

楊欣哲、陳信仲 (2019),『於 5G mmWave 網路架構上設計新的 RR 排程方法以改善處理績效之研究』, 中華民國資訊管理學報, 第二十六卷, 第三期, 頁 349-378。

於 5G mmWave 網路架構上設計新的 RR 排程方法以改善處理績效之研究

楊欣哲*

東吳大學資訊管理學系

陳信仲

東吳大學資訊管理學系

摘要

近年來雲端運算技術的成熟與感應裝置的快速發展,物聯網(IoT)應用的普及,2020年IDC在IT預測中IoT的感應裝置可能成長至300億個,這些無所不在的感應裝置所產生的大量資料將使現有的4G LTE網路流量不敷使用,因此催生了下一代5G行動通訊系統,5G可提供比4G LTE更高的網路流量及更低的網路延遲,只是5G以10Gbps資料傳輸速度供給數以萬計的使用者,但仍需要妥善分配無線行動網路之頻寬資源,因此,如何在有限的頻寬下提供較多使用者可靠的與較佳的資料傳輸是當前重要研究的議題之一。於5G網路之B-RR執行資源分配時會面臨到CQI(Channel Quality Indicator)不存在仍使用大量的時間嘗試進行資料傳輸,以及此演算法中關於處理UE的迴圈設計效率不彰的問題,容易造成UE資料傳輸時的網路資源損耗,因此我們提出新的NRRS排程方法以改善B-RR以達成資源的分配較佳化。NRRS方法著重在依CQI報告積數中作為資源分配的依據,其中並整合NOMA的UE配對策略以有效利用網路資源。我們進行模擬實驗並進行NRRS與B-RR之績效比較,於10UEs、20UEs、50UEs、100UEs的實驗結果,其關鍵績效指標(KPIs)顯示,減少Cell延遲的改善率達到27%,11%、15%和13%,在處理能量的改善率達到2%、26%、25%、14%,以及資料錯誤率的改善也有19%、24%、33%和16%,因為資料錯誤率的降低讓資料錯誤重傳次數也下降28%、18%、16%和22%。總之,模擬實驗結果顯示NRRS排程方法相對於B-RR有較低的Cell延遲、較好的處理能量、較低的資料錯誤率及較低的資料錯誤重傳次數。本文所提出的NRRS方法於5G核心網路上有較佳的頻寬資源的分配機制,以及減少傳輸資料錯誤的發生有極大的助益。

關鍵詞: NRRS、無線行動網路、B-RR、5G、4G LTE

* 本文通訊作者。電子郵件信箱: sjyang@csim.scu.edu.tw
2019/03/22 投稿; 2019/05/27 修訂; 2019/06/19 接受

Yang, S.J. and Chen, H.C. (2019), 'Design issues of novel round-robin scheduling scheme to improve processing performance on the 5G mmWave network', *Journal of Information Management*, Vol. 26, No. 3, pp. 349-378.

Design Issues of Novel Round-Robin Scheduling Scheme to Improve Processing Performance on the 5G mmWave Network

Shin-Jer Yang*

Department of Computer Science and Information Management, Soochow University

Hsin-Chung Chen

Department of Computer Science and Information Management, Soochow University

Abstract

Purpose—5G mobile networks provide higher network traffic and lower packet latency. However, 5G can transmit data to thousands of users at a speed of 10 Gbps, but it still requires proper allocation scheduling of bandwidth resources on the 5G mobile networks. Therefore, this paper proposes a new NRRS scheme to achieve more efficient bandwidth resource allocation in the 5G mmWave mobile network.

Design/methodology/approach—The proposed NRRS scheme focuses on the use of a CQI report value as the basis for bandwidth resource allocation and with combination of NOMA UE pairing method. Also, we did perform some simulations to compare and analyze their performance between NRRS and B-RR schemes.

Findings—The experimental results of KPIs indicate that the Cell latency can achieve 27%, 11%, 15%, and 13% improvement ratios; Throughput can attain 2%, 26%, 25%, 14% improvement ratios; the Data failure ratio can get 19%, 24%, 33%, and 16% improvement ratios; and also Data Failure Retransmission can obtain 28%, 18%, 16%, and 22% improvement ratios in 10UEs, 20UEs, 50UEs, and 100UEs, respectively. Finally, the simulation results have proven that the NRRS scheduling scheme can obtain a lower Cell latency, higher Throughput and lower Data failure ratio, which are highly conducive to allocating better bandwidth resources and more efficient data retransmission

* Corresponding author. Email: sjyang@csim.scu.edu.tw
2019/03/22 received; 2019/05/27 revised; 2019/06/19 accepted

in the 5G mmWave mobile network.

Research limitation/implications — Although the transmission speed in 5G mobile network reaches 40 times that of 4G LTE, there is high data traffic on mobile devices or IoT devices which have very high data traffic. Also, the 5G is still in a rapid growing development and bulky data may not be transmitted within a short period of time. Therefore, a better scheduling method is needed to allocate bandwidth resources. A better scheduling method can fairly and quickly allocate mobile network resources to prevent excessive occurrences of retransmissions and put mobile network resources into efficient usages. The main objective of this paper is to only utilize the 5G mmWave mobile network to perform experiments and design novel round robin scheduling to achieve more efficient resource allocation.

Practical implications — The research results have proven that the NRRS scheme can obtain a lower Cell latency, higher Throughput, and lower Data failure and retransmission ratios than B-RR method. Practically, the proposed NRRS are highly conducive to allocating better bandwidth resources and more efficient data transmissions in the 5G mmWave mobile network.

Originality/value — The inefficient issue of data transmission during the Channel Quality Indicator (CQI) value is nonexistent, when B-RR executes resource allocation on the 5G mmWave network. Also, the UE polling continues to be once happened is likely to be encountered, resulting in network resource loss during UE data transmission. Therefore, a new NRRS scheme is proposed in this paper to enhance the B-RR, in order to achieve more efficient bandwidth resource allocation and obtain higher throughput on the 5G mmWave mobile network.

Keywords: NRRS, Mobile Communications Network, B-RR, 5G, 4G LTE

壹、緒論

一、研究背景與動機

雲端運算科技是一種透過網際網路互連使用者裝置與核心應用網路的方式，透過這個方法可以讓世界各地的使用者端裝置傳送資料到核心應用網路，或是由核心應用網路下載資料給使用者端的裝置，近年來使用者端裝置如手機及智慧型穿戴裝置已成為人們生活中不可或缺的產品，這些行動設備都有直接或間接連接無線行動網路的需求並與核心應用進行資料交換，網路使用趨勢也顯示手機使用網路流量的份額逐年提升（Meeker 2015; Andrews et al. 2014），主因是手機輕便可攜以及可以隨時上網特性可以取代筆記型電腦之功能，加上雲端運算科技的應用愈來愈多所以需要透過無線行動網路即時交換訊息與資料與日俱增，雲端的資料可以說就是 Big Data，IoT 則是另一個推升 Big Data 和無線行動網路演進的重要推手，IoT 感應裝置收集的資料則促進 Big Data 的分析更準確，Big Data 能夠分析讓 IoT 感應裝置收集到的資料呈現更大的價值，IDC 的預測中預測至 2020 年將有 300 億個 IoT 感應裝置（Report IDC FutureScape 2018），這些 IoT 感應裝置都需要更快速穩定的網路將大量資料由 IoT 層傳給雲端上的 Big Data 應用平台，例如從廣大的農地將農業種植資料傳送到雲端儲存分析，或是智慧型手錶經由嵌入式用戶身份識別模組 eSIM（Embedded-Subscriber Identity Module）卡可以直接連線到無線行動網路將資料傳輸到雲端，現階段 IoT 的發展有部份會利用霧運算層對大量原生資料（Raw Data）先進行初步處理精簡的步驟，但是各領域的應用爆炸性的成長還是會造成未來超大量資料需要經由無線行動網路傳輸。

4G LTE 的發展已經達到成熟且穩定，現有的 4G LTE 提供了 300Mbit/s 的下載速率和 75Mbit/s 的上傳速率，支援的頻率為 1.4MHz 至 20MHz 的頻段，但根據國際數據資訊顧問公司 IDC（International Data Corporation）在 Worldwide IT Industry 2018 Prediction 報告中說到關於 IoT 感應裝置的快速成長（Report IDC FutureScape 2018），至 2020 年 IoT 感應裝置可能成長至 300 億個，這些無所不在的裝置將會傳送大量的資料至雲端，所以對於未來無線行動網路的高強度需求來說，4G LTE 已經不敷使用，5G 的無線行動網路因其較高的網路處理速度及較低的延遲所以成為下一代移動通訊的技術標準，首先 5G 將採用的 28GHz 超高頻頻段有更多的連續空白頻譜可用，極高頻的頻譜寬度較高因此可提供較快的傳輸速度，限制干擾因素的條件之下應該可以達到 4G LTE 的 40 倍，再來是網路的延遲時間應該是在 1ms 以下以提供更好的資料傳輸速度，下一代行動網路聯盟（Next Generation Mobile Network Alliance）提到 5G 是一完全連接個人行動裝置及社會群體的生態（NGMN Alliance 2015），經由整合已存在及新興的使用實例提供相同

的體驗，並賦與客戶和合作夥伴創造價值的能力，用以實現永續經營的商業模式，其中並列出 5G 架構在傳輸量和網路延遲方面的要求：

1. 可以在人群擁擠的區域例如體育館等提供數十 Mbps 的速率給數以萬計的人使用，也就是應該要可以達到大約 10Gbps 的資料傳輸速率提供給上萬的使用者同時使用。
2. 以 1Gbps 的資料傳輸速率提供同一樓層的使用者。
3. 5G 應比 4G LTE 有更低的延遲。

為了達成以上的需求，極高頻的 mmWave (Millimeter Wave) 行動網路架構因為具有高傳輸量和低網路延遲的能力 (Mezzavilla et al. 2018; Ford et al. 2016) 就被提出，NOMA (Non-Orthogonal Multiple Access) 則因為可以補足 OFDMA (Orthogonal Frequency Division Multiple Access) 在頻段切換給不同的 UE 使用造成的延遲 (Wei et al. 2016; Ding et al. 2016)，是滿足 5G 要求的一個重要的架構，但是因為 NOMA 的模擬程式難以取得及全面結合到 mmWave 網路架構上，所以本研究的目標是使用 5G mmWave 行動網路架構並整合 NOMA 的 UE 配對策略進行實驗並提出 B-RR (Basic-Round Robin) 排程器的優化以期增加資料處理量、減少資料重新傳輸的次數，以期讓使用者得到較佳的網路體驗。

排程方法在無線行動網路的技術架構上是一個重要的資源分配及處理的機制，B-RR 在各種不同的排程器中雖然效能不是最佳但是有極佳的公平性，所以本論文提出 NRRS (Novel Round Robin Scheduling) 做為改善 B-RR 在無線行動網路 5G mmWave 排程頻寬資源的方法。

二、研究範圍與目的

如前所述雖然 5G 無線行動網路的傳輸速度可以達到 4G LTE 的 40 倍，但是行動裝置及 IoT 裝置的資料傳輸量極大而且正處在快速成長期，這些極大的資料量無法在短時間內傳輸完成，所以需要一個好的排程方法來分配頻寬資源，好的排程方法可以公平且快速地分配無線行動網路資源，防止過度的重傳發生並有效率的利用無線行動網路的資源，本研究的主要是使用 5G mmWave 行動網路架構進行實驗並改善 RR 排程器：B-RR (Basic Round Robin) 以提出新的排程方法以期增加資料處理量、減少資料重新傳輸的次數。總之，本論文主要的研究目的如下：

1. 解釋說明 5G mmWave 和 NOMA 的運作原理，並列出 B-RR 可能導致無線行動網路上資料傳輸的問題。
2. 針對 B-RR 在無線行動網路的網路品質不佳時也花費大量資源嘗試資料傳輸及處理 UE 的迴圈設計效率不彰的問題提出新的 NRRS (Novel Round

Robin Scheduling) 演算法，以改善 5G mmWave 下的 B-RR。

3. 利用 5G mmWave 行動網路架構及實現部份 NOMA 功能模擬無線行動網路系統，來模擬本論文所提出的 NRRS 演算法與 B-RR 實際運行環境，以驗證透過 NRRS 演算法能夠有效地減少 Cell 延遲，降低資料錯誤率進而減少資料錯誤重傳次數，並有提佳的處理能量。
4. 總之，於 5G mmWave 行動網路架構上 NRRS 將改善 B-RR 以提升資料的傳輸量及整體行動網路之處理績效。

三、章節結構

本論文的章節架構可分為六章，首先第壹章緒論中說明研究背景與研究目的。第貳章中說明與探討 5G 相關技術與其應用，研讀有關 5G mmWave 和 NOMA 的運作原理與網路架構、以及網路模擬器等相關文獻。第參章將提出研究架構及方法，瞭解 5G mmWave 中的 B-RR 排程方法並找出 B-RR 的資料傳輸和效率瓶頸，針對 B-RR 的效率瓶頸加以改善並加入部份 NOMA 的功能，進而提出新的排程方法：NRRS 並設計其演算法。第肆章說明提出模擬環境建置及進行模擬實驗，設計模擬程序與關鍵績效指標並加以說明。第伍章為研究結果與分析，取得模擬測試之關鍵績效指標 (KPIs) 數值，繪製視覺化分析圖形進行評估 NRRS 與 B-RR 排程方法，並作績效比較與模擬結果分析。最後，第陸章為結論，說明本文所提出的 NRRS 排程方法之研究成果與實際貢獻，以及未來研究方向。

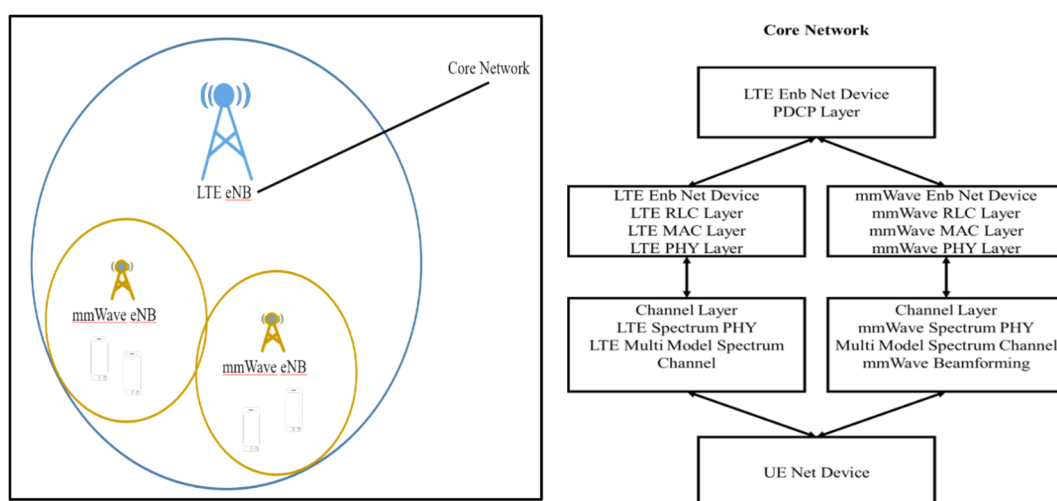
貳、文獻探討及相關研究

一、無線行動網路的趨勢與演進

無線行動網路是一種在移動設備上使用的無線寬頻技術系統，2G (2nd Generation) 行動寬頻系統做為第一代的無線行動網路出現於 1991 年，之後 2001 年出現 3G (3rd Generation) 行動寬頻系統並於 2006 年演進為 4G LTE 行動寬頻系統，2009 年提出為資料傳輸速度更快的 4G LTE Advanced 行動寬頻系統並於 2011 成為 LTE 的增強標準，上列行動寬頻系統使用的頻率皆為 225MHz 至 3700MHz 的頻段，2011 年以來由於各項雲端運算科技與應用的增加和普及、移動設備的快速成長及大量 IoT 裝置的佈署等應用，超大量資料透過無線行動網路傳輸的趨勢已經成形，因此世界上先進國家無不以將無線行動網路由 4G LTE 升級至 5G 為主要的電信產業升級措施，下一代行動網路聯盟在所發表的白皮書中認為 5G 的服務將於 2020 年會陸續推出 (NGMN Alliance 2015)。

本論文提到的 5G 行動網路架構使用的 mmWave 是 3GPP (3rd Generation

Partnership Project) 開發聚焦的重點項目 (3GPP 2018)，同時 mmWave 也由其獨特的傳播技術與可以實現未來無線行動網路所需要的超大量資料傳輸需求 (Mezzavilla et al. 2018; Ford et al. 2016)，mmWave 網路架構堆疊需要至少建立一個 eNB，eNB 與 Gateway 的連接是點對點 (Point to Point) 的方式，該網路架構上每一個 Node 都使用 IPv4 協定分配 IP，mmWave eNB 間和 LTE eNB 間的連接介面是 X2 interface，LTE eNB 與核心網路的連接介面則是 S1 interface，mmWave eNB 都是透過 LTE eNB 將訊號傳給核心網路 (Mezzavilla et al. 2018; Ford et al. 2016)，5G mmWave 網路架構圖如圖 1 所示，圖 1(a)是元件分佈的示意圖表示 mmWave 最終還是透過 LTE 將資料傳送到 Core Network，而圖 1(b)則說明 mmWave 實際上使用 LTE 的 PDCP Layer 和 Core Network 溝通，另外需說明的是因為 UE 的裝置有提供 DC (Dual Connectivity) 功能所以可同時連接到 LTE 和 mmWave 的網路。



(a) 5G mmWave 元件分佈示意圖

(b) 5G mmWave 元件關聯關係圖

圖 1：5G mmWave 網路架構圖

關於 5G mmWave 的網路框架架構說明如下，每一個 Frame 的長度是 1ms，每個 Frame 可以切分成 10 個 Subframes,每個 Subframe 都可得到 100 μ s，這 100 μ s 會再分給 24 個 symbols，每個 symbol 得到 4.16 μ s，在頻率方面每一個 symbol 有 1GHz 的頻寬，可以再細分成 72 個 subbands 每個得到 13.89MHz，每個 subband 則有 48 個 subcarriers 如圖 2 所示，我們就可將 UE 的資料放到這些 subcarriers 上進行資料傳輸，所以待下載處理的 UE 資料就可以依照排程方法利用 Frame 將資料傳輸給 UE (Mezzavilla et al. 2018)。這樣的方式和 CPU 分時處理資料的架構非

常類似，也會使用到和相近的 CPU 排程方法進行資料的傳輸。

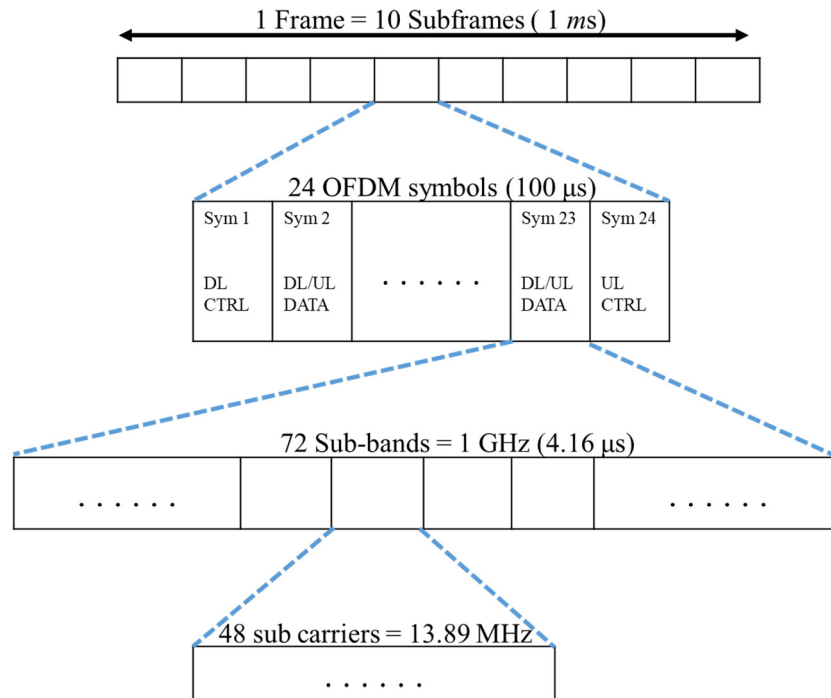


圖 2：5G mmWave 框架架構圖

另一個重要的架構是 NOMA 其使用的是非正交多存取，NOMA 可以補足 OFDMA 因為頻段切換給不同的 UE 使用造成的延遲（Wei et al. 2016; Ding et al. 2016），但是因為 NOMA 的 ns-3 的模擬程式尚未發佈，而且也沒有較有公信力的機構發佈測試模組無法進行模擬測試，所以決定使用原本的 mmWave 測試模組（ns-3 mmWave）實現部份 NOMA 的 UE 配對策略功能，並以 CQI 報告取代功率做為子載波選擇 UE 的依據，選擇 CQI 報告做為選擇依據的原因是 CQI 值和資料傳輸品質（QoS）有關，愈高的 CQI 值代表 QoS 愈好且使用該 UE 的使用者體驗愈好。依照 NOMA 的 UE 配對策略（Song et al. 2017; ZHANG et al. 2017; NAQVI and HASSAN 2016），採用無線行動網路信號品質最好的 UE 會和最差的 UE 配對方法，配對後共用子載波進行資料傳輸，品質較好的 UE 會在每一次的傳輸時被視為是主要用戶，而品質較差的 UE 則不被視為是主要用戶，因此品質較好的 UE 會被分配給較多的無線行動網路資源，藉以提升整個無線行動網路資料傳輸效率。

二、無線行動網路元件

我們提出的 NRRS 是在行動寬頻網路的 RLC 層至 PHY 層的資料傳輸之 B-RR 的改良，無線行動網路的資料傳輸架構圖如圖 3 所示，圖 3 中相關的元件分別說明如下：

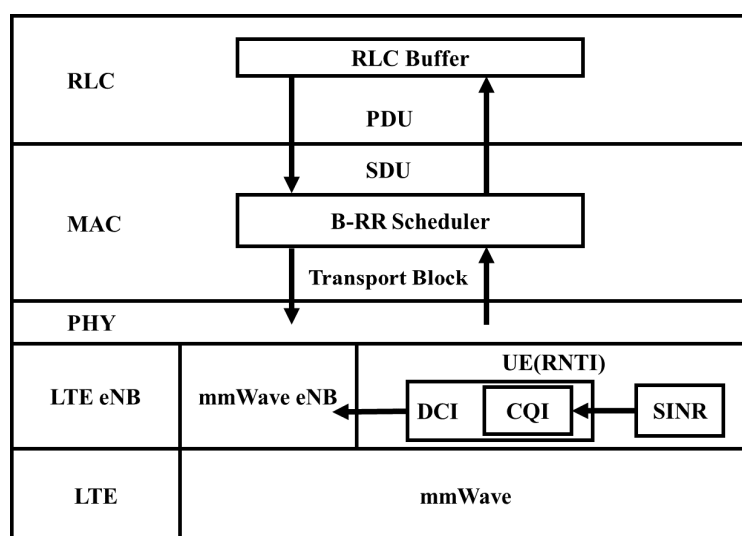


圖 3：無線行動網路的資料傳輸架構圖

1. UE (User Equipment)：UE 是 4G LTE 行動網路架構邏輯上表示為使用者終端網路節點 (Piro et al. 2011)，因 mmWave 延用並改進 4G LTE 的網路架構所以也延用了節點的名稱，在現實生活層面可對應為使用者的隨身行動設備，具備可以與電信公司的基地台連接交換訊息的設備，一般指的是使用者隨身攜帶的手機，也可能是 4G 無線網卡或是使用 eSIM 的穿戴式裝置。
2. eNB (Evolved Node B)：eNB 和 UE 同樣行動網路架構邏輯上的網路節點 (Piro et al. 2011)，是核心網路連接至 UE 的中間網路節點 (網路硬體設備)，在 5G mmWave 網路架構下可分為一般的 eNB 及 mmWave eNB (Mezzavilla et al. 2018; Ford et al. 2016)，一般的 eNB 和 4G LTE 的作用相似，其中的差異是 4G LTE 的 eNB 是與 UE 直接進行訊號的交換，但是 5G mmWave 網路架構下的 eNB 是與 mmWave eNB 進行訊號的交換，而後 mmWave eNB 再與 UE 進行訊號的交換，若是 UE 在一個範圍內無法找到符合條件的 mmWave eNB 則還是可以直接與 eNB 進行訊號的交換。eNB 與 mmWave eNB 的連接使用的是 X2 interface，而與骨幹網路之間的連接使

用的是 S1 interface。因為 5G 使用極高頻的頻段達到較高的傳輸速度，相對需要的付出就是極高頻的傳輸範圍相較 4G LTE 縮小，那這些傳輸範圍較小的 mmWave eNB 就需要透過原本 LTE 的 eNB 來和骨幹網路連接，利用 LTE 的 eNB 較大的涵蓋範圍擴大覆蓋率，這樣就可以提升傳輸速度並將有效的範圍延伸並覆蓋現有 4G LTE 的訊號涵蓋範圍。

3. RNTI (Radio Network Temporary Identifier) : RNTI 的功能是作為 UE 訊息的辨識用途，實際場景中定義有許多不同的 RNTI 類型讓 eNB 可以辨識出同一 UE 傳送的不同功能的訊號，在模擬程式中使用簡單的數字編號作為代表。
4. DCI (Downlink Control Information) : DCI 依據可辨識的 ID (RNTI) 配置實體傳輸通道上的資源，在實體傳輸通道上不同的資料區塊可能會互相參照，這個參照的訊息就會存放在 DCI，UE 是否需要反饋 CQI (Channel Quality Indicator) 也可以經由 DCI 通知，UE 收到通知之後就會在下次上傳時將 CQI 附加在 CSI (Channel State Information) 內傳送給 eNB，DCI 也負責傳送給 UE 關於資料錯誤的資訊。
5. RLC (Radio link control) : 在無線行動網路的資料傳輸上 RLC 層會進行資料的串接、分割與重組，PDCP 層的 PDU 到 RLC 層就變成 RLC 層的 SDU，此時 RLC 層會將此 SDU 切割並組成適當長度的 RLC 層 PDU 往下送給 MAC 層。反向從 MAC 層收到的資料也會重組成 RLC 層的 SDU 再傳給 PDCP 層。為防止 PDU 的遺失 RLC 層使用 ARQ 機制進行資料的重傳。
6. MAC (Media Access Control) : MAC 層會收到 RLC 層的 PDU 當作 MAC 層的 SDU，此時 MAC 層會將此 SDU 切割並組成多工的 Transport Block 傳給 PHY 層，PHY 層再將資料轉成實體的封包傳送。或是將收到 PHY 層傳來的多工的 Transport Block 解成單一或是 MAC 層 SDU 傳給 RLC 層。MAC 層還負責排程 UE 間需求的優先處理順序和資料錯誤重傳的重傳請求，優先權高的 UE 的需求會先被處理，為防止一直由處理高優先權的資料而造成低優先權的資料出現飢餓 (Starvation) 的狀況，而排程方法必須注重其資料處理的公平性。
7. SINR (Signal to Interference and Noise Ratio) : 關於 SINR 應該先提到 SNR (Signal-to-noise ratio) 訊號雜訊比，SNR 的計算是訊號強度與背景噪聲強度的比率，而 SINR 則多加了干擾強度這一因素，所以 SINR 是 UE 計算訊號強度與背景噪聲強度加干擾強度的比率。
8. CQI (Channel Quality Indicator) : 為了確保無線行動網路通訊的品質因此 5G mmWave 和 LTE 同樣使用了 CQI 的回報機制 (FemtoForum 2010)，CQI 是 CSI (Channel State Information) 三種報告中的一種，UE 會會依

SINR 產出計算出 CQI，如前段內容提到 CQI 可由 DCI 訊息觸發，CQI 值（CQI Index）是由 0~15 的數值代表訊號的好壞，0 表示訊號最差或是在可接收到的訊號範圍之外，15 則表示訊號是最好的，一般來說若通訊的品質太差可能會發生資料重傳的現象，資料重傳會在每次傳送新的資料之前執行，太多的資料重傳會佔用傳輸的頻寬導致資料傳送處理效率的降低，CQI 值和效率值（efficiency）參考自（3GPP 2017），並依效率值給予無線行動網路資源分配的百分比（Ratio）如表 1 所示，CQI index 為 0 的給一預設 Ratio 為 0.1，欲模擬實驗之 CQI 值的來源是由 mmWave 模組所產生，其產生的原理是來自 Transport Block 在 4G LTE 的錯誤計量模型（Mezzavilla et al. 2018；Mezzavilla et al. 2012）。

表 1：CQI 數值與網路資料傳輸效率關係表

CQI index	modulation	code rate x 1024	efficiency	Ratio
0	out of range			0.1
1	QPSK	78	0.1523	0.4
2	QPSK	120	0.2344	0.6
3	QPSK	193	0.377	1.1
4	QPSK	308	0.6016	1.7
5	QPSK	449	0.877	2.5
6	QPSK	602	1.1758	3.4
7	16QAM	378	1.4766	4.2
8	16QAM	490	1.9141	5.5
9	16QAM	616	2.4063	7
10	64QAM	466	2.7305	7.9
11	64QAM	567	3.3223	9.6
12	64QAM	666	3.9023	11.3
13	64QAM	772	4.5234	13.1
14	64QAM	873	5.1152	14.8
15	64QAM	948	5.5547	16.8

9. 資料錯誤重傳 (Data Failure Retransmission)：當無線行動網路資料傳輸出現斷線或是無法部份無法辨識時，導致應用層的程式判斷出現錯誤，此時程式會啟動核心網路就會啟動資料錯誤重傳，資料錯誤重傳是將前一次排程資料無法正常傳輸完成的部份再次重傳，資料錯誤重傳會在每次新資料傳輸前先完成。因此愈多的資料錯誤重傳將會影響新資料傳輸的效率，所以應儘可能減少應用層程式啟動資料錯誤重傳的次數，以提高整體的資料傳輸效率。
10. Time Quantum：Time Quantum 可稱為時間量或是時間配額 (Matarneh 2009)，一般是在多工的系統中讓多項工作同時執行時使用的技術，做為 B-RR 中每個執行時間段的時間長度，也就是一種將時間依規則切成固定或是非固定的時間段，接著將不同的時間段分別配置給多個工作用以執行任務，系統會維護一個執行順序列表，系統執行完一個時間段就會繼續執行執行順序列表中的下一個時間段，可能是同一個工作或是不同的工作，這時前次未執行完的工作就會在等待的狀態直到系統再次執行到該工作被配置的時間段，如果工作的執行時間比被配置的時間段更短則剩餘的時間段則可以分配給其他工作使用，只有單一項工作時可視為將時間段全部設置成該工作任務的執行時間直到該項工作執行結束。
11. Frame Structure：本論文使用 5G mmWave 架構並利用 (Mezzavilla et al. 2018; Ford et al. 2016) 提到的 TDD (Time Division Duplex) 及 OFDM (Orthogonal Frequency-Division Multiplexing) 做為 Frame 及 Subframe 的架構設計基礎，並且實作了資料錯誤重傳及預設使用了 B-RR。TDD 分時雙工利用同一個頻率在同一個時間段只能執行一種任務，任務分為上傳資料 (Uplink) 或下載資料 (Downlink) 任務，時間段可以是固定或是變動但同一段時間段內只能有上傳或下載，換句話說就是使用時間進行傳送與接收資料的分隔，單向的資料傳輸不是連續的，TDD 對頻譜的使用效率較佳而同頻率天線也比較好設計。OFDM 為正交式頻分多工，對於 TDD 使用時間段將 Frame 切割成多個 Subframe 之後，OFDM 再對 Subframe 也稱為 Symbol 給予依頻率分割成多個 Sub-band, Sub-band 在這個架構下也被稱為 Slot，接著再依頻率將 Sub-band 分成多個 Sub-carrier 也稱為用以承載資料，將高速的頻率轉換成低速的並行的子通道，子通道可以承載更多資料以增加資料傳輸量，第一個 Symbol 固定為 DL control，最後一個 Symbol 則定為 UL control，剩下的 Symbol 可以是 UL 或是 DL。實際的參數設置是每個 Frame 的長度設定為 1ms 再切分成 10 個 Subframe，每個 Subframe 就是 100us 可以再分為 24 個 OFDM Symbol，每個 OFDM Symbol 為 4.16us 且有 1GHz 的頻寬，每個 OFDM Symbol 還可

以再細為 72 個 Sub-band 也就是 Slot，每個 Sub-band 就有 13.89MHz 的頻寬，最後每個 Sub-band 再往下可以展開 48 個 Sub-carrier。

三、資源分配與排程方法

無線通訊利用電磁波為載體進行訊號的傳遞，電磁波依頻率可以提供不同的頻寬，理論上愈高頻的電磁波其頻寬愈大，也就是同時間可以傳送的訊號愈多，雖然 5G mmWave 使用了極高頻的 28GHz 來進行資料傳輸，但是在人潮擁擠的地區仍會導致頻寬供不應求而出現使用者排隊等待網路服務的狀況，此時就需要一個排程方式確保每個使用者的需求可以被公平且快速的服務。常見的排程方法 (Fataniya & Patel 2018) 如下列所示：

1. FCFS (First Come First Serve)：如字面表示就是需求先到先服務，在需求佇列上可視為需求是先進先出，每一個需求執行完畢後才執行下一個需求。
2. SJF (Shorted Job First)：此方法執行的時候會先依需求執行所需的時間將需求佇列作降冪的排序，再由執行需求時間較短的時間優先的方式依序執行。
3. PS (Priority Scheduling)：將需求記錄至需求佇列時給定一個數值，這個數值對照一個優先層級，優先層級高的需求會先被執行，這個排程方法會將優先層級看得比效率更重要。
4. B-RR (Basic-Round Robin)：B-RR 可看成是將需求佇列的每一個需求依相同的時間量切割成若干個時間段，然後輪流執行每一個需求的第一個時間段，執行到最後一個需求時再從第一個需求開始執行，重複這樣的步驟直到需求都被滿足。

以上的方法中 FCFS 排程方法會遇到的問題是一個較大的需求執行過久造成塞車的狀況，SJF 可能造成執行時間較長的需求的等待時間過長，也可能造成執行時間較長的需求永遠不會被執行的飢餓 (Starvation) 狀態，PS 方法在優先級的認定上難以達到排程方法上的公平性，排程方法的公平性指的是每一個有連線信號的使用者需求都會依順序被執行，B-RR 的優點就是可以公平地執行每一個任務，一開始將需求依 Time Quantum 切割成較小的任務後傳送，在最佳的狀況下可以讓每個使用者感覺不到延遲，而 B-RR 也是 5G mmWave 行動網路架構模擬程式裡預設的排程方法。

四、網路模擬器

本論文所用 ns-3 離散事件模擬器是一個開放式來源的網路模擬器 (ns

simulator wiki; ns-3 tutorial)，主要用於研究及教育方面，ns 系列網路模擬器有 ns-1、ns-2 和 ns-3，ns-1 為 ns 系列第一代網路模擬器從 2001 年起已經不再開發和維護，ns-2 為 ns 系列第二代網路模擬器從 2010 年起也只維護而不再開發（ns simulator wiki），在 ns-3 出現之前知名的是 ns-2 開源網路模擬器已經有許多有維護及沒有維護的分支，ns-2 的程式碼已經超過了 30 萬行之多，ns-3 即是重構 ns-2 的架構並簡化網路的模擬開發，ns-3 由 C++ 開發且不提供向下相容 ns-2 的程式碼。離散事件可以認知是 ns-3 模擬器會在依序執行設定好事件，當二個事件之中有時間差則 ns-3 執行完第一個事件會直接執行到第二個事件，並不會對二個事件中間的時間差做出進行任何的計算，因此在設計事件或是分析模擬結果時應該要考慮到模擬時間與模擬數值是否相符。

參、研究架構與方法

一、NRRS 設計架構與運作流程

本論文使用的 ns-3 模擬軟體並由 github 上取得 5G mmWave 的測試程式（ns-3 mmWave），5G mmWave 網路架構下預設的排程器是 B-RR。B-RR 是一個簡單、公平、容易建置和廣為使用的排程器（Fataniya & Patel 2018），B-RR 的每個 Round 在執行的時候最開始都會看資料錯誤重傳的 buffer 裡是否有需要重傳的資料，這些資料多為傳輸時訊號出現問題所產生，在資料錯誤重傳之後就是正常的資料傳送，不論資料錯誤重傳或是正常的資料傳送都需要參考網路通道品質的 CQI，造成 UE 發出的 CQI 出現差異的因素有遠近、障礙物、速度及方向等，在實際場景中同時都會有多個 UE 與 mmWave eNB 進行資料傳輸，雖然 5G mmWave 網路架構使用了極高頻的頻率提升頻寬理論上可以達到 10Gbps 的資料傳輸速度，每一個 mmWave eNB 卻也需要服務數以萬計的 UE，可以提供傳送的頻寬還是很有限所以需要將每個 UE 的需求依序提供服務。因此，本文所設計之 NRRS 方法在 CQI 不存在或等於 0 時則減少測試資料傳輸的次數，實現方式是依照 CQI 進行 UE 資料分組並整合 NOMA 配對策略的實驗方式，每一個 NRRS 的 Round 開始時都會重新依 CQI 分組，收集最終的資料傳輸結果後比較原本未分組及分組後再配對。此外，實驗的 KPIs 值有 Cell 延遲、處理能量、資料錯誤率和資料錯誤重傳次數，處理能量的數值應該愈高愈好而 Cell 延遲、資料錯誤率及資料錯誤重傳次數應該則儘可能降低。

ns-3 是一個離散事件的網路模擬器，每一個事件都是照著特定的順序逐一執行，在實際應用上就是會將 UE 的需求依特定的順序提供服務，資料錯誤重傳的資料會儲存在 buffer 中，B-RR 每一個 Round 開始的時候先確認資料錯誤重傳 DL buffer 中是否有資料待處理，有就將資料加到重傳的 Slot 裡並清除資料錯誤重傳

DL buffer 中的記錄，詳細的流程是在每個 Round 開始的時候都會先檢查資料錯誤重傳 DL 的 buffer 中是否有需要重傳的資料，Y 就依 UE 檢查 CQI 存在且大於 0 者加入傳送，N 就清除貯存中的資料後就結束這個 Round 的資料錯誤重傳操作，如圖 4 列出資料錯誤重傳的流程圖。

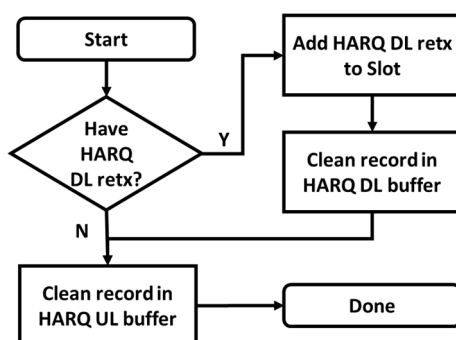


圖 4：資料錯誤重傳的處理流程圖

關於網路通道品質的 CQI 在演算法中的認定如後所述，當 CQI 不存在則表示沒收到該 UE 計算後上傳的 CQI，可能是在服務覆蓋範圍之外或是雖然在服務覆蓋範圍內但是被障礙物阻擋無法傳送 CQI 給 eNB，在確認 CQI 的存在之後還要看 CQI 的值，值為 0 則表示 UE 端的行動裝置收訊能力太差將導致 eNB 無法順利將資料傳送給該 UE，在這個狀況下把資料放入傳送的佇列中會造成 eNB 無法順利將資料傳給 UE，結果是無效率的頻寬佔用及增加資料錯誤重傳 buffer 的數量形成網路資源的浪費，所以當 CQI 為 0 時應該儘量減少嘗試傳輸資料的次數而達到節省網路資源的目的。然而，mmWave 模組裡原有的 B-RR 對 CQI 值的處理流程沒有進行分組處理，只有判斷是否為 0 如圖 5 所示。

B-RR 有二個主要問題，首先對於 CQI 不存在時也會試著進行資料傳輸，第二則是當 UE 的 CQI 不存在及 CQI 指數較低時還是會在迴圈中先處理該 UE 的資料傳輸，造成其他信號品質好的 UE 就在等待，但是可能信號品質差的 UE 會有更多資料錯誤重傳的問題出現，所以在下一個 Round 時又會先處理資料錯誤重傳的資料，如此則導致無線網路資源的耗損，所以我們提出 NRRS 改善 B-RR 以提升網路資源的分配效率，將資料錯誤重傳和 CQI 流程整合表示，改為先依 CQI 的值對 UE 及待傳輸的資料進行分組，並整合 NOMA 的 UE 配對策略，將最高的一組與最低的一組配對進行資料傳輸，由表 1 可以知道不同的網路品質可能透過不同的 Modulation 進行資料傳輸，但是因為 mmWave 行動網路架構有 MIMO beamforming 的技術，可以同時進行不同 Modulation 的傳輸動作，第一組資料配置結束再將剩下的 UE 也依同樣的配對方法進行配對並傳輸，圖 6 則是改善後的

NRRS 操作流程圖。

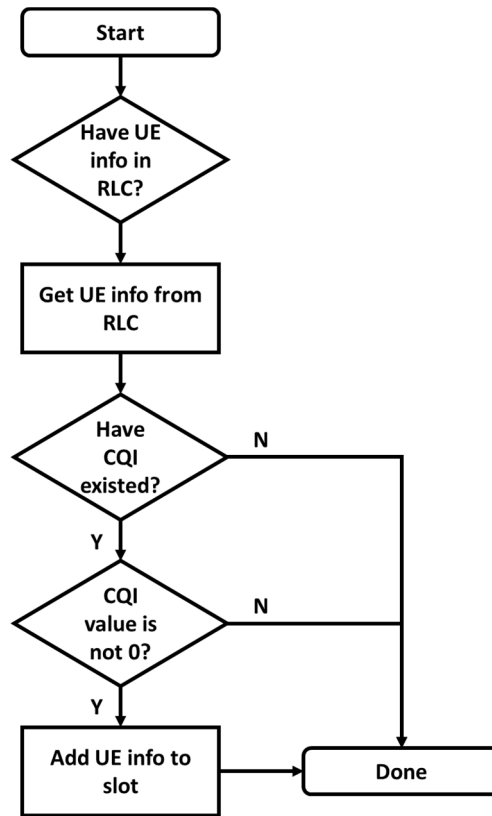


圖 5：原有的 B-RR 之 CQI 的處理流程圖

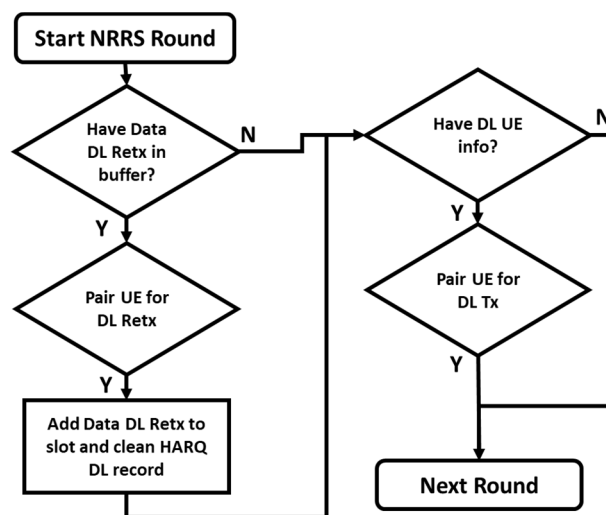


圖 6：NRRS 操作流程圖

一般對於 B-RR 的演算法效能改進多是利用統計指標對 CPU 的 Time Quantum 的計算基礎 (Fataniya & Patel 2018)，但在無線行動網路上因為頻率固定了頻寬，所以在 Time Quantum 的最大值已經是固定值。由於 B-RR 會分配可用資源給每一個 UE，這裡提出的假設是以 CQI 值較高的 UE 先進行資料傳輸，可以獲得更高的網路資源分配效率，並利用 NOMA 的 UE 配對方式將無線行動網路信號品質最佳及最差的配對，配對完之後放到同一個子載波進行資料傳輸以確保提高其處理能量。此外，亦會嘗試對網路信號品質最差的 UE 進行資料傳輸，所以設計 NRRS 依 CQI 來分組 UE 的清單，並採用 NOMA 的 UE 配對方式來提升網路資源分配效率。最後，使用網路模擬器進行實驗透過結果觀察 Cell 延遲、處理能量、資料錯誤率及資料錯誤重傳次數等，依上述本文設計了 NRRS 模擬測試情境如下：

1. mmWave 架構使用 B-RR，隨機 UE 列表。
2. mmWave 架構使用 NRRS，利用 CQI 值將 UE 及待傳輸的資料分 16 組，並整合 NOMA 的 UE 配對策略。

二、NRRS 排程演算法設計

根據圖 6 NRRS 操作流程圖以設計 NRRS 方法，其演算法之虛擬碼如下：

```

Algorithm NRRS () {
Input:
#define addDataDlInfoToDlFlow procedure
#define addDlInfoToDlFlow procedure
#define std::vector < DlDataInfo > dataDlBuffer
#define std::map < Rnti, CqiValue > Cqi

Output: To add DL info to total DL buffer size

Method:
BEGIN {
if (dataDlBuffer.size() != 0) {
    addDataDlInfoToDlFlow();
}

for ( iter = Cqi.begin(); iter != Cqi.end(); iter++ ) { // Pair UE for DL Retx
//and add DL info to DL Flow
    uint8_t itCqi = iter->second;
    if ( itCqi != 0 ) {
        addDlInfoToDlFlow();
    }
}
}
}

```

```

Procedure addDataDIInfoToDIFlow(){ //Add Data into DL flow
for ( iter = DataDIInfo.begin(); iter != DataDIInfo.end(); iter++ ) {
    uint8_t itCqi = iter->second;
    if ( itCqi != 0 ) {
        DIFlow.append(iter);
    }
}
}

Procedure addDIInfoToDIFlow(){
std::vector<uint16_t> rnti_0; // Initialize variables for CQI 0
std::vector<uint16_t> rnti_1; // Initialize variables for CQI 1
std::vector<uint16_t> rnti_2; // Initialize variables for CQI 2
std::vector<uint16_t> rnti_3; // Initialize variables for CQI 3
std::vector<uint16_t> rnti_4; // Initialize variables for CQI 4
std::vector<uint16_t> rnti_5; // Initialize variables for CQI 5
std::vector<uint16_t> rnti_6; // Initialize variables for CQI 6
std::vector<uint16_t> rnti_7; // Initialize variables for CQI 7
std::vector<uint16_t> rnti_8; // Initialize variables for CQI 8
std::vector<uint16_t> rnti_9; // Initialize variables for CQI 9
std::vector<uint16_t> rnti_10; // Initialize variables for CQI 10
std::vector<uint16_t> rnti_11; // Initialize variables for CQI 11
std::vector<uint16_t> rnti_12; // Initialize variables for CQI 12
std::vector<uint16_t> rnti_13; // Initialize variables for CQI 13
std::vector<uint16_t> rnti_14; // Initialize variables for CQI 14
std::vector<uint16_t> rnti_15; // Initialize variables for CQI 15

for (itRlcBuf = m_rlcBufferReq.begin(); itRlcBuf != m_rlcBufferReq.end(); itRlcBuf++) { //Inject variables with DL data
    itUeInfo = ueInfo.find(itRlcBuf->m_rnti);
    std::map<uint16_t, uint8_t>::iterator itCqi = m_wbCqiRxed.find(itRlcBuf->m_rnti);
    uint8_t cqi = 0;
    if (itCqi != m_wbCqiRxed.end()) {
        cqi = itCqi->second;
        if(cqi == 0){
            rnti_0.push_back(itRlcBuf->m_rnti); //Inject rnti_0 variables with DL data
        } else if(cqi == 1){
            rnti_1.push_back(itRlcBuf->m_rnti); //Inject rnti_1 variables with DL data
        } else if(cqi == 2){
            rnti_2.push_back(itRlcBuf->m_rnti); //Inject rnti_2 variables with DL data
        } else if(cqi == 3){
            rnti_3.push_back(itRlcBuf->m_rnti); //Inject rnti_3 variables with DL data
        } else if(cqi == 4){
            rnti_4.push_back(itRlcBuf->m_rnti); //Inject rnti_4 variables with DL data
        } else if(cqi == 5){
            rnti_5.push_back(itRlcBuf->m_rnti); //Inject rnti_5 variables with DL data
        } else if(cqi == 6){
            rnti_6.push_back(itRlcBuf->m_rnti); //Inject rnti_6 variables with DL data
        } else if(cqi == 7){
            rnti_7.push_back(itRlcBuf->m_rnti); //Inject rnti_7 variables with DL data
        } else if(cqi == 8){
            rnti_8.push_back(itRlcBuf->m_rnti); //Inject rnti_8 variables with DL data
        }
    }
}
}

```

```

        }else if(cqi == 9){
            rnti_9.push_back(itRlcBuf->m_rnti); //Inject rnti_9 variables with DL data
        }else if(cqi == 10){
            rnti_10.push_back(itRlcBuf->m_rnti); //Inject rnti_10 variables with DL data
        }else if(cqi == 11){
            rnti_11.push_back(itRlcBuf->m_rnti); //Inject rnti_11 variables with DL data
        }else if(cqi == 12){
            rnti_12.push_back(itRlcBuf->m_rnti); //Inject rnti_12 variables with DL data
        }else if(cqi == 13){
            rnti_13.push_back(itRlcBuf->m_rnti); //Inject rnti_13 variables with DL data
        }else if(cqi == 14){
            rnti_14.push_back(itRlcBuf->m_rnti); //Inject rnti_14 variables with DL data
        }else if(cqi == 15){
            rnti_15.push_back(itRlcBuf->m_rnti); //Inject rnti_15 variables with DL data
        }
        NS_LOG_UNCOND("test_DL_New_Data_cqi:: " << static_cast<int>(cqi));
    }

}

uint16_t x = 0 ;
for (int i = 0; (unsigned)i < rnti_15.size(); i++) { // Pair 15 and 0
    itUeInfo = ueInfo.find(rnti_15[i]);
    DIFlow.append(itUeInfo);
    x++;
    if((x%9) == 0){
        itUeInfo = ueInfo.find(rnti_0[i]);
        DIFlow.append(itUeInfo);
    }
}
x = 0 ;
for (int i = 0; (unsigned)i < rnti_14.size(); i++) { // Pair 14 and 1
    itUeInfo = ueInfo.find(rnti_14[i]);
    DIFlow.append(itUeInfo);
    x++;
    if((x%9) == 0){
        itUeInfo = ueInfo.find(rnti_1[i]);
        DIFlow.append(itUeInfo);
    }
}
x = 0 ;
for (int i = 0; (unsigned)i < rnti_13.size(); i++) { // Pair 13 and 2
    itUeInfo = ueInfo.find(rnti_13[i]);
    DIFlow.append(itUeInfo);
    x++;
    if((x%9) == 0){
        itUeInfo = ueInfo.find(rnti_2[i]);
        DIFlow.append(itUeInfo);
    }
}
}

```

```
x = 0 ;
for (int i = 0; (unsigned)i < rnti_12.size(); i++) {    // Pair 12 and 3
itUeInfo = ueInfo.find(rnti_12[i]);
DIFlow.append(itUeInfo);
x++;
if((x%9) == 0){
    itUeInfo = ueInfo.find(rnti_3[i]);
    DIFlow.append(itUeInfo);
}
}
x = 0 ;
for (int i = 0; (unsigned)i < rnti_11.size(); i++) {    // Pair 11 and 4
itUeInfo = ueInfo.find(rnti_11[i]);
DIFlow.append(itUeInfo);
x++;
if((x%9) == 0){
    itUeInfo = ueInfo.find(rnti_4[i]);
    DIFlow.append(itUeInfo);
}
}
x = 0 ;
for (int i = 0; (unsigned)i < rnti_10.size(); i++) {    // Pair 10 and 5
itUeInfo = ueInfo.find(rnti_10[i]);
DIFlow.append(itUeInfo);
x++;
if((x%9) == 0){
    itUeInfo = ueInfo.find(rnti_5[i]);
    DIFlow.append(itUeInfo);
}
}
x = 0 ;
for (int i = 0; (unsigned)i < rnti_9.size(); i++) {    // Pair 9 and 6
itUeInfo = ueInfo.find(rnti_9[i]);
DIFlow.append(itUeInfo);
x++;
if((x%9) == 0){
    itUeInfo = ueInfo.find(rnti_6[i]);
    DIFlow.append(itUeInfo);
}
}
x = 0 ;
for (int i = 0; (unsigned)i < rnti_8.size(); i++) {    // Pair 8 and 7
itUeInfo = ueInfo.find(rnti_8[i]);
DIFlow.append(itUeInfo);
x++;
if((x%9) == 0){
    itUeInfo = ueInfo.find(rnti_7[i]);
    DIFlow.append(itUeInfo);
}
}
```

```
x = 0;  
}  
  
END NRRS
```

肆、模擬環境之建置與程序設計

一、模擬環境建置與實驗工具說明

本研究中模擬實驗環境必須能夠安裝網路模擬器 ns-3 及進行網路模擬，ns-3 的安裝文件對於 Linux 的發行版 Ubuntu 上安裝 ns-3 模擬器的說明非常詳細，因而本研究選用 Ubuntu 為實驗的作業系統，程式的修改利用整合性開發工具（Eclipse）的 C++版本將 ns-3 程式庫和 5G mmWave 模擬程式匯入後進行修改，但運行模擬時還是經由終端命令行執行，ns-3 雖然支援使用 Python 程式語言顯示模擬時的網路成員及訊號傳輸方向動態圖。此外，5G mmWave 的模擬程式開發時並未加入該功能，最終產出的資料利用 Python 製作彙總表及繪製視覺化分析圖形，其網路模擬系統架構圖如圖 7 所示，以及網路模擬系統環境軟硬體配置表如表 2 所示，表 3 則說明模擬實驗工具。

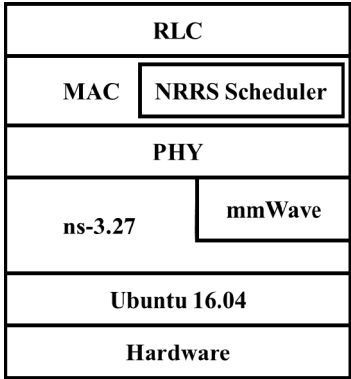


圖 7：網路模擬系統架構圖

表 2：網路模擬系統環境軟硬體配置表

軟硬體配置	軟硬體規格
CPU	Intel® Core™ i7-8750H Processor (9M Cache, up to 4.10 GHz)
Memory	16 GB or above
Virtual Memory	128 GB or above
Disk	150 GB or above
Operation System	Ubuntu 16.04

表 3：模擬實驗工具說明

模擬實驗工具	模擬實驗工具說明
ns-3.27	開源的網路模擬器，由C++開發的離散事件模擬器，可用於有線網路及無線行動網路的開發模擬測試。
Eclipse Oxygen	著名的跨平台開源整合式開發環境，部份模組提供修改C++程式的功能，優點是可以即時提示類別的結構，可提升修改C++程式的速度。
Python 3.5	通用型直譯式的程式語言，具有豐富的資料分析及繪圖程式庫，可以快速處理百萬行的文字資料，有助於處理大的Log檔案。
Microsoft Excel	快速由實驗資料產生實驗結果圖表

二、關鍵績效指標及模擬程序設計

在本論文中，我們將觀察 4 種關鍵績效指標 (KPIs) 如 Cell 延遲、處理能量、資料錯誤率和資料錯誤重傳次數，用來分析模擬實驗的數據如表 4 所示。此外，我們在模擬程式中計算每一次正常傳輸前須將上一次未傳完的資料進行重傳的次數，將這個次數認定為資料錯誤重傳次數。

表 4：關鍵績效指標定義說明

關鍵績效指標(KPIs)	分析目的
Cell延遲(Cell Latency)	RLC層出現的延遲數據並進行收集匯總，此數字為愈低愈好
處理能量(Throughput)	RLC層接資料封包大小數據進行收集匯總，此數字為愈高愈好
資料錯誤率 (DL Data Failure Ratio)	進行DL資料傳輸時發生資料遺失的機率，經收集計算後呈現百分比，此百分比為愈低愈好
資料錯誤重傳次數 (Data Failure Retransmission)	因資料錯誤而產生的資料重送次數，此次數為愈低愈好

本論文所設計之 NRRS 演算法將修改 5G mmWave 網路架構程式裡的 B-RR 方法，並與原本的 B-RR 透過模擬實驗進行比較與分析，NRRS 之設計程序如圖 8 所示，其說明如下：

1. 設計 NRRS 演算法。
2. 利用 NRRS 修改 5G mmWave 的 B-RR 及整合 NOMA 的 UE 配對功能。
3. 執行測試情境並收集實驗產生的資料。

4. 分析 NRRS 的績效：比較 NRRS 與 B-RR 績效
5. 產生 KPIs 數據並繪製視覺化分析圖形，以進行模擬結果分析。

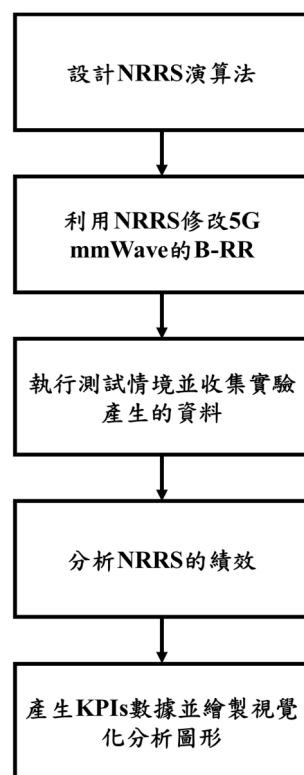


圖 8：NRRS 實驗程序圖

伍、研究結果與分析

一、模擬結果與數據說明

我們於 5G mmWave 行動網路模擬實驗透過在 Ubuntu 16.04 上執行 5G mmWave 程式碼，其模擬實驗實作了二種模擬測試情境如下：

1. 原始的 B-RR，未進行 UE 配對。
2. 改良後的 NRRS，進行 UE 配對。

模擬 NRRS 和原有之 B-RR 之排程方法，分別於 UE 個數 10 個、20 個、50 個和 100 個數目下進行模擬測試，並各自執行十次模擬實驗以取得較客觀的 Cell 延遲、處理能量、資料錯誤率和資料錯誤重傳次數等 4 種關鍵績效指標 (KPIs) 之平均值。Cell 延遲計算方式是由模擬測試後記錄的延遲時間加總後除以 Cell 總數、處理能量計算方式是由模擬測試後記錄的所有下行流量加總後除以 Cell 總

數，資料錯誤率的計算方式是資料錯誤重傳次數佔所有傳輸次數的比例，而資料錯誤率愈低則表示降低了資料錯誤重傳次數進而減少網路資源損耗。經由模擬實驗，其模擬結果比較如圖 9、圖 10、圖 11 和圖 12 所示。

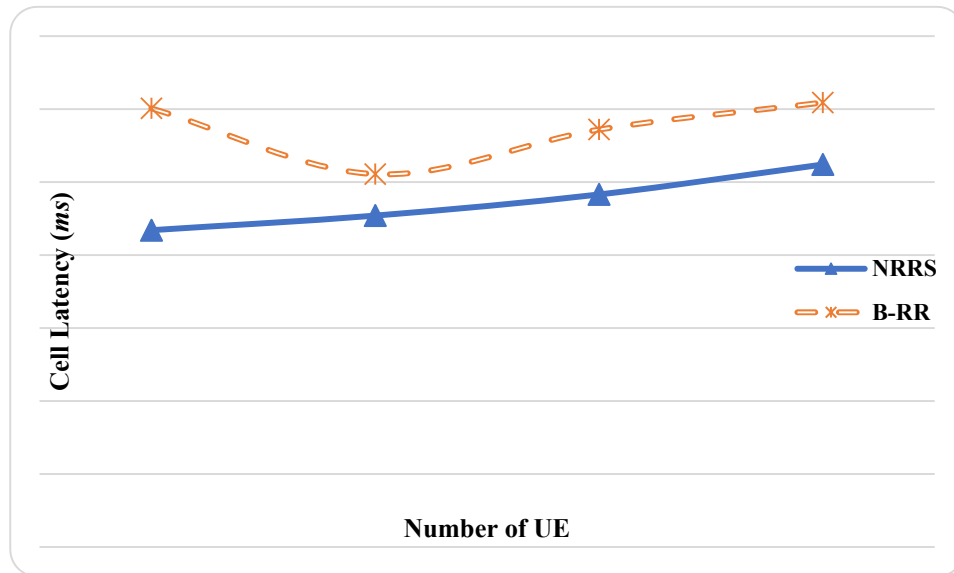


圖 9：Cell 延遲 (Cell Latency)

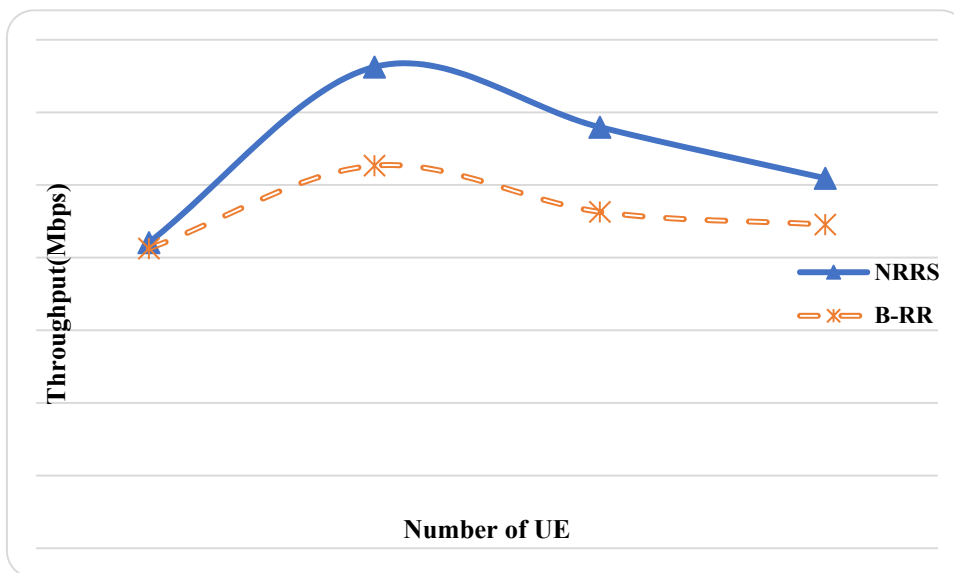


圖 10：處理能量 (Throughput)

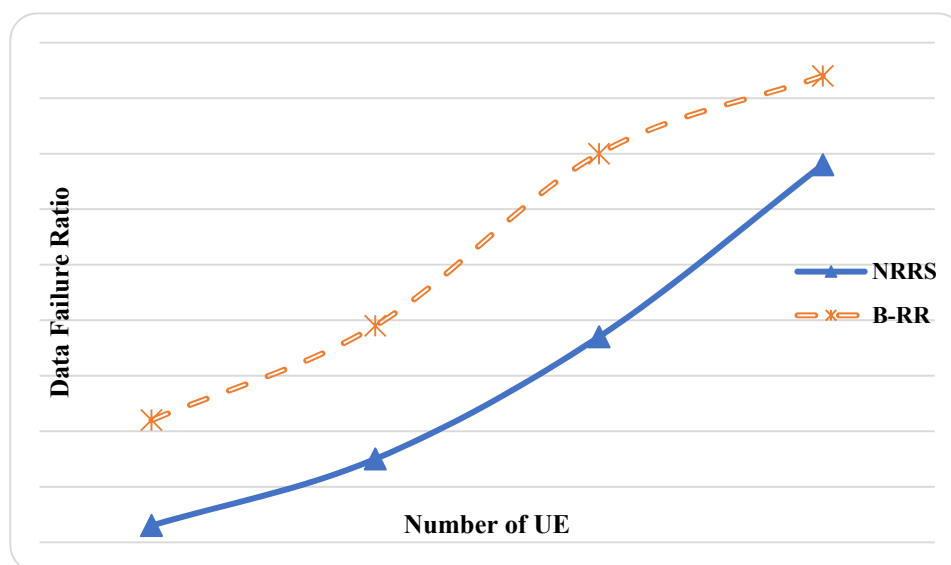


圖 11：資料錯誤率 (Data Failure Ratio)

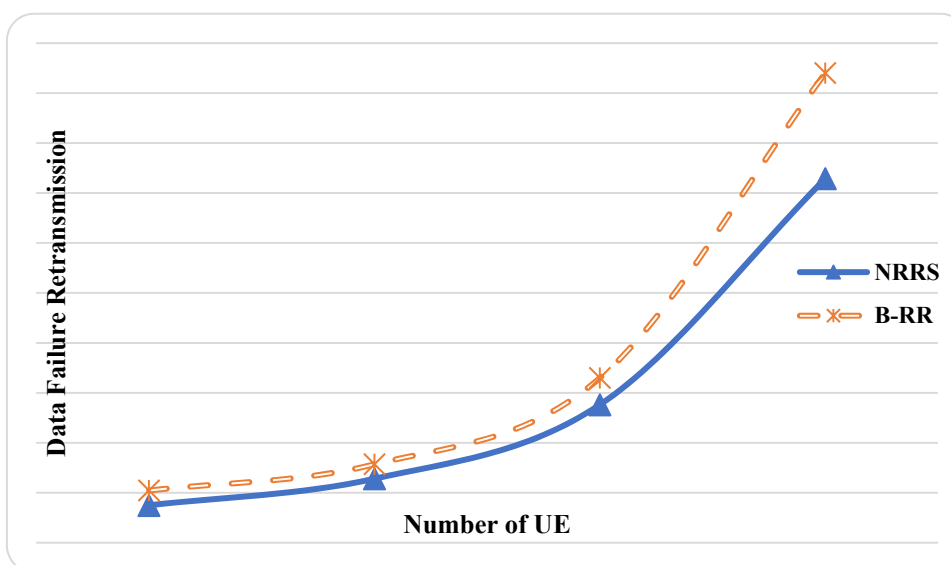


圖 12：資料錯誤重傳次數 (Data Failure Retransmission)

二、結果彙總與整體分析

本論文所提出的 NRRS 方法，在各個模擬測試的 Cell 延遲數據其 NRRS 皆比 B-RR 較好，且隨著 UE 數目的增加，減少的 Cell 延遲會顯著。反觀 B-RR 在不同 UE 數目模擬下，Cell 延遲都比較高。NRRS 排程方法的處理能量數據都比基本的 B-RR 更高，且隨著 UE 數目的增加呈現的結果更為顯著。再者由圖 11 及圖 12 所

示，NRRS 的資料傳輸錯誤率也比 B-RR 更低，改善了資料傳輸錯誤率也導致降低了資料錯誤重傳次數，因此每一個回合進行資料傳輸時都可減少上個回合未傳完的資料之傳輸，讓需要傳輸的資料總數可以更快地消耗。總之，模擬測試情境的實驗結果數據及改善比率（Improving Ratio; IR）顯示於無線行動網路資料傳輸效率上 NRRS 的設計比 B-RR 更好，其實驗結果彙總如表 5 所示。

表5：KPIs實驗結果彙總表

(a) Cell延遲 (Cell Latency)

KPIs UEs	Cell Latency (ms)		
	NRRS	B-RR	IR
10UEs	0.434	0.601	27%
20UEs	0.454	0.511	11%
50UEs	0.483	0.572	15%
100UEs	0.524	0.609	13%

(b) 處理能量 (Throughput)

KPIs UEs	Throughput (Mbps)		
	NRRS	B-RR	IR
10UEs	8418	8253	2%
20UEs	13250	10537	26%
50UEs	11593	9262	25%
100UEs	10186	8912	14%

(c) 資料錯誤率 (Data Failure Ratio)

KPIs UEs	Data Failure Ratio (%)		
	NRRS	B-RR	IR
10UEs	3%	22%	19%
20UEs	15%	39%	24%
50UEs	37%	70%	33%
100UEs	68%	84%	16%

(d) 資料錯誤重傳次數 (Data Failure Retransmission)

UEs \ KPIs	Data Failure Retransmission		
	NRRS	B-RR	IR
10UEs	149577	209586	28%
20UEs	256111	314379	18%
50UEs	552523	660195	16%
100UEs	1457699	1880785	22%

陸、結論

本論文針對 5G mmWave 原有的 B-RR 進行改良而提出 NRRS 方法。原有的 B-RR 於 5G 網路資源分配時會面臨到 CQI 值不存在的問題和處理 UE 時迴圈設計的效率不佳的問題，容易造成 UE 資料傳輸時的網路資源損耗。如果我們能夠減少網路資源損耗，則可將 5G mmWave 行動網路資源分配作較佳化。本論文透過我們所提出的 NRRS 方法，當 CQI 報告不存在時就不進行信號傳輸，並優先判斷 CQI 報告存在再進行信號傳輸，藉此減少 Cell 延遲、提高處理能量、降低資料錯誤率、以及相對地減少資料錯誤重傳次數。實際上，分別於 10UEs、20UEs、50UEs、100UEs 等 UE 個數設定下執行模擬實驗，以比較 NRRS 排程方法與 B-RR 並加以分析，模擬實驗結果其關鍵績效指標 (KPIs) 顯示 Cell 延遲分別可獲得 27%、11%、15%、13% 的改善，處理能量分別可獲得 2%、26%、25%、14% 的改善，資料錯誤率分別可達到 19%、24%、33%、16% 的改善，而資料錯誤率的降低讓資料錯誤重傳次數也下降 28%、18%、16% 和 22% 的改善。

整體而言，本論文達成主要的研究成果如下：

1. 瞭解 5G mmWave 行動網路架構和 NOMA 的運作原理，找出 5G mmWave 可以整合 NOMA 的方式，並找出 B-RR 的資料傳輸以及演算法中關於處理 UE 的迴圈設計之效率不彰等問題。
2. 提出新的 NRRS 方法利用 UE 的 CQI 值對待傳輸資料組進行分組，然後整合 5G mmWave 行動網路與實現 NOMA 的 UE 配對策略，來改善 B-RR 的資料傳輸效率不彰的問題。
3. 經由模擬實驗，其實驗結果均驗證 NRRS 方法於各項績效指標如 Cell 延遲、處理能量、資料錯誤率以及資料錯誤重傳次數等，都比 B-RR 有更好的資料傳輸效率。
4. NRRS 有效地改善 B-RR 以提升 5G mmWave 行動網路資料的傳輸量及整體行動網路之處理績效。

依照我們的模擬實驗得知，NRRS 方法經由有效地結合不同的無線行動網路架構，可以提供更低 Cell 延遲且處理能量更高的 5G mmWave 行動網路，資料傳輸也較正確且有效率，NRRS 方法可應用在各種無線行動網路架構之排程。未來，為了達到無所不在的高品質無線行動網路以及需要整合各種異質架構的網路，可套用 NRRS 方法以整合 5G 行動網路更加順利且達到較佳的服務品質，進而提升整體行動網路的處理績效。

參考文獻

- 3GPP (2018), 'NR and NG-RAN Overall Description - Rel. 15', TS 38.300.
- 3GPP (2017), 'Evolved Universal Terrestrial Radio Access (E-UTRA)', *Physical Layer Procedures (Release 14)*, TS 36.213.
- Andrews, J.G., Buzzi, S., Choi, W., Hanly, S.V., Lozano, A., Soong, A.C.K. and Zhang, J.C. (2014), 'What will 5G be?', *IEEE J. Sel. Areas Communication*, Vol. 32, No. 6, pp. 1065-1082.
- Ding, Z., Fan, P., & Poor, H.V. (2016), 'Impact of User Pairing on 5G Nonorthogonal Multiple-Access Downlink Transmissions', *IEEE Trans. Veh. Technol.*, Vol. 65, No. 8, pp. 6010-6023.
- Fataniya, B. and Patel, M. (2018), 'Survey on Different Method to Improve Performance of The Round Robin Scheduling Algorithm', in *IJSRSET*, Vol. 4, No. 2, pp. 69-77.
- Fataniya, B. and Patel, M. (2018), 'Dynamic Time Quantum Approach to Improve Round Robin Scheduling Algorithm in Cloud Environment', in *IJSRSET*, Vol. 4, No. 4, pp. 963-969.
- FemtoForum 2010, 'LTE MAC Scheduler Interface Specification' v1.11, <http://www.femtoforum.org/femto/technical.php>.
- Ford, R., Zhang, M., Dutta, S., Mezzavilla, M., Rangan, S. and Zorzi, M. (2016), 'A Framework for End-to-End Evaluation of 5G mmWave Cellular Networks in Ns-3', in *Proceedings of the Workshop on Ns-3*, pp. 85-92.
- Piro, G., Grieco, L.A., Boggia, G., Capozzi, F. and Camarda, P. (2011), 'Simulating LTE Cellular Systems: An Open-Source Framework', *IEEE Transaction Vehicle Technology*, Vol. 60, No. 2, pp. 498-513.
- Matarneh, R.J. (2009), 'Self-Adjustment Time Quantum in Round Robin Algorithm Depending on Burst Time of the Now Running Processes', *American Journal of Applied Sciences*, Vol. 6, No. 10, pp. 1831-1837.
- Mezzavilla, M., Miozzo, M., Rossi, M., Baldo, N. and Zorzi, M. (2012), 'A Lightweight

- and Accurate Link Abstraction Model for the Simulation of LTE Networks in ns-3', in *Proc. of 15th ACM Int. Conf. Model. Anal. Simulat. Wireless Mobile Syst.* (MSWiM), pp. 55-60.
- Meeker, M. (2015), 'Internet trends 2015-code Conference', *Glokalde*, Vol. 1, No. 3.
- Mezzavilla, M., Zhang, M., Polese, M., Ford, R., Dutta, S. and Rangan, S. (2018), 'End-to-End Simulation of 5G mmWave Networks', *IEEE COMMUNICATIONS SURVEYS & TUTORIALS*, Vol. 20, No. 3, pp. 2237-2263.
- NGMN Alliance (2015), NGMN 5G White paper, Version 1.
- NAQVI, S.A.R. and HASSAN, S.A. (2016), 'Combining NOMA and mmWave Technology for Cellular Communication', In *Proceedings of IEEE 84th International Conference on Vehicular Technology Conference (VTC-Fall)*, pp. 1-5.
- Report IDC FutureScape : *Worldwide IT Industry 2018 Predictions*.
- Song, L., Li, Y., Ding, Z. and Poor, H.V. (2017), 'Resource management in non-orthogonal multiple access networks for 5G and beyond', *IEEE Network*, Vol. 31, No. 4, pp. 8-14.
- Wei, Z., Yuan, J., Ng, D.W.K., El Kashlan, M. and Ding, Z. (2016), 'A Survey of Downlink Non-orthogonal Multiple Access for 5G Wireless Communication Networks', *ZTE Communication*, Vol. 14, No. 4, pp. 17-25.
- Zhang, D., Zhou, Z., Xu, C., Zhang, Y., Rodriguez, J. and Sato, T. (2017), 'Capacity Analysis of NOMA with mmWave Massive MIMO Systems', *IEEE Journal on Selected Areas in Communications*, Vol. 35, No. 7, pp. 1606-1618.
- ns-3 mmWave, <https://github.com/nyuwireless-unipd/ns3-mmwave>
- ns-3 tutorial, <https://www.nsnam.org/about/what-is-ns-3/>
- ns simulator wiki, [https://en.wikipedia.org/wiki/Ns_\(simulator\)](https://en.wikipedia.org/wiki/Ns_(simulator))

