

利用多層編碼配合 DCCP 形成與 TCP 友善的網路電話

連耀南 林耿誠

國立政治大學資訊科學系

Email: {lien, g10011}@cs.nccu.edu.tw,

黃智賢

中華電信研究所 客服研究室

Email: frank210@cht.com.tw

摘要—壅塞控制是網路管理的重大問題。網路電話大部分使用 UDP 作為傳輸層協定，UDP 不具壅塞控制機制，有礙於網路之和諧共用。要達成網路電話在網路壅塞時有壅塞控制能力，且能有好的通話品質，必須要根據網路的狀況調整封包傳送速度或是封包大小。

本研究利用多層編碼配合 DCCP 形成與 TCP 友善的網路電話。壅塞控制必須針對網路中不同程度的壅塞作出適當程度的反應，因此我們改進語音編解碼器，設計出多層語音編碼並設計出可以與之配合的 DCCP，讓 DCCP 依照網路的狀況送出不同層級的語音封包。

本研究透過實際網路的實驗環境中評估以 CBR over UDP、Flexible Bit-Rate 以及 Scalable Codec 三種方式傳輸網路電話封包的效能。並也評估 Scalable Codec VoIP 與 TCP 同時存在於頻寬不足的網路中，對於頻寬競爭能力的表現。

關鍵詞—壅塞控制機制、DCCP、網路電話、語音編解碼器

一、簡介

網路應用程式除了期望取得足夠的頻寬，也必須要講求頻寬取得的公平性。本研究主要研究的對象是網路應用程式中的網路電話，可歸納為以下兩個研究問題：如何設計供網路電話使用的具有壅塞控制功能以及公平競爭網路頻寬的傳輸協定？如何在網路頻寬不足時可以維持較好的語音品質？

1.1 VoIP over DCCP

傳輸層傳輸協定負責從傳送端傳送封包到接收端，並負責控制封包傳送速率。目前大部分網路應用程式都使用 UDP 或 TCP 做為傳輸協議來傳送封包。UDP 傳輸協定不具有壅塞控制機制，因此其對於網路壅塞沒有調節作用。TCP 傳輸協定則具有壅塞控制機制，當網路狀況改變時，TCP 會透過壅塞控制的機制改變傳送封包的速率，以配合傳送端和接收端之間的網路負載。

假設讓 VoIP 採用具有壅塞控制功能的 TCP

作為傳輸協定，可以達成讓 VoIP 有壅塞控制功能的目的，但是因為 TCP 保證封包一定都要到達，所以有 ACK 以及重送的機制。網路電話可以容忍一定程度的封包遺失，且有較嚴格的時效性（延遲時間不得多於 300ms），因此重送遺失的封包，非但可能因失去時效性而作廢，且非屬接收端重建語音所必需。由於要達到語音通話的即時性，網路電話的封包傳遞時間是分秒必爭的，因此發送端要盡快的將音訊轉成封包並送出於網路上。因此，目前大部分的網路電話都採用 UDP 作為傳輸協議，但 UDP 沒辦法調整封包傳送率，沒有壅塞控制機制，有礙於網路之和諧共用。VoIP 使用 Constant Bit-rate(CBR) 的編解碼器並使用 UDP 做為傳輸協定，以固定的速率向網路送出封包。在網路壅塞時，並未依照網路狀況調整速率，會造成封包遺失率增加，影響語音品質。同樣是非連接導向並且有壅塞控制機制的 DCCP 因而被提出，DCCP 定義於[7]，是有壅塞控制功能的不可靠傳輸協議，沒有重送機制的設計，並且 DCCP 提供多種壅塞控制機制可選擇。

DCCP 有壅塞控制功能，且不需要保證封包到達，因此可以考慮採用 DCCP 做為網路電話的傳輸層協定。如果改而採用 DCCP 作為網路電話的傳輸層協定，雖然可以達到壅塞控制的目的，但是，是否可以維持網路電話所需的傳輸效能？以及在網路頻寬不足時是否可以維持語音通話品質？是本文要研究的問題。

假設 VoIP 改使用 CBR 的編解碼器並配合 DCCP 做為傳輸協定，如圖 2，VoIP 的傳送端在收到音訊後，會盡快封裝成封包送到網路以降低延遲時間。但 DCCP 的壅塞控制機制，是透過調整封包發送間隔來降低傳送速率，當間隔時間多於封包的 Inter-arrived time，語音封包會被暫

存於 buffer 之中，稍後再行送出，因而增加了延遲時間，若到達接收端的時間超過一定時限 (300ms)，封包就被捨棄，大幅降低語音品質。因此，此種將封包暫存於 Buffer 的機制，有礙於 VoIP 的語音品質，必須改弦易轍。

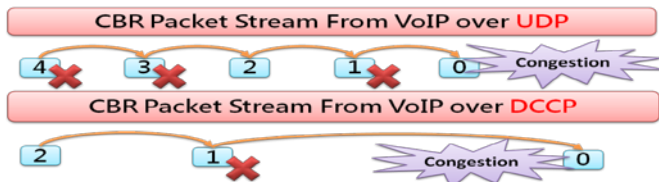


圖 1 VoIP 改使用 Constant Bit-rate 的編解碼器並配合 DCCP 做為傳輸協定的缺點

1.2 Flexible Bit-rate

我們在先前的研究中[9]已提出利用 Flexible Bit-rate 的方式取代一般 VoIP 所使用的 CBR，如圖 3，當 VoIP 改採用 Flexible Bit-rate 的方式時，當網路壅塞發生，不會將封包暫存於 Buffer 中，而是改成降低 Codec 的 Bit-rate 來縮減封包大小，在不拉長延遲時間的情況下，達成降低傳送率的效果，如此，可以減少逾時的封包，並協助減緩網路壅塞的狀況。

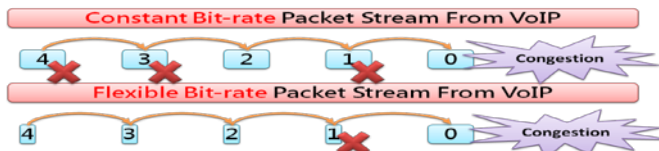


圖 2 VoIP 使用 Flexible Bit-rate 的編解碼器的優點

1.3 Scalable Codec over DCCP

藉由之前章節的討論，我們發現可以考慮採用 DCCP 作為傳輸層協定來達成壅塞控制功能，並且也希望結合 Flexible Bit-rate 的優點，讓 VoIP 在網路壅塞時可以有好的語音品質。但要將 DCCP 配合 Flexible Bit-rate 存在有實際的困難。因此，此研究提出 Scalable Codec 配合 DCCP 的方式。如圖 4 所示，網路七層架構依層次區分，VoIP 位於 Application 層，之後的層次是 Transport 層、Network 層，DCCP 位於 Transport 層。網路七層資訊的控制是單向的，VoIP 在 Application 層將語音資訊送給位於傳輸層的 DCCP，但 DCCP 無法向上傳任何控制訊號，讓 VoIP 調節 Codec 的 Bit-rate。

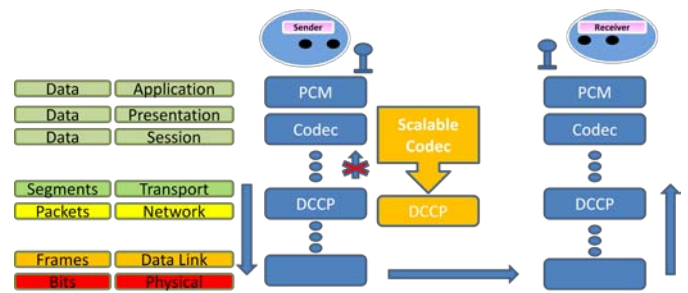


圖 3 VoIP 使用 Scalable Codec 的優點

我們發現網路電話採用 DCCP 作為傳輸層協定來達成壅塞控制功能時，最好搭配 Flexible Bit-rate 來控制資料傳輸率，並維持應有的通話品質，不過因為語音編碼與傳輸協定位於網路七層中不同的階層，且階層之間的傳輸是單向的，而造成 DCCP 知道網路壅塞狀況卻無法告知編碼器做配合的窘境。本研究提出的 Scalable Codec 配合 DCCP 傳輸協定的方法，必須考慮到語音編碼器配合傳輸協定的困難，將語音編碼器的所有層次語音編碼位元都編碼並傳給傳輸層，讓傳輸層依照網路狀況選擇適合網路狀況層次的封包之後再做傳送。如此，可以不需要 DCCP 與編碼器之間的互動。

二、Scalable Codec over DCCP

2.1 需求分析與設計目標

如圖 5 中所示，當網路壅塞發生時，可能造成封包延遲時間拉長或是封包遺失的狀況，封包延遲時間拉長將會造成電話接收端延後聽到語音，或是因逾時而捨棄封包，而封包遺失會造成通話斷斷續續，並且通話內容部分遺失。由此可知減緩網路壅塞的狀況對於提升語音品質有很大的幫助。

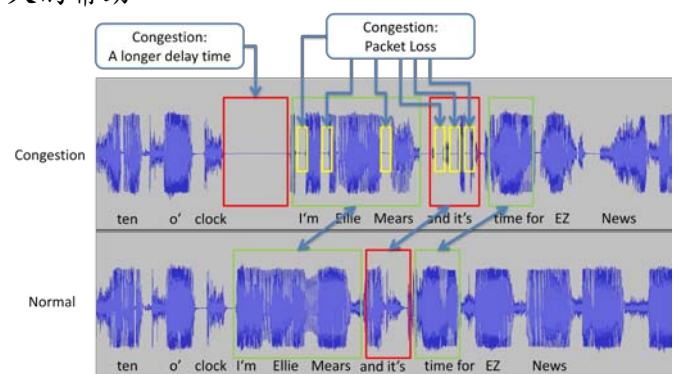


圖 4 網路壅塞時造成造成封包延遲時加拉長或是封包遺失

我們觀察在頻寬瓶頸為 128Kbps 的網路連線中，傳送 10 通網路電話，通話長度約為 15 秒，並分別以不同 Bit-rate 傳送語音，觀察通話品質指標 MOS 的變化[5]。如表 1 所示，我們可以看出，在網路頻寬不足時，採用較低 Bit-rate 的語音品質會比採用高 Bit-rate 的語音品質佳。

表 1 網路頻寬不足時採用不同 Bit-rate 的語音 MOS 變化。

Sender	Receiver	Bit-rate(Kbps)	Packet Loss Rate	品質(MOS)
451	288	42.2	0.3614	1.475
451	385	27.8	0.1463	2.233
451	428	12.8	0.0509	2.79
451	451	5.75	0	3.289

2.2 Scalable Codec over DCCP 基本概念

我們提出多層編碼搭配 DCCP 作為傳輸協定，在通話中多層編碼將所有層次的語音編碼都一併傳送給 DCCP，DCCP 再依照網路的狀況傳送合適速率的語音封包，以保持網路電話的語音品質。DCCP 利用網路的 RTT 以及封包遺失率偵測網路壅塞的情況，當網路壅塞時啟動壅塞控制機制，依照網路狀況選擇較低的語音層次，縮小封包，可以舒緩頻寬，等到網路回復順暢時再回復較高的語音層次，提升通話品質。

2.3 Scalable Codec 設計

我們選定具有多種 Bit-rate 的 Speex 做為 Scalable Codec 的設計原型[10]。為了讓傳送端能依據所感受到的網路壅塞程度，選擇較適合該程度的壅塞控制反應，必須有足夠多的層次供給壅塞控制機制做選擇。我們以 Speex 的 wideband mode 為雛形做修改，結合 narrowband mode 並去除重複的層次，我們將語音層次的範圍訂為 0 到 Max_level，目前語音編碼器的實做細分為 18 個層次。

表 2 Scalable Codec 0~Base_rate

Quality	Bit-rate(Kbps)	Packet Size(Payload only)
0	2.15	6 Byte
1	3.95	10 Byte
2	5.75	15 Byte
3	5.95	15 Byte
4	7.75	20 Byte
5	8	20 Byte
6	9.8	25 Byte
7	11	28 Byte
8	12.8	32 Byte
9	15	38 Byte

此外，一般網路電話常用的 iLBC 的位元率為 13.3Kbps 以及 15Kbps，因此，我們規劃以

15Kbps 的層次 9 的語音編碼封包為基準，並稱呼該層次為 Base_rate，將 18 個層次分為 0~9 的低位元率層次，10~17 的高位元率層次[1]。當偵測到壅塞發生時，立刻改選擇 Base_rate 的語音編碼封包，並且持續偵測網路狀況，如壅塞程度又加劇，則還可持續遞減選擇 Base_rate 層次以下的語音編碼。

表 3 Scalable Codec 10~Max_level

Quality	Bit-rate(Kbps)	Packet Size (Payload only)
10	16.8	42 Byte
11	18.2	46 Byte
12	20.6	52 Byte
13	23.8	60 Byte
14	24.6	62 Byte
15	27.8	70 Byte
16	34.2	86 Byte
17	42.2	106 Byte

三、演算法設計

3.1 壅塞偵測

DCCP 可以選擇的壅塞控制機制中的 CCID3 被建議用來傳送多媒體網路串流封包[3]，透過 RTT 與封包遺失率等參數計算傳輸速率。我們在網路瓶頸為 128Kbps 的網路中觀察壅塞發生時 RTT(封包往返時間)的變化，在網路中每 10 秒增加 1 通電話，直到網路中同時存在 10 通電話後每隔 10 秒結束 1 通電話。隨著通話數的增加，頻寬逐漸不足，網路壅塞也隨之發生。如圖 8、圖 9，可以看出隨著通話數的增加，網路發生壅塞的狀況，RTT 以及封包遺失率也隨之增長，而當網路壅塞狀況舒緩，網路恢復正常時 RTT 以及封包遺失率也隨之減少，可以看出封包傳送到接收端時，RTT 應該在一個固定的時間範圍內；當頻寬不足時，RTT 會拉長，偵測到此現象時可視為網路壅塞。我們假設 RTT 的時間為 t_i ，且服從常態分配，其值以 t_1, t_2, \dots, t_N 表示，平均值為 $\bar{t}(1)$ ，標準差為 $\sigma_t(2)$ 。

$$\bar{t} = \frac{1}{N} \sum_{i=1}^N t_i \quad (1)$$

$$\sigma_t = \sqrt{\frac{1}{N-1} \sum_{i=1}^n (t_i - \bar{t})^2} \quad (2)$$

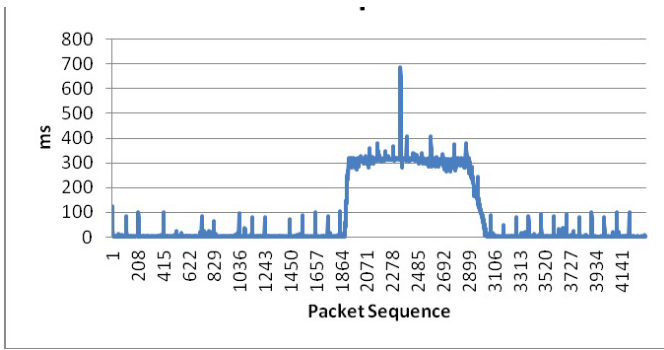


圖 5 網路壅塞時封包來回時間的變化

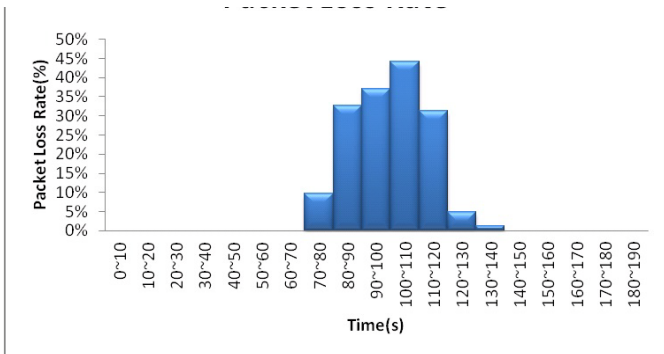


圖 6 網路壅塞時封包遺失率的變化

3.2 High Delay Response 演算法

利用常態分配之經驗法則我們可以推測，如果 RTT 大於 $[\bar{t} + 3\sigma_t]$ 則判斷為網路壅塞，我們訂定調降 Bit-rate 的門檻值 **HDthresh** (High Delay Threshold)，如果連續 **HDthresh** 個封包之 RTT 大於 $[\bar{t} + 3\sigma_t]$ ，且目前的語音編碼層次高於 Base_rate，則將層次立即降低至 Base_rate，如果目前的語音編碼層次低於 Base_rate 且不低於最低層次，則只須降低一個層次，演算法如圖 10 所示。

3.3 High Loss Response 演算法

TFRC 利用 RTT 與封包遺失率可以立即得知網路狀況，迅速調降封包傳輸率，因此傳輸速率會下降的比其他傳輸協定快速。而 TCP 傳輸協定則是要等到 Timeout 才降低速率。因此使用 TFRC 的 DCCP 無法取得合理的頻寬。由以上的描述我們可以知道 DCCP 在得知壅塞發生時的快速反應並不適切。因為 DCCP 較快降低傳送速率，網路狀況得到舒緩，但 TCP 因此沒有感受到網路的壅塞因而持續增加 CWND，會因此造成 DCCP 持續退讓而其他協定沒有感受到的後果。我們設想當網路壅塞狀況發生時，DCCP 可以得知網路狀況，但是持續保持比適切速率較高

一些，就可以維持網路擁塞的訊號，而其他傳輸協定偵測到網路壅塞的狀況後也會因此做調整。並且雖然網路保持輕微的壅塞狀況，但對於可以忍受少量封包遺失的網路應用程式(例如網路電話)而言並不影響。

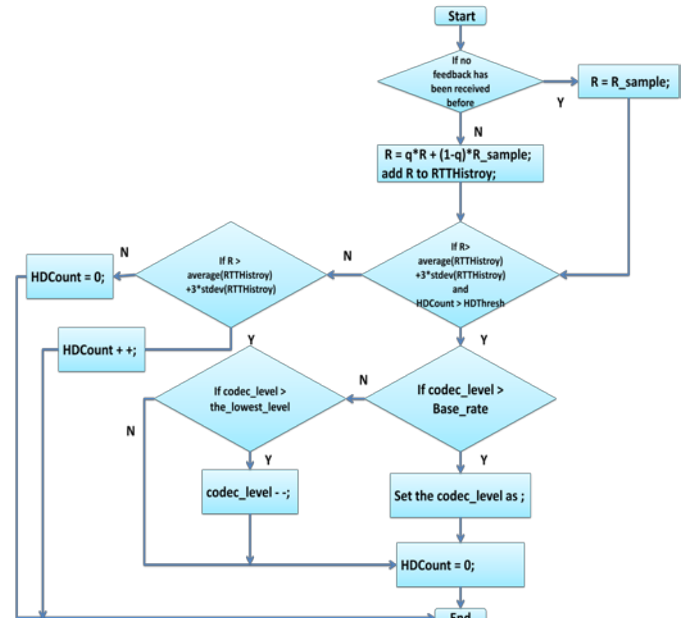


圖 7 High Delay Response 演算法 流程圖

由以上論述，我們設定 **HLthresh**(High Loss Threshold)，**HLthresh** 必須依照應用程式類型所能忍受的封包遺失率訂定。當封包遺失率小於 **HLthresh** 表示尚能忍受少量封包遺失的壅塞程度，並不作壅塞控制反應，讓網路持續保持小壅塞訊息，讓 TCP 可以感受到網路壅塞的狀況並降緩其傳送封包的數量。當封包遺失率大於 **HLthresh** 時表示，壅塞程度已經讓封包遺失量到達已經不能忍受的程度了，將採取壅塞控制反應，演算法如圖 11 所示。

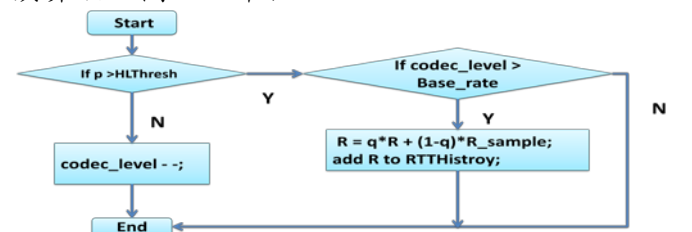


圖 8 High Loss Response 演算法 流程圖

3.4 Low Delay Response 演算法

利用常態分配之經驗法則我們可以推測，如果 RTT 小於 $[\bar{t} - 3\sigma_t]$ 則判斷為網路壅塞狀況舒緩，並且我們訂定提升 Bit-rate 門檻值 **LDthresh**

(Low Delay Threshold), 如果連續 LDthresh 個封包之 RTT 小於 LDthresh, 則將 Bit-rate 提升一個層次, 演算法如圖 12 所示。

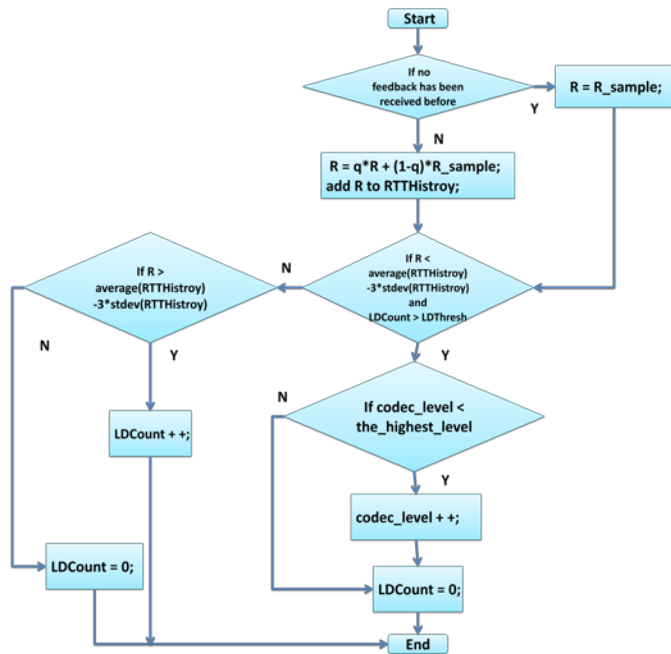


圖 9 Low Delay Response 演算法流程图

四、實驗與效能評估

4.1 評估指標

MOS(Mean Opinion Score)是直接利用人耳對於語音的感覺評分, 分為 Excellent(5)、Good(4)、Fair(3)、Poor(2)及 Bad (1)等五個等級[5]。但有限的人力難以進行大量的評估工作, 故大部分採用 E-model, 以客觀的方式來評估網路電話品質, 語音傳輸過程中可能有許多原因會造成語音品質降低, 如 Packet Delay、Packet Loss 和經編碼器(Codec)壓縮後的失真程度。E-Model 透過公式(3)可以客觀的計算出該語音的品質分數, 而 E-Model 計算出的 R-factor 可再透過公式(4)轉換成 MOS。

$$R = R_o - I_s - I_d - I_e + A \quad (3)$$

$$MOS = 1 + 0.035 * R + 7 * 10^6 * R * (R - 60) / (100 - R) \quad (4)$$

4.2 實驗一

在頻寬瓶頸為 2Mbps 的網路情境下, 分別以 CBR over UDP、Flexible Bit-Rate 以及 Scalable Codec 三種方式傳輸 VoIP。每次的實驗方式為每隔 10 秒增加 10 通 VoIP 語音通話進入網路, 直到網路中有著 100 通 VoIP 同時進行傳輸為止, 之後每隔 10 秒鐘減少 10 通電話, 直到所有通話

結束。實驗時間共 190 秒。

4.2.1 三種傳輸方式實驗結果與分析

圖 12、圖 13 顯示了 CBR over UDP、Flexible Bit-Rate 以及 Scalable Codec 三種方式傳輸 VoIP 的效能比較, 分別為封包遺失率、通話品質(MOS)。使用 Flexible Bit-rate 以及 Scalable Codec 來傳輸 VoIP 通話時, 可以看出相較於 CBR 搭配 UDP, 他們都可以擁有比較好的 MOS 以及較低的封包遺失率。不過在比較壅塞狀況時, Scalable Codec 因為利用 DCCP 偵測網路狀況的 RTT 以及封包遺失率來做網路偵測, 而且有較細緻的語音層次, 所以可以馬上得知網路的狀況, 並且做最適當的反應, 因此封包遺失率都來的比 Flexible Bit-rate 低, 而且對於控制整體網路狀況的穩定性也比 Flexible Bit-rate 來的好。

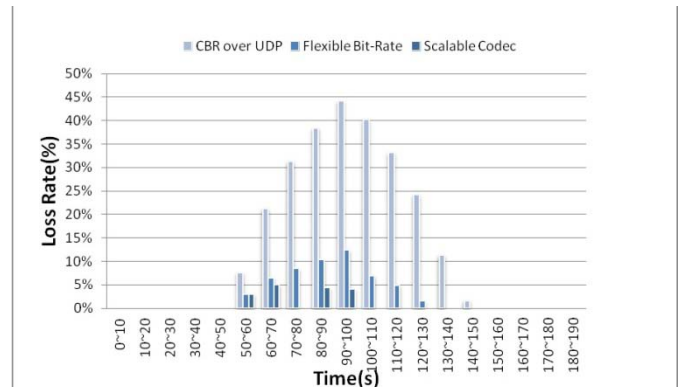


圖 10 三種方式傳輸 VoIP 的封包遺失率比較

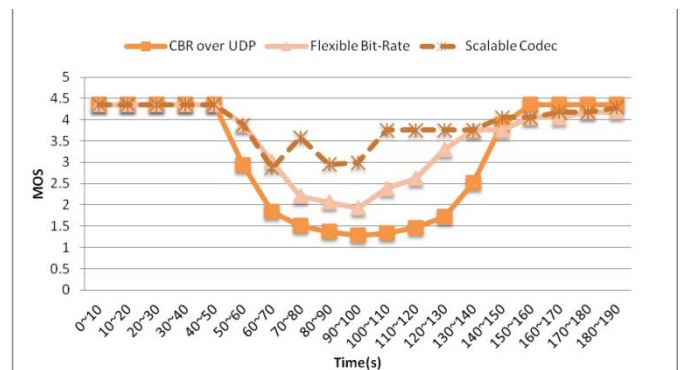


圖 11 三種方式傳輸 VoIP 的 MOS 比較

4.3 實驗二

在頻寬瓶頸為 256Kbps 的網路情境下, 分別以 CBR over UDP 以及 Scalable Codec 在頻寬不足時和 TCP 同時存在於網路中, 比較對於頻寬的競爭能力的表現。我們分別觀察 VoIP 先進入網路以及 TCP 先進入網路的頻寬競爭的狀況。

VoIP 先進入網路實驗方式是在開始時傳送 5 通 VoIP，在實驗第 10 秒時，在網路中再加入 5 條 TCP 傳送檔案資料流，在實驗第 90 秒時，結束所有 TCP 資料流，並在實驗第 100 秒時結束實驗。TCP 先進入網路的狀況則實驗方法類似，不同處是將 VoIP 以及 TCP 進入的時間對調。

4.3.1 頻寬競爭能力實驗結果與分析

圖 15 中可看出 VoIP 先進入情景下改為 Scalable Codec VoIP 的狀況下，頻寬競爭評比指標 Jain's Index 幾乎都高於使用 CBR over UDP VoIP 的狀況，大約高於 12% 左右，可知擁有較公平的 Throughput 分配比例。在 TCP 先進入的狀況下，圖 16 中也可看出使用 Scalable Codec VoIP 的狀況 Jain's Index 也有 15% 左右的改善，由此可表示，此研究提出的方法可以在頻寬不足的情況下友善且公平的競爭頻寬。

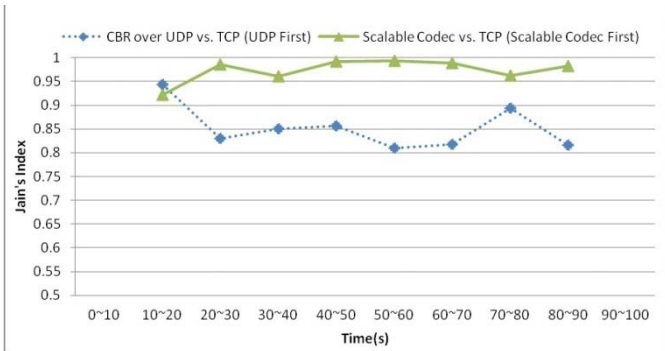


圖 12 頻寬競爭評比指標 Jain's Index 變化 - CBR over UDP 或 Scalable Codec VoIP 先進入

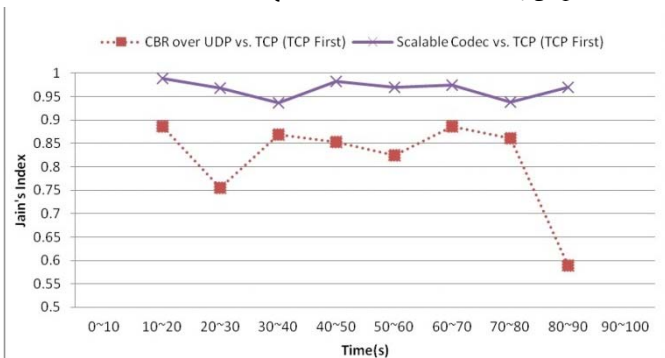


圖 13 頻寬競爭評比指標 Jain's Index 變化 - TCP 先進入

五、結論與未來研究

此研究提出的 TCP-friendly VoIP By Scalable Codec over DCCP，是利用 DCCP 偵測網路狀況的封包來回時間以及封包遺失率來判斷網路壅塞的程度，並設計出細緻的語音層次，讓網路的壅塞時可以依照壅塞程度有更適切的反應。本研

究透過實際網路的實驗環境中評估以 CBR over UDP、Flexible Bit-Rate 以及 Scalable Codec 三種方式傳輸網路電話封包的效能，結果顯示透過 Scalable Codec 方法，能有效降低網路電話的封包遺失率，維持通話品質，而且相較於 Flexible Bit-Rate 可以有更準確的壅塞偵測以及更合適的壅塞控制反應。研究結果顯示在網路壅塞最嚴重的情況下可以和一般的 CBR 搭配 UDP 的封包遺失率達到 40% 左右的改善，與 Flexible Bit-Rate 方法也的達到 8% 以上的改善，可以有較佳的語音品質，並且整體網路狀況更為穩定。而在和 TCP 頻寬競爭實驗中可看出，在頻寬不足的情況下，此研究提出的方法頻寬競爭評比指標 Jain's Index 也有 15% 左右的改善，顯示可以和 TCP 友善且公平的競爭頻寬。本研究中僅粗略的讓語音編碼配合 DCCP，由於 DCCP 尚未常見於一般傳輸層協定，許多 DCCP 配合僅依照[3]中所定義的壅塞偵測方法做實做，未來將針對語音編碼器與 DCCP 做更細微的配合。

六、參考文獻

- [1] S. Andersen and A. Duric, "Internet Low Bit Rate Codec (iLBC)," IETF RFC 3951, Dec. 2004.
- [2] S. Floyd and E. Kohler, "Profile for DCCP Congestion Control ID 2: TCP-like Congestion Control," IETF draft-ietf-dccp-ccid2-08, <http://www.ietf.org/internet-drafts/draft-ietf-dccp-ccid2-10.txt>, Mar. 2005.
- [3] S. Floyd, E. Kohler, and J. Padhye, "Profile for DCCP Congestion Control ID 3: TFRC Congestion Control," IETF draft-ietf-dccp-ccid3-11, <http://www.ietf.org/internetdrafts/draft-ietf-dccp-ccid3-11.txt>, Mar. 2005.
- [4] M. Handley, "TCP Friendly Rate Control (TFRC): Protocol Specification," IETF RFC 3448, Jan. 2003.
- [5] ITU-T Rec. P.800, "Methods for Subjective Determination of Transmission Quality," Aug. 1996.
- [6] ITU-T Rec. G.107, "The E-Model, A Computational Model for Use in Transmission Planning," May 2000.
- [7] E. Kohler, M. Handley and S. Floyd, "Datagram Congestion Control Protocol (DCCP)," IETF RFC 4340, Mar. 2006.
- [8] Eddie Kohler, Mark Handley, and Sally Floyd, "Designing DCCP: Congestion Control Without Reliability," SIGCOMM 06, Sep. 2006, Pisa, Italy, pp. 27-38.
- [9] Y.N. Lien, Y.C. Ding, "Congestion Control Enabled VoIP by Flexible Bit-rate" M.A. thesis, University of Chengchi, Taiwan, 2010
- [10] Xiph.org Foundation, "The Speex Codec Manual Version 1.2 Beta 3", 2007.