

# 在 NEMO 環境下群組式網路編碼

李龍盛

國立嘉義大學資訊工程系  
sheng@mail.ncyu.edu.tw

李育德

國立嘉義大學資訊工程系  
s1010422@mail.ncyu.edu.tw

## 摘要

近年來由於攜帶型裝置的盛行，以及網路編碼成為許多研究學者關注的議題，因此我們根據網路編碼的理論，提出了一個新的方法，我們稱為群組編碼，並且實作在行動網路上，透過將網路編碼做改良，由節點路由器透過來源位址和目的位址配對編碼兩個訊框，我們透過編碼增益和時間複雜度來做評估，結果顯示，透過群組編碼，在訊框數較多的情況下，能夠有較好的執行效益，且大大的降低節點路由器的運算負擔。

**關鍵詞：**網路編碼、群組編碼、行動網路。

## Abstract

In recent years, due to the prevalence of portable devices and network coding be the subject of attention of many researchers. So, we based on network coding theory, proposed a new approach. We called "The Auto Network Coding(ANC)" and implemented on a network mobility, through the improvement of network coding by the node router intelligent mix to packets. We evaluate by the coding gain and time complexity. The results showed that the auto network coding(ANC), in the case of large number of packets can be better throughput, and greatly reduce the calculation load of the node router.

**Keywords:** network coding, the auto network coding(ANC), network mobility.

## 1. 前言

近年來由於行動式裝置的盛行，對於無線網路的需求越來越高，而無線網路卻遭受低流通量的問題，在先前的研究中，已有研究學者提出網路編碼 COPE[1]方法，來改進流通量的問題，原始網路編碼方法主要理論說明在圖

1。

假設一個狀況，Alice 和 Bob 想要交換一對訊框經由一個節點路由器，Alice 必須發送訊框到節點路由器，再由節點路由器轉發訊框到 Bob，且 Bob 也一樣必須透過節點路由器來轉發訊框到 Alice，因此 Alice 和 Bob 交換一對訊框需要 4 次的傳輸，而 COPE 提出一個方法，透過節點路由器來智能混合編碼訊框，對訊框做 XORs 運算，之後廣播編碼完成的訊框，因此 Alice 和 Bob 可以獲得彼此的訊框，藉由利用自己所傳送出去的訊框再做一次 XORs 運算，這樣就可以用 3 次的傳輸來代替原始 4 次的傳輸。

然而在 COPE[1]方法，為了能實現較大的網路流通量，必須讓節點監聽所有傳輸且儲存所監聽到的訊框在一個短暫的期間，但這將會讓節點路由器的運算負擔加重。

在本文，我們希望將網路編碼實作在行動網路，因為目前市面上的行動式裝置配備，比較無法處理一個較大量的複雜運算，因此，我們提出一個群組編碼方法，將網路編碼進行改良，以兩個訊框作為一個群組，因此節點路由器，被控制在每次只編碼兩個訊框，這將大大的降低節點路由器的運算負擔，並且在訊框數較多時，也能夠有較好的編碼增益。

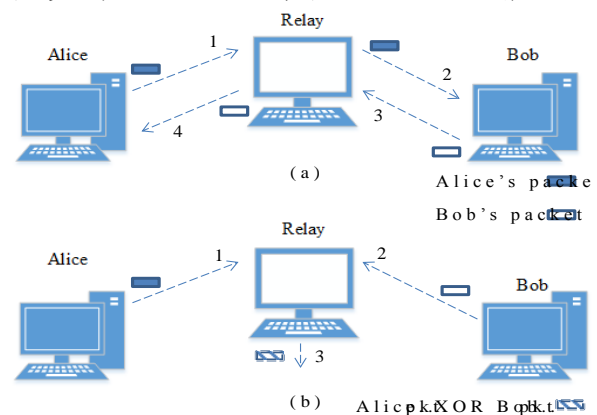


圖 1 如何增加網路流通量(a)原始方法(b)網路編碼方法

## 2. 相關工作

網路編碼的研究起於 Ahlswede[2]等人的研究，文中顯示了路由器混合資訊在不同的訊息中，允許溝通透過路由器來實現多播能力，而在 Li[3]等人的研究中說明，線性編碼可以達到最大容量的傳輸上載和下載，而 Koetter 和 Medard[4]提出了多項式時間演算法來讓路由器做編碼和解碼，Ho[5]等人延伸了這些結果來做隨機編碼，然後在 Lun[6]等人的研究得出內部交談網路編碼最小化的溝通問題可以被配置作為線性規劃然後以分佈式的方式來解決，而在[7]得知，線性編碼是不足以來最佳化誇交談網路編碼，在[1]中致力於開發一個實際尋優式功能的網路編碼理論並提供操作協定，而 Wu[8]等人提出了一個演算法來作佳化混合訊框，Liu[9]等人分析理論網路流通量增益，被用來延伸到物理層[10][11][12][13]。

而上述這些研究都致力於如何才能獲得最高的網路流通量增益，並且實作在無線網狀網路中，而本文透過將網路編碼來實作在行動網路中，因為是實作在行動網路上，因此上述的方法套用在攜帶式裝置上時，將會讓攜帶式裝置有較大的負擔，因此本文將網路編碼作改良，透過我們的群組編碼方法，將可以讓攜帶式裝置得到網路編碼的網路流通量增益，並且大大的降低攜帶式裝置的運算負擔。

## 3. 群組編碼

在我們的方法中，透過 Source Address 與 Destination Address 配對方式，修改原始 IEEE 802.11 訊框如圖 2。

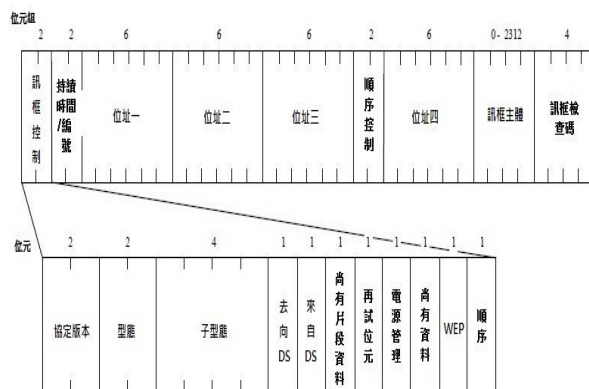


圖 2 原始 IEEE 802.11 訊框

我們將原始 IEEE 802.11 訊框新增欄位，並使用保留訊框如圖 4，且使用一個虛擬佇列，來達到群組編碼方法。

在圖 4 中，我們新增了位址五、位址六、

位址七、位址八和位址九，其中位址五固定填入節點自己的 MAC 位址，我們將資料訊框中子型態的保留欄位 1000 和 1001 用作判斷編碼訊框，若收到訊框的資料訊框子型態保留欄位設為 1000 代表為已編碼訊框，若為 1001 則代表未編碼訊框，在表 1 我們新增欄位，位址六和位址七存放該解碼編碼訊框節點的 MAC 位址，若為位址七，則可從位址二和位址三得知真正的目的端和來源端，若為位址六，則可從位址八和位址九得知真正的目的端和來源端，在下面方法我們將使用這些欄位來達到我們群組編碼的目的。

表 1 新增欄位說明

訊框解碼節點對應目的端和來源端	目的端 (原始訊框)	來源端 (原始訊框)
位址七	位址二	位址三
位址六	位址八	位址九

### 3.1 傳送訊框流程

在節點要傳送訊框時(包含轉送)，根據原始 IEEE 802.11 規則，根據 ToDS、FromDS 欄位如表 2，來決定資料訊框中的位址一、位址二、位址三和位址四。

表 2 欄位填放規則

	去向 DS	來自 DS	位址一 (接收端)	位址二 (傳送端)	位址三	位址四
IBSS	0	0	DA	SA	BSSID	不使用
來自 AP	0	1	DA	BSSID	SA	不使用
去向 AP	1	0	BSSID	SA	DA	不使用
WDS (橋接器)	1	1	RA	TA	DA	SA

之後將自己的 MAC 位址填入位址五欄位，並將此訊框儲存到虛擬佇列後，送出訊框，圖 3 為傳送流程圖。

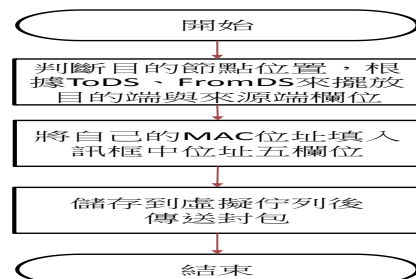


圖 3 傳送流程圖

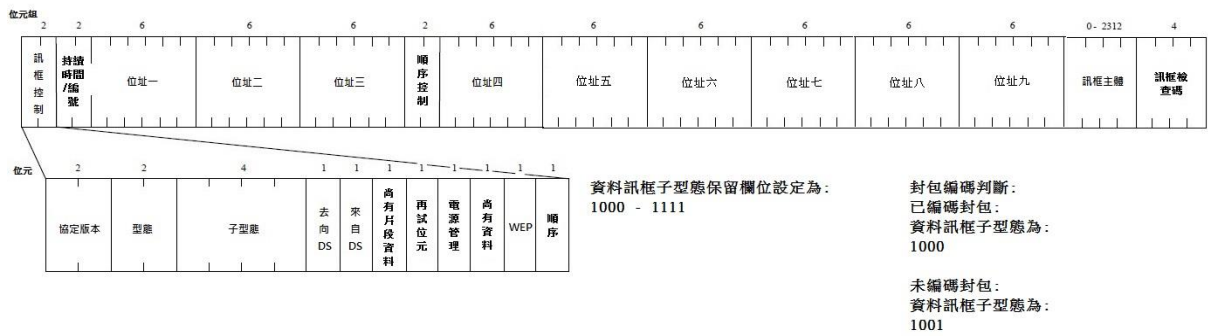


圖 4 群組編碼訊框

### 3.2 接收訊框流程

當節點接收到訊框後先查看判斷欄位，看是否為編碼訊框，如果是，則進行解碼，解碼完畢後，如果自己的 MAC 位址在位址六則看位址八來判斷自己是否為真正的目的端，如果自己的 MAC 位址在位址七則看位址二，若自己的 MAC 欄位在位址八或位址二，則收下訊框，接下來必須判斷是由哪個節點送來的訊框，如果自己的 MAC 位址在位址六，可從位址九得知誰送來的，如果自己的 MAC 位址在位址七，可從位址三得知誰送來，若位址八和位址二都不是自己的 MAC 欄位，則代表自己是轉送節點，新的訊框位址五填入自己的 MAC 位址，如果自己的 MAC 位址在位址六，則將位址八作為目的端，位址九作為來源端，如果自己的 MAC 位址在位址七，則將位址二作為目的端，位址三作為來源端，完成訊框後，進入群組編碼方法，而若此訊框不是編碼訊框，則先根據 ToDS、FromDS 查看目的端欄位是否為自己，是，則收下訊框，否，則進入群組編碼方法，接收流程圖如圖 5。

### 3.3 群組編碼

當進入群組編碼方法，首先檢查虛擬佇列裡，是否有任一訊框來源位址欄位和收到訊框中的目的位址欄位相同，且此虛擬佇列訊框目的位址欄位是否和收到訊框訊框中的來源位址欄位相同，若虛擬佇列內訊框都沒有相同，則將此訊框儲存在虛擬佇列後，直接傳送訊框，若都相同，則等待來源位址與此訊框目的位址相同，且目的位址與此訊框來源位址相同的訊框到來，此時只要有訊框進來則先判斷是否為編碼訊框，不是的話先檢查此訊框來源位址，是否和等待編碼訊框目的位址相同，且目的位址與等待編碼訊框來源位址相同，若都沒有相同的訊框，且超過等待訊框編碼時間，將此訊框儲存在虛擬佇列後，直接傳送訊框，

若相同，則進行編碼，以 IBSS 的方式去傳送 (ToDS=0,FromDS=1)，將第一個收到訊框的位址五和第二個收到訊框的位址五放到新訊框的位址六和位址七，將位址六的目的端放在位址八，來源端放在位址九，將位址七的目的端放在位址二，來源端放在位址三，將資料部分以 XORs 運算進行編碼，新訊框的資料訊框子形態設為 1000，將兩訊框原始訊框儲存在虛擬佇列，之後廣播訊框，群組編碼流程如圖 6。

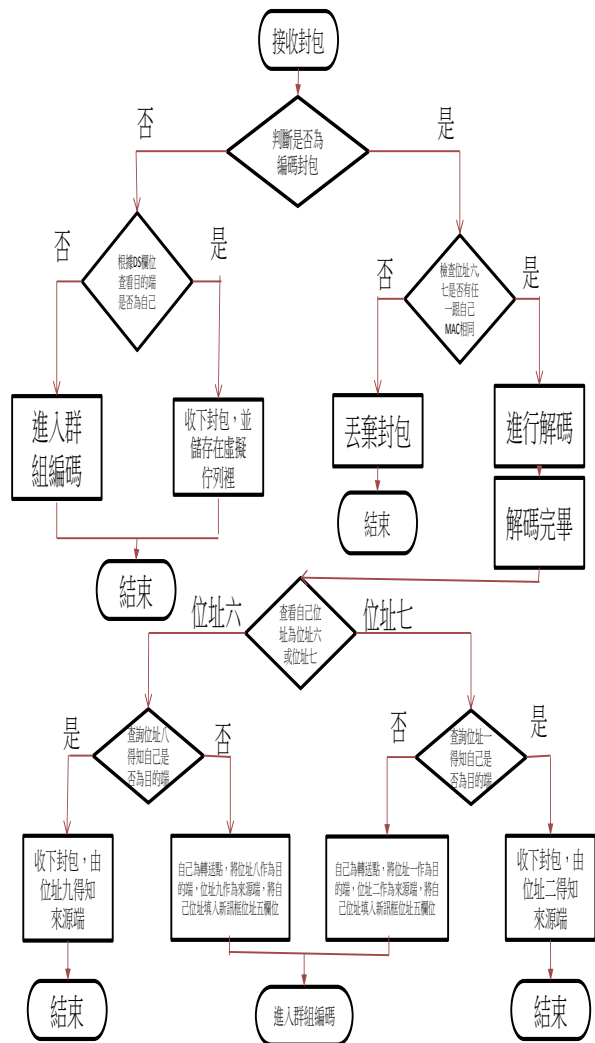


圖 5 接收流程圖



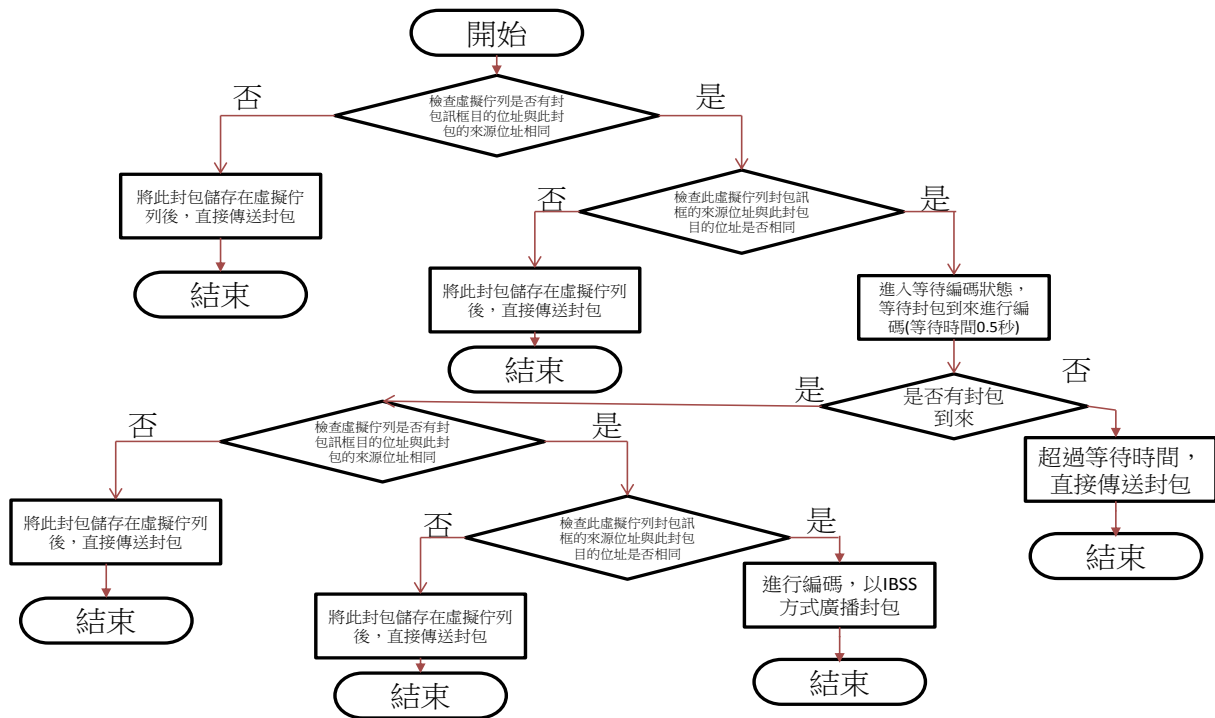


圖 6 群組編碼流程圖

### 3.4 群組編碼模擬

在這個部分，我們將以圖 7 來模擬說明我們所提出的群組編碼方法，在中圖 7 我們假設節點 X 與節點 Y 溝通，我們這邊主要將節點 B 作為編碼節點來做說明，且圖中的訊框位址一到位址四的擺放規則，根據表 2，IEEE 802.11 規則擺放，而我們假設節點 X 已有傳送一訊框 M1 傳送至節點 Y，並在節點 A、B 和 C 中虛擬佇列裡儲存此訊框 M1，由於 M1 訊框經由節點 A 做轉傳至節點 B，因此，在節點 B 中的 M1 訊框的訊框位址五為節點 A 的 MAC 位址如圖 7(a)，而此時，節點 Y 要發送一訊框 N1 給節點 X 如圖 7(b)，N1 訊框經由節點 C，因此節點 C 儲存 N1 訊框到虛擬佇列中，而在轉送的過程中經過節點 B 時，節點 B 收到 N1 訊框，先檢查此訊框是否為編碼訊框，而此訊框不是編碼訊框，因此節點 B 檢查此訊框目的位址是否為節點 B 自己，是的話則收下訊框，並儲存在虛擬佇列，而此訊框 N1 目的位址為節點 X，因此節點 B 進入群組編碼方法，首先節點 B 先檢查虛擬佇列裡是否有訊框目的位址與 N1 訊框來源位址相同，我們可以看到在節點 B 虛擬佇列裡的 M1 訊框的目的位址與收到的 N1 訊框的來源位址都同為 Y，因此在檢查節點 B 虛擬佇列裡 M1 訊框的來源位址與收到的 N1 訊框的目的位址是否相同，我們可以看到 M1 訊框的來源位址和 N1 訊框的目的位址都同為

X，因此進入等待編碼狀態，此時只要有訊框到達節點 B，節點 B 都會先檢查是否到達節點 B 的訊框可以跟目前等待編碼的 N1 訊框進行編碼，我們可以看到在圖 7(c)，節點 X 要發送一訊框 M2 到節點 Y，我們可以看到當此訊框 M2 從節點 A 要轉送到節點 B 時的訊框 M2 位址五為節點 A 的 MAC 位址並儲存 M2 到節點 A 虛擬佇列，因為我們的方法在做轉送或傳送時都必須將自己的 MAC 位址填入到位址五，我們看到圖 7(d)，當節點 B 收到 M2 訊框後，先檢查此訊框的目的位址是否和等待的 N1 訊框來源位址相同，可以由圖 7(d)看到同為 Y，因此在檢查 M2 訊框的來源位址，是否和等待編碼的 N1 訊框目的位址相同，可以看到圖 7(d)都同為 X，因此將 M2 訊框，與 N1 訊框進行編碼，我們看到圖 7(e)中，我們先將 N1 訊框中的目的端位址和來源端位址根據表 2 IEEE 802.11 規則放入新訊框的位址一到位址四，並將 N1 訊框的位址五填入新訊框的位址六，然後將 M2 訊框的目的位址填入新訊框的位址八，來源位址填入新訊框的位址九，並將 M2 訊框的位址五欄位，填入新訊框的位址七，節點 B 將自己的 MAC 位址填入新訊框的位址五，之後將新訊框的資料訊框子形態設為 1000 代表為編碼訊框，最後將 M2 訊框和 N1 訊框的資料部分進行 XORs 運算，節點 B 將 M2 訊框和 N1 訊框儲存在虛擬佇列後，節點 B 將編碼完畢後的訊框以 IBSS 方式做廣播。

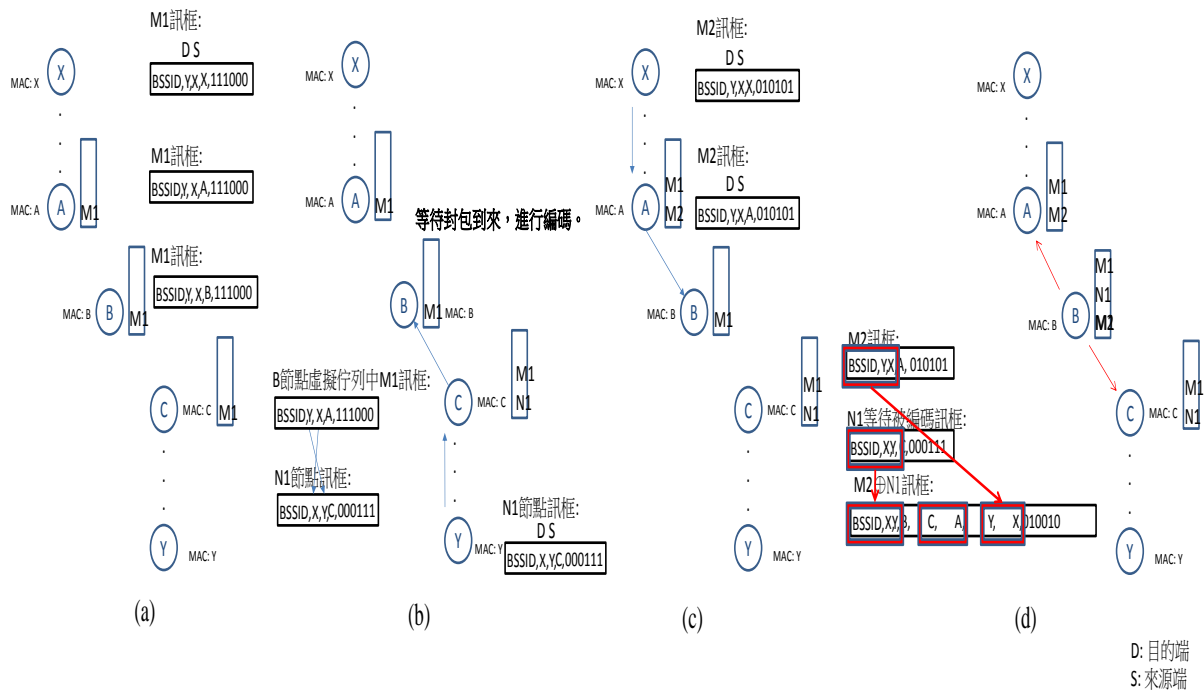


圖 7 群組編碼模擬說明

接下來我們透過圖 8 的部分來說明當節點接收到已編碼訊框後的解碼動作。

在圖 8 (a)中，為節點 A 收到廣播訊框，當節點 A 收到廣播訊框後，首先檢查是否為編碼訊框，若為編碼訊框，先檢查位址六或位址七其中一欄位是否為自己的 MAC 位址，由圖 8 (a)中可以看到廣播訊框的位址七為節點 A 的 MAC 位址，因此，將此廣播訊框進行解碼，節點 A 從虛擬佇列裡找出剛剛所傳送出去的 M2 訊框資料，與此廣播訊框資料做 XORs 運算進行解碼，得到資料 000111，節點 A 接下來根據表 1，檢查位址二是否為節點 A 的 MAC 位址，由圖 8 (a)中可以看到廣播訊框的位址二為節點 X 的 MAC 位址，因此節點 A 準備進行做轉傳，節點 A 根據表 1 得知解碼後訊框的來源端為節點 Y，目的端節點 X，以及資料為 000111，因此，將此做為轉傳訊框的依據，進行轉傳，如圖 8 (b)。

在圖 8 (c)中，為節點 C 收到廣播訊框，當節點 C 收到廣播訊框後，首先檢查是否為編碼訊框，若為編碼訊框，先檢查位址六或位址七其中一欄位是否為自己的 MAC 位址，由圖 8 (c)中可以看到廣播訊框的位址六為節點 C 的 MAC 位址，因此，將此廣播訊框進行解碼，節點 C 從虛擬佇列裡找出剛剛所傳送出去的 N1 訊框資料，與此廣播訊框資料做 XORs 運算進行解碼，得到資料 010101，節點 C 接下來

根據表 1，檢查位址八是否為節點 C 的 MAC 位址，由圖 8 (c)中可以看到廣播訊框的位址八為節點 Y 的 MAC 位址，因此節點 C 準備進行做轉傳，節點 C 根據表 1 得知解碼後訊框的來源端為節點 X，目的端為節點 Y，以及資料為 010101，因此，將此做為轉傳訊框的依據，進行轉傳，如圖 8 (d)。

#### 4. 評估

在這一節，我們評估透過編碼增益(Coding Gain)[1]和節點路由器運算處理的時間複雜度所需承受負載來做評估。

##### 4.1 編碼增益(Coding Gain)

我們評估編碼增益(Coding Gain,CG)，計算編碼所獲得的效益，透過以未使用過編碼的傳輸次數( $\alpha$ )，除以使用編碼的傳輸次數( $\beta$ )，來得到經過編碼後的效益如式 1 所示。

式 1:編碼增益(Coding Gain)

$$CG = \frac{\alpha}{\beta} \dots\dots\dots(1)$$

在編碼增益(Coding Gain)評估中，我們設定每個訊框傳輸需花費 0.1 秒，且假設 COPE[1]每要多編碼一個訊框，虛擬佇列需多等待 0.1 秒，而群組編碼為固定編碼兩個訊框，因此組編碼固定花費編碼秒數為 0.2 秒。

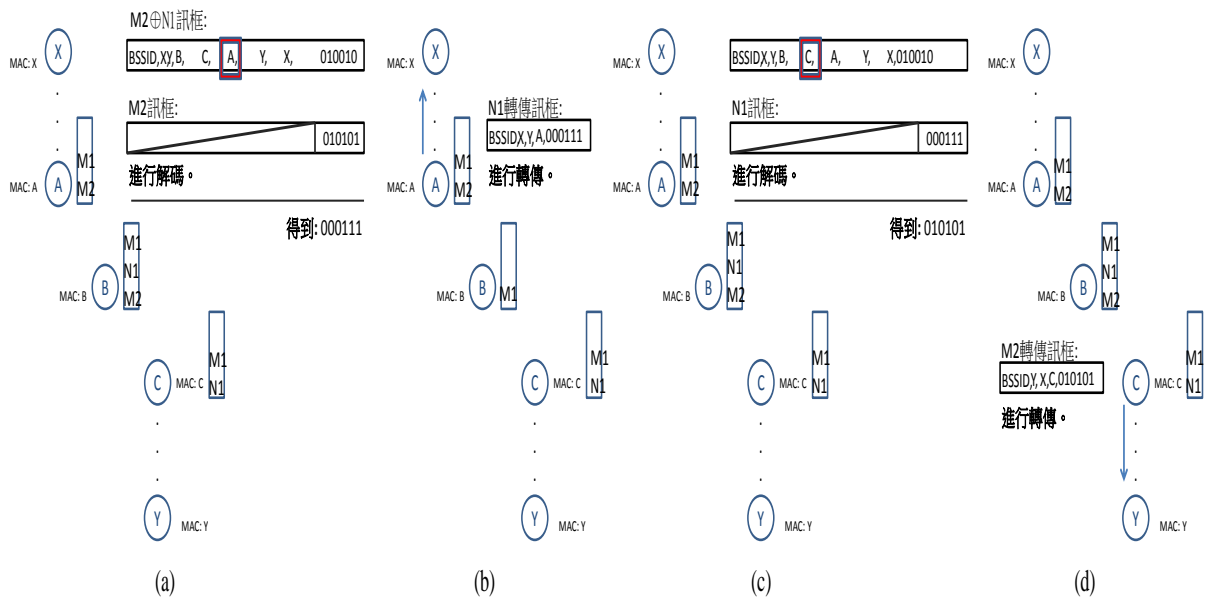


圖 8 解碼編碼訊框

我們的編碼增益模擬拓樸如圖 9，每個節點發送一訊框與對角線節點做溝通，模擬節點數為 2、4、6、8、10。

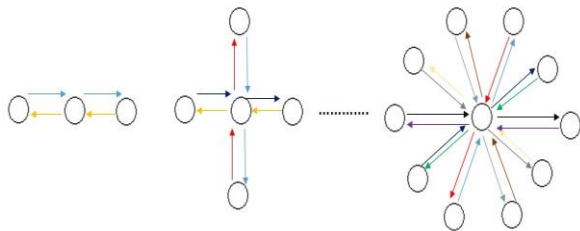


圖 9 編碼增益模擬拓樸

我們可以從圖 10 看到當節點數增加時群組編碼所獲的編碼增益(Coding Gain)，優於 COPE 編碼方法。

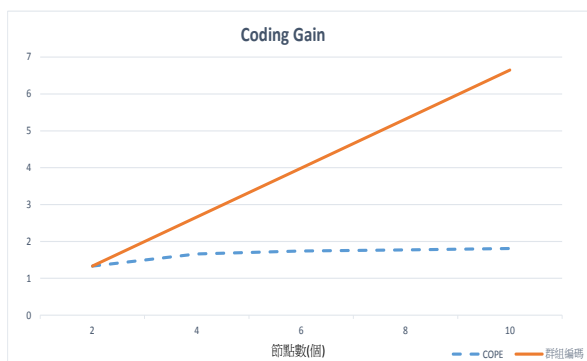


圖 10 編碼增益評估，X 軸為節點數，Y 軸為編碼增益

#### 4.2 節點路由器運算時間複雜度

我們計算時間複雜度透過頻率計數如圖 11，藉由頻率計數的方式來計算 COPE[1]方法的時間複雜度，以及我們群組編碼的時間複雜度，來比較節點路由器運算處理的所需負載程度。

我們首先對 COPE 的編碼程序做頻率計數計算如圖 13，由圖 13 得知，COPE 的編碼程序的頻率計數為  $2N^2 + 4N + 11$ ，接下來我們對群組編碼程序做頻率計數計算，如圖 12，由圖 12 得知，群組編碼程序的頻率計數為  $N + 7$ ，因此我們總結 COPE 和群組編碼頻率計數如表 3，由表 3 得知，COPE 節點路由器每次的編碼動作，都必須比群組編碼來的多負載  $2N^2 + 3N + 4$  的執行時間(N 指的是虛擬佇列長度)。

種類	頻率計數	說明
註解	0	註解是不可執行的程式碼
宣告程式碼	0	宣告程式碼也是不可執行的程式碼，例如：變數宣告，程式區塊的大括號
指定敘述和運算式	1	不含函數呼叫的指定敘述和運算式
執行函數	1	函數執行次數是 1，遞迴函數需視其遞迴執行的次數而定
條件敘述	視情況	if、switch 條件敘述的頻率計數是條件運算式本身的判斷次數，和程式區塊中程式碼頻率計算的總和
迴圈敘述	視情況	for、while 和 do/while 迴圈敘述的頻率計數分為控制部分(即迴圈條件和程式區塊，如果迴圈執行 n 次，則控制部分為 n+1 次(多一次結束迴圈的判斷)，程式區塊是 n 次)

圖 11 頻率計數說明

```

1. PN:原始封包
2. Pi:取得虛擬佇列中的封包
3. PE:編碼封包
4. PN.D:原始封包目的位址
5. PN.S:原始封包來源位址
6. Pi.D:取得虛擬佇列中封包目的位址
7. Pi.S:取得虛擬佇列中封包來源位址
8. PE.D:原始封包目的位址
9. PE.S:原始封包來源位址
10. Queue.length:虛擬佇列長度
11. judge: 用來判斷是否找到可編碼封包，一箇布林值
12. Virtual queue Q(PN):尋找虛擬佇列封包目的位址與原始封包來源位址相同封包，回傳一封信包
13. do
14. {
15. if (i < Queue.length) && (PE == NULL) then
16.   Pi = Virtual queue Q(PN)
17.   if PN.D == Pi.S then
18.     PE = PN @ Pi
19.     judge = true
20.   end if
21. end if
22. i++
23. }while(judge);
24. return PE

```

列數	計數
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	N/2
14	0
15	N/2
16	2
17	1
18	1
19	1
20	0
21	0
22	1
23	0
24	1

頻率計數總和:  
N+7

圖 12 群組編碼程序頻率計數

1 Coding Procedure

```

1. Pick packet p at the head of the output queue.
2. Natives = {p}
3. Nexthops = {nextHop(p)}
4. if size(p) > 100 bytes then
5.   which.queue = 1
6. else
7.   which.queue = 0
8. end if
9. for Neighbor i = 1 to M do
10.  Pick packet pi, the head of virtual queue Q(i, which.queue)
11.  if ∀n ∈ Nexthops ∪ {i}, Pr[n can decode p ⊕ pi] ≥ G then
12.    p = p ⊕ pi
13.    Natives = Natives ∪ {pi}
14.    Nexthops = Nexthops ∪ {i}
15.  end if
16. end for
17. which.queue = !which.queue
18. for Neighbor i = 1 to M do
19.  Pick packet pi, the head of virtual queue Q(i, which.queue)
20.  if ∀n ∈ Nexthops ∪ {i}, Pr[n can decode p ⊕ pi] ≥ G then
21.    p = p ⊕ pi
22.    Natives = Natives ∪ {pi}
23.    Nexthops = Nexthops ∪ {i}
24.  end if
25. end for
26. return p

```

列數	計數
1	0
2	1
3	2
4	1
5	1
6	0
7	0
8	0
9	N+1
10	N <sup>2</sup>
11	N
12	1
13	1
14	1
15	0
16	0
17	1
18	N+1
19	N <sup>2</sup>
20	N
21	1
22	1
23	1
24	0
25	0
26	1

頻率計數總和:  
 $2N^2+4N+11$

2、4、6、8、10。



圖 14 時間複雜度模擬拓樸

我們可以從圖 15 得知，當傳遞的訊框數越來越多時，群組編碼節點路由器所需承擔負載的運算遠低於 COPE 方法。

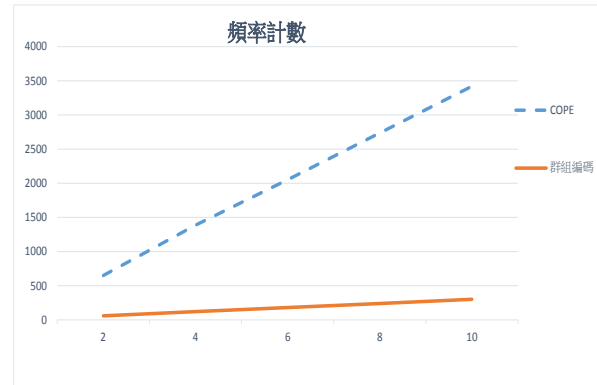


圖 15 頻率計數評估，X 軸為各節點傳遞訊框數，Y 軸為頻率計數之時間複雜度

圖 13 COPE 編碼程序頻率計數

表 3 群組編碼與 COPE 頻率計數，N 為虛擬佇列長度

群組編碼	COPE
N+7	$2N^2+4N+11$

我們評估節點路由器運算處理的時間複雜度所需承受負載拓樸如圖 14，並假設中間為節點路由器，虛擬佇列已有八個訊框可進行選擇編碼(N 設為 8)，且節點傳遞訊框數分別為

### 5. 結論

在我們所設計的群組編碼方法，若在越多的節點數以及越多訊框數的溝通下，將會有較好的網路流通率，而對於原始網路編碼節點路由器所需運算處理的負載程度而言，群組編碼方法大大的降低節點路由器所需運算處理的負載程度，而這將讓網路編碼理論透過我們的群組編碼方法，更優於 COPE 實作在行動網路上。

## 參考文獻

- [1] Katti, Sachin., Rahul, Hariharan., Wenjun, Hu., Katabi, Dina., Médard, Murie., Senior Member, IEEE, and Crowcroft, Jon., Fellow, IEEE, “**XORs in the Air: Practical Wireless Network Coding**”, IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 16, NO. 3, JUNE 2008”.
- [2] Ahlswede, R., Cai, N., Li, S. R., and Yeung, R. W., “**Network information flow**,” IEEE Trans. Inf. Theory, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.
- [3] Li, S. R., Yeung, R. W., and Cai, N., “**Linear network coding**,” IEEE Trans. Inf. Theory, vol. 49, no. 2, pp. 371–381, Feb. 2003.
- [4] Koetter, R. and Médard, M., “**An algebraic approach to network coding**,” IEEE/ACM Trans. Networking, vol. 11, no. 5, pp. 782–795, Oct. 2003.
- [5] Ho, T., Médard, M., Shi, J., Effros, M., and Karger, D., “**On randomized network coding**,” presented at the 41st Annu. Allerton Conf. Communication, Control, and Computing, Monicello, IL, Oct. 2003.
- [6] Lun, D. S., Ratnakar, N., Koetter, R., Médard, M., Ahmed, E., and Lee, H., “**Achieving minimum cost multicast: A decentralized approach based on network coding**,” in Proc. IEEE INFOCOM, Miami, FL, Mar. 2005, pp. 1607–1617.
- [7] Dougherty, R., Freiling, C. and Zeger, K., “**Insufficiency of linear coding in network information flow**,” IEEE Trans. Inf. Theory, vol. 51, no. 8, pp. 2745–2759, Aug. 2005.
- [8] Wu, Y., Padhye, J., Chandra, R., Padmanabhan, V. and Chou, P. A., “**The local mixing problem**,” presented at the Information Theory and Applications Workshop, San Diego, CA, Feb. 2006.
- [9] Liu, J., Goeckel, D. and Towsley, D., “**Bounds on the gain of network coding and broadcasting in wireless networks**,” in Proc. IEEE INFOCOM, May 2007, pp. 724–732.
- [10] Popovski, P. and Yomo, H., “**Bi-directional amplification of throughput in a wireless multi-hop network**,” in Proc. IEEE 63rd Vehicular Technology Conf. (VTC 2006-Spring), 2006, pp. 588–593.
- [11] Aulin, T. and Xiao, M., “**A physical layer aspect of network coding with statistically independent noisy channels**,” in Proc. IEEE Int. Conf. Communications (ICC), Jun. 2006, vol. 9, pp. 3996–4001.
- [12] Zhang, S., Liew, S. C. and Lam, P. P., “**Physical-layer network coding**,” presented at the ACM MobiCom, Los Angeles, CA, Sep. 2006.
- [13] Katti, S., Gollakota, S. and Katabi, D., “**Embracing wireless interference: Analog network coding**,” in Proc. ACM SIGCOMM, Kyoto, Japan, Aug. 2007, pp. 397–408.