好的現象，因此較為合理的方式應該是設定一個門檻值，在可接受的成本下檢測封包，以期能夠達到合理可接受的檢測成功率，接下來我們將分析不同的檢測率下，對於封包成功檢測的影響。

三、模擬與結果分析

(一) 參數設定

我們建立了一個離散時間模擬模型，以用來分析在不同的參數下，執行深度封包檢測的效能。主要有以下的參數：

- t_x ：代表一個服務所執行的時間，取系統允許使用者使用服務的時間及使用者自己所宣告的使用時間的最小值，可用 $\min\{t_m, t_l\}$ 表示。
- t_m ：代表系統允許使用者使用服務的時間， t_m 是一個隨機變數，期望值為 $1/\eta$ ，變異數為 v_m 。
- t_l ：代表使用者所需要使用的時間， t_l 是一個隨機變數，期望值為 $1/\sigma$ ，變異數為 v_l 。
- t_{on} ：代表封包內包含應用程式特徵值的時間區間， t_{on} 是一個隨機變數，期望值為 $1/\mu_1$ ，變異數為 v_1 。
- t_{off} ：代表封包內不包含應用程式特徵值得時間區間， t_{off} 是一個隨機變數，期望值為 $1/\mu_2$ ，變異數為 v_2 。
- t_d ：代表兩次 DPI 的間隔時間， t_d 是一個

隨機變數，期望值為 $E[t_d]$ ，變異數為 v_d 。

- P ：代表服務剛開始的封包內包含應用程式的特徵值的機率， $P=\{0,0.5,1\}$ ，其中 0 代表剛開始的封包均不包含特徵值，1 代表剛開始的封包均包含特徵值。

在這個實驗當中，我們主要測得以下幾個結果：

- $E[N]$ ：代表在一個服務開始後，經過幾次 DPI 才能夠偵測出使用者目前使用何種應用程式。
- $E[T_s]$ ：代表在一個服務開始後，經過多久的時間才能夠偵測出使用者目前使用的為何種應用程式。
- Success Detected Rate：代表在所有被建立的服務中，有多少比率的服務能夠被偵測出是屬於何種應用程式。

(二) 實驗結果分析

a) 服務時間長度 t_x 與 DPI 偵測頻率之影響

圖 4 及圖 5 分別繪出在不同的 t_m 和 t_l 時間下對於 DPI 的結果的影響，從圖 4 中可以看到，當 t_l 大於 t_m 時，因為 $t_x = \min\{t_m, t_l\}$ ，所以 t_x 會被設為 t_m ，因此雖然使用者預計使用的時間為 t_l 但依然會被設為 t_m ，因此可以從圖 4 中看到， t_l 大於 t_m 的結果與 t_l 等於 t_m 的結果是相同的，而當 t_l 小於 t_m 時，因為 t_x 會等於 t_l ，因為 t_l 的時間會變得較短，因為能夠進行 DPI 檢測的時間將會變低，所以可能會造成部分的 session 在尚未被檢測出特徵值的情況下就結束了，因此 DPI 成功率會較服務時間較長的 session 低。

而從圖 4 和圖 5 也可以觀察到，當 t_{on} 的時間較長時，因為特徵封包出現的頻率較高，所以有較高的機會能夠偵測到 session 所使用的應用程式。

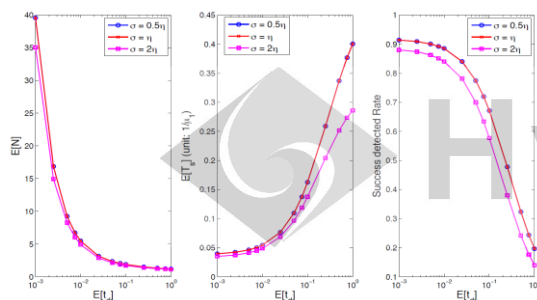


圖 4 服務時間長度 t_x 與 DPI 偵測頻率之影響 ($\mu_1 = \mu_2 = 10\eta$)

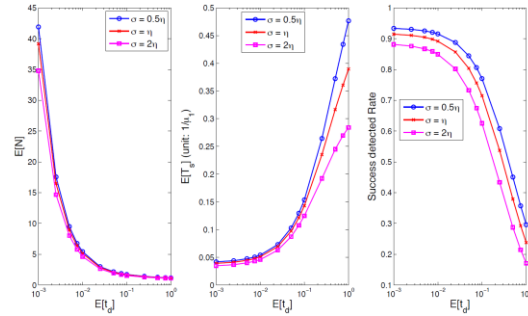


圖 5 服務時間長度 t_x 與 DPI 偵測頻率之影響 ($\mu_1 = 5\eta, \mu_2 = 10\eta$)

b) t_{on} 與 t_{off} 之影響

圖 6 繪出當不同長度的 t_{on} 與 t_{off} 對於 DPI 之影響。從圖 6 可以看到，當 t_{on} 的時間越長時，特徵封包被檢測出來的機率會更高，這是因為當 t_{on} 的時間越長代表特徵封包出現的時間越長，也因此我們有更多的機會可以檢測到戴有特徵值的封包。

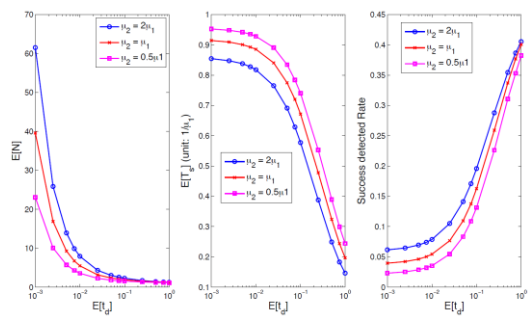


圖 6 t_{on} 與 t_{off} 之影響 ($\mu_1 = 10\eta, \eta = \sigma$)

而從圖中可以觀察到，當 t_{on} 長度為 t_{off} 之兩倍時， $E[N]$ 之間的差距也達到 2/3，因此，我們可以得出 $E[N] = \mu_1 / (\mu_1 + \mu_2)$ ，即在進行 DPI 檢測時，我們有 2/3 的機率可以檢測到帶有特徵值的封包。

c) 不同之 P 對於 DPI 之影響

圖 7 繪出當特徵值越快出現，即 P 越大，對於 DPI 結果之影響，首先，我們可以看到當 P 越大時，即特徵封包較快出現在服務開始的時候，則 $E[N]$ 越小，這代表越快可以檢測出封包的特徵值，這是因為當封包的特徵值越快出現時，我們越有機會在較前面幾次 DPI 檢測中發現封包的特徵值，因此所需要的檢測次數就會下降，也因此需要經過的檢測時間 $E[T_s]$

也越短。而相對來說當封包特徵值出現的時間越早，意味著我們有更多的時間可以找出封包的特徵值，因此成功偵測特徵封包的機率也會較為上升。

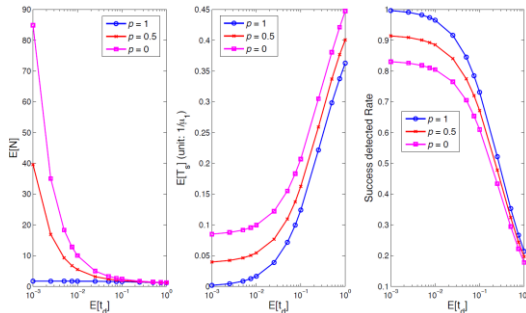


圖 7 不同之 P 對於 DPI 之影響

$$(\mu_1 = \mu_2 = 10\eta, \eta = \sigma)$$

d) 變動與固定的 $E[t_d]$ 對於 DPI 之影響

圖 8 繪出當 $E[t_d]$ 時間長度為指數分布或是固定時間對於 DPI 之影響。我們可以看到，無論 $E[t_d]$ 是指數分布或是固定時間，對於 $E[N]$ 均沒有太大的影響，這是因為在相同的平均值下，指數分布和固定時間的在固定的服務時間長度下，所進行的 DPI 次數是相近的，但是 $E[T_s]$ 卻有所不同，這是因為固定時間的每一次 DPI 的間隔時間會大於指數分布的間隔時間，因此雖然進行 DPI 次數相同，但所經過的時間會有所不同，而在成功率也可以看到，使用固定時間的成功率較指數時間為低，這是因為使用固定時間的偵測時間較長，則有可能部分的服務已經終止了，所以成功率會略為下降，然而由於固定時間的設計較為簡單，而成功率的差距並不是很大，因此也還是可以做為設計的參考。

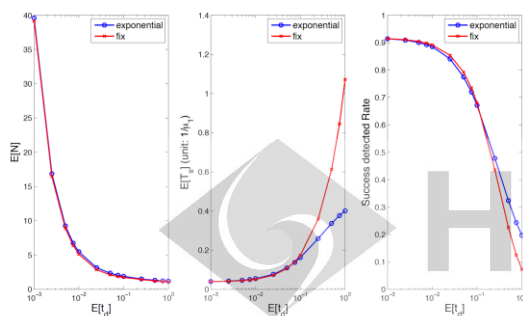


圖 8 變動與固定的 $E[t_d]$ ($\mu_1 = \mu_2 = 10\eta, \eta = \sigma$)

四、結論

在本論文中我們提出深度封包檢測的分析模型，並對於此模型進行了效能分析。我們可以從實驗中觀察到，當進行深度封包檢測的次數越多能夠越有機會找出具有特徵值的封包，以找出適合該服務的 QoS 規則，然而越頻繁的深度封包檢測將會提高檢測的成本，這是使用者必須要注意的部分。而當具有特徵值的封包出現的機率較高時，我們也可以有較高的機率在較短的時間內找到該封包，因此，在設計系統時，應該要盡可能的尋找各種可能的特徵值，如此將會有較高的機會找到特徵封包，給予不同的 QoS 規則。

而從實驗中也可以看到，兩次深度封包檢測的間隔時間如果是指數分布，對於封包偵測的成功率將會比固定時間長度來得好，這是因為指數分布的間隔時間會有機會在較短的時間內進行更多次的偵測，因此可以提高偵測的成功率，然而，在系統的設計上將會變得較為複雜，這是在設計上必須要作權衡的部分。

本論文主要是針對深度封包檢測的成本與偵測成功率進行分析，而在論文中我們建立了一個分析模型，電信營運商將可以利用這個分析模型來對其電信網路中的深度封包檢測功能，進行效能分析，以期能夠在較低的成本控制下達到較高的偵測成功率，而在未來我們將會對分析模型提出數學分析的工具，以期能夠更有效率的對深度封包檢測進行成本與效能的檢測。

參考文獻

- [1] CISCO, Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2013–2018 (2014), http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.html.
- [2] C. Janssen, “Over-the-Top Application (OTT),” <http://www.techopedia.com/definition/29145/over-the-top-application-ott>.
- [3] D. York, “What is an Over-The-Top (OTT) Application or Service? - A Brief Explanation,” <http://www.disruptivetelephony.com/2012/07/what-is-an-over-the-top-ott-application-or-service-a-brief-explanation.html>.
- [4] K. Kimbler, M. Taylor, “Value added mobile broadband services Innovation driven transformation of the ‘smart pipe’,” *IEEE 2012 16th International Conference on Intelligence in Next Generation Networks (ICIN)*, pp. 30-34, 2012.
- [5] M. Lott, J. Karppinen, J. Kross, “Service differentiation for enhanced consumer experience,”

- riences”, *IEEE 2011 15th International Conference on Intelligence in Next Generation Networks (ICIN)*, pp. 58-63, 2011.
- [6] Allot Communications, Digging Deeper Into Deep Packet Inspection (DPI), 2007.
- [7] C. Parsons, “Deep Packet Inspection and Its Predecessors,” <http://www.christopher-parsons.com/Main/wp-content/uploads/2013/02/DPI-and-Its-Predecessors-3.5.pdf>.
- [8] M. Finsterbusch, C. Richter, E. Rocha, J.-A. Muller, K. Hanssger, “A Survey of Payload-Based Traffic Classification Approaches,” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 1135 - 1156, May, 2014.
- [9] P.-C. Lin, Y.-D. Lin, Y.-C. Lai, T.-H. Lee, “Using String Matching for Deep Packet Inspection,” *IEEE Computer*, vol. 41, no. 4, pp. 23-28, April, 2008.
- [10] M. Muellera, H. Asghari, “Deep packet inspection and bandwidth management: Battles over BitTorrent in Canada and the United States,” *Telecommunications Policy*, vol. 36, no. 6, pp. 462–475, July, 2012.
- [11] 3GPP, “3rd Generation Partnership Project; Technical Specification Group Service and System Aspects; Policy and charging control architecture,” 2014, technical Specification 3GPP TS 23.203 Version 13.0.1 (2014-06), 2014.
- [12] 3GPP, “3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Study on Application Based Charging (ABC),” 2013, Technical Report 3GPP TR 23.800 Version 12.0.0 (2013-03), 2013.
- [13] S.-I. Sou, Y.-B. Lin, C.-L. Luo, “Cost Analysis of Short Message Retransmissions,” *IEEE Transactions on Mobile Computing*, vol. 9, no. 2, pp. 215 - 225, Feb, 2010.

