

國立臺灣師範大學  
資訊工程研究所碩士論文

指導教授：蔡榮宗 博士

利用網路編碼改善無線感測網路下傳輸資料的  
可靠性

Improving Reliability by Using Network Coding in  
Wireless Sensor Networks

研究生：何君慧 撰

中華民國 一百零一 年 一 月

## 摘要

無線感測網路(Wireless Sensor Networks)廣泛應用在環境偵測、遠距離看護系統、軍事監測、災難救援偵測等等。無線感測網路受到資源上的限制，例如：電量、頻寬、記憶體以及計算速度。然而，無線感測網路在運作過程中由於電量上的限制，在加上無線網路中訊號強度的改變，電波干擾等因素影響使得感測器失去功能進而產生封包遺失的問題。是以無線感測網路的可靠性是一個仍須探討的議題。網路編碼(Network Coding)，近年來為一熱門研究主題，企圖利用中繼點將所收到的封包加以編碼，使得接收端在收到足夠的封包時自己便能自行解碼。

在此篇論文中，我們考慮在不使用傳統的 ARQ 機制下，利用有限度的泛播(Flooding)以及有限度的重傳次數，改善無線感測網路資料傳輸的可靠性。並且結合網路編碼在多個中繼點編碼，達到資料擴散至節點之間的目的。是以，透過時間分集(time diversity)以及空間分集(space diversity)的方式，進而提升無線感測網路的可靠性。實驗結果顯示，我們所提出的方法的有效提升封包到達率。

## 關鍵詞

無線感測網路、網路編碼、封包遺失、傳送次數、封包到達率

## **Abstract**

Wireless sensor networks have been widely used in many applications such as environmental monitoring, health-care monitoring, military monitoring, disaster management, etc. Sensor nodes are constrained by energy, computational capability and storage space. Such constraints, signal strength, and interference may cause node failure, link failure and result in the problem of packet loss. Therefore, how to provide a reliable data transmission in wireless sensor networks is still a challenging issue. Network coding, a popular research topic for years, is one of techniques attempting to encode packets for transmission at a node or relay station. After the relay station sends the encoded packet, a receiver can successfully decode the packet when it received enough encoded packets.

In this thesis, we consider limited flooding and multiple transmissions techniques to counter the loss of reliability when traditional ARQ techniques are not employed. Furthermore, we apply network coding to encode buffered packets at relay stations. Taking advantage of time diversity from limited flooding and multiple transmission and Taking advantage of space diversity from network coding, we thus can improve the reliability in data transmission in wireless sensor networks. We show simulation results on packet delivery ratio to validate the proposed methods.

## **Key Words**

wireless sensor networks, network coding, packet loss probability, number of transmissions, packet delivery ratio

## 誌謝

首先，我要感謝我的指導教授 蔡榮宗 老師，由於老師的細心的指導，讓我獲益良多，並且由於老師對研究的拘謹，讓我學習到對研究以及學術上的應有的嚴謹態度。這篇論文也是由衷的感謝老師，幫我字句斟酌、細心修改，才能有如此的面貌。同時也要感謝口試委員台灣科技大學電機工程學系 陳俊良 教授、清華大學資訊工程學系 林華君 教授百忙之中撥冗審查本論文，並惠賜許多寶貴意見，讓本論文更加充實與完備。

此外，感謝網路實驗室的所有成員，感謝昇倫學長、耀鴻學長給我在課業上的指導，重慶學長、宇鎮學長給我研究上的意見與指導，以及卉美、怡婷、勇諭、彥成這些日子一起努力，所給予的幫助與照顧。謝謝你們！

最後，我要感謝我親愛的家人，謝謝你們的體諒與關心，因為有你們的支持使我能無後顧之憂，專心於研究上而順利完成學業。

祝福所有的人健康快樂！

# 目 錄

附表目錄.....	vi
附圖目錄.....	vii
<b>第一章 簡介.....</b>	<b>1</b>
第一節 研究背景.....	1
第二節 研究動機.....	2
第三節 論文架構.....	4
<b>第二章 相關研究探討.....</b>	<b>5</b>
第一節 無線感測網路.....	5
第二節 感測網路路由協定.....	6
第三節 網路編碼機制.....	9
<b>第三章 路由協定.....</b>	<b>11</b>
第一節 系統模型.....	11
第二節 Level-Based 路由協定.....	12
第三節 K-based transmission .....	13
第四節 網路編碼.....	14
<b>第四章 模擬結果與討論.....</b>	<b>18</b>
第一節 模擬環境設定.....	18
第二節 模擬結果與討論.....	19
<b>第五章 結論與未來研究方向.....</b>	<b>38</b>
參考著作.....	39

## 附表目錄

表格 1 模擬實驗相關參數設定 .....	19
表格 2 模擬實驗相關參數設定 .....	34

## 附圖目錄

圖 2-1 感測網路基本架構示意圖 .....	6
圖 2-2(a) Interest and Gradient setting .....	7
圖 2-2(b) Data deliver along reinforcement path .....	7
圖 2-3 SPIN 協定路由建立及資料傳送示意圖([12]) .....	8
圖 2-4 網路編碼(Network Coding)概念示意圖 .....	10
圖 3-1 感測網路節點、基地台分佈示意圖 .....	11
圖 3-2 Level-Based 路由協定示意圖 .....	13
圖 3-3 線性網路編碼示意圖 .....	15
圖 3-4 感測網路系統流程圖 .....	17
圖 4-1(a) $\alpha=0.5$ , 50Nodes, Packet Delivery Ratio .....	22
圖 4-1(b) $\alpha=0.5$ , 50Nodes, Number of transmissions .....	22
圖 4-2(a) $\alpha=0.6$ , 50 Nodes, Packet Delivery Ratio .....	23
圖 4-2(b) $\alpha=0.6$ , 50 Nodes, Number of transmissions .....	23
圖 4-3(a) $\alpha=0.7$ , 50 Nodes, Packet Delivery Ratio .....	24
圖 4-3(b) $\alpha=0.7$ , 50 Nodes, Number of transmissions .....	24
圖 4-4(a) $\alpha=0.8$ , 50 Nodes, Packet Delivery Ratio .....	25
圖 4-4(b) $\alpha=0.8$ , 50 Nodes, Number of transmissions .....	25
圖 4-5(a) $\alpha=0.9$ , 50 Nodes, Packet Delivery Ratio .....	26
圖 4-5(b) $\alpha=0.9$ , 50 Nodes, Number of transmissions .....	26
圖 4-6(a) $\alpha=0.5$ , 100 Nodes, Packet Delivery Ratio .....	28
圖 4-6(b) $\alpha=0.5$ , 100 Nodes, Number of transmissions .....	29
圖 4-7(a) $\alpha=0.6$ , 100 Nodes, Packet Delivery Ratio .....	29
圖 4-7(b) $\alpha=0.6$ , 100 Nodes, Number of transmissions .....	30
圖 4-8(a) $\alpha=0.7$ , 100 Nodes, Packet Delivery Ratio .....	30
圖 4-8(b) $\alpha=0.7$ , 100 Nodes, Number of transmissions .....	31
圖 4-9(a) $\alpha=0.8$ , 100 Nodes, Packet Delivery Ratio .....	31
圖 4-9(b) $\alpha=0.8$ , 100 Nodes, Number of transmissions .....	32
圖 4-10(a) $\alpha=0.9$ , 100 Nodes, Packet Delivery Ratio .....	32
圖 4-10(b) $\alpha=0.9$ , 100 Nodes, Number of transmissions .....	33
圖 4-11(a) 50 Nodes, One-based v.s.K-based transmission, Packet Delivery Ratio .....	35
圖 4-11(b) 50 Nodes, One-based v.s.K-based transmission, Number of transmissions .....	35
圖 4-12(a) 100 Nodes, One-based v.s.K-based transmission, Packet Delivery Ratio .....	36

圖 4 -12(b) Nodes, One-based v.s.K-based transmission , Number of  
transmissions.....37



# 第一章簡介

## 第一節研究背景

無線感測網路是由大量的感測器所組成，其感測器具有處理資訊、通訊以及感測能力。這些感測器分佈於感測區內，用來偵測環境的變化或者感應其目標物，並且將所收集到的資料以無線傳輸的方式傳遞至匯集點(Sink)或者基地台(Base station) [1][2][3]。

無線感測網路的應用上範圍甚廣，例如：軍事、環境監控、健康醫療、商業應用…等等。感測器通常是大量隨機佈置於艱困的環境之中，難以回收、充電，加上成本考量使得感測器本身資源有限，不論是運算能力、記憶體容量，以及電力 [1][13]。

如何更有效利用感測器的電量，一直是為熱門的研究方向。為了使感測器的生命週期能提升，在路由協定上大多採用多躍式(multi-hop)的資料傳遞方式，減少傳送遠距離而導致的電量大幅消耗 [4]。

感測網路的目的在於傳遞所偵測到的資料。然而，感測器所在的環境以及感測器本身的資源限制等因素，使得感測器本身的損毀或者通訊失敗，而導致所偵測到的資料在傳遞過程中遺失。此外，在傳輸品質很差的節點或者鏈結的環境之下，傳輸效能將會快速下降，使得資料將無法有效地傳送至匯集點(Sink)或者基地台(Base station) [5]。

過去許多研究為了解決此種情形，提供多種可靠性的傳輸方式，例如：自動請求回復(ARQ)、多徑傳輸(Multipath Forwarding)。自動請求回復(ARQ)，是為

一種傳統的錯誤控制機制。其方法主要為，當傳送端傳送訊息後，將等待接收端是否收到訊息，若傳送端在等待一段時間之內，沒有收到接收端的確認訊息，將會再重新發送。

多徑傳輸(Multipath Forwarding)，主要則是尋找幾條不同的傳送端至目的端的路徑，並且將其中之一的路徑視為主要路徑，而其他條路徑則是備份路徑。當主要路徑不通時，則可利用備份路徑來傳送。此外也能同時複製傳送端訊息，並將其訊息透過不同的路徑傳輸訊息至目的端。利用多條不同路徑傳送訊息，進而增加封包成功抵達基地台的可能性，使得提升其傳輸的可靠性 [5][6][7]。

網路編碼(Network Coding)[3][6][10][11]，是一種透過中繼節點(relay station)，對所接收到的訊息進行編碼後再傳送給接收端，而接收端在接收到足夠多的訊息量時，即可自行解碼，而得到所需的訊息。相對於傳統上，中繼點只負責訊息的儲存轉送，網路編碼技術在網路傳遞過程中，將資訊擴散於節點之間。近年來，在提升可靠性傳輸的方式的研究，也開始利用編碼方式試圖提升其傳播的可靠性。

## 第二節 研究動機

無線感測網路，是利用其區域內的感測器所偵測到的訊息，經由各種傳送方式匯集至基地台，進而使得使用者能夠監控其感測的區域。其中感測器傳送資料的方式，可分為時間驅動型(continuous based)、事件驅動型(event based)。時間驅動型(continuous)則是將感測器所偵測到的資料週期性的傳回匯集點或者基地台。而事件驅動型(event based)為當偵測到事件發生時，感測器才將訊息傳遞回基地台。

然而，由於感測器本身資源的限制，以及有時所偵測的區域環境的頻帶(channel)並不是那麼良好，在此環境之下，使得所偵測到的事件可能無法有效地將訊息傳送回基地台，存在著高度的訊息遺失風險。是以，許多研究也著重於如何有效的提高其傳輸的可靠性。

許多關於可靠性的研究，目的都是為了能夠使感測器在經過一層層的傳輸過程後，基地台能成功收到感測器的訊息。儘管，許多研究都著重於，自動請求回復(ARQ)或者多徑傳輸(Multipath Forwarding)，在頻帶狀況不佳的區域，進行重傳或者重複相同的訊息以不同的傳輸路徑傳送，確實能夠提升其可靠性。

倘若感測器所分佈的區域環境並不是很好，其頻帶錯誤率相當高亦或其環境使得感測器故障，存在高度的封包遺失風險。此時，若能利用網路編碼(Network Coding)之方式，將傳送出去的訊息，經中繼點加以編碼(encode)，再予以傳送，使得封包能擴散於節點之間，那麼封包在抵達基地台(Base Station)後，只要基地台收集足夠的封包，即可自行解碼，得到所需的訊息。

但利用網路編碼(Network Coding)，存在一個風險就是倘若經過編碼的封包，封包有所遺失時，有可能使得許多封包無法得以解碼，進而使得所真正收集到的訊息反而大幅下降。為了避免此種情形，我們經過機率運算，計算出在某些成功機率以及所存在的傳送失敗率後，所需的傳送次數。企圖使感測器傳送訊息時，能有效的提升鄰居收到的成功率，進而減少封包遺失率。是以，透過時間分集(time diversity)以及空間分集(space diversity)的方式，進而提升無線感測網路的可靠性。

### 第三節 論文架構

本論文章節架構如下所述。第二章，我們將介紹本論文的相關研究。首先簡單的介紹感測網路(Sensor Networks)，接著介紹著名的感測網路路由協定(routing protocol)的相關研究，並且本章末概述網路編碼(Network Coding)及其應用。而在第三章中，將會介紹我們使用的方法以及系統架構。第四章則將呈現所模擬實驗的結果(Simulation Results)。最後，第五章將引申討論此論文的結果以及未來可進行的方向。

## 第二章相關研究探討

### 第一節無線感測網路

由於微型機械電子系統、數位訊號處理以及無線通訊技術的進步，使得無線感測網路有許多進步的空間，並且有廣泛的應用。無線感測網路系統，是由許多感測器所組成，可自行組成一個網路，互相溝通並且傳遞資料。具有造價成本低、無需基礎建設的無線通訊、體積小、佈建容易、低耗電等優點。

然而由於感測器是大量的佈置在一區域內，且其區域環境之地形等因素之下，感測器難以回收或補充電力，在成本考量之下，無線感測網路擁有許多限制：

[1][2]

- 感測器電量、運算能力以及記憶體容量有相當大的限制。
- 無線感測器網路常因感測器之電力消耗完畢或因故障，使得網路拓撲常常改變。
- 當感測器大量的同時傳送資料時，容易造成頻寬及電力的消耗及浪費。

無線感測網路主要的目的，就是將在環境中所偵測到的資料或者目標物的資料，傳送至匯集點或者基地台，使得能加以監控其所偵測區域。無線感測網路之基本架構如圖 2-1 所示：

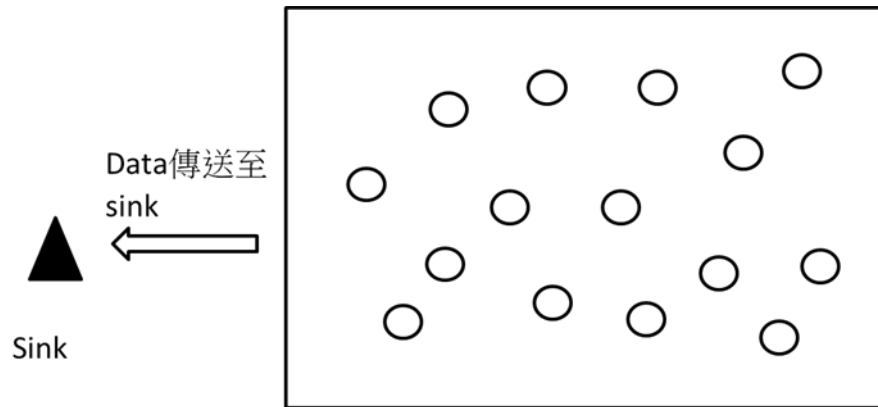


圖 2 - 1 感測網路基本架構示意圖

## 第二節感測網路路由協定

由於電力的消耗與傳輸距離的平方至四次方及傳輸資料量大小成正比，資料通訊為無線感測網路中最耗電力的部分，為了節省電力的消耗，無線感測網路多採用短距離的傳輸方式，進而藉由多點跳躍(multi-hop)路由方式建立無線感測網路路由，藉由多點跳躍式的方式將資料層層傳送至基地台。

許多研究中，都已提出利用多點跳躍路由方式來傳遞資料的方法，並以降低資料傳輸時的電力消耗為原則，來設計無線感測網路路由協定，其中最簡單的方式為泛播基礎 (Flooding Based) 架構。泛播機制(Flooding Based)是最為簡單的路由方式，不需要維護網路拓樸架構和路由資訊，其節點在產生或第一次收到資料後，以廣播方式轉送給所有的鄰近節點，藉由不斷的廣播傳送，將資料送達目的地。

Directed Diffusion [8][9]協定則是以資料為中心 (Data-centric) 的方法，每一節點不以位址來區分，而是以它所感測到的資料來定址。使用者透過匯集點 (Sink)發送關注訊息(Interest Message) 進行查詢 (Query)，收到此訊息的節點會先朝向發送端建立梯度 (Gradient)，做為訊息回傳的有效路徑，接著轉送此查詢

訊息，使其擴散到網路中所有節點，進而建立起多條通往匯集點(Sink)的路徑，若先前已接收過此訊息則不予處理，如圖 2-2(a)。一旦有節點符合此次查詢所需資料時，會沿著先前建立的路徑向匯集點回報，匯集點會依據一些訊息進行評估後(如：根據節點收到的時間延遲)，挑選出一條最佳路徑作為資料回傳使用，其他路徑則予以中斷，如圖 2-2(b)。

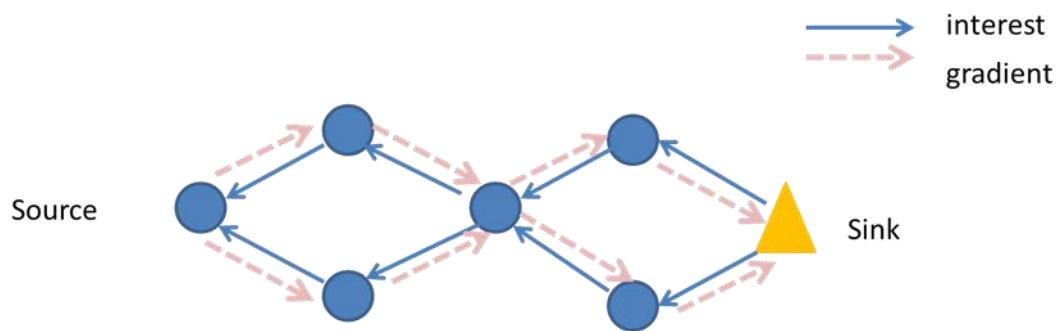


圖 2- 2(a) Interest and Gradient setting

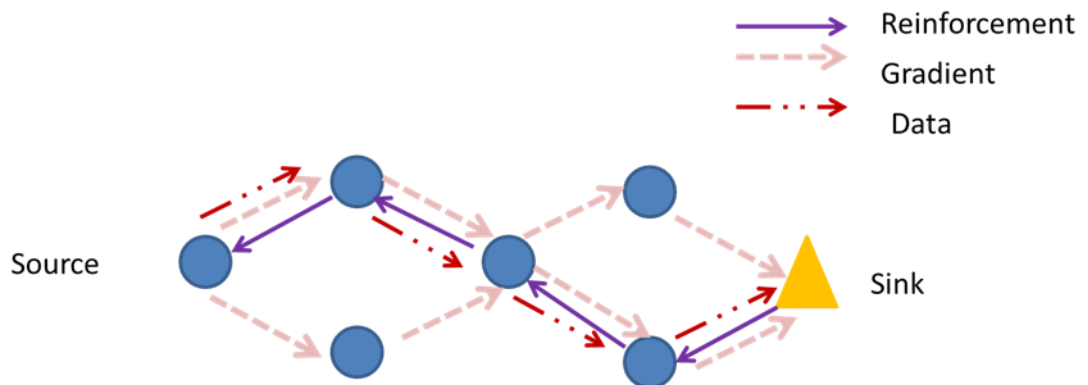


圖 2-2(b) Data deliver along reinforcement path

另一種則是 SPIN (Sensor Protocol for Information via Negotiation) [12]協定主要利用協商 (Negotiation) 及資源調適 (Resource Adaptation) 來解決泛播機制中傳送所導致的相關問題。當網路中任一節點要傳送資料給它的鄰近節點時，先藉由節點用來描述資料本身特性的超資料 (Meta-data) 來進行協商，若鄰近節點已經接收過此資料並存於記憶體中，則不予理會；反之，則向傳送超資料(Meta

Data)的節點要求真正的資料。如此，可避免傳送相同的資料，如圖 2-3。此外，每一個節點都有自己的資源管理者 (Resource Manager) 來記錄資源的消耗情形，在傳送資料前，會先檢查自己的資源狀態再行決定傳送與否。

此兩種方式都是先藉由廣播方式以及相互溝通後傳遞資料，在這邊，我們之後的方式會利用此概念，來建立我們的傳送表(Forwarding Table)，進一步的我們藉由一層層傳的概念建成樹。

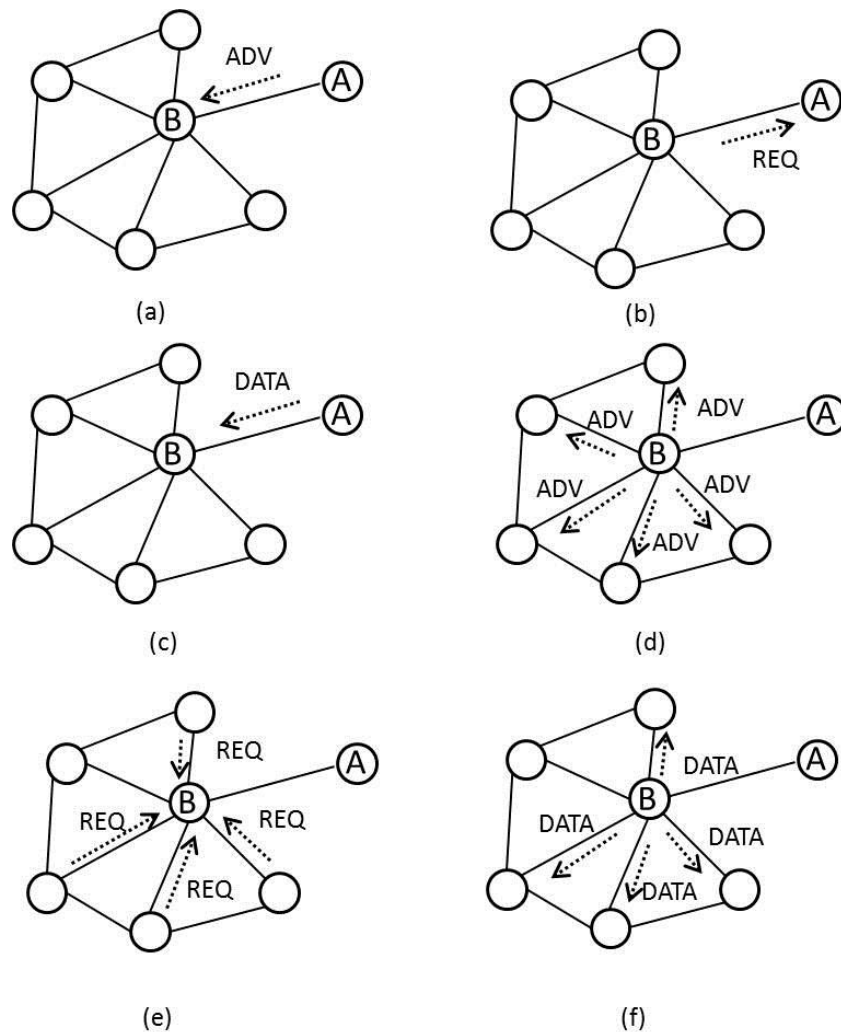


圖 2-3 SPIN 協定路由建立及資料傳送示意圖[12]



### 第三節網路編碼機制

傳統的路由器或者中繼點(relay station)收到封包後，都是先儲存再轉送(store and forward)。但若擁有網路編碼的能力，則可先將所收到的封包先與以編碼，再轉送出去，使其擁有儲存、編碼再轉送的能力，此即為網路編碼(Network Coding)的基本概念 [10][11]，目的在於達到傳送資料時的最大傳輸率(throughput)，即是攜帶最大資料量。下圖 2-4 可說明如何利用網路編碼(Network Coding)的機制，增加傳輸率。圖中假設 R 以及 R2 將接收來自於來源端 S 以及 S2 所送出的資料 a 以及 b。而中繼點 x，對所收到的封包 a 與 b 經過編碼處理(在此為 Xor 運算)後再傳送出去。而節點 R 以及 R2 在分別收到(a, (a xor b))以及(b, (a xor b))後，可分別自行經由 Xor 運算得到 a 與 b 兩筆資料。此外，封包的編碼方式不只有 Xor 的運算方式，也可利用線性網路編碼(linear network coding)方式。若有 n 個封包需要加以編碼，則產生 n 個係數，將此 n 個封包以及係數經兩兩相乘並且累加起來，並將乘積以及係數傳遞出去，接收端在收到足夠的 n 個獨立係數以及所對應的乘積後，即可解碼回來。

由於網路編碼有利於一對多的情形之下，例如、群播或者廣播機制。因此，大部份的研究也大多利用這個優勢，應用在許多層面上。本文則是以多對一的架構，試圖在此環境下提升傳輸可靠性。

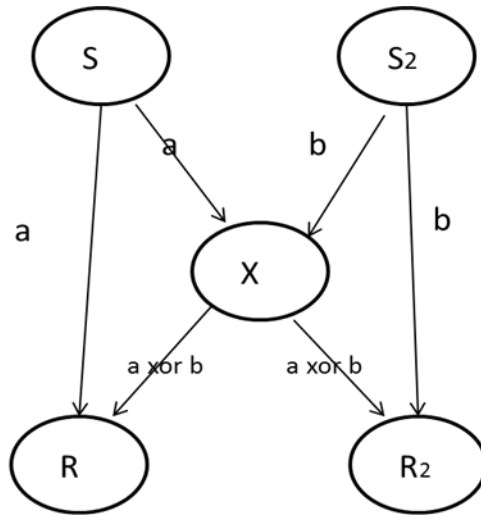


圖 2-4 網路編碼(Network Coding)概念示意圖[14]

## 第三章路由協定

### 第一節系統模型

- 所有節點為同質性的感測器，感測器電力皆相同。
- 感測器感測範圍以感測器為中心，半徑  $r$  所圍成的圓形區域。
- 基地台位於區域中正下方。
- 感測器隨機分佈於固定區域，分佈完後便不會再移動，並且每一個感測器，都至少有一條路徑可將資料傳送到匯集點(Sink)。如圖 3-1 所示
- 無線感測網路可以用圖形  $G(V, E)$ ，表示其拓撲模型(topology model)， $V = \{v_1, v_2 \dots v_n\}$  表示有限個節點(感測器)，節點分佈在固定區域， $E = \{\{v_i, v_j\} \mid v_i, v_j \in V\}$  表示節點的邊的集合， $e_{ij} = \{v_i, v_j\} \in E$ ， $e_{ij}$  為 1 表示  $v_i$  與  $v_j$  互相在彼此傳送範圍內，相反地  $e_{ij}$  為 0 表示兩個節點為不連通。

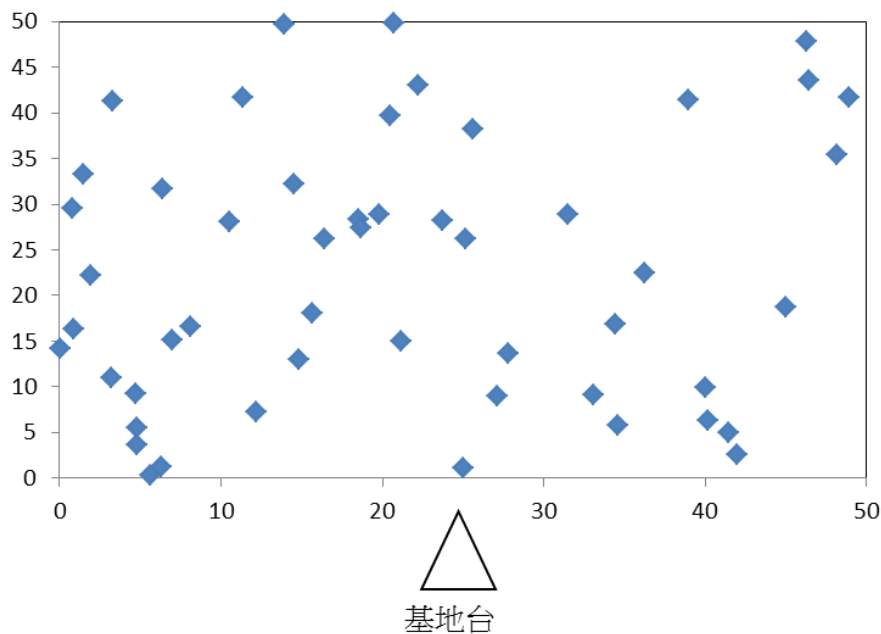


圖 3-1 感測網路節點、基地台分佈示意圖

- 當感測器偵測到事件時，將所偵測到的資訊傳回基地台。傳送的方式是根據所建立的傳送表(forwarding table)，將資訊傳給至鄰居。並且所有節點會根據所收集的資料個數，決定是否做網路編碼(Network Coding)，再傳送出去，直到匯集點(Sink)。
- 傳送封包時，存在遺失機率  $p$ 。

## 第二節 Level-Based 路由協定

Level-Based 路由協定，主要概念來自於 Directed Diffusion[8][9]，匯集點(Sink)會廣播(Broadcast)關注訊息(interest)給鄰居，此訊息包含了現在時間、持續時間、傳送者的編號(ID)以及傳送者的跳躍數(hop count)。而系統一開始設定，匯集點(Sink)的編號(ID)以及跳躍數(hop count)為 0，所有節點的跳躍數(hop count)為無限大。當匯集點(Sink)廣播關注訊息(interest)，在傳送範圍內的節點收到訊息後，會比較自身的跳躍數(hop count)以及其關注訊息(interest)內的跳躍數(hop count)，若本身的跳躍數比較大，將會更新本身的跳躍數為關注訊息(interest)內的跳躍數再加一。並且將關注訊息內的節點編號加入至傳送表(forwarding table)。節點再將其更新過後的跳躍數以及本身節點編號更新至關注訊息並且廣播其訊息。如此，收到訊息的節點便能依次更新其本身的跳躍數，直到訊息的被刪除為止。其建立方式如圖 3-2 所示。

如此建立好傳送表(Forwarding Table)後，當節點偵測到事件時，便可將其訊息傳送給傳送表內的鄰居，而每個節點只會接收並且傳送比自身跳躍式還高的鄰居所傳送的封包。每個節點透過多跳躍(multi-hop)的方式，一層層朝向匯集點傳送。

而為了減少傳送過程中，傳送到相同封包，每個節點收到訊息時，將會檢查

自身的暫存器 (Buffer) 中，是否有相同的封包，若有相同，則將重複的封包刪除，否則儲存其訊息。

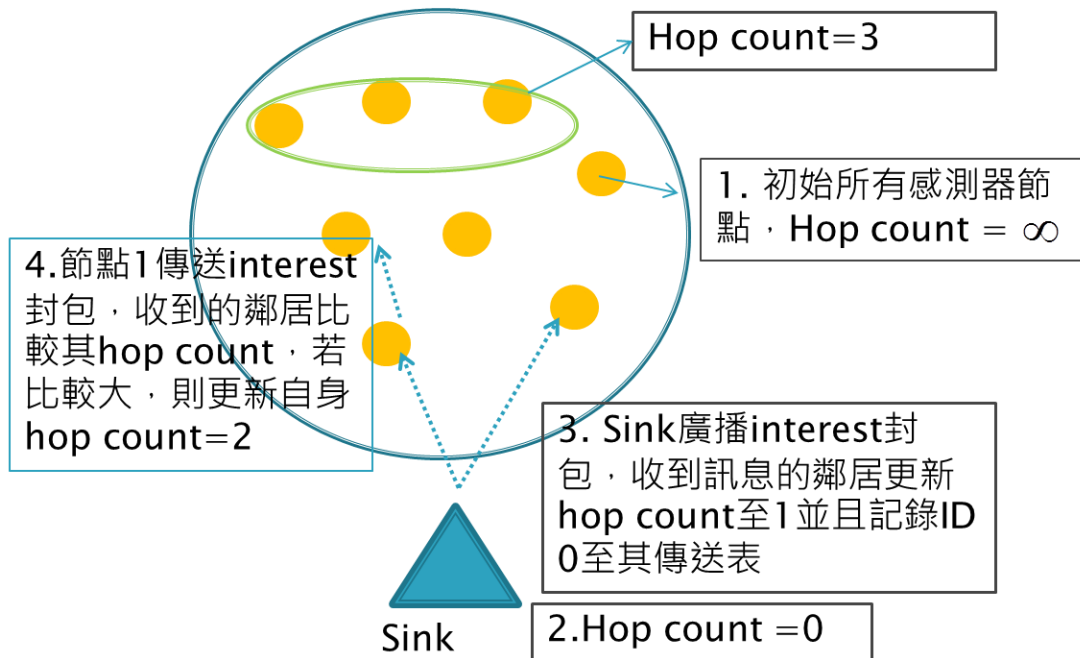


圖 3-2 Level-Based 路由協定示意圖

### 第三節 K-based transmission

由於無線感測網路的節點失敗或者頻帶問題而造成所送出的封包，無法有效地抵達目的地，匯集點(Sink)無法收到節點資訊，造成隨著封包遺失機率的提升可靠度下降的問題。是以，假若在傳送封包時，存在遺失機率  $p$ ，並且節點擁有  $n$  個鄰居，倘若我們希望至少有一個傳送節點之鄰居收到的機率為  $\alpha$ ，則所需要傳送  $k$  次才能使得傳送節點的鄰居成功收到機率大於  $\alpha$ 。

$$P(S \geq 1) \geq \alpha \quad (1)$$

$$P(S \geq 1) = (1 - p^n) + p^n(1 - p^n) + \dots + p^n(1 - p^n)^k \quad k \in \mathbb{N} \quad (2)$$

$$= 1 - p^{kn}$$

$$1 - p^{kn} \geq \alpha \quad (3)$$

$$k \geq \frac{\log(1-\alpha)}{n \log p} \quad (4)$$

$$k \geq n^{-1} \log_p(1 - \alpha) \quad (5)$$

藉由式子(5)，每個節點便可以藉由已知的遺失機率  $p$  以及傳送表內的鄰居個數，來決定當希望鄰居能夠收到的機率大於  $\alpha$  時，所需的傳送次數  $k$ 。

## 第四節 網路編碼

### ■ 線性網路編碼 (Linear Network Coding)

線性網路編碼的方式，是以線性組合的方式將多個封包進行編碼，而線性組合的係數是來自於有限體 (Finite Field)。如果節點有  $n$  個封包，則可以從有限體中選出  $n$  個係數與  $n$  個封包配對，形成線性組合，再將編碼過的封包傳遞出去。

當節點  $S$  有  $n$  個封包  $P_1, P_2, \dots, P_n$ ，節點  $A$  將從有限體選出  $n$  個係數與  $n$  個封包做線性組合，而從有限體選出的  $n$  個係數，是為編碼向量 (coding vector)，而經過編碼完成的封包為  $X$  (coded packet)， $X = C_1P_1 + C_2P_2 + \dots + C_nP_n$ 。當  $R$  接收到  $X$  和  $\vec{C}$ ， $\vec{C} = [C_1, C_2, \dots, C_n]$ ，可得知有  $n$  個封包 ( $P_1 \sim P_n$ )，是以若  $R$  收到  $m$  個封包，只要  $m \geq n$ ，且其中  $n$  個的編碼向量 (coding vector) 彼此互相獨立，則  $R$  節點即可解碼回完整的  $P_1, P_2, \dots, P_n$  封包。反之，若  $m < n$ ，則無法解出唯一解  $P_1, \dots, P_n$ 。圖 3-3 為編碼過程示意圖。

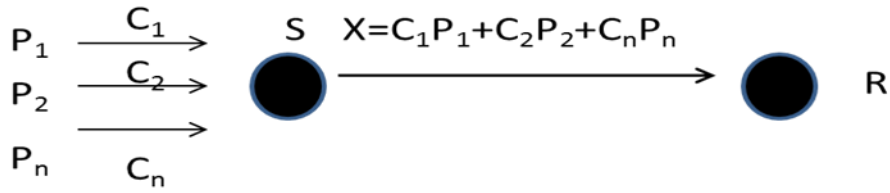


圖 3-3 線性網路編碼示意圖

## ■ 線性隨機編碼

由於感測器節點是將所感測的資訊傳送至匯集點，每一個感測器要傳送封包時大多傳送一筆資訊不需要作分割，但由於傳送方式是多跳躍(multi-hop)，是以許多感測器都有機會收到大於一個以上的封包數，若傳送時能將這些封包經過編碼再傳送出去，使得即使在傳送過程中有所遺失，也有機會藉由其他感測器所編碼的封包經匯集點(Sink)或者基地台解碼而還原成原來的封包。大部份的研究大多將一份資料切割成多個封包傳送出去，在方法上需與傳統上不太相同。當節點需做傳送時，會檢測待傳送的封包若有  $m$  個， $m \geq 1$

，則將隨機組合  $n$  個編碼向量(coding vector)，係數屬於  $G(2^1)$  元素，與待傳送的封包做線性組合傳送。如式子(1)所示

$$x = \sum_{i=1}^n c_i \times P_i \quad \cdots(1)$$

$P_i$  為所待傳送的的  $i$  個封包。

此外，倘若我們編碼封包的個數過多，易造成一旦有所遺失時，會使得匯集點無法成功解碼，而讓我們遺失封包的個數跟著增加。例如：我們將 8 個封包編碼成 8 個編碼後的封包，一旦遺失 1 個封包的時候，剩下的 7 個封包無法成功解碼，實際上的遺失從 1 個封包增加為 7 個。為了降低可能無法解碼所付出的成本，我們將限制每次編碼所取的封包個數為  $S$ ， $S = \min\{m, D\}$ ，其中  $D$  為我們的參數，決定  $S$  值最大為多少。

由於隨機從  $G(2^1)$  選取編碼向量，若不慎控制易造成所編碼的封包之間會有編碼向量的相依性(dependent)，使得匯集點無法成功解碼。這個問題，我們使用簡單的方式來解決。

假若編碼  $m$  個封包，需編碼  $m$  次產生  $m$  個已編碼的封包。第一次編碼時，我們從  $m$  個封包中選取出  $m$  個封包，其中  $m$  個封包之係數都為 1，編碼成

$$x_1 = C_1P_1 + C_2P_2 + \dots + C_mP_m, C_1 \sim C_m = 1, C_1 + C_2 + \dots + C_m = m。$$

而第二次編碼時，我們從前一次編碼中封包係數為 1 的封包， $P_1、P_2 \dots P_m$  中隨機選  $m-1$  個封包的係數為 1

$$剩下沒選中的封包係數為 0 編碼為  $x_2 = C_1P_1 + C_2P_2 + \dots + C_mP_m, C_1 + C_2 + \dots + C_m = m - 1,$$$

直到第  $m$  次編碼時，則從  $m-1$  次中所取的封包中隨機選取 1 個封包的係數為 1 剩

下沒選中的封包係數為 0 編碼  $x_m$ 。這樣即可使得每一個編碼過後的封包  $x_1 \sim x_m$  之間

的編碼向量可線性獨立。以下為簡單的程序來達到  $x_1 \sim x_m$  封包之間的線性獨立。

```

t is the nth times to encode 、g is the number of packets
Input: m packets , P1、P2...Pm
Output: x1,x2...xm , xt is the encoded packet at t time
g=m
For each t=1~m
    If t=1{
        Select g packets, all of the g packet's coefficient Ci=1 , i=1~m
        xt=C1P1+C2P2+...CmPm
        C1 + C2 + ... + Cm = g
        g-1
        return xt }
    else if t>1{
        random pick g packets through the packet which coefficient is 1 at t-1
        time and all the coefficient of selected packets Ci=1, the remaining
        m-g packets coefficient Ci=0 , i=1~m
        Xt=C1P1+C2P2+...CmPm
        C1 + C2 + ... + Cm = g
        g-1
        return xt }
end for

```



最後當匯集點收到資料時，會週期性的以高斯-喬丹消去法進行解碼。

圖 3-4 為此系統運作的流程圖。

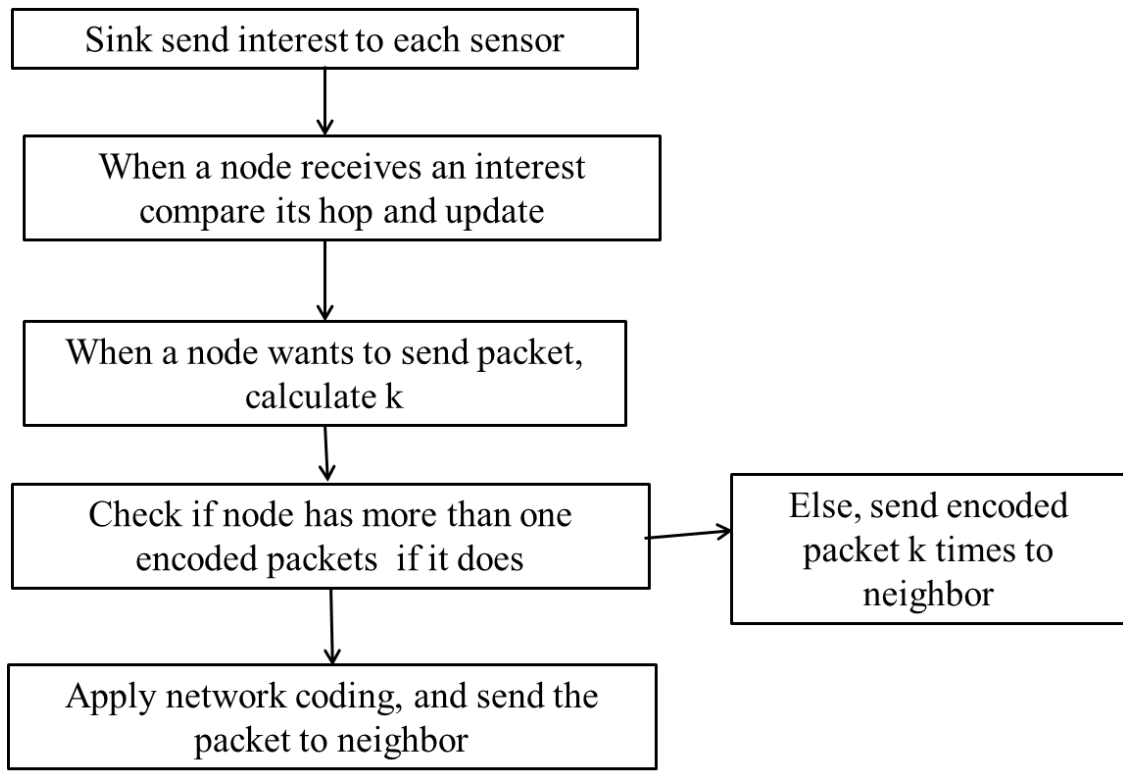


圖 3-4 感測網路系統流程圖

## 第四章模擬結果與討論

### 第一節模擬環境設定

本章節將展示並討論我們所提出的 K-based transmission 網路編碼(Dynamic transmission\_NC)實驗模擬結果，我們在不同的成功率  $\alpha=0.5$ 、 $\alpha=0.6$ 、 $\alpha=0.7$ 、 $\alpha=0.8$  以及  $\alpha=0.9$  的情形下，比較 K-based transmission 結合網路編碼以及單純 K-based transmission。再將 K-based transmission 以及傳送一次的方法做比較。

在我們的模擬實驗中，感測器隨機分布於固定大小的地圖之中。感測器分布完後，就不再移動，匯集點(Sink)固定在地圖的正下方。每個感測器都有自己的編號(ID)，並且有固定的最大傳送範圍(r)，在傳送範圍以內的感測器才能接收到資料。一開始，匯集點(Sink)會廣播關注訊號(interest)，節點會建立其傳送表(Forwarding Table)，並且依據第三章第三節的 k 值決定傳送資料時所需要的傳送次數。所有的節點都將產生一筆資料將傳送至匯集點(Sink)，並且每隔一個時間單位節點可以傳送一筆資料，每個節點傳過一次資料時稱為一回合。我們測試 1000 回合時匯集點所收能成功解碼的封包數以及全部節點的傳送次數。我們模擬每張地圖執行 1000 回合，最後依據 100 張不同的地圖計算其平均值。

我們先說明模擬實驗中的計算方式：

- 全部的傳送次數(Total Number of transmission)：全部的傳送次數為模擬時間中，整個網路總共產生的封包傳送次數。
- 封包到達率(Packet Delivery Ratio, PDR)：

$$PDR = \frac{N_{\text{sink}}}{N_{\text{source}}} \quad (6)$$

$N_{\text{sink}}$  代表模擬時間中，最後 sink 接收到的封包數總和，其中每個封包都來自於

不同的節點，並且不重複計算，倘若匯集點收到來自於相同節點封包兩個，只會計算收到一個；Nsource 代表所有來源感測器所產生的封包數總和，例如在模擬時間中，50 個感測器各自產生一個封包傳送至匯集點，則 Psource = 50。

接下來的章節我們將呈現在 50 以及 100 個節點下不同的  $\alpha$  值中比較網路編碼以及無網路編碼的實驗模擬數據。

## 第二節模擬結果與討論

在此章節我們先比較在成功率  $\alpha$  值為 0.5、0.6、0.7、0.8 以及 0.9，比較網路編碼的有無，對可靠性(Reliability)的影響。我們分別模擬在不同的封包遺失機率(packet loss ratio)下，地圖為(50m×50m)分佈 50 個節點以及地圖(100m×100m)分佈 100 個節點。相關參數設定如下表 1 所示。

地圖大小	50m×50m	100m×100m
地圖數量	100 張	100 張
傳送半徑	r=10m	r=10m
感測器數量	50	100
模擬回合	1000 回(每回合每個節點 傳送一次資料)	1000 回(每回合每個節點 傳送一次資料)
D 最大編碼個數	4、8	4、8
Packet loss ratio	0.05、0.1、0.2、0.3、 0.4、0.5	0.05、0.1、0.2、0.3、 0.4、0.5
Buffer size	120、1000	120、1000

表格 1 模擬實驗相關參數設定

## ■ 50 nodes, 50m×50m

圖 4-1、圖 4-2、圖 4-3、圖 4-4 及圖 4-5 分別展示在 50 個節點的密度，不同的成功率  $\alpha$  下使用我們所提出的 K-based transmission 結合網路編碼  $D=4$  且 Buffer 大小為 1000、K-based transmission 結合網路編碼  $D=4$  且 Buffer 大小為 120、K-based transmission 結合網路編碼  $D=8$  且 Buffer 大小為 1000，以及 K-based transmission 無網路編碼四種情形下所比較的結果(子圖(a)所呈現的是封包到達率(Packet Delivery Ratio)；子圖(b)所呈現的是傳送次數)。

在圖 4-1，成功率  $\alpha$  為 0.5 時，伴隨著不同的封包遺失率，四種方法的節點都只需傳送一次就能使得其傳送成功之機率大於  $\alpha$  值，使得在各不同的封包遺失率之下，封包到達率都隨著封包遺失率(Packet Delivery Ratio)的上升而下降。但由於，網路編碼所產生的多餘性(Redundancy)較高，使得其封包到達率雖然在封包遺失率 0.05 時略遜於無網路編碼的方式，但由於網路編碼所擁有的擴散性，使得在封包遺失率 0.2 至 0.5，隨著封包遺失率的遞增而加入網路編碼的方式使得封包到達率高於無編碼的方式。尤其在封包遺失率 0.5 時，加入網路編碼的方式能使得其封包到達率高達 58%，相較於無網路編碼的方式只有 44%，高出 14% 的封包到達率。當節點收到相同的封包的時候會檢查緩衝器(Buffer)是否有相同的封包，若已收到則刪除多餘的封包，緩衝器(Buffer)的大小會影響傳送次數以及最後封包送達率。圖 4-1(a)可看到加入網路編碼的方式 Buffer 為 120 的封包到達率以及比 Buffer 為 1000 低，傳送次數在封包遺失率為 0.05 時也較為低。並且，加入網路編碼的方式使得原本相同的幾個封包在經過編碼後收到封包的節點在檢查時判斷為不相同的封包不會加以刪除，這使得封包傳送次數高於無網路編碼的方式，但在封包遺失率較高之情形下，其所付出的傳送次數只略高於無網路編碼的方式。

而限制最大編碼個數 4 以及 8，從圖 4-1(a)，得到的結果幾乎相同跟所預期的當編碼個數較大可能會使得遺失更多封包的風險增加較為不同。

在圖 4-2，成功率  $\alpha$  值變成 0.6，使得有些節點的所需的傳送在封包遺失率高達 0.5 時所需的傳送次數高達兩次，是以使得封包遺失率在 0.5 時，封包到達率反而提升。而比較加入網路編碼的方式也是在封包遺失機率 0.05 時略遜於無網路編碼的方式，但在封包遺失機率 0.2 之後，加入網路編碼的方式所得到的封包到達率高於無網路編碼的方式(圖 4-2(a))。而相對應的傳送次數(圖 4-2(b))也在封包遺失率 0.5 時由於傳送次的改變而上升。而  $D=4$  以及  $D=8$  依然相差不大，Buffer 大小為 120 的封包到達率依然比 Buffer 大小為 1000 低。

在圖 4-3，成功率  $\alpha$  值變成 0.7，因為所要求的成功率的提升，使得節點所需傳送次數的改變，導致在封包遺失率 0.4 時封包到達率(圖 4-3(a))以及傳送次數(圖 4-3(b))都反而上升。並且可以看出加入網路編碼的方式依然在封包遺失機率提升時，亦即封包遺失率 0.2、0.3、0.4 以及 0.5 時的封包到達率，依然高於無網路編碼的方式，但是加入網路編碼方式的傳送次數卻遠高於無網路編碼方式。

在圖 4-4，成功率  $\alpha$  值變成 0.8 後，由於所要求的成功率提升，使得節點所需的傳送次數在封包遺失率為 0.3 時增加為 2，使得封包到達率以及傳送次數都增加，而在封包遺失率為 0.5 時傳送次數再度增加為 3，再次使得封包到達率以及傳送次數都上升。並且除了封包遺失率 0.05 之外，加入網路編碼方式的封包到達率依然高於無編碼的方式，但加入網路編碼方式的傳送次數(圖 4-4(b))卻遠高於無網路編碼方式。在圖 4-5，成功率  $\alpha$  值變成 0.9 後，傳送次數在成功率為 0.2 以及 0.4 時發生改變。但因重傳次數的提升無編碼方式與加入編碼方式的封包到達率有縮小差異，加入編碼的方式封包到達率在 0.2 時只略高於無編碼方式。而因

為增加了累贅性(Redundancy)使得加入網路編碼方式的傳送次數遠高於無網路編碼方式。

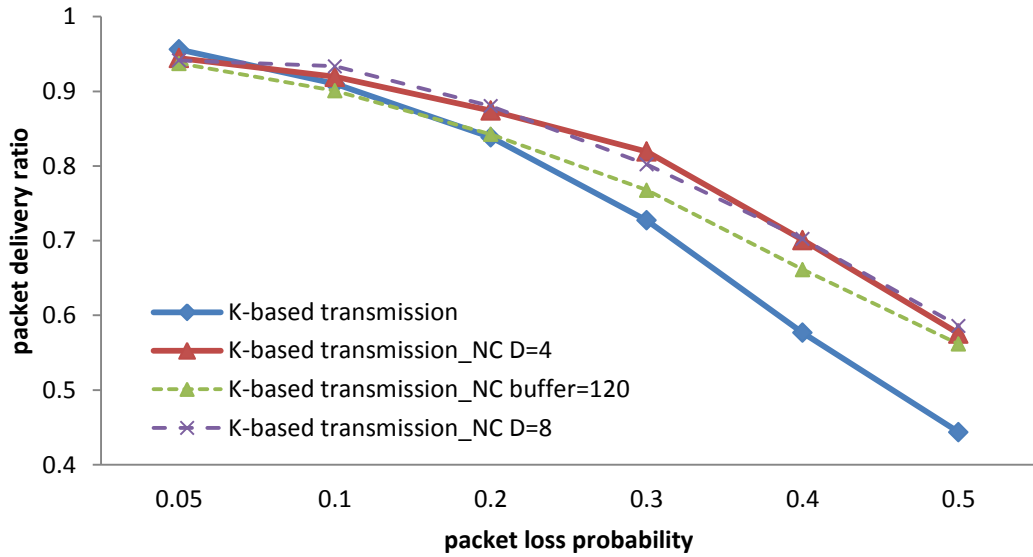


圖 4-1(a)  $\alpha=0.5$  , 50Nodes , Packet Delivery Ratio

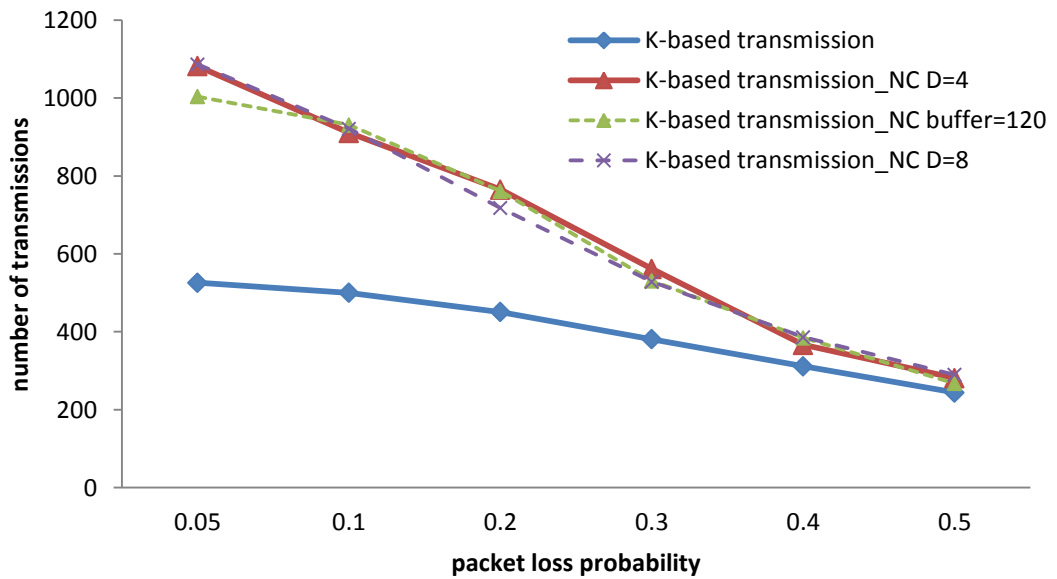


圖 4-1(b)  $\alpha=0.5$  , 50Nodes , Number of transmissions

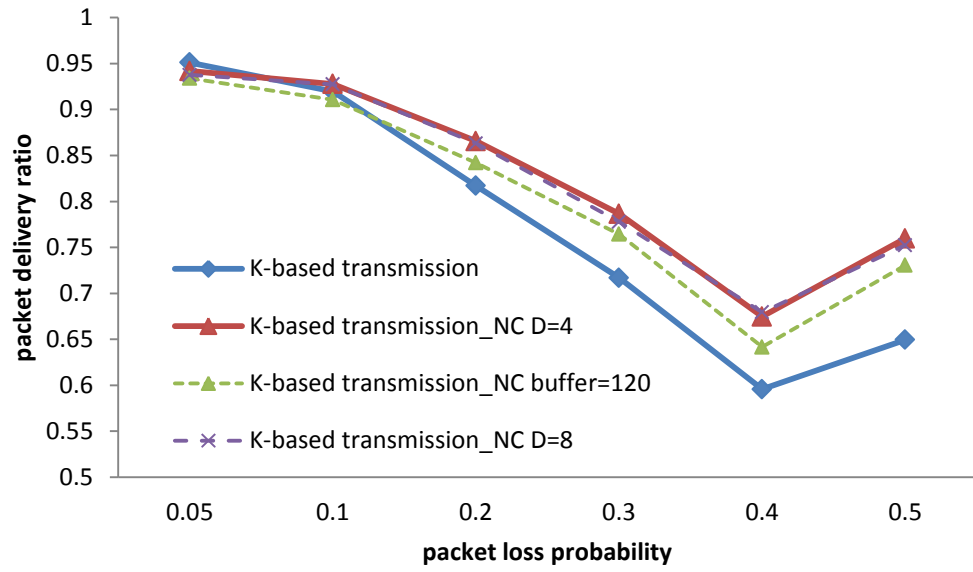


圖 4-2(a)  $\alpha=0.6$  , 50 Nodes , Packet Delivery Ratio

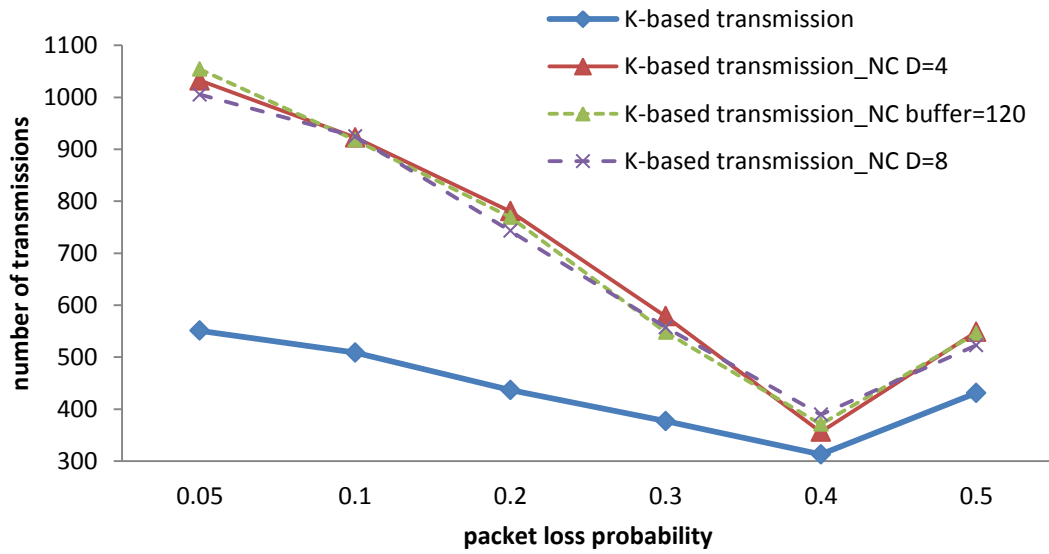


圖 4-2(b)  $\alpha=0.6$  , 50 Nodes , Number of transmissions

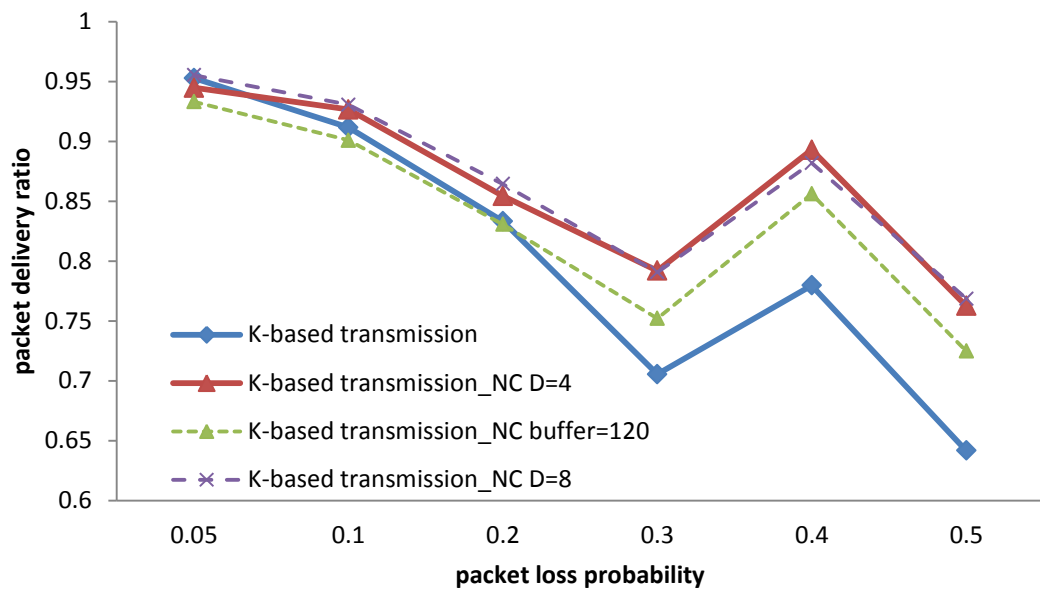


圖 4-3(a)  $\alpha=0.7$  , 50 Nodes , Packet Delivery Ratio

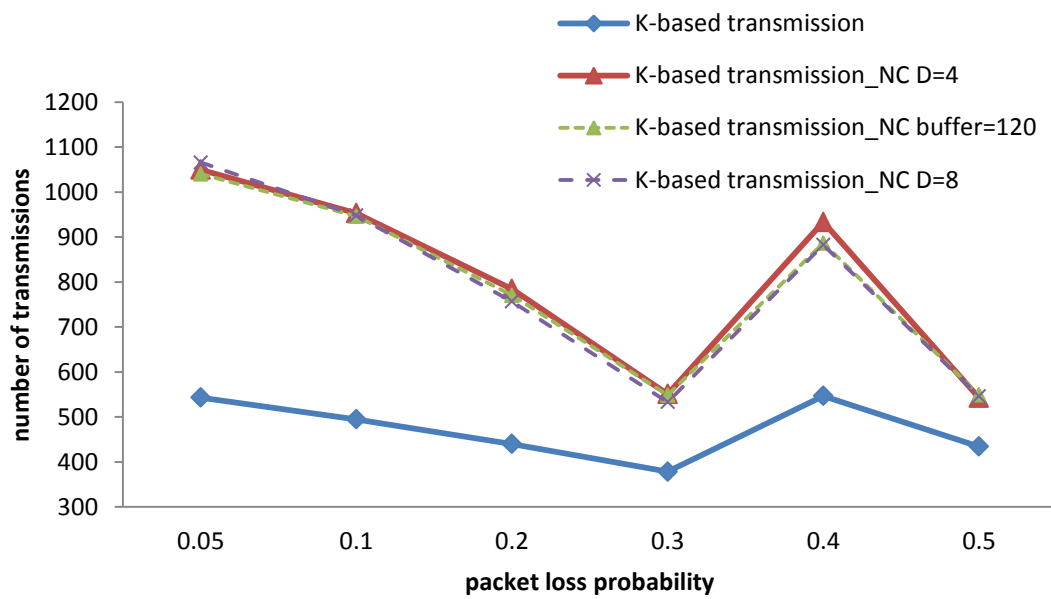


圖 4-3(b)  $\alpha=0.7$  , 50 Nodes , Number of transmissions



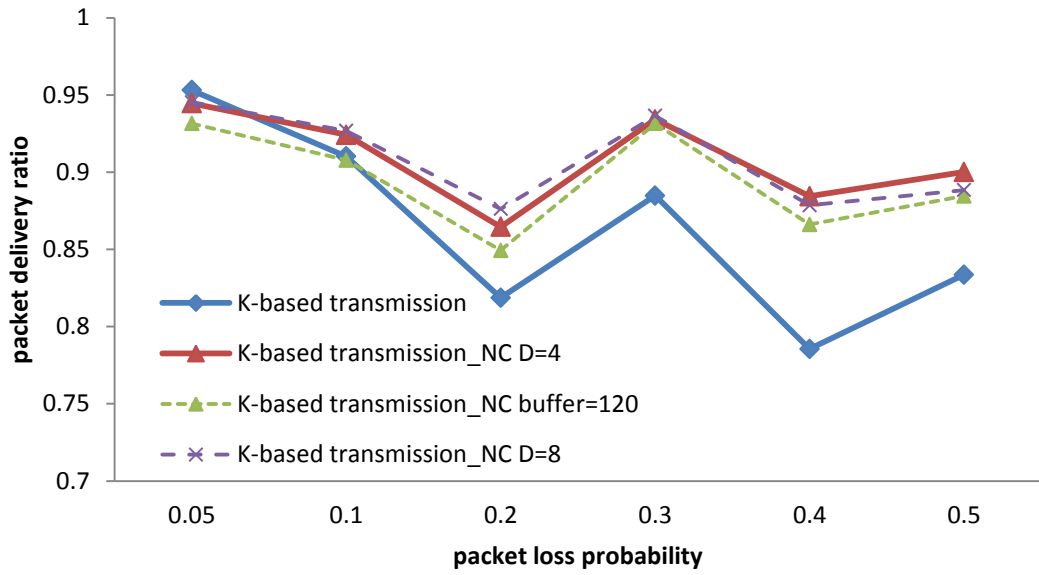


圖 4-4(a)  $\alpha=0.8$  , 50 Nodes , Packet Delivery Ratio

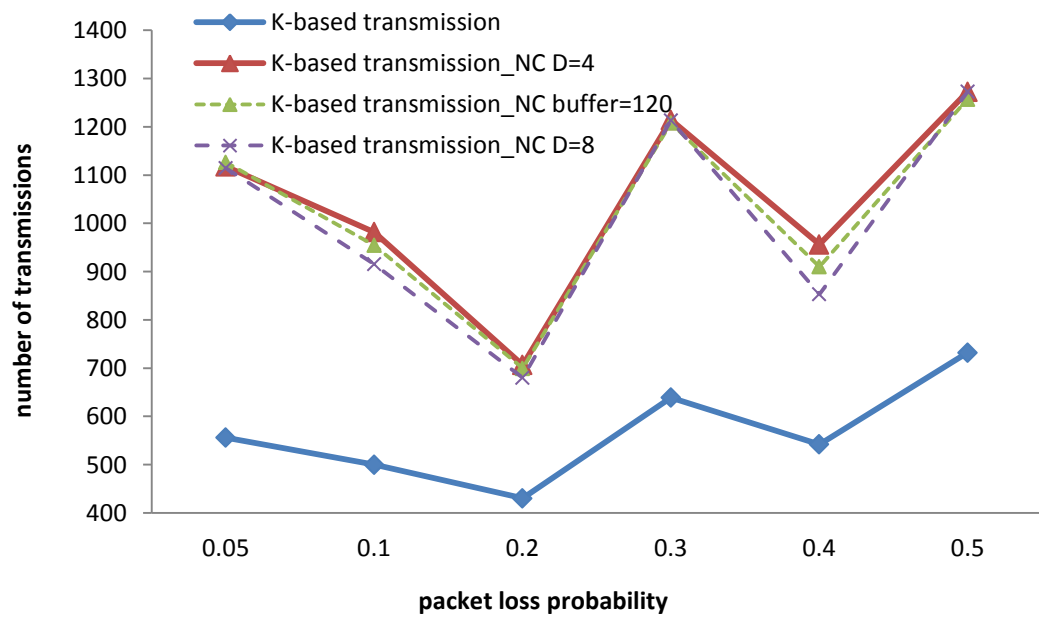


圖 4-4(b)  $\alpha=0.8$  , 50 Nodes , Number of transmission

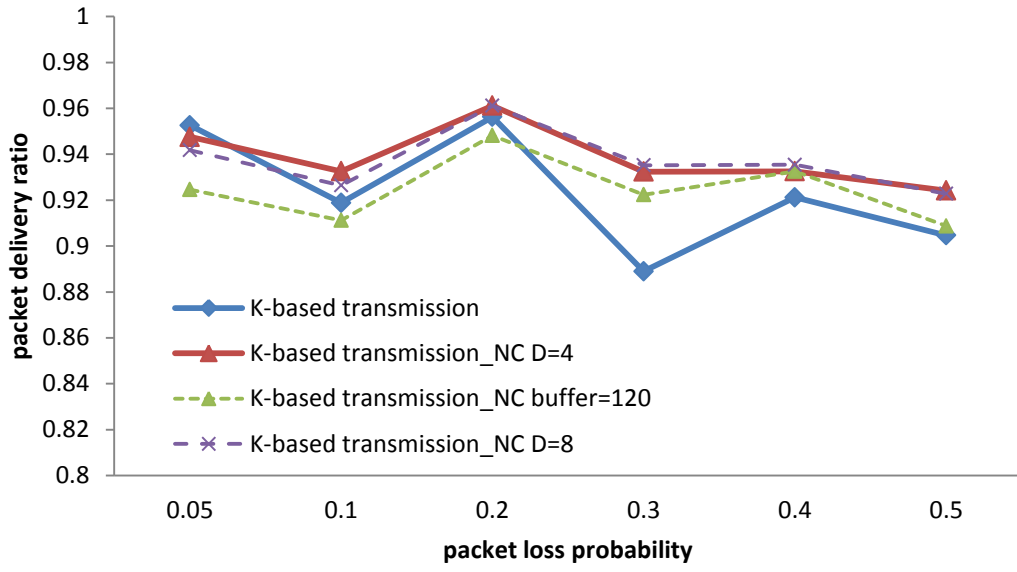


圖 4-5(a)  $\alpha=0.9$  , 50 Nodes , Packet Delivery Ratio

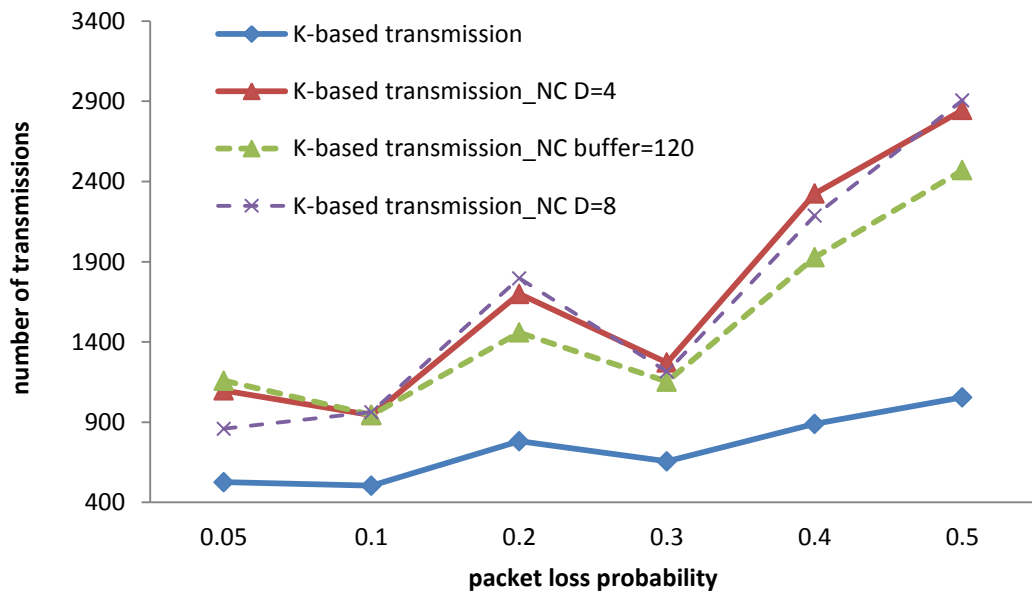


圖 4-5(b)  $\alpha=0.9$  , 50 Nodes , Number of transmission

## ■ 100 Nodes, 100m×100m

圖 4-6、4-7、4-8、4-9 以及 4-10，為在 100m×100m 的地圖 100 個節點情形，不同的成功率  $\alpha$  下，使用我們所提出的 K-based transmission 結合網路編碼 D=4 且 Buffer 大小為 1000、K-based transmission 結合網路編碼 D=4 且 Buffer 大小為 120、K-based transmission 結合網路編碼 D=8 且 Buffer 大小為 1000，以及 K-based transmission 無網路編碼四種情形下所比較的結果(子圖(a))所呈現的是封包到達率(Packet Delivery Ratio)；子圖(b)所呈現的是傳送次數)。

在圖 4-6，成功率  $\alpha=0.5$ ，和模擬 50 nodes 分佈於 50m×50m 類似，在封包遺失機率 0.05 下，K-based transmission\_NC 的方式相較於無網路編碼的方式，封包到達率略低，然而隨著封包遺失機率的上升，當  $p=0.2$  以後，加入網路編碼的方式的封包到達率(圖 4-6(a))漸漸與無網路編碼方式差異增大，到  $p=0.5$  時，加入網路編碼的方式的封包到達率高達 60%，無網路編碼方式的封包到達率只有 43%，使得網路編碼方式的封包到達率在  $p=0.5$  時，高於無網路編碼方式的封包到達率 17%。但由於緩衝器(Buffer)大小的影響，使得在加入網路編碼方式且 Buffer 為 120 的封包到達率在較低的封包遺失率下效果低於其他三種情形。由於節點會檢查緩衝器(Buffer)中是否擁有相同的封包並且刪除相同的封包，導致使用網路編碼累贅(Redundancy)增加，使得在低遺失機率時所需付出的傳送次數(圖 4-6(b))遠高於無網路編碼的方式。然而在高遺失機率的情形下，加入網路編碼方式的傳送次數只略高於無網路編碼方式。

而圖 4-7、圖 4-8、圖 4-9 以及圖 4-10，類似於 50m×50m 的地圖時，但加入網路編碼方式且 Buffer 大小為 120，由於節點數增加 Buffer 大小相較於在 50m×50m 的地圖影響更大，在封包遺失率 0.05、0.1 時反而因為封包過多接收點無法

成功接收其封包，導致最後匯集點無法成功解碼，使得整體的封包到達率反而不如無網路編碼方式。無網路編碼方式，隨著傳送次數的增加使得封包到達率漸漸提升，直到成功率  $\alpha=0.9$  時，封包到達率已經高達 89%(圖 4-10(a))，加入網路編碼方式，由於所增加的累贅並使得此方法得到助益。然而由於所增加的累贅使得所付出的傳送次數(圖 4-10(b))遠高於無網路編碼方式。

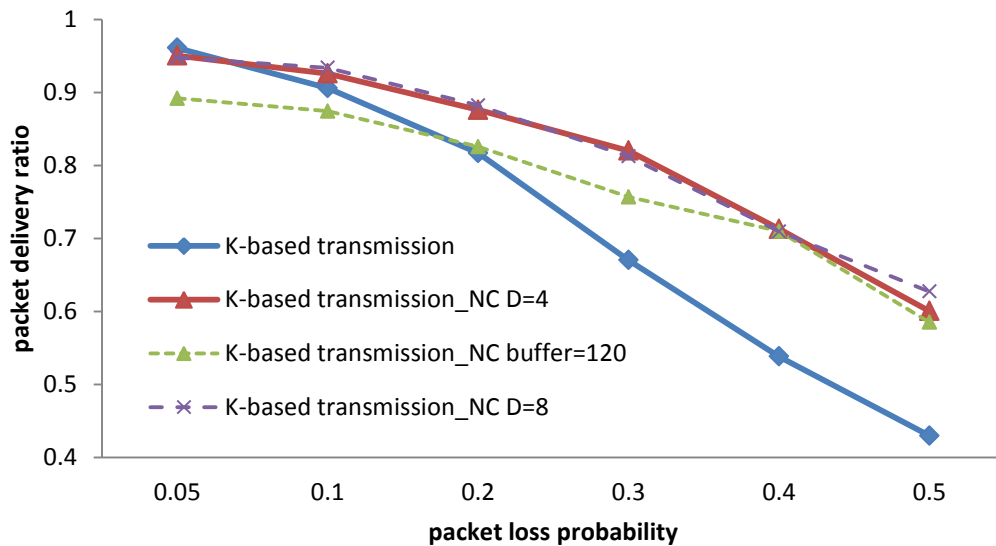


圖 4-6(a)  $\alpha=0.5$  , 100 Nodes , Packet Delivery Ratio

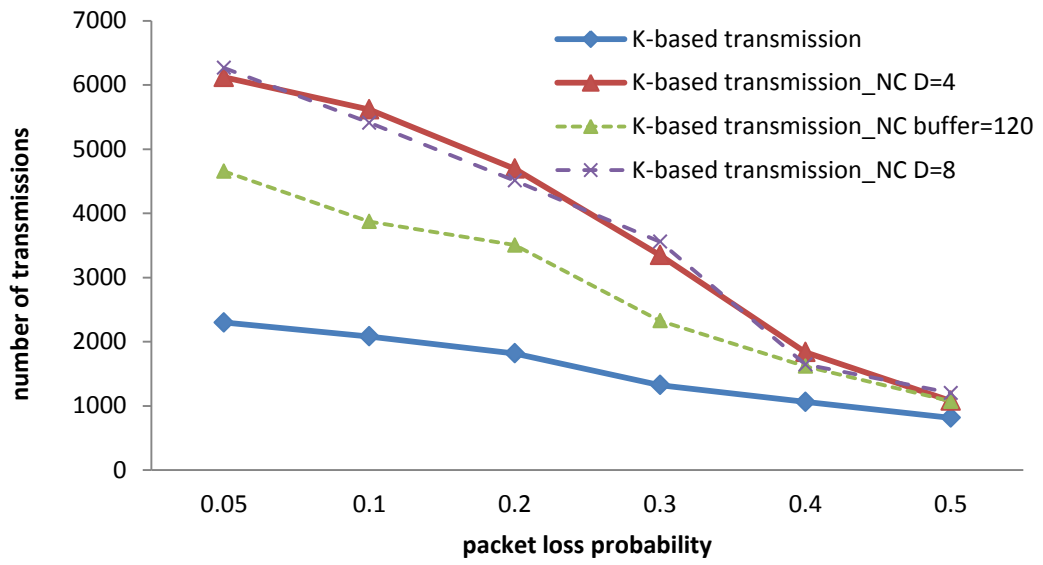


圖 4-6(b)  $\alpha=0.5$  , 100 Nodes , Number of transmissions

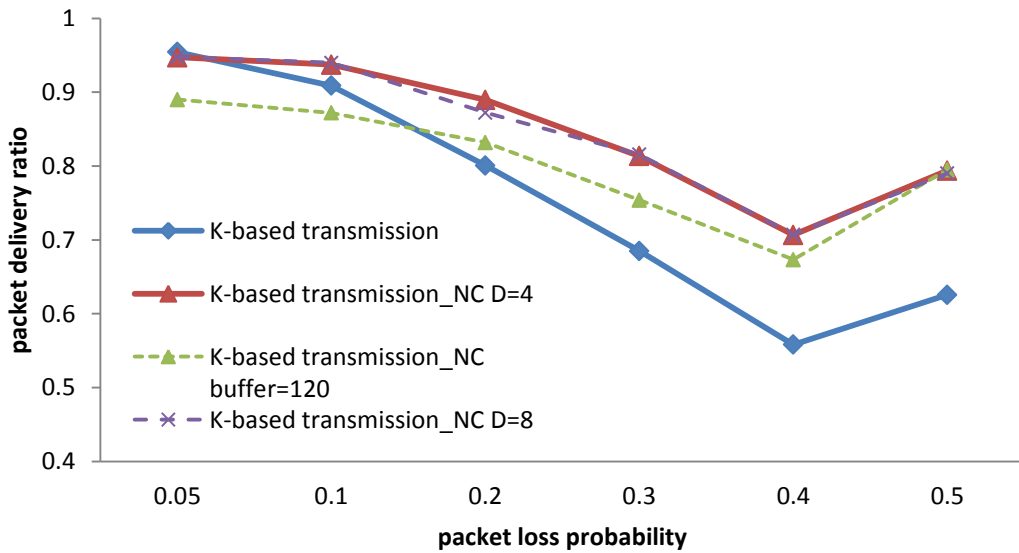


圖 4-7(a)  $\alpha=0.6$  , 100 Nodes , Packet Delivery Ratio

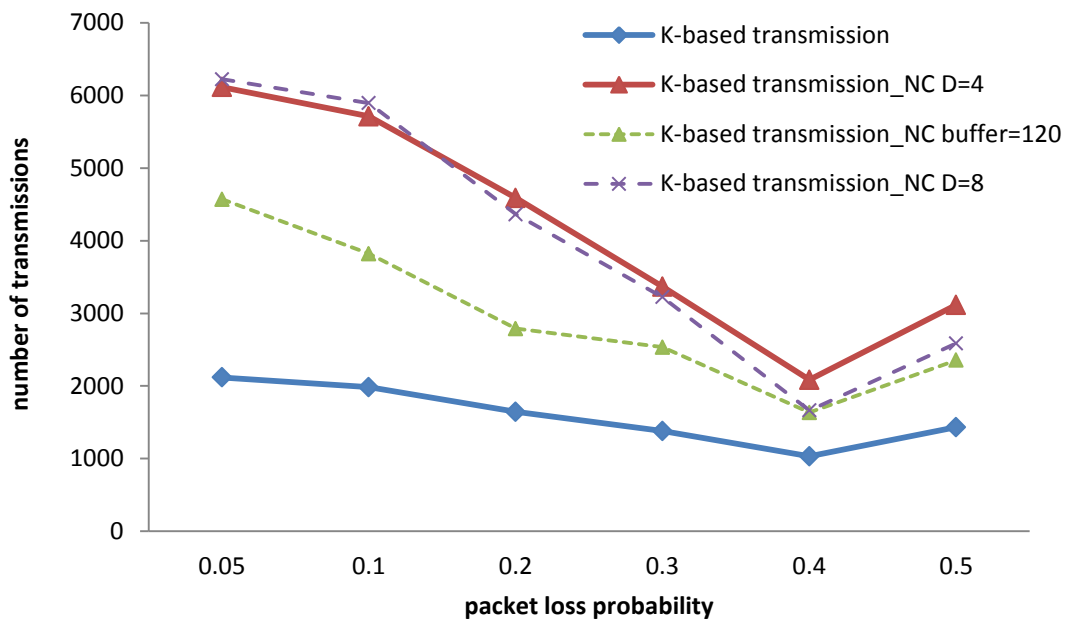


圖 4-7(b)  $\alpha=0.6$  , 100 Nodes , Number of transmissions

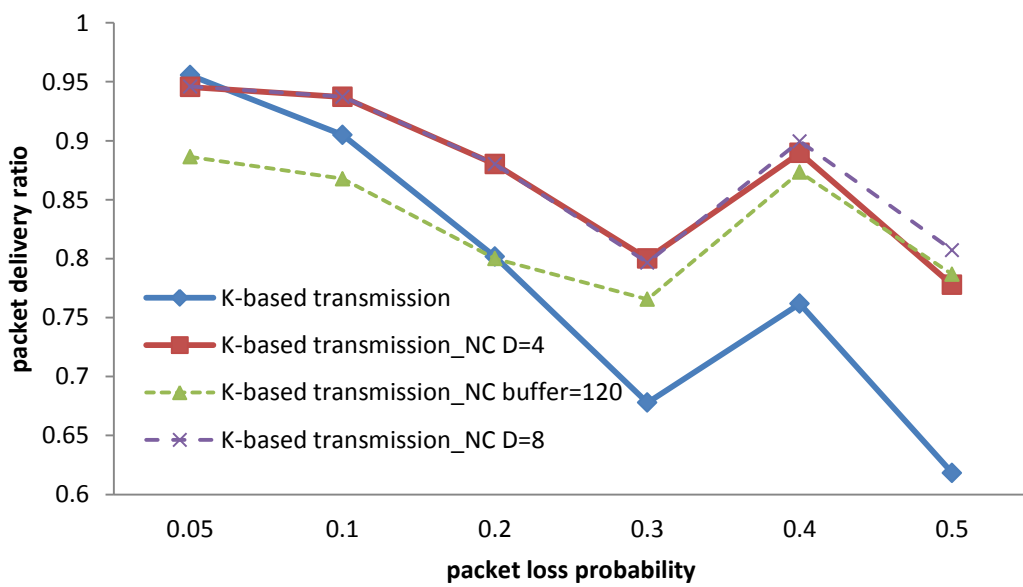


圖 4-8(a)  $\alpha=0.7$  , 100 Nodes , Packet Delivery Ratio

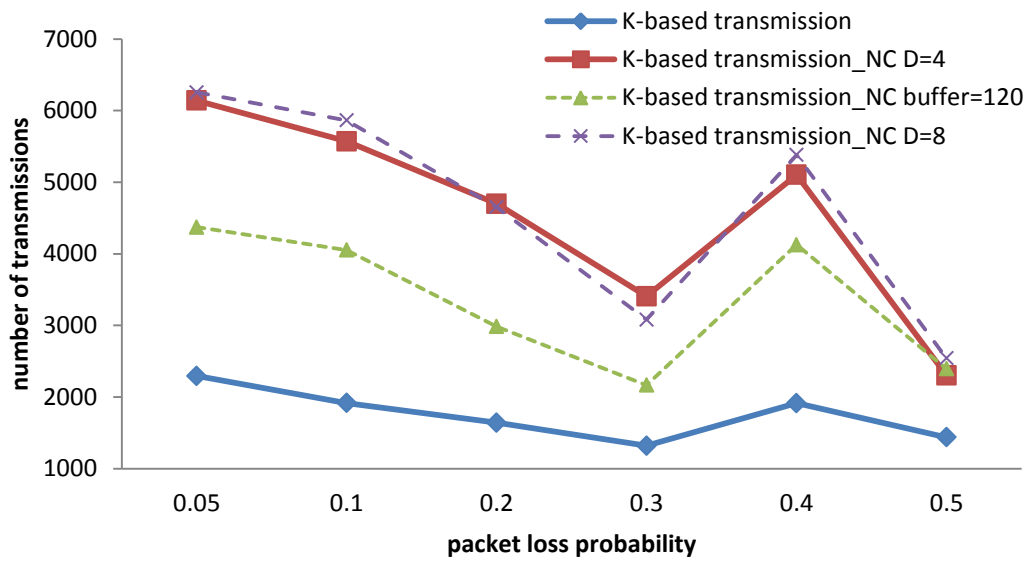


圖 4-8(b)  $\alpha=0.7$  , 100 Nodes , Number of transmissions

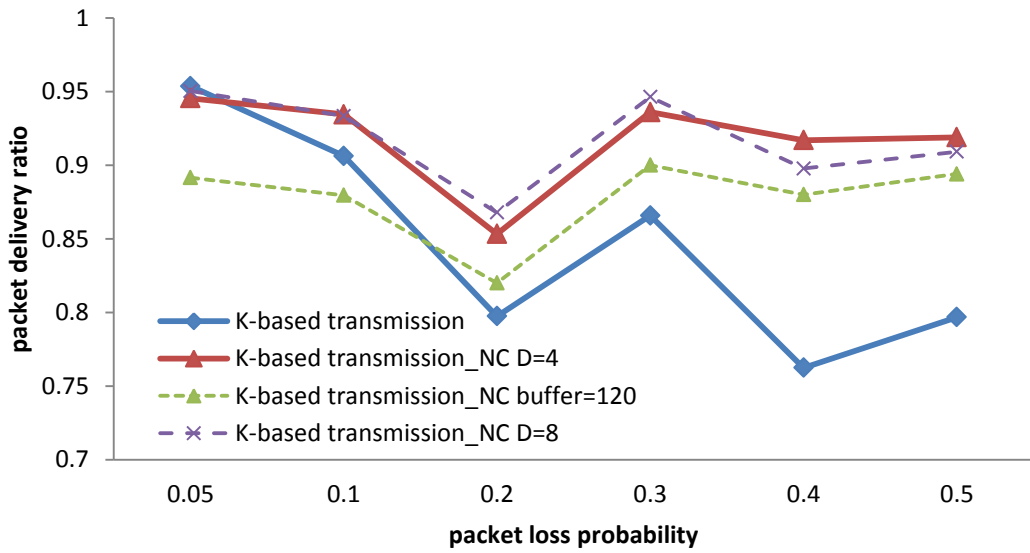


圖 4-9(a)  $\alpha=0.8$  , 100 Nodes , Packet Delivery Ratio

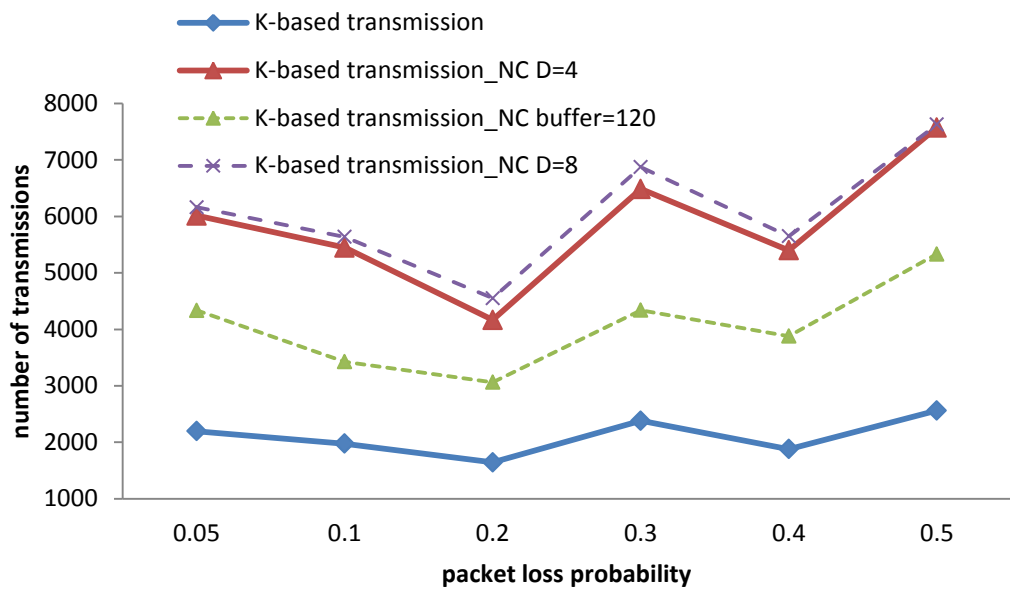


圖 4-9(b)  $\alpha=0.8$  , 100 Nodes , Number of transmissions

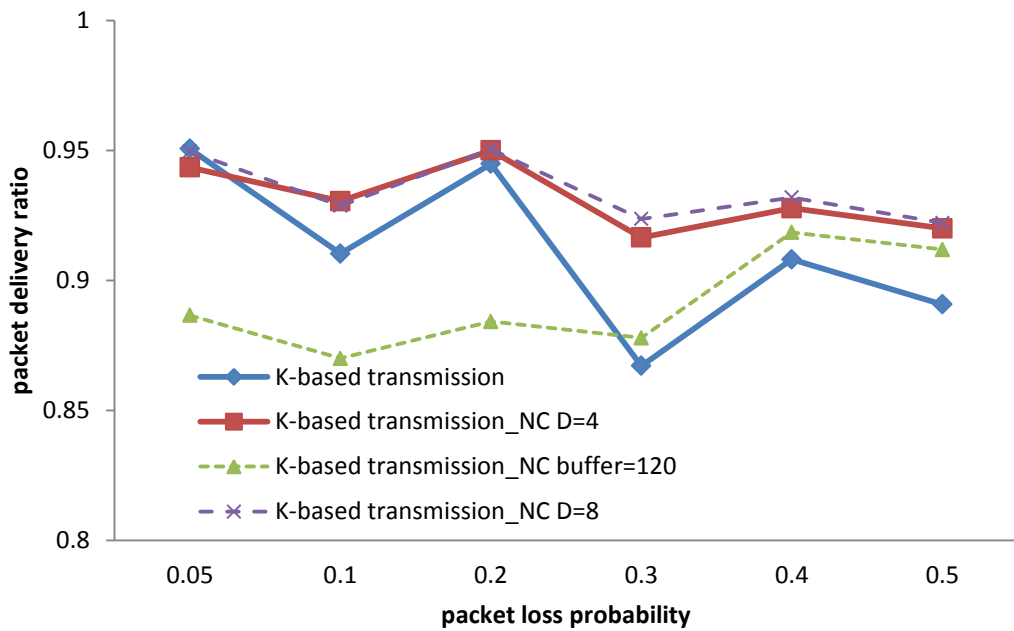


圖 4-10(a)  $\alpha=0.9$  , 100 Nodes , Packet Delivery Ratio



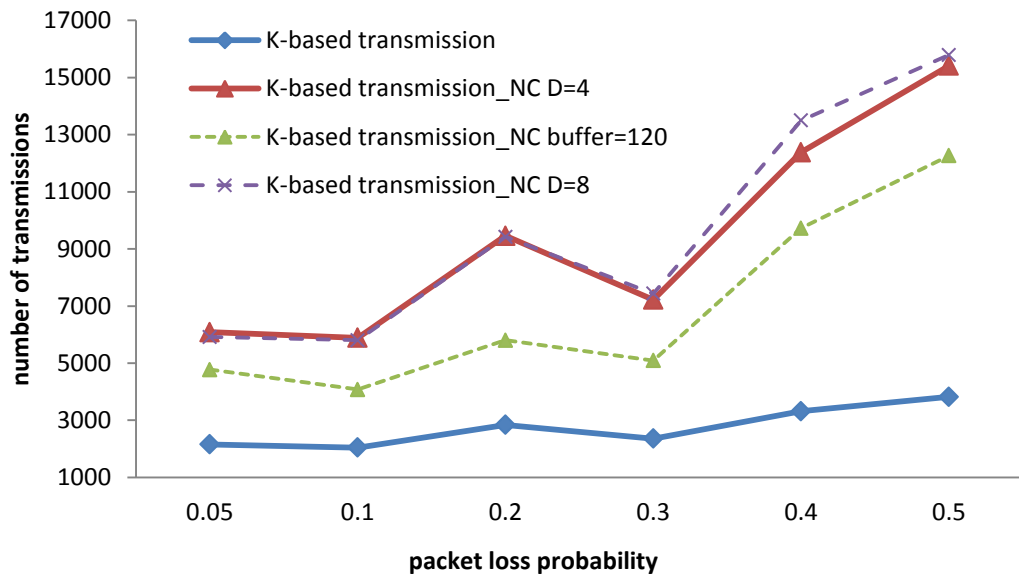


圖 4-10(b)  $\alpha=0.9$  , 100 Nodes , Number of transmissions

接著圖 4-11 以及圖 4-12，我們將展示所提出的 K-based transmission 與 One-based transmission 的模擬實驗數據，其中我們比較成功率在 0.5、0.6、0.7 時，K-based transmission 與 One-based transmission 的封包到達率以及全部節點的傳送次數。

相關參數設定如下表 2 所示。

地圖大小	50m×50m	100m×100m
地圖數量	100 張	100 張
傳送半徑	r=10m	r=10m
感測器數量	50	100
模擬回合	1000 回(每回合每個節點 傳送一次資料)	1000 回(每回合每個節點 傳送一次資料)
Packet loss ratio	0.05、0.1、0.2、0.3、 0.4、0.5	0.05、0.1、0.2、0.3、 0.4、0.5
Buffer size	120	120

表格 2 模擬實驗相關參數設定

## ■ 50 Nodes, 50m×50m

在圖 4-11，為 50 個節點分佈於 50m×50m 的地圖範圍。圖 4-11(a)中當  $\alpha$  值在 0.5 時，K-based transmission 與 One-based transmission 的封包到達率相差無幾，然而隨著所要求的成功率  $\alpha$  值的提升使得 K-based transmission 的封包到達率提升。但所要求的成功率  $\alpha$  值得提升也伴隨著傳送次數(圖 4-12(b))的提升，K-based 的全部傳送次數自然也提升不少， $\alpha$  值高達 0.7 封包遺失率為 0.5 時所需的傳送次數 One-based transmission 的 2 倍。

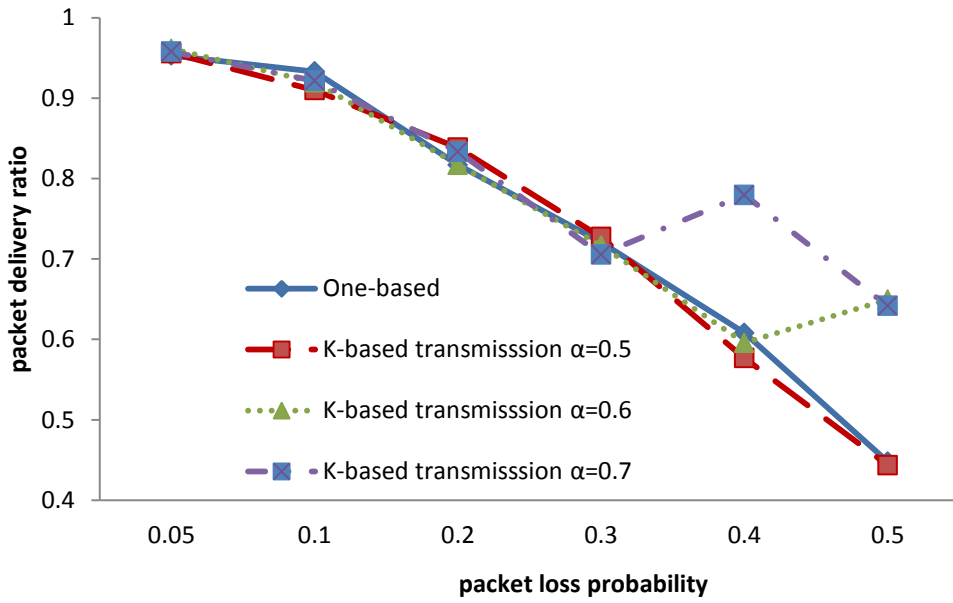


圖 4 -11(a) 50 Nodes, One-based v.s.K-based transmission , Packet Delivery Ratio

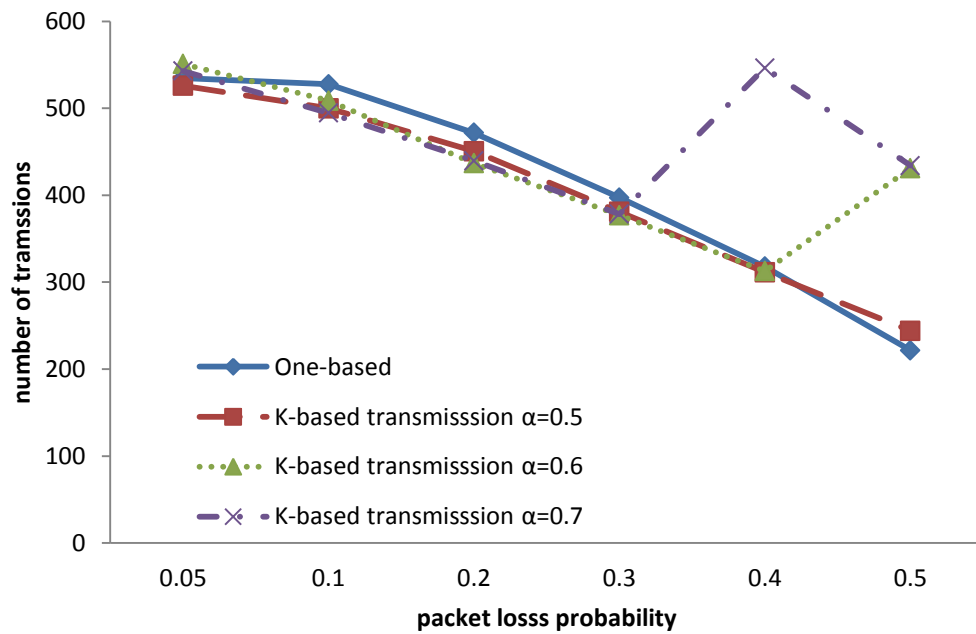


圖 4 -11 (b) 50 Nodes, One-based v.s.K-based transmission , Number of Transmissions

## ■ 100 Nodes, 100m×100m

圖 4-12，為 100 個節點分佈於 100m×100m 地圖範圍之內，相似於 50 個節點時，圖 4-12(a)中當  $\alpha$  值在 0.5 時，K-based transmission 與 One-based 的封包到達率相差無幾，然而隨著所要求的成功率  $\alpha$  值 0.6 以及 0.7 使得 K-based transmission 在封包遺失率 0.4 以及 0.5 時的封包到達率相對於只傳送一次提升許多。然而由於節點的傳送次數上升，使的全部傳送次數在遺失機率 0.4 以及 0.5 時提升許多(圖 4-12(b))。

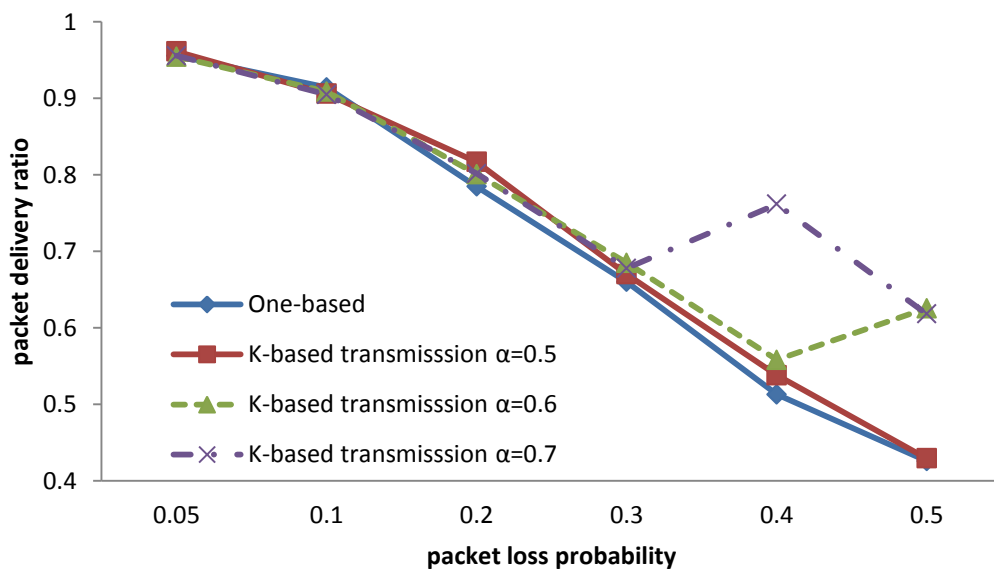


圖 4-12(a) 100 Nodes, One-based v.s.K-based transmission, Packet Delivery Ratio

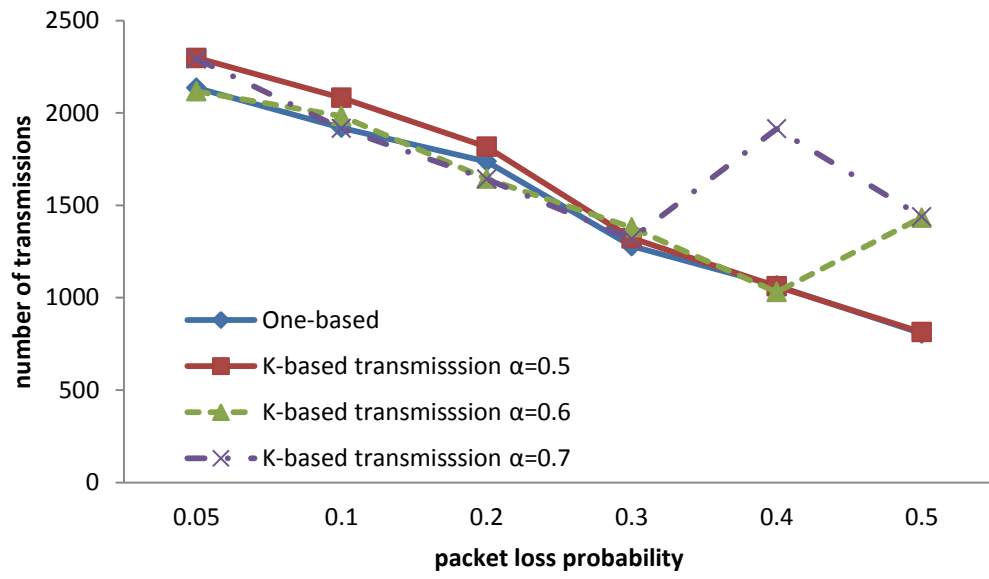


圖 4 -12(b) 50 Nodes, One-based v.s.K-based transmission , Number of Transmissions

## 第五章 結論與未來研究方向

本篇論文主要的貢獻是透過一個簡單的方式，提供一個利用網路編碼並且結合傳送次數的變更概念來提升無線感測網路的可靠性。Level-Based 路由協定，是以集中式的方式由匯集點主動發散訊息建立傳送表(Forwarding Table)。並且再根據建立好的傳送表，得知所希望能夠成功收到的鄰居個數，決定傳送次數，並且若有中繼點所收到的封包數超過 2 個以上時，可藉由網路編碼方式，先行編碼再傳送至鄰居。在此模擬實驗中，使用 K-based transmission 結合網路編碼，在所需的成功率  $\alpha=0.5$  至 0.9 時，在一開始的封包遺失機率 0.05 的情形下，無網路編碼的封包到達率將略高於加入網路編碼的方式。然而當遺失機率提升時，使用網路編碼的方式在  $\alpha=0.5$  時的封包到達率將高於無使用網路編碼的方式 17%(100 nodes)。然而由於網路編碼所增加的累贅(Redundancy)，使得他所付出的傳送次數也高於無網路編碼的方式。但在較高的封包遺失機率並且所要求的成功率 0.5 的情形之下，使用網路編碼將有效提升其封包到達率，並且不會增加太多的傳送次數。

根據實驗結果，傳送次數增加雖然能提升封包到達率，但同時也需付出較多的傳送次數，而網路編碼的方式的傳送次數也依據各別節點的傳送次數的提升而大幅上升，在感測器有限的電量下，傳送次數過多易造成感測器的失效，如何在可靠性的問題以及傳送次數之間達到一個平衡點將是一個可以努力的目標。

## 參考著作

- [1] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal, "Wireless sensor network survey", *Computer Networks*, 52(12):2292-2330, 2008.
- [2] Stefan Dulman, Tim Nieberg, Jian Wu, and Paul Havinga, "Trade-Off between Traffic Overhead and Reliability in Multipath Routing for Wireless Sensor Networks", in *Wireless Communications and Networking*, vol.3, pp.1918 - 1922, March 2003
- [3] Lorenzo Keller, Emre Atsan, Katerina Argyraki, and Christina Fragouli, "SenseCode: Network Coding for Reliable Sensor Networks", Oct 2009.
- [4] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris, "A high-throughput path metric for multi-hop wireless routing", Proceedings of the 9th annual international conference on Mobile computing and networking, pp. 134-146., 2003
- [5] Deb, B., Bhatnagar, S., Nath, B., "ReInForM: reliable information forwarding using multiple paths in sensor networks", in *Proceedings. 28th Annual IEEE International Conference on Local Computer Networks*, 406 - 415, Oct. 2003.
- [6] Crisostomo, S., Barros, J., Bettstetter, C., "Flooding the Network: Multipoint Relays versus Network Coding", in *4th IEEE International Conference on Circuits and Systems for Communications*, 119 – 124, May 2008
- [7] Deepak Ganesan, Ramesh Govindan, Scott Shenker, and Deborah Estrin, "Highly-resilient, energy-efficient multipath routing in wireless sensor networks", in *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol 5 no 4, pp. 11 – 25, October 2001.
- [8] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin, John Heidemann, and Fabio Silva, "Directed Diffusion for Wireless Sensor Networking", in *IEEE/ACM Transactions on Networking*, Vol 11 no 1, pp. 2 – 16, Feb 2003.
- [9] Kemal Akkaya, and Mohamed Younis, "A survey on routing protocols for wireless sensor networks", *Ad Hoc Networks*, Vol 3, no. 3, pp. 325-349, May 2005.
- [10] Christina Fragouli, Jean-Yves Le Boudec, and Jörg Widmer, "Network coding: an

instant primer”, in *ACM SIGMOBILE Mobile Computing and Communications Review*, January 2006.

[11] Li, S.-Y.R.,Yeung, R.W.,Ning Cai, “Linear network coding”, in *IEEE Transactions on Information Theory*, Vol 49, no 2, pp. 371 – 381, Feb. 2003.

[12] W. Heinzelman, J. Kulik, and H. Balakrishnan, “Adaptive protocols for information dissemination in wireless sensor networks,” in Proc. MOBICOM , pp. 174–185,1999.

[13] Sukun Kim,Fonseca, R.,Culler, D.,”Reliable Transfer on Wireless Sensor Networks”, *Sensor and Ad Hoc Communications and Networks*, 449 - 459 ,Oct. 2004.

[14] Philipp M. Glatz, Reinhold Weiss,” LINDONCS: localized in-network detection of network coding structures in wireless sensor networks”, PM2HW2N '09 Proceedings of the 4th ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks,2009