

國立交通大學

網路工程研究所

碩士論文

位元率-失真最佳化網路視訊串流技術—H.264
可調視訊編碼於實時傳輸協定進行接收端驅動
重傳

Rate-Distortion Optimization of H.264 SVC Internet
Streaming via RTP with Receiver-driven Retransmission

研究生：胡嘉錡

指導教授：邵家健 博士

中華民國九十八年十月

位元率-失真最佳化網路視訊串流技術—H.264 可調視訊編碼於實時

傳輸協定進行接收端驅動重傳

Rate-Distortion Optimization of H.264 SVC Internet Streaming via RTP
with Receiver-driven Retransmission

研 究 生：胡嘉錡

Student : Chia-Chi Hu

指 導 教 授：邵家健 博士

Advisor : Dr. John Kar-Kin Zao

國 立 交 通 大 學

網 路 工 程 研 究 所

碩 士 論 文

A Thesis

Submitted to Institute of Network Engineering

College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of Master in Computer Science

October 2009

Hsinchu, Taiwan, Republic of China

中華民國九十八年十月

位元率-失真最佳化網路視訊串流技術—H.264 可調 視訊編碼於實時傳輸協定進行接收端驅動重傳

學生：胡嘉錡

指導教授：邵家健 博士

國立交通大學網路工程研究所 碩士班

摘要

欲達到位元率-失真最佳化的視訊傳輸，一套可靠的估測模型將扮演重要角色。此研究目的為建立一套位元率與失真的估測模型，在特定的傳輸機制下，給予封包遺失率與傳輸策略，即可計算出視訊在傳輸後的位元率與失真之期望值。此估測模型考量了在有線網際網路上以實時傳輸協定傳輸視訊時，實際上所需考慮的細節，適用的傳輸機制包含接收端驅動重傳以及接收端積極重傳，且應用於傳輸可調視訊編碼。

研究結果發現，當封包遺失率愈大，不同傳輸策略所造成的差異愈明顯。且最多重傳次數愈大時，失真與位元率的變化愈趨於平緩。

透過此模型，我們提出了最佳化傳輸策略的演算法，給予有限的頻寬之下找出最佳傳輸策略，使得傳輸後的視訊具有最小的失真期望值。

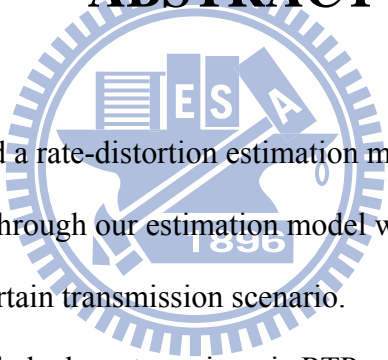
Rate-Distortion Optimization of H.264 SVC Internet Streaming via RTP with Receiver-driven Retransmission

Student: Chia-Chi Hu

Advisor: Dr. John Kar-Kin Zao

Institute of Network Engineering
National Chiao Tung University

ABSTRACT



In this thesis, we aim to build a rate-distortion estimation model. The expected rate and distortion can be calculated through our estimation model with a given packet loss rate and a transmission policy under certain transmission scenario.

Many details need to be tackled when streaming via RTP over the wired Internet are taken into account in our estimation model. This model is build for SVC video streaming under receiver-driven retransmission and receiver-driven aggressive retransmission.

The results show that the difference of RD points between different transmission policies become larger when the packet loss rate increase. As the maximum retransmission times increase, the improvement of distortion becomes less obvious.

We also propose transmission policy optimization algorithms. For a given bit rate, we can find the optimal transmission policy with minimum expected distortion through our optimization algorithms.

致 謝

感謝邵家健老師兩年多以來對我的指導與教誨，讓我在學術研究上能有所進步。老師花了很多時間和我討論，指引我研究方向並修正我的步伐，最後才能將此論文完成。於待人處事上，我更從老師身上如何與人相處，如何把事做對且做好。這些獲得都是彌足珍貴的。

感謝口試委員林嘉文教授和彭文孝教授撥冗來參加我的論文口試，並且給予相當實在的建議，使得此研究能夠有在未來延續時能針對這些缺失的部分做補強，讓整個研究更加完整。

感謝時常給予指導的彭文孝老師，在我研究上碰到困難時，彭老師的建議總能讓研究更加順利。一起開過會的雪婷學姐、哲永、澤瑋與閔旭，從你們身上我也學了許多有關視訊壓縮的知識，謝謝你們。感謝幫助我壓製視訊的學弟們，皓偉、李翰、柏崴以及力行，因為有你們，這篇論文才能順利完成。

此外，我想感謝實驗室一同相處過的夥伴，國晉、哲民、輔國和芳伯學長，感謝你們，不管是學業上的討論或是待人處事的道理，你們的建議都很有幫助。感謝一同努力六年的星閔，從大學到研究所，從十舍到九舍到六舍，和你相處學習到了不少東西。感謝勝焜、梅瑛、博政和彥霖，兩年多的研究生生活有了你們變得更精彩了。還有明昌學長、Martin、子晉、志明、鈞凱、俊瑋、嘉妤，謝謝你們對我在研究上的幫助，也是我平常在實驗室聊天的好夥伴。

最後，我要謝謝我的家人，在我求學的路上支持著我，我才能有今日的成就。還有謝謝我的女朋友，佳欣，也是陪我經歷了許多的歲月，願意和我一同分享快樂，一同分擔哀愁，互相扶持，這段時間，感謝妳給我的鼓勵與支持。

要感謝的人太多了，真的謝謝你(妳)們對我的支持與鼓勵，謝謝！

目 錄

摘要	i
ABSTRACT	ii
致 謝	iii
目 錄	iv
圖目錄	vi
表目錄	viii
第 1 章 概論	1
1.1 問題陳述	1
1.2 研究方法	2
1.3 論文大綱	3
第 2 章 背景知識	4
2.1 可調視訊編碼(Scalable Video Coding, SVC)	4
2.2 RTP封包封裝SVC視訊方式	4
2.3 位元率-失真最佳化視訊傳輸	6
第 3 章 研究方法	7
3.1 錯誤-花費基本模型	7
3.2 傳送單一多媒體資料單元	8
3.2.1 接收端驅動重傳	8
3.2.1.1 錯誤期望值	9
3.2.1.2 花費期望值	12
3.2.2 接收端驅動積極重傳	13
3.2.2.1 錯誤機率期望值	15
3.2.2.2 花費期望值	15
3.3 傳送多媒體資料單元群組	19
3.3.1 失真期望值	19
3.3.2 頻寬花費期望值	21
3.3.3 多個多媒體資料單元放在同一個封包	22
3.3.4 一個多媒體資料單元中部份的畫面解開所降低的失真	22

3.4	傳輸策略最佳化	24
3.4.1	傳輸策略最佳化方法 1	24
3.4.2	傳輸策略最佳化方法 2	26
3.4.3	強迫相依視訊層重傳	26
3.5	相關研究	27
3.5.1	基本模型	27
3.5.2	傳送單一多媒體資料單元	28
3.5.3	傳送多媒體資料單元群組	28
3.5.4	傳輸策略最佳化	29
第 4 章	估測結果與分析	30
4.1	SVC 視訊格式設定	30
4.2	失真與頻寬花費期望值估測	32
4.3	傳輸策略最佳化	38
4.3.1	傳輸策略最佳化演算法 1	38
4.3.2	傳輸策略最佳化演算法 2	39
4.3.3	傳輸策略最佳化結果與分析	41
第 5 章	結語	44
5.1	研究成果	44
5.2	未來方向	44
參考文獻	45
附錄	46

圖目錄

圖 1: RTP封包的裝載量	5
圖 2: 接收端驅動重傳之錯誤事件示意圖	9
圖 3: 接收端驅動重傳下傳送一個封包之錯誤事件示意圖	9
圖 4: 接收端驅動重傳下傳送二個封包之錯誤事件示意圖	10
圖 5: 接收端驅動重傳下傳送三個封包之錯誤事件示意圖	11
圖 6: 接收端驅動重傳下傳送一個封包之花費示意圖	12
圖 7: 接收端驅動重傳下傳送二個封包之花費示意圖	12
圖 8: 接收端驅動積極重傳之錯誤事件示意圖	14
圖 9: 接收端驅動積極重傳下傳送一個封包之花費示意圖	16
圖 10: 接收端驅動積極重傳下傳送一個封包之花費簡化示意圖	16
圖 11: 接收端驅動積極重傳下傳送二個封包之花費示意圖	17
圖 12: 目標視訊層為 $S(1,1)$ 時出現多個可解視訊層表示格式的情況	20
圖 13: 目標視訊層為 $S(1,2)$ 時出現多個可解視訊層表示格式的情況	20
圖 14: SVC視訊畫面之間的相依關係	23
圖 15: 最佳化過程可選擇增加最大重傳次數的視訊層	25
圖 16: 多媒體資料單元群組織傳輸策略示意圖	25
圖 17: SVC視訊相依關係與QP設定	30
圖 18: 未經傳輸的原始RD柵籬圖	31
圖 19: 不同傳輸策略於封包遺失率為百分之二十的RD柵籬圖	32
圖 20: 封包遺失率為百分之二十，RD點於不同最多重傳次數的變化	33
圖 21: 傳輸策略一在不同封包遺失率下的RD柵籬圖	35
圖 22: 封包遺失率為百分之二十，四種不同傳輸策略的RD柵籬圖	36
圖 23: 封包遺失率為百分之三十，四種不同傳輸策略的RD柵籬圖	36
圖 24: 封包遺失率為百分之四十，四種不同傳輸策略的RD柵籬圖	37

圖 25: 封包遺失率為百分之五十，四種不同傳輸策略的RD柵籬圖 37

圖 26: 不同位元率限制下，最佳化演算法 2 找出的最佳傳輸策略對應的RD曲線 41

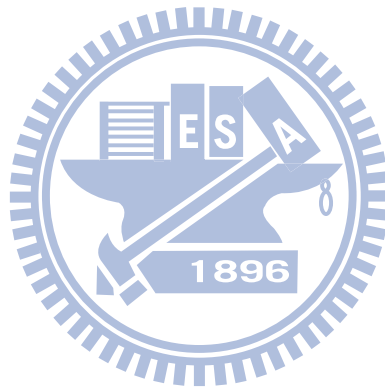
圖 27: 目標視訊層為(4,3)時不同最佳化演算法演化過程中的部分RD曲線..... 42

圖 28: 不同位元率限制下，不同最佳化演算法找出的最佳傳輸策略對應的部分RD曲線
..... 43



表目錄

表格 1: 傳輸策略最佳化方法 1 的最佳化過程.....	25
表格 2: 傳輸策略最佳化方法 2 的最佳化過程.....	26
表格 3: Foreman在CIF尺寸螢幕每秒播放三十張的失真MSE與位元率.....	31
表格 4: 最高重傳次數為三次的四種傳輸策略.....	34
表格 5: 傳輸策略最佳化演算法 1.....	39
表格 6: 傳輸策略最佳化演算法 2.....	40



第 1 章 概論

1.1 問題陳述

在網路上進行視訊串流時，如何在有限的頻寬下，讓使用者播放視訊時能有最佳的畫面品質，即達到位元率-失真最佳化傳輸，一直以來為視訊串流系統中相當重要的議題。Chou和Miao將視訊傳輸中估測失真與位元率的問題轉化成估測多媒體資料單元的錯誤率期望值與其頻寬花費的增長倍數，原來要考慮的“失真-位元率”(distortion-rate)估測變成了“錯誤-花費”(error-cost)估測，並藉由該估測模型來研究如何達到位元率-失真最佳化傳輸[1]。然而實際上在使用RTP傳輸SVC視訊時，仍然有些細節必須考慮進去，像是如何將H.264 視訊封裝至RTP封包中與SVC視訊的畫面解碼方式，這些細節其實對估測模型的建立有很大的影響。除了針對視訊在傳輸過程中做位元率-失真最佳化，考慮如何適當地編碼與解碼，對於位元率-失真最佳化扮演著相當重要的角色[2]。經過位元率-失真最佳化編碼的SVC視訊，在網路上傳輸後，各視訊層的位元率與失真之變化情形亦是我們感興趣的。

此研究旨在建立一套可應用於網際網路上RTP 視訊串流的頻寬花費與失真之估測模型，此模型適用於接收端驅動重傳與接收端驅動積極重傳的傳輸機制下傳輸 SVC 視訊。只要給予封包遺失率、傳輸機制與傳輸策略，便能計算出視訊經過網路傳輸後的頻寬花費與失真的期望值。更進一步，藉此估測模型來建立傳輸策略最佳化演算法，找出不同頻寬下最佳的傳輸策略，達到位元率-失真最佳化的視訊傳輸。

1.2 研究方法

為了達到位元率-失真最佳化的視訊傳輸，首先建立頻寬花費與失真的估測模型。接著下來才以估測模型和傳輸策略最佳化演算法來找出最有效率的傳輸策略。

在建立頻寬花費與失真的估測模型時，我們步驟包含：

- 推導傳送單一多媒體資料單元時，在給定的封包遺失率、傳輸機制與傳輸策略下，經過傳輸後的花費期望值與錯誤期望值。
- 推導傳送含有多個多媒體資料單元的群組時，在給定的封包遺失率、傳輸機制與傳輸策略下，經過傳輸後整體頻寬花費期望值與失真期望值。
- 考量使用 RTP 來做視訊串流時，對 H.264 視訊所規範的封包封裝方式，以及 SVC 視訊的相依關係影響其解碼方式。

找出最佳傳輸策略的步驟有：

- 建立傳輸策略最佳化的演算法，透過頻寬花費與失真的估測模型，以改善最多失真而頻寬耗費最少的規則，找出最佳的傳輸策略。
- 延伸上述的演算法，在進行最佳化時多考慮一個準則，將 SVC 視訊各視訊層彼此之間的相依關係及重要性也加入考量，找出最佳的傳輸策略。

1.3 論文大綱

在接下來的章節中，首先第二章將會介紹 SVC 的特性、RTP 封包封裝方式以及 Chou 在位元率-失真最佳化傳輸研究中所提出的概念。第三章描述我們如何推導出估測花費與失真的模型，以及如何找到最佳傳輸策略。第四章展示 SVC 視訊透過我們建立的估測模型計算，其傳輸後的花費與失真，還有以傳輸策略最佳化演算法，找出在不同的頻寬限制下最佳的傳輸策略。第五章說明此研究的貢獻與未來可延伸的研究方向。



第 2 章 背景知識

2.1 可調視訊編碼(Scalable Video Coding, SVC)

SVC為H.264/AVC編碼標準的延伸，一個SVC視訊由多層視訊層所組成，包含一層基本層(base layer)與多層增強層(enhancement layer)，視訊層(scalable layer)之間具有相依關係(dependency) [3]。增強層參照至基本層，只有在基本層成功解碼的情況下，增強層才能解開。

SVC 具有時間(temporal)、空間(spatial)與雜訊比(SNR)三種可調性。在這裡時間指的是每秒播放張數，而空間代表視訊的尺寸大小，雜訊比則是代表失真量的多寡。可調視訊編碼可以這三種可調性來適應不同的網路情況與使用者裝置，依照需求調整每秒播放張數、視訊的尺寸與雜訊比高低符合網路的頻寬或是使用者裝置的螢幕尺寸與計算能力等。因為 SVC 包含許多視訊層，使用者裝置可依自己需求來接收特定數層視訊層來解碼，故一個 SVC 視訊能滿足多種不同使用者裝置的需要。SVC 視訊會封裝成網路抽取層單位(Network Abstraction Layer unit, NAL unit)以利於網路上傳輸，使用者裝置依照所收到的 NAL unit 來進行解碼。

SVC 視訊多層次的結構，使其在頻寬變化的情況下傳輸仍能依照使用者的網路情形提供最佳化的視訊傳輸品質，然而，多層視訊層的相依關係也使得傳輸上變得較為複雜。因此，傳輸 SVC 視訊欲達到位元率-失真最佳化比傳輸單層視訊要來得困難。

2.2 RTP封包封裝SVC視訊方式

在網路上以RTP傳輸H.264 視訊時，必須考慮視訊是如何封裝至RTP封包當中的。將SVC 視訊封裝至RTP封包的封裝方式仍在制定中[4]，與封裝H.264/AVC視訊稍有不同，但主要仍遵照封裝H.264 視訊的規範，亦即RFC3984 [5]。SVC視訊會先封裝成NAL unit的格式，然後才裝至封包當中。在SVC的封裝方式規範中，將NAL unit封裝至RTP封包的方

式有封裝單一NAL unit、封裝多個NAL unit以及封裝NAL unit的片段(fragment)。將多個位元組大小較小的NAL unit共同封裝至一個封包內，可以節省封包頭的頻寬花費。且同一封包的多個NAL unit的播放時間可以不一樣。當一個RTP封包不足以封裝一個過大的NAL unit，此時就會將該NAL unit分成多個片段，分別封裝於多個封包內。須注意的是，雖然一個封包可以封裝多個NAL unit，但是不能放入多個NAL unit的片段。

一個封包雖然可裝載多個NAL unit，但在網際網路上傳輸時，該封包的位元組大小仍要有限制，使得整個IP封包的容量不超過網路的最大傳輸單位(Maximum Transmission Unit)，如此才不會造成IP封包過大而分成多個片段。考量此限制，網際網路的MTU為1500 個位元組，如圖 1所示，一個RTP封包所能裝載的NAL unit的大小為 1460 個位元組。

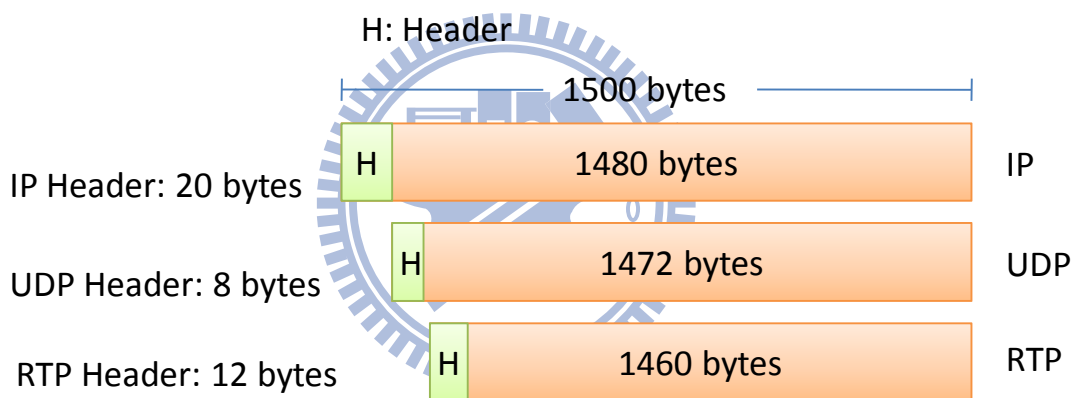


圖 1: RTP 封包的裝載量

2.3 位元率-失真最佳化視訊傳輸

在網路上進行視訊串流時，如何在有限的頻寬下，讓使用者播放視訊時能有最佳的畫面品質，即達到位元率-失真最佳化傳輸，一直以來為視訊串流系統中相當重要的議題。

Chou和Miao將視訊傳輸中估測失真與位元率的問題轉化成估測多媒體資料單元的錯誤率期望值與其頻寬花費的增長倍數，原來要考慮的“失真-位元率”(distortion-rate)估測變成了“錯誤-花費”(error-cost)估測，並藉由該估測模型來研究如何達到位元率-失真最佳化傳輸。

Chou和Miao將位元率-失真最佳化視訊傳輸此議題簡化為對單一多媒體資料單元傳輸做錯誤-花費最佳化，此概念使得失真的計算得以量化[1]。並且，針對多種傳輸機制建立錯誤-花費模型，包括單一QoS (Quality of Service)、多種QoS、FEC (Forward Error Correction)、傳送端驅動重傳與接收端驅動重傳等，依此推算在不同的傳輸策略下頻寬花費與視訊失真的期望值。



第 3 章 研究方法

3.1 錯誤-花費基本模型

視訊經過傳輸後的頻寬花費與失真可藉由封包遺失率、傳輸機制與傳輸策略來加以估測。當封包在傳輸途中遺失，造成部分的視訊無法完整地解碼，失真值 MSE 便會上升。而針對遺失的封包再次重傳，如果最多可重傳次數愈大，該封包成功傳送至接收端的機率也愈大，代表失真值 MSE 會降低，同時頻寬花費也會變大。

在視訊串流系統中，壓縮後的多媒體資料被封裝成多媒體資料單元後才放入封包中於網路上傳輸。我們定義一個多媒體資料單元 $\mathfrak{N}_G(L, T)$ 為 SVC 的某一視訊層 $\mathfrak{N}(L, T)$ 於第 G 個 GOP 的所有畫面，每個多媒體資料單元所包含的畫面數可能不同，且假設一個多媒體資料單元中每張畫面的位元組大小是相同的，解得各畫面能降低的失真也相同。當傳輸的視訊格式為 SVC 時，各個視訊層之間的相依關係也會反應在封裝成多媒體資料單元後彼此的相依關係上。多媒體資料單元彼此間的相依關係可以表示成有向無循環圖 (directed acyclic graph)。

各個多媒體資料單元同時也儲存了各自的位元組大小 B_i 、解碼後所能改善的失真 Δd_i 以及解碼的最後期限 T_{DTS} 。各個多媒體資料單元的位元組大小不一定相同，其解碼後所能降低的失真也不一樣，通常來說解開基本層的多媒體資料單元降低的失真會比解開增強層的多媒體資料單元來得多， Δd_i 的值越大，則代表此多媒體資料單元愈重要。隨著播放時間的不同，各個多媒體資料單元解碼的最後期限也不同。在此計算失真與頻寬花費時， $\mathfrak{N}_G(L, T)$ 的位元組大小與改善的失真值是以整個視訊層 $\mathfrak{N}(L, T)$ 中平均一個多媒體資料單元去計算的，因此在任何 GOP 中， $\mathfrak{N}_G(L, T)$ 的位元組大小與改善的失真值皆相同。要完成估算失真與頻寬花費，首先建立不同的傳輸機制在傳送單一多媒體資料單元時的錯誤-花費模型，藉此模型，建立傳送多媒體資料單元群組的失真-頻寬花費模型。

我們假設在網路上各個封包遺失的事件彼此獨立，且封包遺失率短時間內是不變的。從

傳送端至接收端稱之為順向通道(forward channel)，接收端至傳送端為反向通道(backward channel)，以 RTP 進行傳輸時，在順向通道傳遞多媒體資料使用的是 UDP，而負責傳輸雙方控制訊息的 RTCP 也是架構在 UDP 上，故在順向通道與反向通道上的封包皆有一定的封包遺失率，分別為 ϵ_F 與 ϵ_B ，在此我們假設兩封包遺失率相等。

3.2 傳送單一多媒體資料單元

考慮傳送單一個多媒體資料單元時，在該多媒體資料單元解碼的最後期限前，假設會有 N 次的傳送機會，分別為 $t_{0,l}, t_{1,l}, \dots, t_{N-1,l}$ 這些時間點，且兩相鄰傳送機會的時間間隔為一秒鐘，此時間遠大於一個封包的封包往返時間 RTT (Round Trip Time)。因此，我們將多媒體資料單元會發生錯誤的原因簡化為封包遺失，不包括因延誤而沒有及時解碼的錯誤。在此假設下，若是此時間點發出重傳要求，在下個時間點仍未收到，代表該多媒體資料單元已經遺失，必須再次重傳才有機會收到該多媒體資料單元。

3.2.1 接收端驅動重傳

在此我們所考慮的傳輸機制為接收端驅動重傳，表示整個傳輸程序以接收端為主導方，若接收端欲接收某一多媒體資料單元，接收端會發出要求請傳送端將該多媒體資料單元傳給接收端，如果到了下一個傳送時間點接收端仍未收到，那麼接收端會依照傳輸策略來決定該不該再送出重傳要求給傳送端，直到接收端收到該多媒體資料單元或是時間到達該多媒體資料單元的解碼最後期限。另外我們特別假設在 $t_{0,l}$ 時，接收端不需發出重傳要求，傳送端就會自動將預設該傳送的多媒體資料單元傳給接收端，當接收端在下個傳送時間點 $t_{1,l}$ 仍未收到該多媒體資料單元時，接收端才需要發出重傳要求來要求多媒體資料單元。

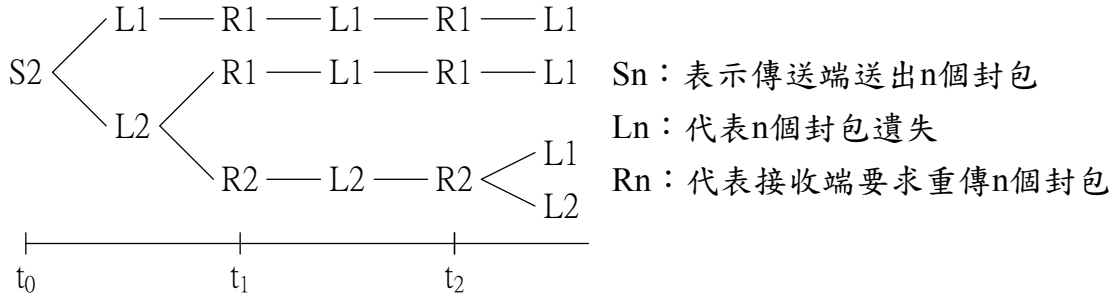


圖 2: 接收端驅動重傳之錯誤事件示意圖

圖 2 為接收端驅動重傳的所有錯誤事件示意圖。一開始在時間 t_0 傳送端傳送兩個封包給接收端，發生錯誤的事件有兩種，分別為遺失一個和兩個封包。傳送 i 個封包而遺失 j 個封包的機率為

$$P_{SL}(i, j) = \binom{i}{j} \epsilon_F^j (1 - \epsilon_F)^{i-j}$$

接著在時間 t_1 ，若有兩個封包遺失，則接收端向傳送端發出要求重傳兩個封包，其錯誤的事件分為三種，傳送端收到接收端的要求，並且送出兩個封包，結果遺失一個封包與兩個封包兩種，以及傳送端沒有收到接收端的要求，因此不重傳，接收端仍然沒有收到那兩個封包。接收端要求重傳 i 個封包而遺失 j 個封包的機率為

$$P_{RL}(i, j) = \begin{cases} \binom{i}{j} (1 - \epsilon_B) \epsilon_F^j (1 - \epsilon_F)^{i-j} & , \text{for } i \neq j \\ \binom{i}{j} (\epsilon_B + (1 - \epsilon_B) \epsilon_F^j) & , \text{for } i = j \end{cases}$$

有了以上的假設，便可以推導出底下的錯誤期望值與花費期望值。

3.2.1.1 錯誤期望值

首先僅考慮重傳的部分，亦即 t_1 以後的狀態。

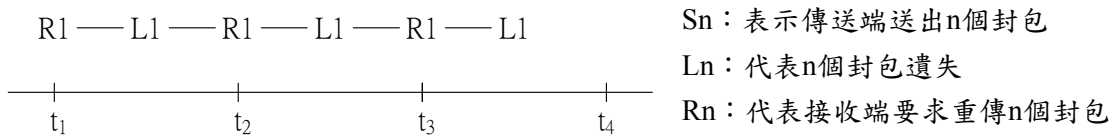


圖 3: 接收端驅動重傳下傳送一個封包之錯誤事件示意圖

令 $F_i(r)$ 為接收端要求 i 個封包，經過至多 r 次重傳後沒有完全收到 i 個封包的機率。

當接收端要求一個封包的情況下，如圖 3，不同最多重傳次數的錯誤期望值如下：

$$r = 1, F_1(1) = P_{RL}(1,1)$$

$$r = 2, F_1(2) = (P_{RL}(1,1))^2$$

$$r = 3, F_1(3) = (P_{RL}(1,1))^3$$

可推得接收端要求一個封包，經過至多 r 次重傳後沒有完全收到一個封包的機率為

$$F_1(r) = (P_{RL}(1,1))^r$$

當接收端要求二個封包的情況下，如圖 4，不同最多重傳次數的錯誤期望值如下：

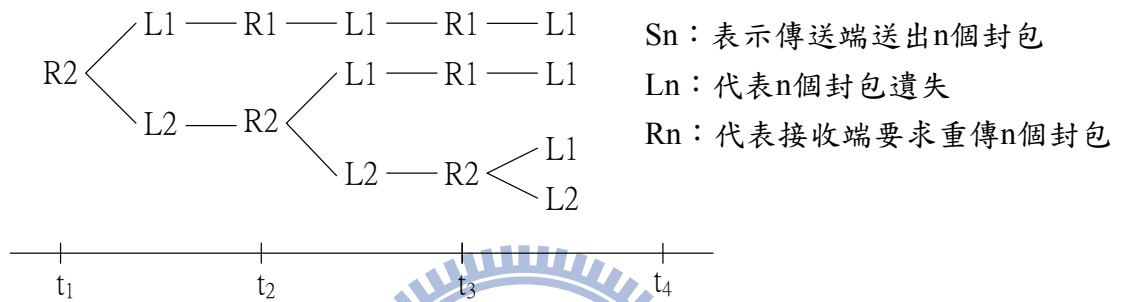


圖 4: 接收端驅動重傳下傳送二個封包之錯誤事件示意圖

$$r = 1, F_2(1) = \sum_{j=1}^2 P_{RL}(2, j)$$

$$r = 2, F_2(2) = P_{RL}(2,1)P_{RL}(1,1) + P_{RL}(2,2) \sum_{j=1}^2 P_{RL}(2, j)$$

$$r = 3, F_2(3) = P_{RL}(2,1)(P_{RL}(1,1))^2 + P_{RL}(2,2)(P_{RL}(2,1)P_{RL}(1,1) + P_{RL}(2,2)F_2(1))$$

可推得接收端要求二個封包，經過至多 r 次重傳後沒有完全收到二個封包的機率為

$$F_2(r) = P_{RL}(2,1)F_1(r-1) + P_{RL}(2,2)F_2(r-1)$$

當接收端要求三個封包的情況下，如圖 5，不同最多重傳次數的錯誤期望值如下：

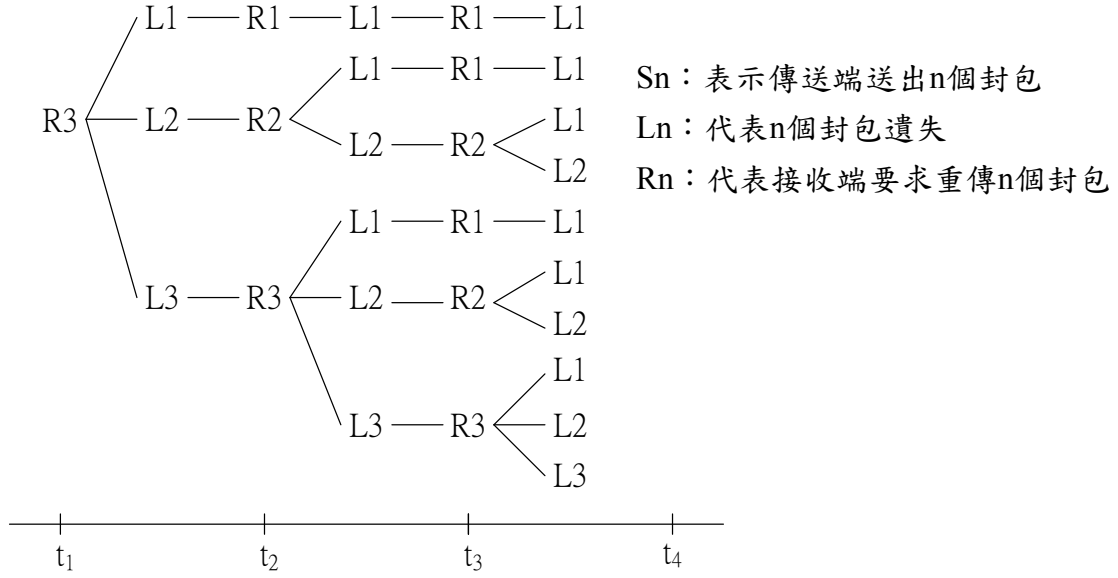


圖 5: 接收端驅動重傳下傳送三個封包之錯誤事件示意圖

$$r = 1, F_3(1) = \sum_{j=1}^3 P_{RL}(3, j)$$

$$r = 2, F_3(2) = P_{RL}(3,1)P_{RL}(1,1) + P_{RL}(3,2)P_{RL}(2,2) + P_{RL}(3,3)F_3(1)$$

$$r = 3, F_3(3) = P_{RL}(3,1)F_1(2) + P_{RL}(3,2)F_2(2) + P_{RL}(3,3)F_3(2)$$

可推得接收端要求3個封包，經過至多 r 次重傳後沒有完全收到3個封包的機率為

$$F_3(r) = P_{RL}(3,1)F_1(r-1) + P_{RL}(3,2)F_2(r-1) + P_{RL}(3,3)F_3(r-1)$$

以上的推導可看出 $F_i(r)$ 具有遞回的關係。將 t_0 時由傳送端傳封包給接收端的部分也一起考慮時，可推導出當多媒體資料單元封裝成 N 個封包，而傳輸策略 π 最多可重傳 r 次，則傳送此單一多媒體資料單元的錯誤機率期望值為

$$\epsilon_{\pi}(r) = \sum_{i=1}^N P_{SL}(N, i) F_i(r) \quad (1)$$

其中

$$F_i(r) = \begin{cases} \sum_{j=1}^i P_{RL}(i, j) F_j(r-1) & , \text{for } r > 0 \\ 1 & , \text{for } r = 0 \end{cases}$$

3.2.1.2 花費期望值

同樣先考慮重傳的部分，亦即 t_1 以後的狀態。令 $G_i(t)$ 為接收端要求 i 個封包，於時間 t 重傳的花費期望值。

當接收端要求一個封包的情況下，如圖 6，於不同時間重傳的花費期望值如下：

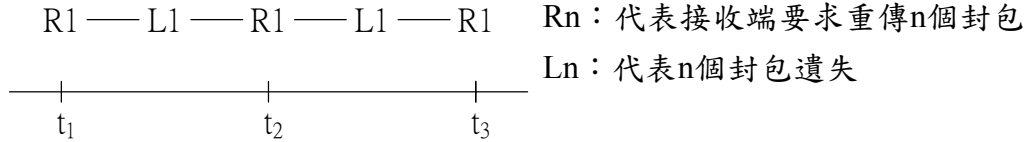


圖 6: 接收端驅動重傳下傳送一個封包之花費示意圖

$$t = 1, G_1(1) = (1 - \epsilon_B)$$

$$t = 2, G_1(2) = (\epsilon_B + (1 - \epsilon_B)\epsilon_F)(1 - \epsilon_B) = P_{RL}(1,1)(1 - \epsilon_B)$$

$$t = 3, G_1(3) = (P_{RL}(1,1))^2(1 - \epsilon_B)$$

當接收端要求二個封包的情況下，如圖 7，於不同時間重傳的花費期望值如下：

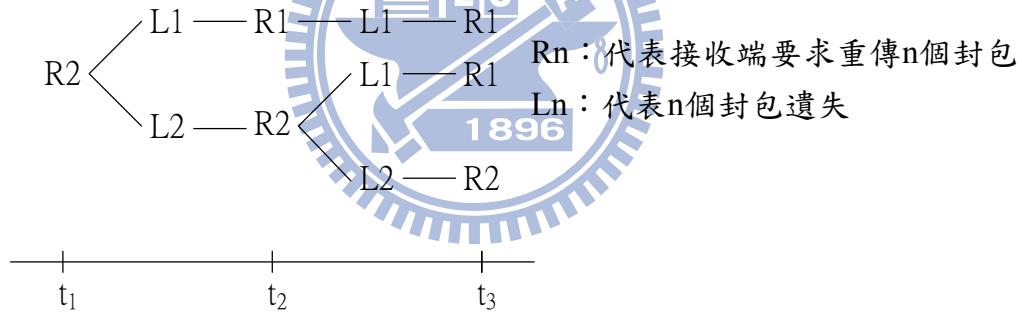


圖 7: 接收端驅動重傳下傳送二個封包之花費示意圖

$$t = 1, G_2(1) = 2(1 - \epsilon_B)$$

$$t = 2, G_2(2) = P_{RL}(2,1)(1 - \epsilon_B) + P_{RL}(2,2) \cdot 2(1 - \epsilon_B)$$

$$t = 3, G_2(3) = P_{RL}(2,1)P_{RL}(1,1)(1 - \epsilon_B) + P_{RL}(2,2)(P_{RL}(2,1)(1 - \epsilon_B) + P_{RL}(2,2) \cdot 2(1 - \epsilon_B))$$

可推得接收端要求二個封包，於時間 t 重傳的花費期望值為

$$G_2(t) = P_{RL}(2,1)G_1(t - 1) + P_{RL}(2,2)G_2(t - 1)$$

以上的推導可看出 $G_i(t)$ 也同樣具有遞回的關係。將 t_0 時由傳送端傳封包給接收端的部分也一起考慮，可推導出當多媒體資料單元封裝成 N 個封包，傳送此單一多媒體資料單元

於時間 t 的花費期望值為

$$\rho(t) = \begin{cases} \frac{1}{N} \sum_{i=1}^N P_{SL}(N, i) G_i(t) & , \text{for } t > 0 \\ 1 & , \text{for } t = 0 \end{cases}$$

其中

$$G_i(t) = \begin{cases} \sum_{j=1}^i P_{RL}(i, j) G_j(t-1) & , \text{for } t > 1 \\ i \cdot (1 - \epsilon_B) & , \text{for } t = 1 \end{cases}$$

若傳輸策略 π 最多可重傳 r 次，則其總共花費期望值為

$$\rho_{\pi}(r) = \sum_{t=0}^r \rho(t) \quad (2)$$

3.2.2 接收端驅動積極重傳

在實際的實時視訊串流系統中，接收端的暫存區僅容納約三秒左右的多媒體資料[6]，代表任多媒體資料單元從傳送端送出後，能夠再重傳的次數並不多。目前設定的接收端驅動重傳情況下，傳送端傳送多媒體資料單元的時機除了在 $t_{0,i}$ 時自動送出多媒體資料單元之外，只有在收到來自接收端的重傳要求才會再次傳送該多媒體資料單元，當接收端送出的重傳要求沒有被傳送端收到，那麼傳送端將不做任何動作。在重傳次數不多的情況下，此做法會較沒有效率。為把握次數不多的重傳機會進行更有效的傳輸，我們將原傳輸機制做了修改，成為積極重傳(aggressive retransmission)。積極重傳同樣是在 $t_{0,i}$ 時自動送出多媒體資料單元，若是接收端收到多媒體資料單元則回覆ACK(Acknowledgement)訊息，表示已成功收到；沒收到則發出重傳要求。接著，傳送端收到ACK代表不需再傳該多媒體資料單元，收到重傳要求則再次傳送該多媒體資料單元，(若是在該收到來自接收端訊息的時間，卻沒收到任何訊息)，若是沒收到任何來自接收端的訊息，而時間也超過訊息從接收端至傳送端平均所需的時間，那麼情況便假定為訊息遺失，此時，傳送端會再將該多媒體資料單元最近一次所傳輸的封包進行重傳。以上動作會持續至時

間到達解碼最後期限。

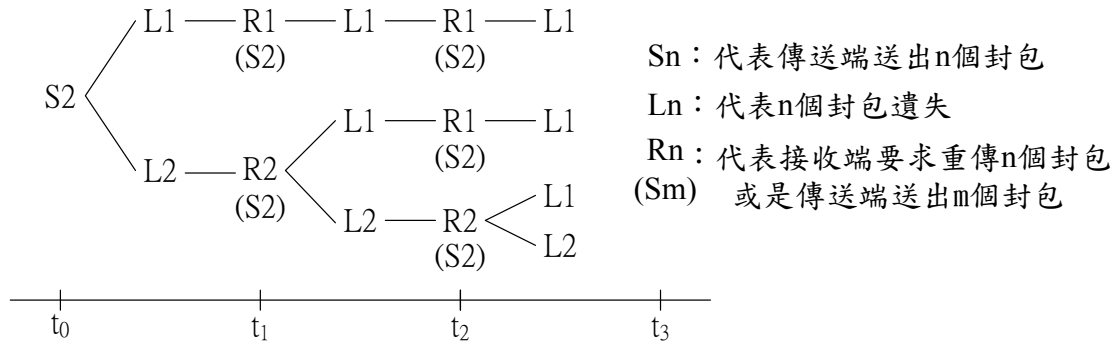


圖 8: 接收端驅動積極重傳之錯誤事件示意圖

圖 8 為接收端驅動積極重傳的所有錯誤事件示意圖。一開始在時間 t_0 傳送端傳送兩個封包給接收端，發生錯誤的事件有兩種，分別為遺失一個和兩個封包。傳送 i 個封包而遺失 j 個封包的機率為

$$P_{SL}(i, j) = \binom{i}{j} \epsilon_F^j (1 - \epsilon_F)^{i-j}$$

接著在時間 t_1 ，若有一個封包遺失，則接收端向傳送端發出要求重傳一個封包，若是傳送端收到接收端的要求，就傳送接收端要求的封包；若是接收端送給傳送端的要求遺失了，那麼傳送端在沒收到要求情況下，便會假定最近一次傳送的封包全部遺失，將上一次接收端要求的封包再重傳一次，如此一來雖然可能多傳送了不必要的資訊，但也確保在短時間內能將要求的封包傳給接收端。在此情況，無論傳送端是否收到接收端的要求，傳送端都會傳送接收端可能缺少的封包，因此，接收端要求一個封包，然後傳送端有可能傳送一個封包(成功收到接收端要求)或傳送二個封包(沒有收到接收端要求)，最終錯誤情況就是遺失一個封包。在此情況下反向通道的封包遺失率並不影響傳送端重傳與否。接收端要求重傳 i 個封包而遺失 j 個封包的機率為

$$P_{RL}(i, j) = \binom{i}{j} \epsilon_F^j (1 - \epsilon_F)^{i-j}$$

有了以上的假設，便可以推導出底下的錯誤期望值與花費期望值。

3.2.2.1 錯誤機率期望值

推導過程同接收端驅動重傳的錯誤機率期望值一樣，可推得當多媒體資料單元封裝成 N 個封包，而傳輸策略 π 最多可重傳 r 次，則傳送此單一多媒體資料單元的錯誤機率期望值為

$$\epsilon_{\pi}(r) = \sum_{i=1}^N P_{SL}(N, i) F_i(r) \quad (3)$$

其中

$$F_i(r) = \begin{cases} \sum_{j=1}^i P_{RL}(i, j) F_j(r-1) & , \text{for } r > 0 \\ 1 & , \text{for } r = 0 \end{cases}$$

3.2.2.2 花費期望值

令 $G_i(t)$ 為一開始欲傳送 i 個封包，於時間 t 重傳的花費期望值，而 $G_{i,j}(t)$ 為欲傳送 i 個封包而遺失 j 個的情況下，在時間 t 重傳的花費期望值。

當接收端要求一個封包的情況下，如圖 9，不同時間重傳次數的花費期望值如下：

$$t = 0, G_1(0) = 1$$

$$t = 1, G_1(1) = P_{SL}(1,0)(\epsilon_B \cdot 1) + P_{SL}(1,1)(1) = P_{SL}(1,0)(\epsilon_B \cdot 1) + P_{SL}(1,1)G_1(0)$$

$$\begin{aligned} t = 2, G_1(2) &= P_{SL}(1,0)(\epsilon_B^2 \cdot 1) + P_{SL}(1,1)(P_{SL}(1,0)(\epsilon_B \cdot 1) + P_{SL}(1,0)(1)) \\ &= P_{SL}(1,0)(\epsilon_B^2 \cdot 1) + P_{SL}(1,1)G_1(1) \end{aligned}$$

$$\begin{aligned} t = 3, G_1(3) &= P_{SL}(1,0)(\epsilon_B^3 \cdot 1) \\ &+ P_{SL}(1,1) \left(P_{SL}(1,0)(\epsilon_B^2 \cdot 1) + P_{SL}(1,1)(P_{SL}(1,0)(\epsilon_B \cdot 1) + P_{SL}(1,0)(1)) \right) \\ &= P_{SL}(1,0)(\epsilon_B^3 \cdot 1) + P_{SL}(1,1)G_1(2) \end{aligned}$$

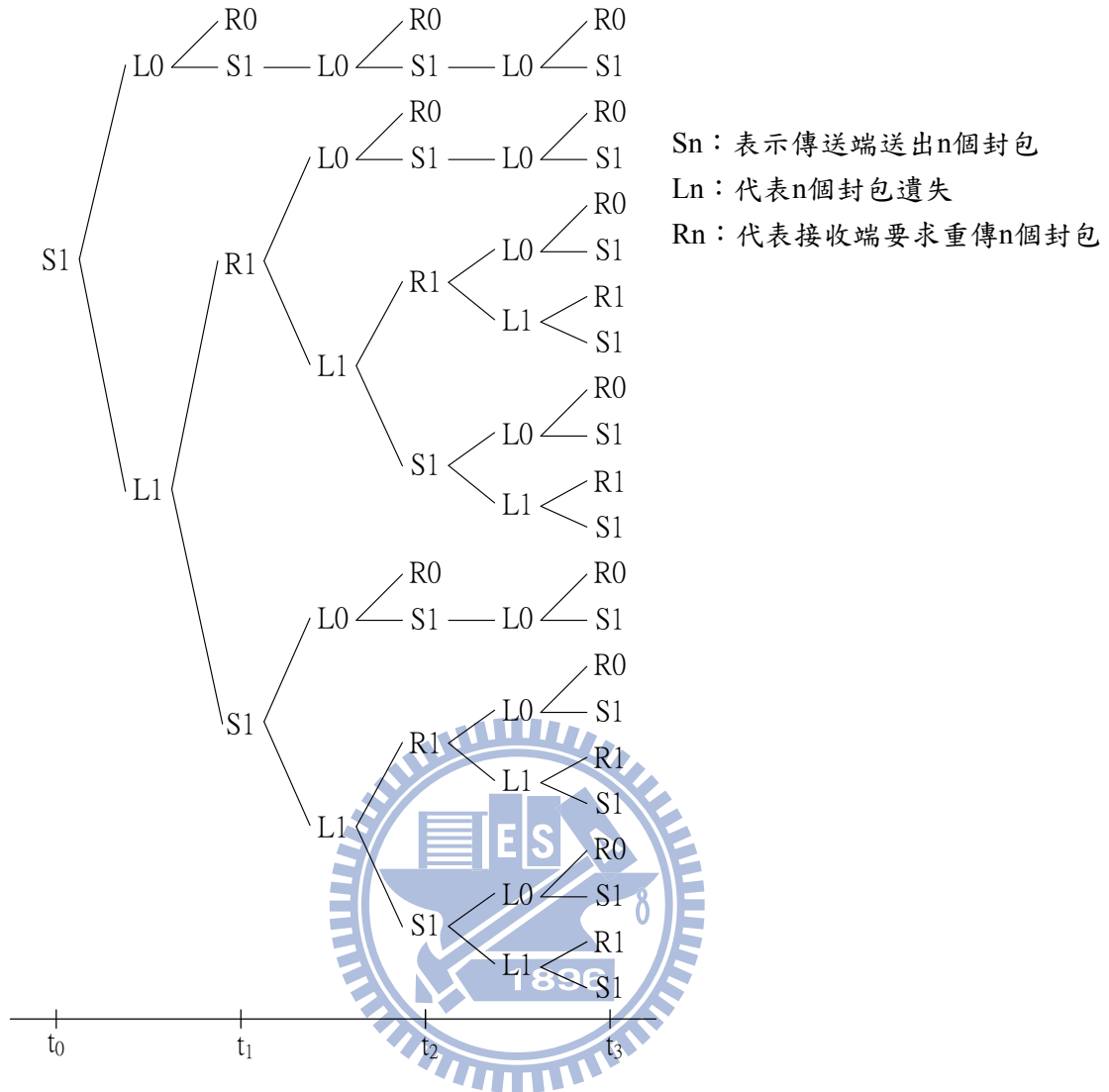


圖 9: 接收端驅動積極重傳下傳送一個封包之花費示意圖

圖 9對稱的架構，經過簡化後即成為底下圖 10。

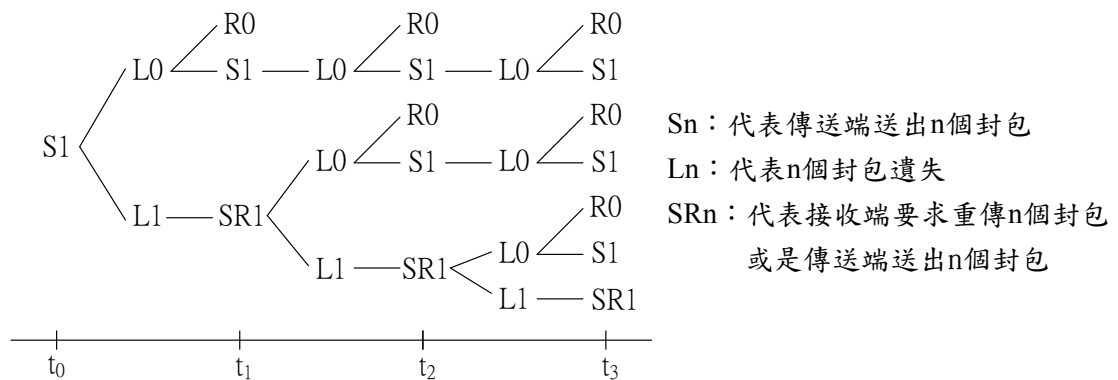


圖 10: 接收端驅動積極重傳下傳送一個封包之花費簡化示意圖

當接收端要求二個封包的情況下，如圖 11，不同時間重傳的花費期望值如下：

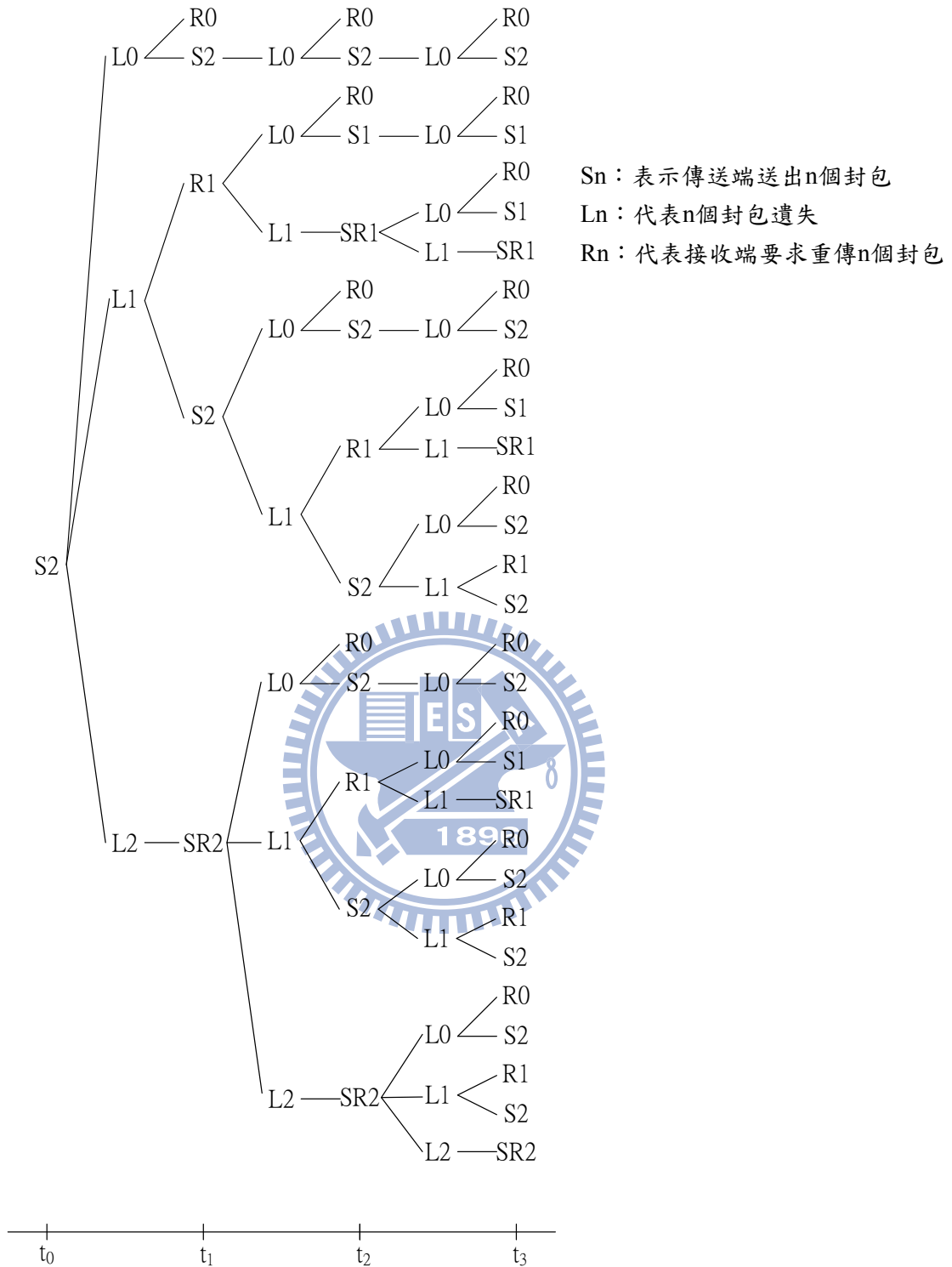


圖 11: 接收端驅動積極重傳下傳送二個封包之花費示意圖

$$t = 0, G_2(0) = 2$$

$$t = 1, G_2(1) = P_{SL}(2,0)(\epsilon_B \cdot 2) + P_{SL}(2,1)((1 - \epsilon_B) \cdot 1 + \epsilon_B \cdot 2) + P_{SL}(2,2)(2)$$

$$\begin{aligned}
t = 2, G_2(2) &= P_{SL}(2,0)(\epsilon_B^2 \cdot 2) \\
&+ P_{SL}(2,1) \left((1 - \epsilon_B)(P_{SL}(1,0)(\epsilon_B \cdot 1) + P_{SL}(1,1)(1)) \right. \\
&+ \left. \epsilon_B (P_{SL}(1,0)(\epsilon_B \cdot 2) + P_{SL}(1,1)((1 - \epsilon_B) \cdot 1 + \epsilon_B \cdot 2)) \right) \\
&+ P_{SL}(2,2) \left(P_{SL}(2,0)(\epsilon_B \cdot 2) + P_{SL}(2,1)((1 - \epsilon_B) \cdot 1 + \epsilon_B \cdot 2) \right. \\
&+ \left. P_{SL}(2,2)(2) \right) = P_{SL}(2,0)G_{2,0}(2) + P_{SL}(2,1)G_{2,1}(2) + P_{SL}(2,2)G_2(1)
\end{aligned}$$

$$\begin{aligned}
t = 3, G_2(3) &= P_{SL}(2,0)(\epsilon_B^3 \cdot 2) \\
&+ P_{SL}(2,1) \left((1 - \epsilon_B)G_1(2) + \epsilon_B (P_{SL}(1,0)(\epsilon_B^2 \cdot 2) + P_{SL}(1,1)G_{2,1}(2)) \right) \\
&+ P_{SL}(2,2)G_2(2) = P_{SL}(2,0)G_{2,0}(3) + P_{SL}(2,1)G_{2,1}(3) + P_{SL}(2,2)G_2(2)
\end{aligned}$$

可推得接收端要求2個封包，在時間 t 重傳的花費期望值為

$$G_2(t) = P_{SL}(2,0)G_{2,0}(t-1) + P_{SL}(2,1)G_{2,1}(t-1) + P_{SL}(2,2)G_2(t-1)$$

以上的推導可看出 $G_i(t)$ 與 $G_{i,j}(t)$ 具有遞回的關係。由此可推導出當多媒體資料單元封裝成 N 個封包，傳送此單一多媒體資料單元於時間 t 的花費期望值為

$$\rho(t) = \frac{1}{N} (G_N(t))$$

其中

$$G_N(t) = \begin{cases} \left(\sum_{i=0}^{N-1} P_{SL}(N,i)G_{N,i}(t) \right) + P_{SL}(N,N)G_N(t-1) & , \text{for } t > 0 \\ N & , \text{for } t = 0 \end{cases}$$

$$G_{N,i}(t) = \begin{cases} (1 - \epsilon_B)G_i(t-1) + \epsilon_B \sum_{j=0}^i P_{RL}(i,j)G_{N,j}(t-1) & , \text{for } t > 1 \\ (1 - \epsilon_B)G_i(t-1) + \epsilon_B G_N(t-1) & , \text{for } t = 1 \end{cases}$$

若傳輸策略 π 為連續重傳 r 次，則重傳 r 次後的花費期望值為

$$\rho_\pi(r) = \sum_{t=0}^r \rho(t) \quad (4)$$

3.3 傳送多媒體資料單元群組

有了傳送單一多媒體資料單元的錯誤期望值與花費期望值，接著考慮傳送一組包含多個多媒體資料單元的群組時，藉由各多媒體資料單元的錯誤期望值與花費期望值來推算多媒體資料單元群組的失真期望值與頻寬花費期望值。

我們將一個多媒體資料單元群組定義為一個視訊層表示格式(scalable layer representation) $S(L, T)$ 於一個 GOP(Group Of Pictures)內所有的多媒體資料單元，也就是 $\{\aleph_G(L_i, T_i) | \aleph(L_i, T_i) \in S(L, T), i = 1, 2, \dots, L\}$ ，代表在第 G 個 GOP 中，所有的屬於此視訊表示格式的多媒體資料單元。

3.3.1 失真期望值

首先推算一組多媒體資料單元經過傳輸後，因不同的封包遺失率與傳輸策略所造成不同的失真期望值。

假設 D_0 代表完全沒收到任何多媒體資料單元時的失真 MSE 值， $\Delta D_{(L', T')}$ 為收到 $\aleph_G(L', T')$ 時所能改善的失真， $\epsilon_{\pi(L'', T'')}$ 為 $\aleph_G(L'', T'')$ 以傳輸策略 $\pi(L'', T'')$ 傳輸時的錯誤率，可參照公式(1)或公式(3)，那麼以傳輸策略 π 來傳輸 target layer 為 $S(L, T)$ 的多媒體資料單元群組時，其失真期望值為

$$D_{(L, T)}(\pi) = D_0 - \sum_{\aleph_G(L', T') \in S(L, T)} \Delta D_{(L', T')} \prod_{\aleph_G(L'', T'') \preceq \aleph_G(L', T')} (1 - \epsilon_{\pi(L'', T'')}) + C_{(L, T)} \quad (5)$$

其中 $\aleph_G(L'', T'') \preceq \aleph_G(L', T')$ 表示 $\aleph_G(L', T')$ 直接或間接參照到 $\aleph_G(L'', T'')$ ，而 $C_{(L, T)}$ 代表當部分多媒體資料單元遺失而出現有多個可解碼的視訊層表示格式(scalable layer representation)時，擇取失真較低的 $S(L, T)$ 解碼後，針對那些不屬於 $S(L, T)$ 的多媒體資料單元所降低的失真補償回來的值。部分多媒體資料單元雖然收到了卻沒有實質上地改善失真，因此要把相對應所降低的失真量補回來。

例如目標視訊層為 $S(1, 1)$ 時，會出現一種具有多個可解碼的視訊層表示格式的情況，如圖 12，此時要將失真補償回來的值為

$$C_{(1,1)} = (1 - \epsilon_{\pi(0,0)})(1 - \epsilon_{\pi(0,1)})(1 - \epsilon_{\pi(1,0)})\epsilon_{\pi(1,1)} \cdot (\Delta D_{(0,0)} + \Delta D_{(0,1)} + \Delta D_{(1,0)} - \max\{\Delta D_{(0,0)} + \Delta D_{(0,1)}, \Delta D_{(0,0)} + \Delta D_{(1,0)}\})$$

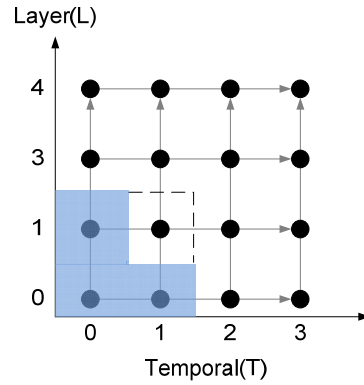


圖 12: 目標視訊層為 $S(1,1)$ 時出現多個可解視訊層表示格式的情況

當目標視訊層為 $S(1,2)$ 時，具有多個可解碼的視訊層表示格式的情況有三種，如圖 13，計算 $C_{(1,2)}$ 需將這三種情況都考慮進去。

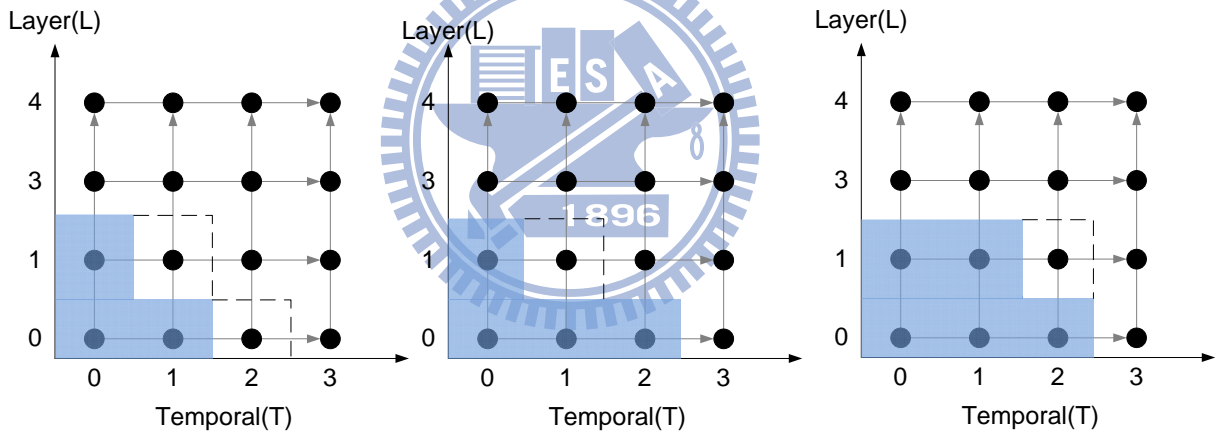


圖 13: 目標視訊層為 $S(1,2)$ 時出現多個可解視訊層表示格式的情況

$$\begin{aligned}
\mathcal{C}_{(1,2)} = & (1 - \epsilon_{\pi(0,0)})(1 - \epsilon_{\pi(0,1)})(1 - \epsilon_{\pi(1,0)})\epsilon_{\pi(0,2)}\epsilon_{\pi(1,1)} \left(\Delta D_{(0,0)} + \Delta D_{(0,1)} + \Delta D_{(1,0)} \right. \\
& - \left. \left\{ \sum_{\mathfrak{N}_G(L,T) \in \mathcal{S}(0,1)} \Delta D_{(L,T)}, \sum_{\mathfrak{N}_G(L,T) \in \mathcal{S}(1,0)} \Delta D_{(L,T)} \right\}_{max} \right) \\
& + (1 - \epsilon_{\pi(0,0)})(1 - \epsilon_{\pi(0,1)})(1 - \epsilon_{\pi(0,2)})(1 - \epsilon_{\pi(1,0)})\epsilon_{\pi(1,1)} \left(\Delta D_{(0,0)} \right. \\
& + \Delta D_{(0,1)} + \Delta D_{(0,2)} + \Delta D_{(1,0)} \\
& - \max \left\{ \sum_{\mathfrak{N}_G(L,T) \in \mathcal{S}(0,2)} \Delta D_{(L,T)}, \sum_{\mathfrak{N}_G(L,T) \in \mathcal{S}(1,0)} \Delta D_{(L,T)} \right\} \\
& + (1 - \epsilon_{\pi(0,0)})(1 - \epsilon_{\pi(0,1)})(1 - \epsilon_{\pi(0,2)})(1 - \epsilon_{\pi(1,0)})(1 \\
& - \epsilon_{\pi(1,1)})\epsilon_{\pi(1,2)} \left(\Delta D_{(0,0)} + \Delta D_{(0,1)} + \Delta D_{(0,2)} + \Delta D_{(1,0)} + \Delta D_{(1,1)} \right. \\
& - \max \left\{ \sum_{\mathfrak{N}_G(L,T) \in \mathcal{S}(0,2)} \Delta D_{(L,T)}, \sum_{\mathfrak{N}_G(L,T) \in \mathcal{S}(1,1)} \Delta D_{(L,T)} \right\} \left. \right)
\end{aligned}$$

由以上例子可推得當目標視訊層為 $S(L, T)$ 時失真補償值為

$$\mathcal{C}_{(L,T)} = \sum_{\lambda \in \Lambda(L,T)} P_\lambda \left(\sum_{\mathfrak{N}_G(L,T) \in \lambda_r} \Delta D_{(L,T)} - \max_{i \in [1 \dots k]} \left\{ \sum_{\mathfrak{N}_G(L,T) \in \mathcal{S}(L_i, T_i)} \Delta D_{(L,T)} \right\} \right) \quad (6)$$

其中 $\Lambda(L, T)$ 代表目標視訊層為 $S(L, T)$ 時，因部分多媒體資料單元遺失而出現 k 個可解的視訊層表示格式的所有事件，而 λ 代表其中的一個事件， P_λ 為發生事件 λ 的機率， λ_r 代表在事件 λ 中收到的多媒體資料單元的集合。

3.3.2 頻寬花費期望值

一組多媒體資料單元經過傳輸後，其頻寬花費期望值和各個多媒體資料單元的位元組大小以及不同封包遺失率與傳輸策略所產生的花費期望值有關係。

假設 $B_{(L', T')}$ 代表多媒體資料單元 $\mathfrak{N}_G(L', T')$ 的位元組大小， H 為封包頭的位元組大小，

$N_{(L',T')}$ 為多媒體資料單元 $\mathfrak{N}_G(L',T')$ 所封裝成 RTP 封包的數量， $\rho_{\pi(L',T')}$ 代表 $\mathfrak{N}_G(L',T')$ 以傳輸策略 $\pi(L',T')$ 傳輸時的花費期望值，或位元組大小增長的倍率，可參照公式(2)或公式(4)，那麼以傳輸策略 π 來傳輸目標視訊層為 $S(L,T)$ 的多媒體資料單元群組，其頻寬花費期望值為

$$R_{(L,T)}(\pi) = \sum_{\mathfrak{N}_G(L',T') \in S(L,T)} \left(B_{(L',T')} + H \cdot N_{(L',T')} \right) \rho_{\pi(L',T')} \quad (7)$$

3.3.3 多個多媒體資料單元放在同一個封包

當一組多媒體資料單元中有多個多媒體資料單元的位元組大小不大時，為了省去封包頭所耗費的頻寬，會將多個小的多媒體資料單元封裝至同一個封包。如此一來，放在同一個封包上的多媒體資料單元皆擁有相同的錯誤期望值與花費期望值。在算頻寬花費與失真的期望值時，只要收到該封包，就代表封包上所有多媒體資料單元都傳輸成功。若是該封包丟失，代表每個封包中的多媒體資料單元皆傳輸失敗。

假設 $\mathfrak{N}_G(L,T)$ 和 $\mathfrak{N}_G(L_i, T_i) \big|_{i=1}^k$ 共 $k+1$ 個多媒體資料單元放在同一個封包，那麼這些多媒體資料單元會有相同的錯誤期望值和花費期望值。

$$\epsilon_{(L,T)} = \epsilon_{(L_i, T_i) \big|_{i=1}^k}$$

$$\rho_{\pi(L,T)} = \rho_{\pi(L_i, T_i) \big|_{i=1}^k}$$

且收到 $\mathfrak{N}_G(L,T)$ 和 $\mathfrak{N}_G(L_i, T_i) \big|_{i=1}^k$ 所能降低的失真為

$$\left(\Delta D_{(L,T)} + \sum_{i=1}^k \Delta D_{(L_i, T_i)} \right) \prod_{\mathfrak{N}_G(L',T') \in \mathfrak{N}_G(L,T)} (1 - \epsilon_{\pi(L',T')})$$

而傳輸 $\mathfrak{N}_G(L,T)$ 和 $\mathfrak{N}_G(L_i, T_i) \big|_{i=1}^k$ 所需花費的頻寬為

$$\left(B_{(L,T)} + \sum_{i=1}^k B_{(L_i, T_i)} + H \right) \rho_{\pi(L,T)}$$

3.3.4 一個多媒體資料單元中部份的畫面解開所降低的失真

封裝成 N 個封包的多媒體資料單元，在傳送過程中可能會封包丟失，最後只有部份的封

包能成功傳送至接收端。在此情況下，僅收到多媒體資料單元部份的封包，能否解開並播放呢？若是該多媒體資料單元包含不只一張畫面，那麼收到部份封包，仍有解碼的可能。

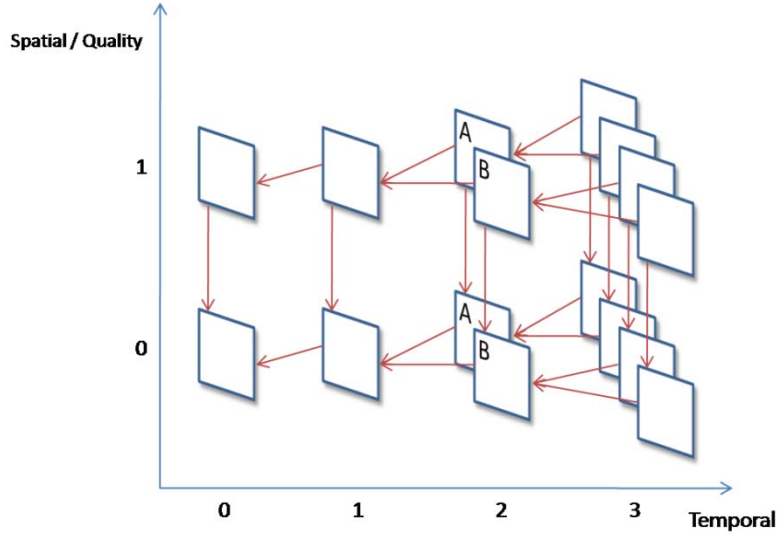


圖 14: SVC 視訊畫面之間的相依關係

假設 $\zeta(0,2,t_1)$ 與 $\zeta(0,2,t_2)$ 兩NAL units代表 $\mathfrak{N}_G(0,2)$ 中於時間 t_1 播放的畫面A與時間 t_2 播放的畫面B，如圖 14，且兩張畫面的錯誤機率與降低的失真量相等。令 $\hat{\epsilon}_{\pi(0,2)}$ 代表 $\mathfrak{N}_G(0,2)$ 中特定一張畫面的錯誤機率，則 $\hat{\epsilon}_{\pi(0,2)} = \epsilon_{\pi(0,2,t_1)} = \epsilon_{\pi(0,2,t_2)}$ ，又 $\mathfrak{N}_G(0,0)$ 與 $\mathfrak{N}_G(0,1)$ 同樣只包含一張畫面，故 $\hat{\epsilon}_{\pi(0,1)} = \epsilon_{\pi(0,1)}$ ， $\hat{\epsilon}_{\pi(0,2)} = \epsilon_{\pi(0,2)}$

$$\begin{aligned}
 D_{(0,2)}(\boldsymbol{\pi}) &= D_0 - \Delta D_{(0,0)}(1 - \epsilon_{\pi(0,0)}) - \Delta D_{(0,1)}(1 - \epsilon_{\pi(0,0)})(1 - \epsilon_{\pi(0,1)}) \\
 &\quad - \frac{\Delta D_{(0,2)}}{2}(1 - \epsilon_{\pi(0,0)})(1 - \epsilon_{\pi(0,1)})(1 - \epsilon_{\pi(0,2,t_1)}) \\
 &\quad - \frac{\Delta D_{(0,2)}}{2}(1 - \epsilon_{\pi(0,0)})(1 - \epsilon_{\pi(0,1)})(1 - \epsilon_{\pi(0,2,t_2)}) + \mathcal{C}_{(0,2)} \\
 &= D_0 - \Delta D_{(0,0)}(1 - \hat{\epsilon}_{\pi(0,0)}) - \Delta D_{(0,1)}(1 - \hat{\epsilon}_{\pi(0,0)})(1 - \hat{\epsilon}_{\pi(0,1)}) \\
 &\quad - \frac{\Delta D_{(0,2)}}{2}(1 - \hat{\epsilon}_{\pi(0,0)})(1 - \hat{\epsilon}_{\pi(0,1)})(1 - \hat{\epsilon}_{\pi(0,2)}) \cdot 2 + \mathcal{C}_{(0,2)} \\
 &= D_0 - \sum_{\mathfrak{N}_G(L',T') \in \mathcal{S}(0,2)} \Delta D_{(L',T')} \prod_{\mathfrak{N}_G(L'',T'') \preccurlyeq \mathfrak{N}_G(L',T')} (1 - \hat{\epsilon}_{\pi(L'',T'')}) + \mathcal{C}_{(0,2)}
 \end{aligned}$$

當考慮某些多媒體資料單元僅部分的畫面能解開，其失真期望值為

$$D_{(L,T)}(\boldsymbol{\pi}) = D_0 - \sum_{\mathfrak{N}_G(L',T') \in \mathcal{S}(L,T)} \Delta D_{(L',T')} \prod_{\mathfrak{N}_G(L'',T'') \preccurlyeq \mathfrak{N}_G(L',T')} (1 - \hat{\epsilon}_{\pi(L'',T'')}) + \mathcal{C}_{(L,T)} \quad (8)$$

3.4 傳輸策略最佳化

影響視訊傳輸表現的主要因素除了傳輸機制與封包遺失率外，就是傳輸策略的設計了。我們在接收端驅動重傳的傳輸機制下，可透過找出最佳的傳輸策略來達到位元率-失真最佳化的視訊傳輸，亦即在有限的頻寬之下，以最佳的傳輸策略來傳送視訊使得接收端收到視訊並解碼後能有最佳的畫面品質。

視訊串流系統進行位元率-失真最佳化時，視訊編碼解碼與網路傳輸兩方面都相當重要。在視訊編碼解碼方面，播放SVC視訊時，使用者裝置會依據本身需求來擷取並解碼所需的視訊層，而依據相依關係來擷取這些視訊層的序列即為SVC視訊的擷取路徑(extraction path)。SVC擷取路徑遵守連續改進(successive refinement)的規範時，當網路突然頻寬降低，使用者裝置可以僅擷取部分的視訊層來解碼，以適應頻寬的變動[2]。另外在網路群播(multicast)時，也能傳送相同的視訊層集合來滿足多種使用者裝置。在編碼端達成位元率-失真最佳化需考慮到SVC視訊擷取路徑的安排，遵守連續改進規則的最佳擷取路徑，使用者裝置在播放時可達到位元率-失真最佳化的表現。

考慮到在網路上傳輸時，隨著可用頻寬的增加，傳輸的目標視訊層與其相對的傳輸策略也跟著改變，以提供更好的失真表現。我們希望能夠在不同的頻寬限制下找到最佳傳輸策略，使得其對應的位元率與失真能夠近似於SVC視訊最佳擷取路徑的表現。

3.4.1 傳輸策略最佳化方法 1

當給定了限制的頻寬與目標視訊層，我們會逐漸將各個視訊層傳輸策略慢慢增加，直到位元率期望值達到了限制的頻寬。在此最佳化方法中，會考慮到SVC視訊的擷取路徑來決定哪些視訊層的最大重傳次數要先增加。每次增加最大重傳次數的視訊層為現有擷取路徑所包含的視訊層，或是擷取路徑延伸至下一個可解視訊層時所需要的視訊層。例如目前已經將 $\mathcal{N}(0,0)$ 的傳輸策略增加的情況下，如圖 15(a)，下一步增加最大重傳次數的視訊層可選擇 $\mathcal{N}(0,0)$ ，或是選擇當擷取路徑延伸至下一個可解視訊層 $S(0,1)$ 或 $S(1,0)$ 時所需的視訊層，也就是 $\mathcal{N}(0,1)$ 或 $\mathcal{N}(1,0)$ 。若是選擇了增加 $\mathcal{N}(0,1)$ 的最大重傳次數，如圖 15(b)，

接下來，下一步要增加最大重傳次數的視訊層選擇有 $\aleph(0,0)$ 、 $\aleph(0,1)$ ，或是擷取路徑延伸至下一個可解視訊層 $S(0,2)$ 所需的 $\aleph(0,2)$ ，以及 $S(1,1)$ 所需的 $\aleph(1,0)$ 與 $\aleph(1,1)$ ；如果選擇了 $\aleph(1,0)$ 與 $\aleph(1,1)$ ，那麼此二層視訊層會一起傳輸，其傳輸策略會同時增加。

假設目標視訊層為 $S(1,2)$ ，最大重傳次數限制為八次，其最佳化過程可參考圖 16與表格 1。

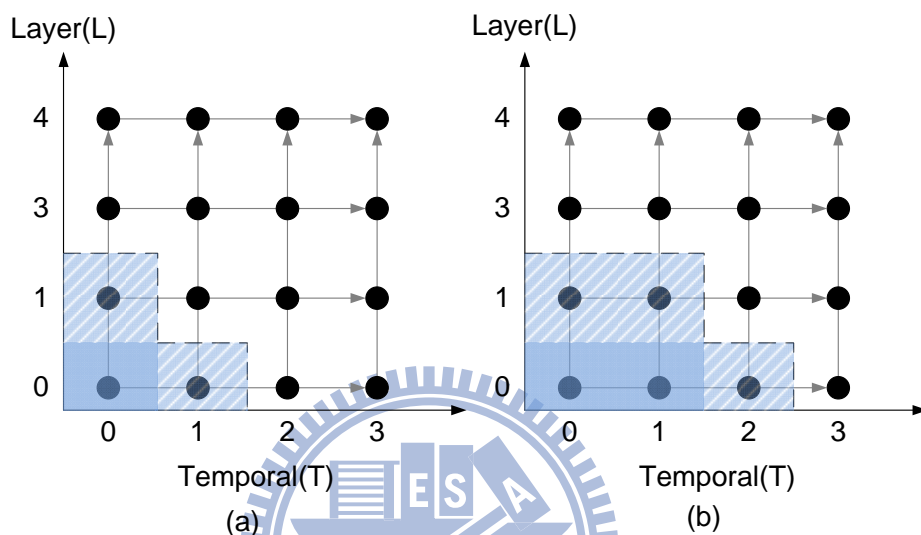


圖 15: 最佳化過程可選擇增加最大重傳次數的視訊層

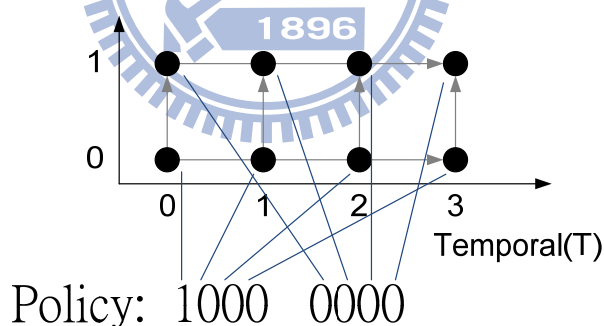


圖 16: 多媒體資料單元群組織傳輸策略示意圖

Optimization algorithm 1	最佳化結果
round 8 ...	8000 0000
round 9 8100 0000	8100 0000
8000 1000	
round 10 8200 0000	8200 0000
8110 0000	
8100 1100	

表格 1: 傳輸策略最佳化方法 1 的最佳化過程

3.4.2 傳輸策略最佳化方法 2

由於傳輸策略最佳化方法 1 被 SVC 視訊的擷取路徑所限制，可能同時必須增加多個視訊層的最大重傳次數，較無法細微地去設定個別視訊層的傳輸策略，於是我們試著將此限制拿掉，作為第二種傳輸策略最佳化方法。在此方法中，每一步欲增加最大重傳次數的視訊層只有一層，不像方法 1 可能為多層，因此可細微地調整傳輸策略，如表格 2。此外，增加最大重傳次數的視訊層可任意選擇，沒有最佳化方法 1 的限制，選擇性也相對變多。

Optimization algorithm 2		最佳化結果
round 0	Initialization	0000 0000
round 1	1000 0000 0100 0000 ⋮ 0000 0001	1000 0000
round 2	2000 0000 1100 0000 ⋮ 1000 0001	2000 0000

表格 2: 傳輸策略最佳化方法 2 的最佳化過程

3.4.3 強迫相依視訊層重傳

SVC 視訊層之間彼此相依的特性，使得欲解碼某一視訊層時，連同其底下所參照到的視訊層也必須可解，上面的視訊層才能夠成功解碼。倘若收到上面的視訊層，而底下的視訊層尚未收到，為了能順利解開已收到的視訊層，此時不管下面視訊層的傳輸策略為何，皆將最多重傳次數改為最大的限制值，亦即是最後解碼期限前，只要有重傳機會就可以重傳，來增加已收到的上層解開之機率，如此能夠有效使用頻寬改善失真，避免收到上面的視訊層卻解不開的情況。

然而，這樣的作法會使得各個多媒體資料單元是否成功收到的事件變得相關，並非我們建立估測模型時所假設的彼此是獨立的。因此，此方法需要更複雜的數學模型，或是需

要以模擬來實現，此部分為此研究未來延伸的方向。

3.5 相關研究

Chou和Miao將位元率-失真最佳化視訊傳輸此議題簡化為對單一多媒體資料單元傳輸做錯誤-花費最佳化，此概念使得失真的計算得以量化[1]。並且，針對多種傳輸機制建立錯誤-花費模型，依此推算在不同的傳輸策略下頻寬花費與視訊失真的期望值。底下將介紹Chou和Miao的研究方法，以及部分我們在考量與做法上的相異之處。

3.5.1 基本模型

在視訊串流系統中，壓縮後的多媒體資料被封裝成多媒體資料單元然後放入封包中在網路上傳輸。Chou將多媒體資料單元封裝至封包設定為一對多的關係，過大的多媒體資料單元會封裝至多個封包，然而一個封包只能擺放一個多媒體資料單元，即使該封包仍然能容納其他較小的多媒體資料單元。

各個多媒體資料單元同時也儲存了各自的位元組大小 B_l 、解碼後所能降低的失真 Δd_l 以及解碼的最後期限 T_{DTS} 。各個多媒體資料單元的位元組大小不一定相同，其解碼後所能降低的失真也不一樣，通常來說解開基本層的多媒體資料單元降低的失真會比解開增強層的多媒體資料單元來得多， Δd_l 的值越大，則代表此多媒體資料單元愈重要。隨著播放時間的不同，各個多媒體資料單元解碼的最後期限也不同。多媒體資料單元彼此間的相依關係可以表示成有向無循環圖(directed acyclic graph)。

在Chou的研究中，各個封包遺失的事件是獨立的，且封包遺失的事件包含了封包在傳輸過程中不見與封包太晚傳至接收端，以致於過了該封包解碼的最後期限而當作封包遺失。假設封包的去程時間 FTT(forward trip time)的機率密度為 p_F ，把封包在順向通道(forward channel)傳輸過程中遺失設定為 $FTT = \infty$ ，那麼封包遺失的機率為

$$P\{FTT > \tau\} = \epsilon_F + (1 - \epsilon_F) \int_{\tau}^{\infty} p_F(t|\text{not lost})dt$$

而反向通道(backward channel)也是相似的設定，封包的回程時間 BTT(backward trip time)

的機率密度為 p_B ，在反向通道封包遺失的機率為

$$P\{\text{FTT} > \tau\} = \epsilon_B + (1 - \epsilon_B) \int_{\tau}^{\infty} p_B(t|\text{not lost})dt$$

由以上順向和反向通道的封包遺失率，可推得來回一趟的封包遺失率為

$$P\{\text{RTT} > \tau\} = \epsilon_F + (1 - \epsilon_F)\epsilon_B + (1 - \epsilon_F)(1 - \epsilon_B) \int_{\tau}^{\infty} p_R(t|\text{not lost})dt$$

其中 p_R 為封包往返時間 RTT(round trip time)的機率密度。以上提到的 p_F 、 p_B 和 p_R 並沒有特定的機率分布模型。

3.5.2 傳送單一多媒體資料單元

在接收端驅動重傳的機制下，讓 r_0, r_1, \dots, r_{N-1} 為 N 個可重傳的機會，各重傳機會的時間間隔為50毫秒。而 a_0, a_1, \dots, a_{N-1} 為相對在各個重傳機會所做的動作，也就是所謂的傳輸策略， $a_i=1$ 代表在 r_i 時要求重傳， $a_i=0$ 則相反。一旦接收端收到該封包後，便忽略後面的動作，不再對該封包要求重傳。此情況下，考慮到有可能傳出去的封包稍微受到延誤，只需再等一小段時間即可收到，故傳輸策略的設定上可先不急著重傳，隔了數個重傳機會仍未收到才再次要求重傳。

在接收端驅動重傳的傳送機制下，傳送單一多媒體資料單元的錯誤率期望值為

$$\prod_{i:a_i=1} P\{\text{RTT} > r_{\text{DTS}} - r_i\}$$

而傳送單一多媒體資料單元的花費期望值為

$$\sum_{i:a_i=1} \left(\prod_{j<i:a_j=1} P\{\text{RTT} > r_i - r_j\} \right) P\{\text{BTT} < \infty\}$$

對於牽涉到反向通道的傳輸機制，Chou將各個時間點要求重傳與否的動作當作馬可夫決策過程(Markov decision process)中各個狀態如何做轉移至下個狀態的依據。透過馬可夫決策過程，便能將錯誤與花費期望值再做進一步的推導。

3.5.3 傳送多媒體資料單元群組

在傳送包含 L 個多媒體資料單元的群組時， π_l 為多媒體資料單元 $l \in \{1, \dots, L\}$ 的傳輸策略，

而 $\boldsymbol{\pi} = \{\pi_1, \dots, \pi_L\}$ 為整個多媒體資料單元群組的傳輸策略。那麼傳送多媒體資料單元群組的頻寬花費期望值為

$$R(\boldsymbol{\pi}) = \sum_l B_l \rho(\pi_l)$$

B_l 為多媒體資料單元 l 的位元組大小， $\rho(\pi_l)$ 為傳輸策略 π_l 的花費期望值。

假設完全沒收到任何多媒體資料單元的視訊失真為 D_0 ， $\epsilon(\pi_{l'})$ 為傳輸策略 $\pi_{l'}$ 的錯誤率期望值，則傳送多媒體資料單元群組的失真期望值為

$$D(\boldsymbol{\pi}) = D_0 - \sum_l \Delta D_l \prod_{l' \leq l} (1 - \epsilon(\pi_{l'}))$$

其中 $l' \leq l$ 代表多媒體資料單元 l 參照到多媒體資料單元 l' ，必須解開 l' 才能順利解開 l 。

3.5.4 傳輸策略最佳化

Chou 將傳輸單一多媒體資料單元的最佳化問題簡化為以 Lagrangian 乘子法找出傳輸策略 π 使得 $J_\pi \equiv \epsilon_\pi + \lambda' \rho_\pi$ 為最小值，解決此問題的手法有許多種，例如透過動態規畫 (dynamic programming) 來達成，在此將這些手法皆稱為 X1 演算法，當解決傳輸多個多媒體資料單元的最佳化問題時會借助 X1 演算法。

傳輸多媒體資料單元群組的最佳化問題同樣簡化為以 Lagrangian 乘子法找出傳輸策略 $\boldsymbol{\pi}$ 使得 $J(\boldsymbol{\pi}) = D(\boldsymbol{\pi}) + \lambda R(\boldsymbol{\pi})$ 為最小值，由於同時要考慮多個 π_l ，便利用 ISA (Iterative Sensitivity Adjustment) 演算法搭配 X1 演算法來找出傳送多媒體資料群組時的最佳傳輸策略組。

第 4 章 估測結果與分析

此章節將會介紹我們所估測影片的詳細資訊，以及其失真與位元率的估測結果。不同的傳輸機制、封包遺失率、傳輸策略的差異皆會討論分析，最後則是傳輸策略最佳化的結果與比較。

4.1 SVC視訊格式設定

我們使用影片Foreman 來做為估測的SVC視訊，且依照[2]中的設定來壓製。以JSVM軟體來壓製視訊，每個GOP有八張畫面，每秒播放張數為 3.75 張至 30 張。雜訊比可調性為CGS (Coarse Granularity Scalability)。以下為該影片的詳細資訊。圖 17為其相依關係與QP設定值。

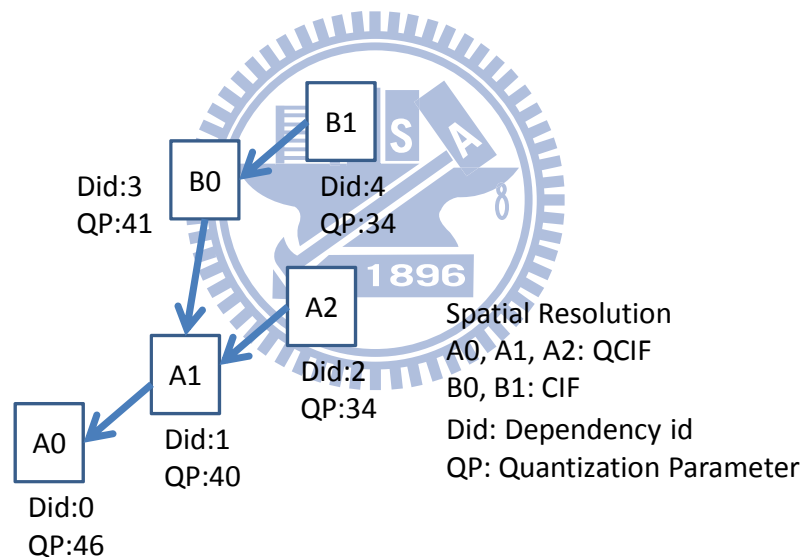


圖 17: SVC 視訊相依關係與 QP 設定

量測各個視訊層表示格式的失真時皆以CIF的視訊尺寸，每秒三十張畫面來測量，在此失真值以平均平方誤差(Mean Square Error, MSE)表示。經過我們的測量，完全沒有收到任何多媒體資料單元而重建的視訊其失真MSE值為 3536.066。我們目標視訊層訂為B1，欲解碼B1所需的其他視訊層之失真值與位元率如表格 3。過小的多媒體資料單元會封裝至同一封包，如 $\mathcal{N}_G(0,0)$ 、 $\mathcal{N}_G(0,1)$ 、 $\mathcal{N}_G(0,2)$ 與 $\mathcal{N}_G(0,3)$ 會裝在同一封包， $\mathcal{N}_G(1,0)$ 、 $\mathcal{N}_G(1,1)$ 、 $\mathcal{N}_G(1,2)$ 與 $\mathcal{N}_G(1,3)$ 會裝在同一封包， $\mathcal{N}_G(3,0)$ 與 $\mathcal{N}_G(3,1)$ 會裝在同一封包， $\mathcal{N}_G(3,2)$ 與 $\mathcal{N}_G(3,3)$

會裝在同一封包。過大的多媒體資料單元會封裝至多個封包，如 $\mathcal{N}_G(4,0)$ 會裝至二個封包。

Did	TL	Rate	Distortion	Δ Rate	Δ Distortion
0	0	8.82	406.894	8.82	3129.172
0	1	12.5927	284.698	3.7727	122.196
0	2	16.5916	222.699	3.9989	61.999
0	3	21.1565	207.431	4.5649	15.268
1	0	22.6151	371.573	13.7951	35.321
1	1	30.4257	232.178	4.0379	17.199
1	2	39.3738	155.451	4.9492	14.728
1	3	49.3427	135.192	5.404	4.991
3	0	54.2822	342.967	31.6671	28.606
3	1	72.1646	181.99	10.0718	21.582
3	2	93.6641	89.4113	12.5514	15.8517
3	3	117.538	62.7922	13.905	6.3601
4	0	143.355	340.457	89.0728	2.51
4	1	185.361	162.513	24.1236	16.967
4	2	234.549	57.0151	27.6885	12.9192
4	3	290.175	23.4258	31.7521	6.9702

表格 3: Foreman 在 CIF 尺寸螢幕每秒播放三十張的失真 MSE 與位元率
這些失真值與相對的位元率可建立成 RD 柵籬圖(Rate-Distortion Trellis)，如圖 18。

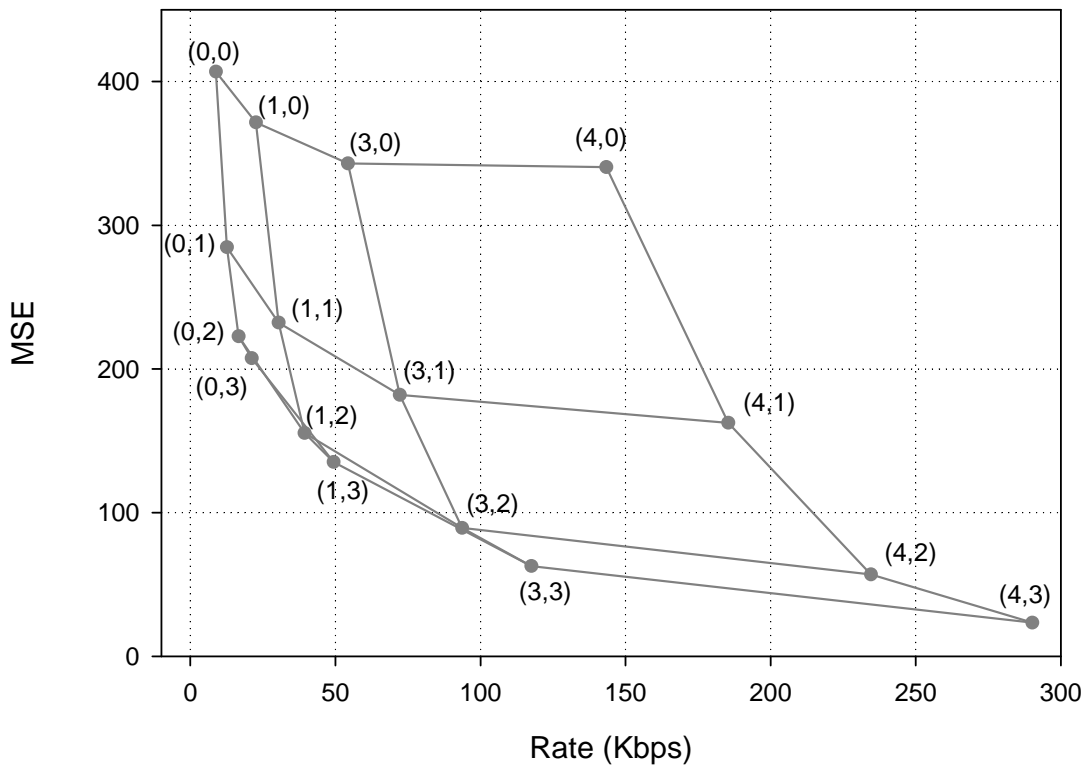


圖 18: 未經傳輸的原始 RD 柵籬圖

4.2 失真與頻寬花費期望值估測

首先我們觀察在接收端驅動重傳的機制下，封包遺失率設定為百分之二十，並且比較不同的傳輸策略所帶來的影響。每個重傳機會的間隔為一秒，並假定因時間上的限制使得重傳機會最多可以有八次。多媒體資料單元群組中每個多媒體資料單元的傳輸策略皆設定成相同的最大重傳次數，比較所有的視訊層最多重傳一次至八次其RD(Rate-Distortion)點變化的差異。由圖 19可看到經過傳輸後的各個RD點所組成的RD柵籬圖皆有不同程度的扭曲與位移。

傳輸策略為最多重傳一次的 RD 柵籬圖，因為頂多重傳一次，多媒體資料單元成功傳送給接收端的機率較低，導致失真較大，向上的位移最大。但也因為只重傳一次，在頻寬花費上不會太多，向右位移的幅度最小。當傳輸策略增加為最多重傳八次，頻寬花費的期望值會升高，RD 柵籬圖的位移往右偏移量最多；多媒體資料單元成功傳送給接收端的機率相較於最多重傳一次來得高，失真期望值較低，最接近未經傳輸的失真值，向上的位移較小。

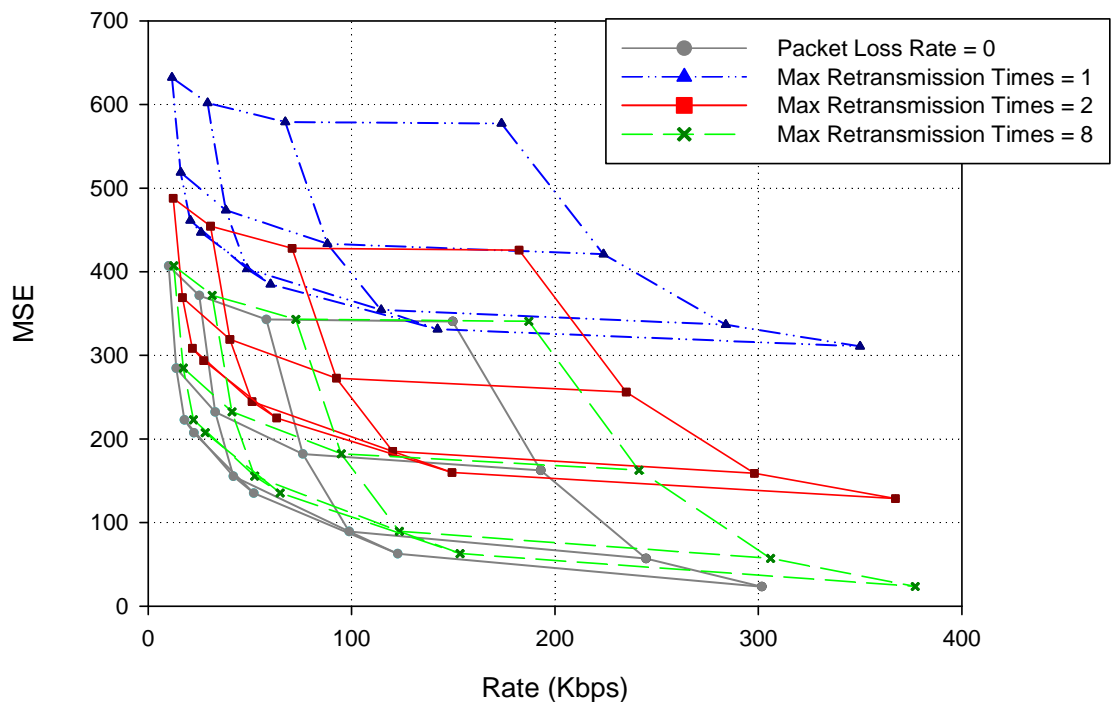


圖 19: 不同傳輸策略於封包遺失率為百分之二十的 RD 柵籬圖

接下來比較不同的視訊層在最多重傳一到八次時RD點的變化，如圖 20。當Did (Dependency id)愈高，則視訊層的變化曲線愈歪斜，代表越大的重傳次數需要愈多的頻寬。而各視訊層從最多重傳一次至最多重傳八次，剛開始失真改善最大，到了最多重傳次數愈大，則失真改善愈不明顯。而頻寬花費期望值的增加量則是隨著最多重傳次數的上升而愈趨減緩。

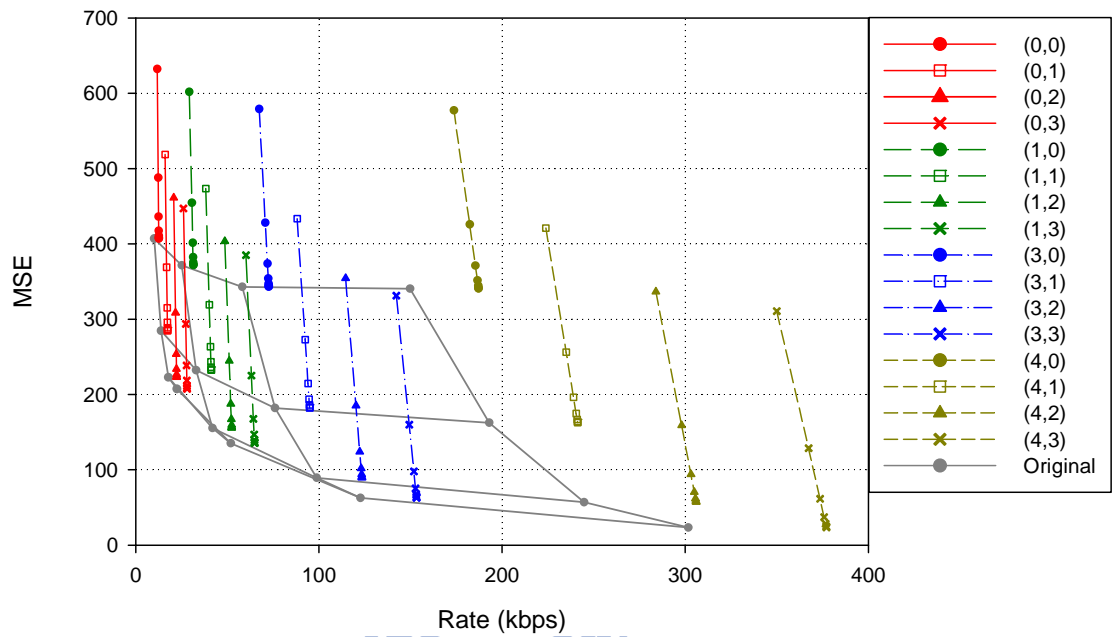


圖 20: 封包遺失率為百分之二十，RD 點於不同最多重傳次數的變化

當考量到實時視訊串流時，由於無法有太長的時間慢慢重傳，因此使用接收端積極重傳的傳輸機制以求在重傳次數不多的情況下達到較好的表現。

我們設計了四種合理的傳輸策略，在最多重傳次數為三次的限制下，並考慮各視訊層的重要性來設定傳輸該視訊層的傳輸策略，如表格 4。

Did	TL	Policy 1	Policy 2	Policy 3	Policy 4
0	0	3	3	3	3
0	1	3	3	3	3
0	2	3	3	3	3
0	3	3	3	3	3
1	0	3	3	3	2
1	1	3	3	3	2
1	2	3	3	3	2
1	3	3	3	3	2
3	0	3	3	2	2
3	1	3	3	2	2
3	2	3	3	2	2
3	3	3	3	2	2
4	0	3	2	1	1
4	1	3	2	1	1
4	2	3	2	1	1
4	3	3	2	1	1

表格 4: 最高重傳次數為三次的四種傳輸策略

傳輸策略一設定為不計頻寬花費而欲使失真值降到最低的策略，或是在封包遺失率較高時所適用的策略。傳輸策略四則是設定為節省頻寬花費，但失真值的表現就沒有傳輸策略一來得好。

接下來分析在接收端驅動積極重傳之下RD變化。在圖 21中，隨著封包遺失率的增加，整個RD 柵籬圖也越往右上方移動。封包遺失率的不同會影響到傳輸單一多媒體資料單元的錯誤率期望值與花費期望值，進而影響到傳輸多媒體資料單元群組的失真與頻寬花費期望值。因此相同的傳輸策略，在不同的封包遺失率之下，其失真與頻寬花費期望值皆會有所變化。

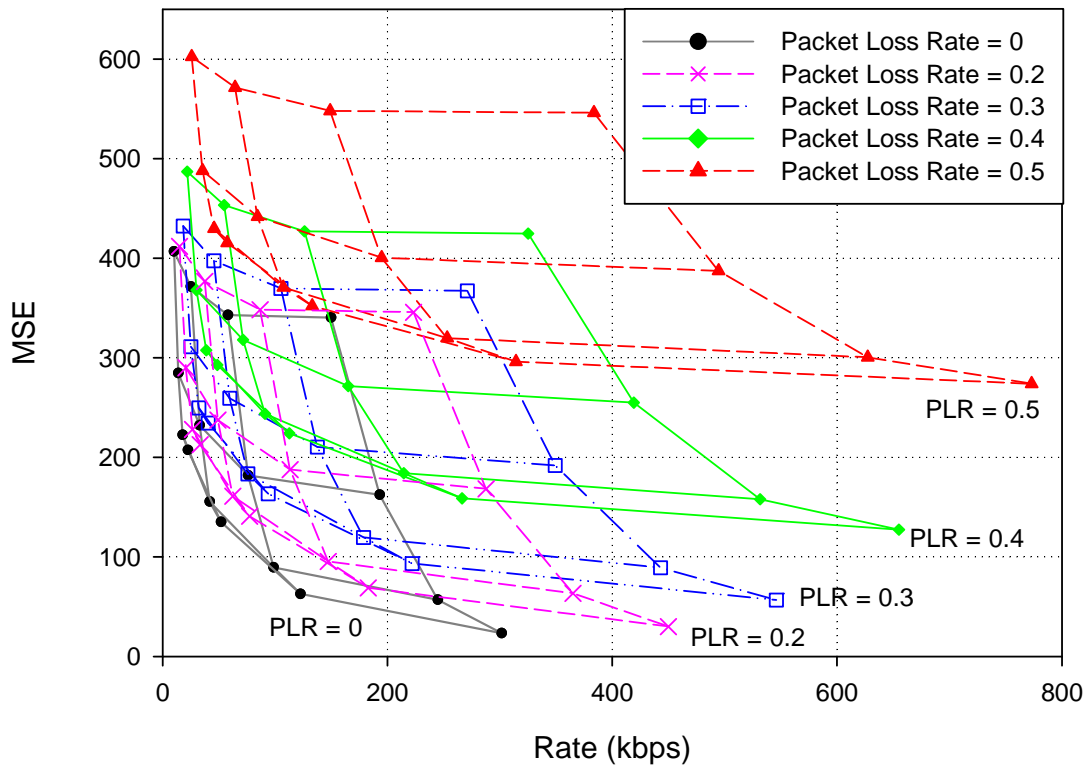


圖 21: 傳輸策略一在不同封包遺失率下的 RD 柵籬圖

四種合理的傳輸策略在不同封包遺失率下的 RD 變化如圖 22、圖 23、圖 24 與圖 25，傳輸策略一和傳輸策略二的結果較相近，而傳輸策略三和傳輸策略四較相近。隨著封包遺失率變大，各傳輸策略的 RD 柵籬圖變化也愈大。

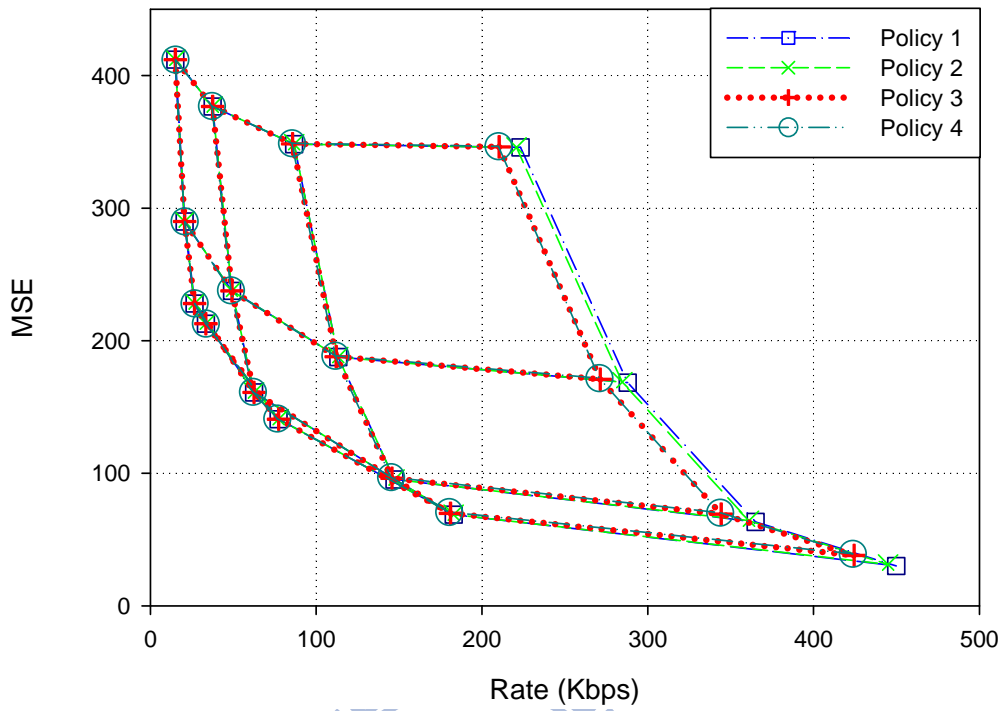


圖 22: 封包遺失率為百分之二十，四種不同傳輸策略的 RD 柵籬圖

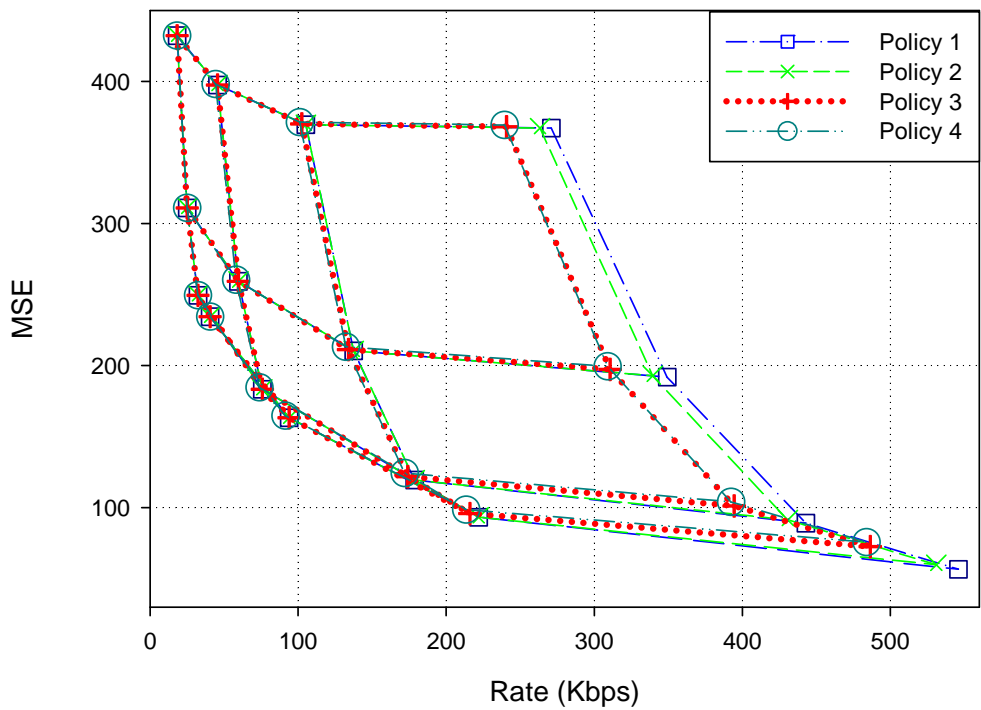


圖 23: 封包遺失率為百分之三十，四種不同傳輸策略的 RD 柵籬圖

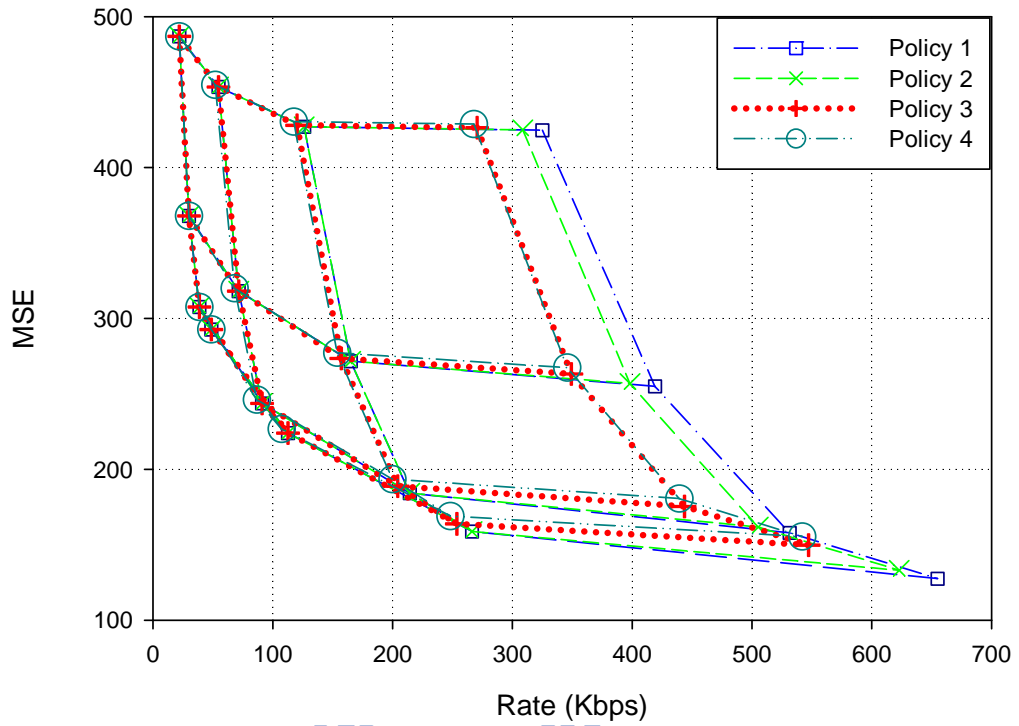


圖 24: 封包遺失率為百分之四十，四種不同傳輸策略的 RD 柵籬圖

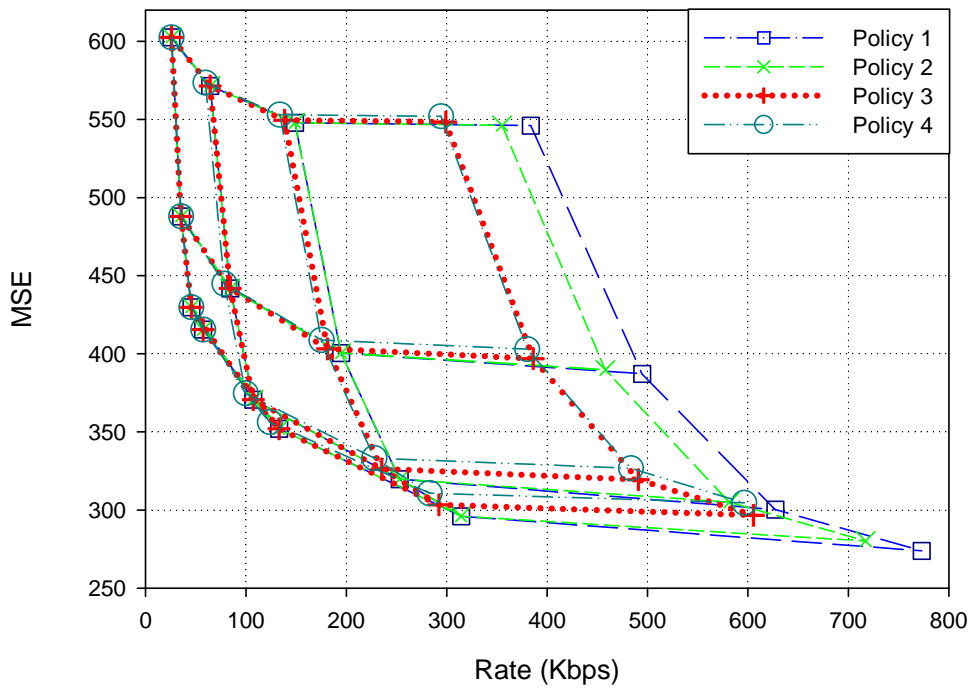


圖 25: 封包遺失率為百分之五十，四種不同傳輸策略的 RD 柵籬圖

4.3 傳輸策略最佳化

當傳輸的 SVC 視訊包含的視訊層愈多，傳輸多媒體資料單元群組時，須考慮的視訊層之傳輸策略也變多，影響傳輸效率的因素變多，欲達成位元率-失真最佳化傳輸並非容易的事。

配合頻寬花費與視訊失真的估測模型，我們建立了找出最佳傳輸策略的演算法，只要給予限制的頻寬，即能找出使用頻寬最有效率的傳輸策略，使得視訊失真降到最低。在傳輸策略最佳化演算法中，由初始完全不重傳的傳輸策略開始，每一步都找出對視訊的失真改善與頻寬花費最有效率的傳輸策略，直到找到最佳傳輸策略。

4.3.1 傳輸策略最佳化演算法 1

此方法採用貪婪演算法的精神，每一步都找最的傳輸策略有效益，最後找出最佳的傳輸策略。首先初始的傳輸策略 $\boldsymbol{\pi}^{(0)} = \{\pi_1^{(0)}, \pi_2^{(0)}, \dots, \pi_L^{(0)}\}$ 為每個視訊層都不重傳，即最大重傳次數為零， $\pi_i^{(0)} = 0$ for $i = [1..L]$ 。將暫時最佳傳輸策略定為初始傳輸策略，接著下一步，假設要增加最大重傳次數的視訊層有 k 種選擇，從目前的傳輸策略演變出 k 種可能的新傳輸策略，評估 k 種傳輸策略之中，頻寬使用不超過給予的限制頻寬，且失真改善量除以頻寬增加量的值為最大的當作這一回的暫時最佳傳輸策略 $\boldsymbol{\pi}^{(1)}$ ，然後以此暫時最佳傳輸策略重複以上的動作，直到每個視訊層的重傳次數達到限制或所有新的傳輸策略皆使用超過限制的頻寬。最後經過 n 次最佳化過程的傳輸策略 $\boldsymbol{\pi}^{(n)}$ ，即為在我們給予的頻寬限制之下，達到位元率-失真最佳化的傳輸策略。

以下為傳輸策略最佳化演算法 1 的虛擬碼：

INPUT	: A given limited bitrate R_{Given}
OUTPUT	: The optimized transmission policy $\hat{\boldsymbol{\pi}}$
Gain:	Distortion improvement divide by bandwidth increase
$S(L_i, T_i)$:	Next decodable scalable layer representation
P :	Possible next policy choices
$\hat{\boldsymbol{\pi}} \leftarrow [0, \dots, 0]$	


```

WHILE  $R_{\hat{\pi}} \leq R_{\text{Given}}$ 
   $P \leftarrow \emptyset$ 
  FOR each scalable layer  $\{l | \hat{\pi}[l] \neq 0\}$ 
     $\pi_l \leftarrow \hat{\pi}$ 
    IF  $\pi_l[l] < \text{Max\_Retransmission\_Times}$ 
       $\pi_l[l] + 1$ 
      IF  $R_{\pi_l} \leq R_{\text{Given}}$  THEN
         $P \leftarrow P \cup \{\pi_l\}$ 
      END IF
    END IF
  END FOR
  FOR each  $S(L_i, T_i), 1 \leq i \leq k$ 
     $\pi_i \leftarrow \hat{\pi}$ 
    FOR each scalable layer  $l \in S(L_i, T_i)$  AND  $\{l | \hat{\pi}[l] = 0\}$ 
      IF  $\pi_i[l] < \text{Max\_Retransmission\_Times}$ 
         $\pi_i[l] + 1$ 
      END IF
    END FOR
    IF  $R_{\pi_i} \leq R_{\text{Given}}$  THEN
       $P \leftarrow P \cup \{\pi_i\}$ 
    END IF
  END IF
  IF  $P \neq \emptyset$  THEN
     $\hat{\pi} \leftarrow \max_{\text{Gain}}\{\pi \in P\}$ 
  ELSE
    RETURN  $\hat{\pi}$ 
  END IF
END WHILE

```

表格 5: 傳輸策略最佳化演算法 1

4.3.2 傳輸策略最佳化演算法 2

此方法同樣採用貪婪演算法。首先初始的傳輸策略 $\boldsymbol{\pi}^{(0)} = \{\pi_1^{(0)}, \pi_2^{(0)}, \dots, \pi_L^{(0)}\}$ 為每個視訊層都不重傳，即最大重傳次數為零， $\pi_i^{(0)} = 0$ for $i = [1..L]$ 。將暫時最佳傳輸策略定為初始傳輸策略，接著分別將 L 個視訊層的重傳次數加一，從目前的傳輸策略演變出 L 種可能的新傳輸策略，評估 L 種傳輸策略之中，頻寬使用不超過給予的限制頻寬，且失

真改善量除以頻寬增加量的值為最大的當作這一回的暫時最佳傳輸策略 $\pi^{(1)}$ ，然後以此暫時最佳傳輸策略重複以上的動作，直到每個視訊層的重傳次數達到限制或所有新的傳輸策略皆使用超過限制的頻寬。最後經過 n 次最佳化過程的傳輸策略 $\pi^{(n)}$ ，即為在我們給予的頻寬限制之下，達到位元率-失真最佳化的傳輸策略。

以下為傳輸策略最佳化演算法 2 的虛擬碼：

```

INPUT    : A given limited bitrate  $R_{\text{Given}}$ 
OUTPUT   : The optimized transmission policy  $\hat{\pi}$ 
Gain:    Distortion improvement divide by bandwidth increase
 $S(L, T)$ : Target layer
 $P$ : Possible next policy choices

 $\hat{\pi} \leftarrow [0, \dots, 0]$ 
WHILE  $R_{\hat{\pi}} \leq R_{\text{Given}}$ 
   $P \leftarrow \emptyset$ 
  FOR each scalable layer  $l \in S(L, T)$ 
     $\pi_l \leftarrow \hat{\pi}$ 
    IF  $\pi_l[l] < \text{Max Retransmission Times}$ 
       $\pi_l[l] + 1$ 
      IF  $R_{\pi_l} \leq R_{\text{Given}}$  THEN
         $P \leftarrow S \cup \{\pi_l\}$ 
      END IF
    END IF
  END FOR
  IF  $P \neq \emptyset$  THEN
     $\hat{\pi} \leftarrow \max_{\text{Gain}}\{\pi \in P\}$ 
  ELSE
    RETURN  $\hat{\pi}$ 
  END IF
END WHILE

```

表格 6: 傳輸策略最佳化演算法 2

4.3.3 傳輸策略最佳化結果與分析

首先我們觀察在不同頻寬限制之下傳輸策略最佳化演算法 2 的結果。圖 26 為頻寬限制從 10 Kbps 至 380 Kbps，以最佳化演算法 2 所找到的最佳傳輸策略所對應的 RD 點。在不同頻寬下的最佳傳輸策略所對的位元率與失真構成的曲線大致上貼近由 SVC 最佳擷取路徑所構成的 RD 曲線。

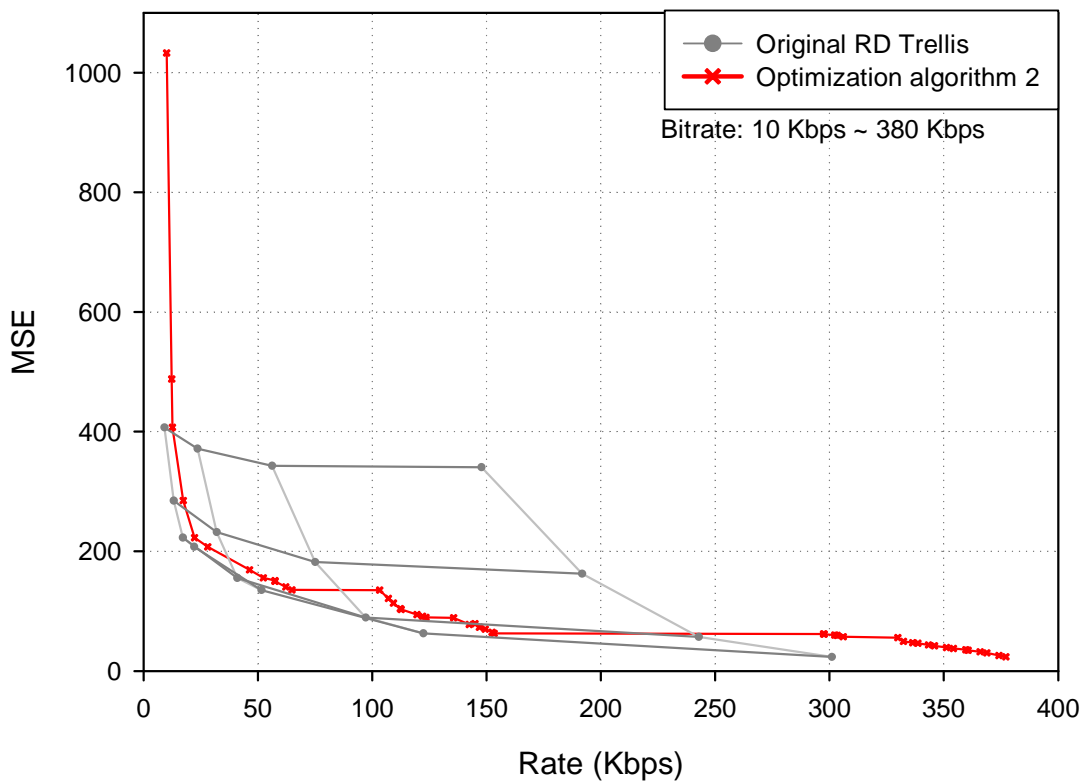


圖 26: 不同位元率限制下，最佳化演算法 2 找出的最佳傳輸策略對應的 RD 曲線

接下來比較兩傳輸策略最佳化演算法的結果差異。兩種演算法最佳化過程中的 RD 點大致上相似，在此我們只呈現有明顯差異的部分。

當目標視訊層為 $S(4,3)$ 時，兩種演算法最佳化過程的部分 RD 曲線如圖 27。

因為不需考慮到 SVC 視訊的擷取路徑，完全以改善失真的量除以頻寬增長的比例來決定要增加哪個視訊層的最多重傳次數。最佳化演算法 2 會先增加 $S(4,1)$ 這個視訊層的最多重傳次數，然後才增加 $S(4,0)$ 的最多重傳次數，原因是因為讓前者多重傳的效益會比

較好。沒有擷取路徑的限制，最佳化演算法 2 表現的比最佳化演算法 1 來得好。

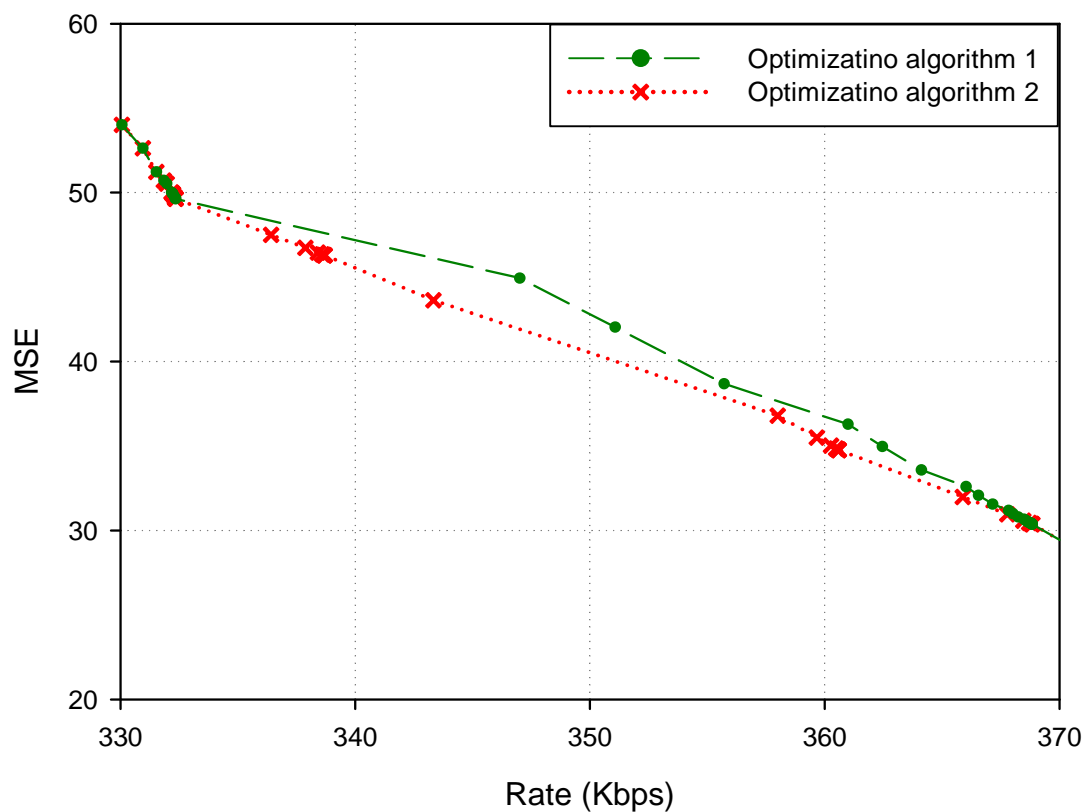


圖 27: 目標視訊層為(4,3)時不同最佳化演算法演化過程中的部分 RD 曲線

除了比較目標視訊層為S(4,3)的最佳化過程，我們還比較了在不同的頻寬限制下，兩種最佳化演算法找到的最佳傳輸策略所對應到的RD曲線。兩種演算法最佳化過程中的RD點大致上相似，在此我們只呈現有明顯差異的部分，如圖 28。在某些頻寬限制下，最佳化演算法 2 所找到的失真值比最佳化演算法 1 的結果還要低。兩最佳化演算法的比較和上面的結果相同，沒有SVC視訊擷取路徑限制的最佳化演算法 2 表現得比最佳化演算法 1 來得好。

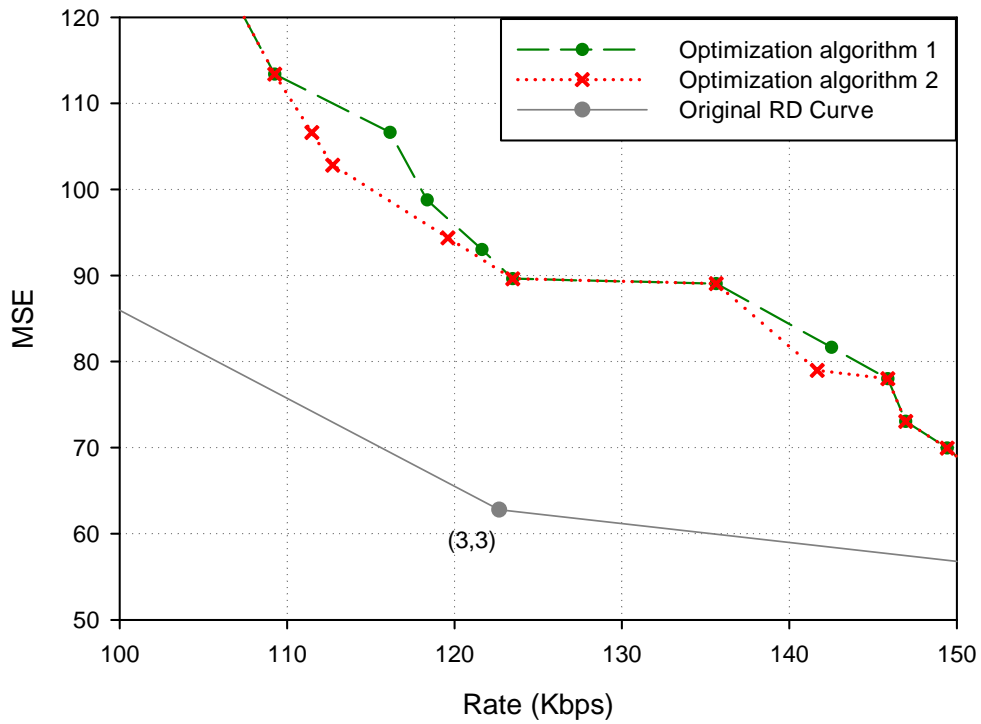


圖 28: 不同位元率限制下，不同最佳化演算法找出的最佳傳輸策略對應的部分 RD 曲線



第 5 章 結語

5.1 研究成果

此研究建立了一個估測失真與位元率期望值的模型，此模型考量了許多實際在網際網路以實時傳輸協議傳輸 SVC 視訊所需注意到的細節，適用的傳輸機制為接收端驅動重傳與接收端積極重傳。且透過此估測模型，我們建立了傳輸策略最佳化的演算法，能有效找出近似最佳的傳輸策略，使得在限制的頻寬下能有最小的失真期望值。

5.2 未來方向

此研究可延伸的方向如下：

- 除了時間限制住最多重傳次數，將分配給重傳的頻寬限制也考慮進去
- 將估測模型由考慮單點傳播延伸為多點傳播
- 將實際的 SVC 視訊來模擬網路傳輸



參考文獻

- [1] P. A. Chou, Z. Miao, “Rate-Distortion Optimized Streaming of Packetized Media,” *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 390–404, Apr. 2006.
- [2] W.-H. Peng, J. K. Zao, H.-T. Huang, T.-W. Wang, and L.-S. Huang, “A rate-distortion optimization model for SVC inter-layer encoding and bitstream extraction,” *J. Vis. Comun. Image Represent.*, vol. 19, no. 8, pp. 543–557, 2008.
- [3] Heiko Schwarz, Detlev Marpe, Thomas Wiegand “Overview of the Scalable Video Coding Extension of the H.264/AVC Standard” *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 17, No. 9, Sep. 2007.
- [4] S. Wenger, Y.-K. Wang and T. Schierl, “RTP payload format for SVC video,” IETF Internet Draft draft-ietf-avt-rtp-svc-19.txt, Sep, 2009.
- [5] S. Wenger, M. M. Hannuksela, T. Stockhammer, M. Westerlund, and D. Singer, “RTP payload format for H.264 video,” IETF RFC 3984, Feb. 2005.
- [6] C.-N. Wang, C.-Y. Tsai, H.-C. Lin, H.-C. Chuang, Y.-C. Lin, J.-H. Chen, K. L. Tong, F.-C. Chang, C.-J. Tsai, S.-Y. Lee, Tihao Chiang, and H.-M. Hang, “FGS-Based Video Streaming Test Bed for MPEG-21 Universal Multimedia Access and Digital Item Adaptation,” *IEEE International Symp. Circuits and Systems*, vol. 2, pp. 364-367, May 2003.

附錄

符號表

$S(L, T)$	一個可調視訊層表示格式(scalable layer representation)，其中 $L = D \parallel Q$ ， D 為 dependency 識別符， Q 為 quality 識別符，而 T 為 temporal 識別符
$\aleph(L, T)$	一個可調視訊層(scalable layer)，其中 $L = D \parallel Q$ ， D 為 dependency 識別符， Q 為 quality 識別符，而 T 為 temporal 識別符
$\aleph_G(L, T)$	視訊層 $\aleph(L, T)$ 在第 G 個 GOP 的多媒體資料單元
$\varsigma(L, T, t)$	視訊層 $\aleph(L, T)$ 在時間 t 播放的 NAL unit
$P_{SL}(i, j)$	傳送 i 個封包而遺失 j 個封包的機率
$P_{RL}(i, j)$	接收端要求重傳 i 個封包而遺失 j 個封包的機率
$F_i(r)$	接收端要求 i 個封包，經過至多 r 次重傳後沒有完全收到 i 個封包的機率
$\epsilon_\pi(r)$	多媒體資料單元封裝成 N 個封包，而傳輸策略 π 最多可重傳 r 次，傳送此單一多媒體資料單元的錯誤機率期望值
$G_i(t)$	接收端要求 i 個封包，於時間 t 重傳的花費期望值。
$G_{i,j}(t)$	傳送 i 個封包而遺失 j 個的情況下，在時間 t 重傳的花費期望值
$\rho(t)$	多媒體資料單元封裝成 N 個封包，傳送此單一多媒體資料單元於時間 t 的花費期望值
$\rho_\pi(r)$	多媒體資料單元封裝成 N 個封包，而傳輸策略 π 最多可重傳 r 次，傳送此單一多媒體資料單元的花費期望值
$D_{(L,T)}(\pi)$	以傳輸策略 π 來傳輸目標視訊層為 $S(L, T)$ 的多媒體資料單元群組之失真期望值
D_0	完全沒收到任何多媒體資料單元時的失真 MSE 值
$\mathcal{C}_{(L,T)}$	成功接收卻未改善失真多媒體資料單元之失真補償值
$\Lambda_{(L,T)}$	目標視訊層為 $S(L, T)$ 時，因部分多媒體資料單元遺失而出現多個可解的視訊

	層表示格式的所有事件
λ	$\Lambda_{(L,T)}$ 中的一個事件
λ_r	在事件 λ 中收到的多媒體資料單元的集合
P_λ	發生事件 λ 的機率
$R_{(L,T)}(\boldsymbol{\pi})$	以傳輸策略 $\boldsymbol{\pi}$ 來傳輸目標視訊層為 $S(L,T)$ 的多媒體資料單元群組之頻寬花費期望值
$B_{(L,T)}$	多媒體資料單元 $\mathfrak{N}_G(L,T)$ 的位元組大小
H	封包頭的位元組大小
$N_{(L,T)}$	多媒體資料單元 $\mathfrak{N}_G(L,T)$ 所封裝成 RTP 封包的數量
$\rho_{\pi(L,T)}$	$\mathfrak{N}_G(L,T)$ 以傳輸策略 $\pi(L,T)$ 傳輸時的花費期望值
$\epsilon_{\pi(L,T)}$	$\mathfrak{N}_G(L,T)$ 以傳輸策略 $\pi(L,T)$ 傳輸時的錯誤率
$\hat{\epsilon}_{\pi(L,T)}$	$\mathfrak{N}_G(L,T)$ 以傳輸策略 $\pi(L,T)$ 傳輸時特定一張畫面的錯誤率

