

Wireless Network PaperReading Report

108064535 陳文遠

UAVs Traffic Control Based on Multi-Access Edge Computing

Oussamma Bekkouche, Tarik Taleb and Miloud Bagaa

I. INTRODUCTION & MOTIVATION

□ 為何研究無人機

無人機這項新興的科技隨著技術的進展，越來越多的問題得到了解決，因此它所能應用的領域有變得越來越廣，像是軍事偵察、監視基礎建設、智慧農業、交通管理、邊境巡邏、貨物運送等（如 Fig 1.）。而上面所提及的領域其實已經被真實應用，雖然目前還有很多問題需要被克服。除了我所提及的領域以外，還有更多的應用場景正等待著學界業界去挖掘。



Fig 1. UAV 的常見應用場景

隨著無人機的技术越來越進步以及應用領域越來越廣，其市場價值也理所當然地逐年上漲（如 Fig 2.），在 2017 年時全球的無人機市場價格來到了 180 億美元，而預估到 2025 年時的全球市場價值可以高達 458 億美元（大約 1.4 兆台幣）。這也是為何無人機技術在學界業界越來越受到重視。

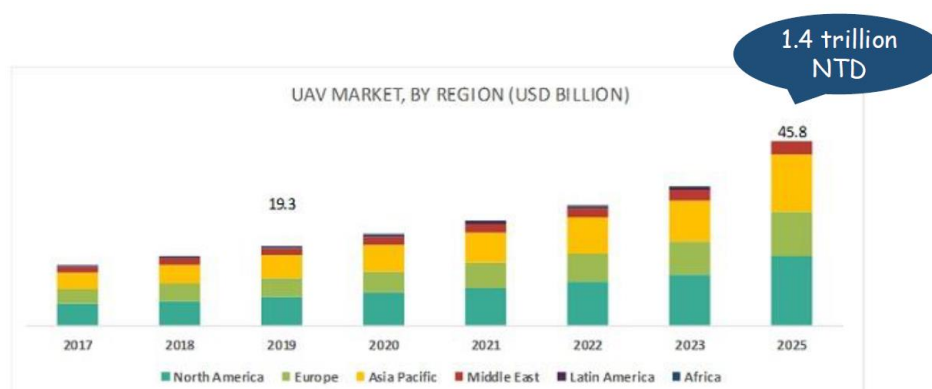


Fig 2. UAV 的市場價值

□ Air Traffic Management System (ATM)

無人機最初開始發展時，其交通流量的控制是沿用一般飛機所使用的空中交通管理系統 (Air Traffic Management, ATM)，在傳統的 ATM 系統中，每一台無人機都是透過操作員的手動控制，或者使用預定義的程式來操縱無人機。而在比較有挑戰性的環境中 (例如災難救援)，就需要操作員在 Visual-Line-of-Sight 中來操縱無人機，才能使任務完成的機率提高。因此，在傳統的 ATM 系統中，每一台無人機之間的交通管理是透過飛行員之間的聲音通訊來進行，所以可常見到如 Fig 3. 中，坐了一整排的管理者來進行通訊與管理交通。

從上述對傳統 ATM 的描述中你可以發現，它並不適合用來處理當今世代高密度的無人機交通，因為在一個小系統中，無人機動輒數千數萬台，不可能依靠聲音的通訊來進行交通流量管理。



Fig 3. ATM 交通流量管理

□ UAV Traffic Management System (UTM)

為了因應高密度的無人機交通管理需求，因而產生了專為無人機所打造的無人機交通管理系統 (UAV Traffic Management, UTM)。目前世界上最優良的 UTM 系統架構是由美國聯邦航空總署 (FAA) 以及美國國家航空暨太空總署 (NASA)。下面的 Fig 4. 是一個由 FAA 所提出的 UTM 簡單架構。

Fig 4. 的架構大致是說明，使用者 (sUAS pilot) 可以在客戶端 (UTM Client) 上設定一些重要的參數 (例如想要無人機做甚麼事、完成甚麼任務等)，接著會把這些要求 (Request) 送到後端 (UTM service) 來做出因應的處理，而最後方則有管理者負責管理維護系統或者提供更多資訊給 UTM service。

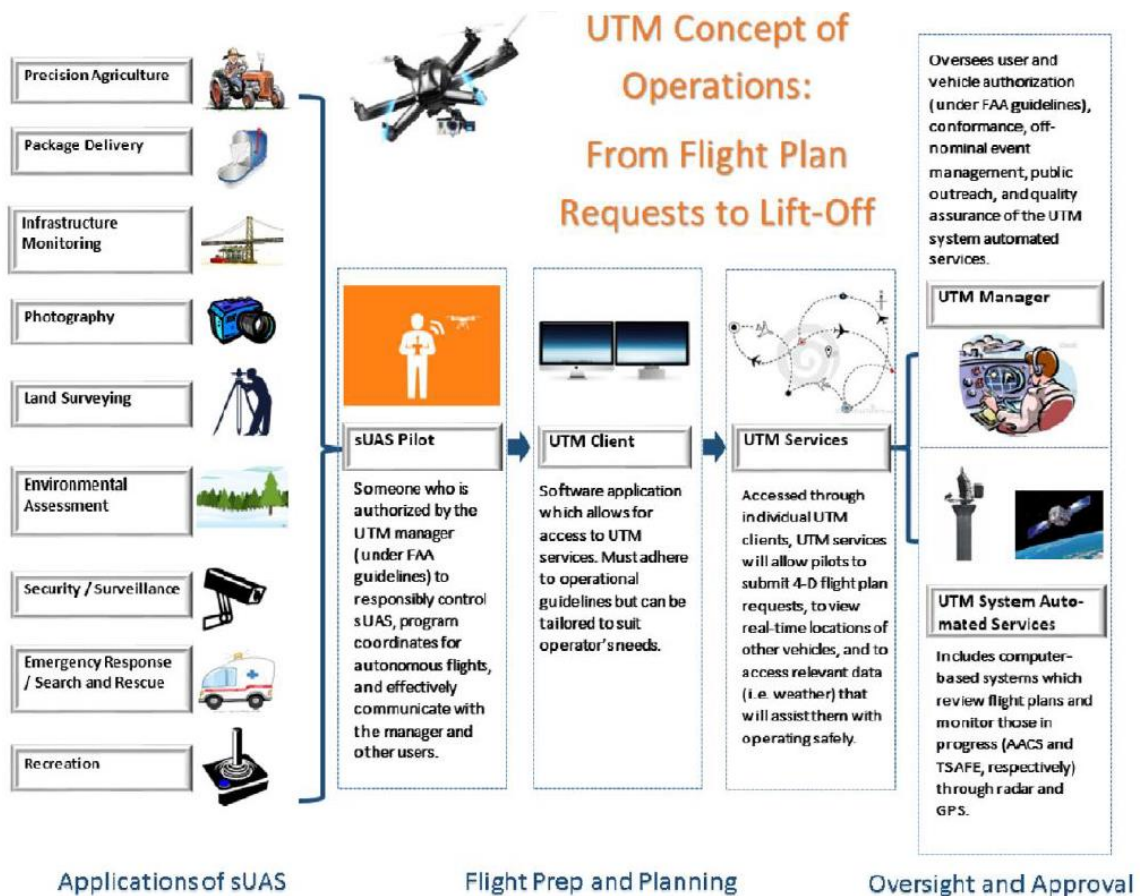


Fig 4. FAA 所提出的 UTM 簡單架構

在最新的 UTM 中系統中，有人提出一種結合舊有 Mobile Cellular Network 的架構 (Fig 5.)，其優點是可以保有 Mobile Cellular Network 的益處，例如可擴展性 (Scalable)、安全性 (Secure)、隨時可用 (Ready-for-use)。因為原有的 Mobile Cellular Network 已經在安全性問題上研究多年，因此可以讓我們的 UTM 直接沿用其安全性優點，而且基地台 (BS) 一直都在運作，並且現今的覆蓋範圍也很廣，更有助於操作無人機。

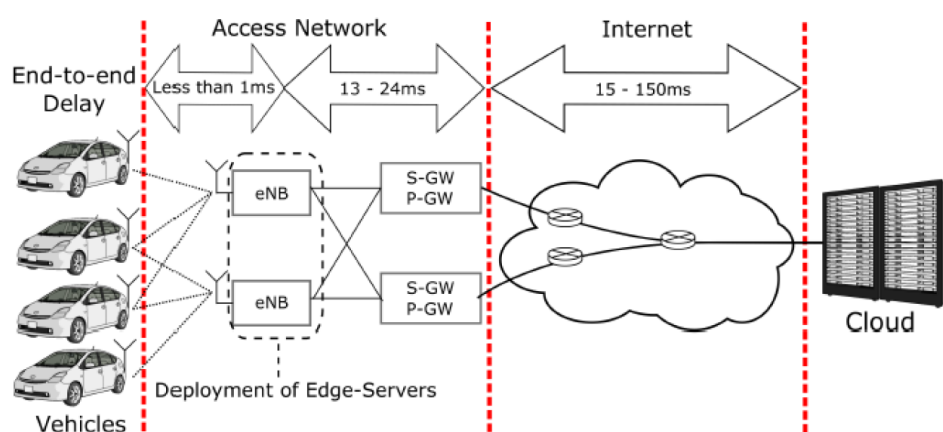


Fig 5. 5G Cellular-Connected Devices

Multi-Access Edge Computing (MEC)

在上方所提及的 Mobile Cellular Network 結合 UTM 系統的架構，在現今的 4G 網路中的封包延遲太大，因為當連接到基地台的設備發送訊息到基地台時，基地台會將其訊息經由中間的骨幹網路以及電信核心網路送到後面的 Cloud 做處理，Cloud 做完處理耗材送回到設備，因此

一來一往之間就會導致封包大量的延遲，大量的延遲也會影響到無人機的控制。多重存取邊緣運算 (Multi-Access Edge Computing, MEC) 便是為了這個資料傳輸延遲問題而生，下圖 Fig. 6. 就一個簡單的 MEC 架構圖。

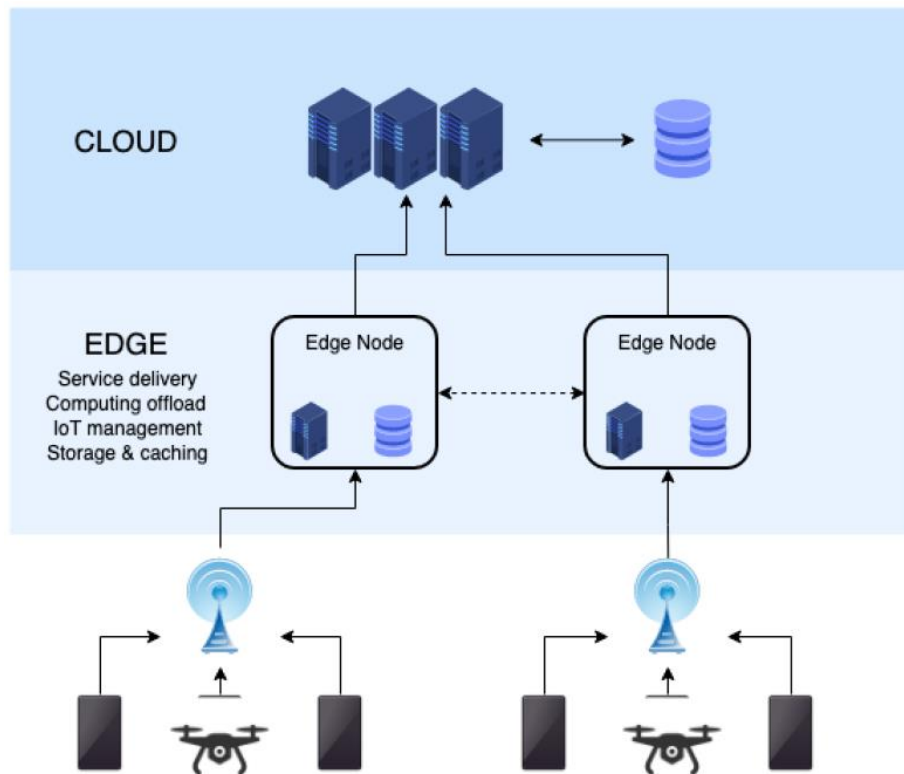


Fig 6. MEC 架構圖

MEC 的概念就是，當裝置送回資料到基地台時，基地台並不會立即將資料全部送到最後方的雲端去，而是先將一大部分的工作負載卸在安置於基地台後方的 Edge Node 上計算，目的是為了減輕 Cloud 的工作量，使得封包傳輸的延遲被大幅降低。因此，當系統中使用了 MEC 時，其封包傳輸的延遲可以降到小於 1ms。

II. EXPERIMENT

□ 實驗介紹

無人機的管理是透過 UTM 幫無人機定義一個可飛行的範圍，並且將其控制在範圍內，藉此來防止無人機之間發生碰撞，同時也可以提高其完成任務的機率。我們都知道封包的丟失以及延遲會導致通訊的可靠度下降，進而影響對無人機的控制。因此設計了一個實驗來加以探討封包丟失以及延遲對無人機的影響有多大 (Fig 7.)。



Fig 7

□ 實驗方法

作者設置了如下圖 Fig 8. 的實驗場景，接著調整不同的封包延遲以及封包丟失率，並且盡可能地讓無人機在預定義的飛行範圍邊界上飛行。途中的紅色虛線代表我們希望無人機飛行的路徑，而藍色的虛線代表無人機實際上飛行的軌跡，紅色虛線與藍色虛線之間的距離 M_i 為無人機的出界距離。

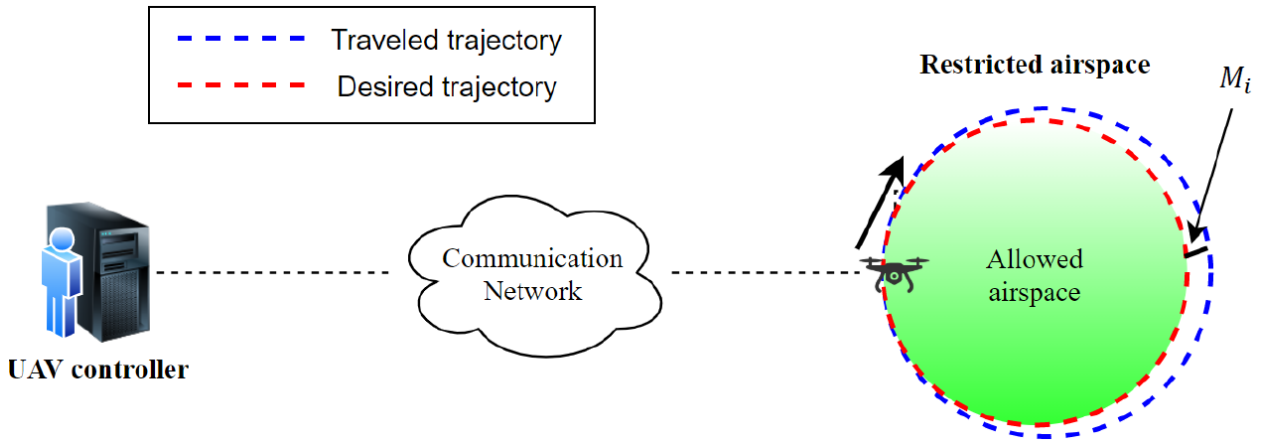


Fig 8. 實驗場景

接著作者為了說明其無人機的控制程度，因此定義了兩個參數來判斷飛行的穩定度，也就是平均出界距離 C_{avg} (The average of crossing rate) 以及標準差 D_r (The deviation ratio)。其中 M_i 為無人機與邊界的距離； Nb_m 為樣本數量； O_D 為在邊界以外飛行的距離； T_D 為總共飛行的距離。

$$C_{avg} = \frac{\sum_{i=1}^{Nb_m} M_i}{Nb_m} \quad D_r = \frac{O_D \times 100}{T_D}$$

□ 實驗設置

下圖 Fig 9. 為實驗中所用到的工具。這次的實驗用了三台 Virtual Machine 來分別模擬 UAV Controller、Communication Network 以及 UAV operation。UAV Controller 是以 Python 所撰寫的，其中使用了 Python drone-kit API，這個 API 後面用的是 MAVLink Protocol，這是一個是用於無人機通訊的輕量級通訊協定。UAV Controller 與無人機之間的通訊網路是使用一般的 Network Emulator 來做，你可你在上面調整你想要的封包延遲以及封包丟失率。最後則是使用了 Software-In-the-Loop emulator (SITL) 來模擬無人機的真實飛行狀況。

EXPERIMENT TOOL	
UAV Controller (First VM)	UAV Controller is developed using the Python drone-kit API which is an implementation of the MAVLink protocol, used for the communication with drones.
Network (Second VM)	Through network emulator to simulate the communication between UAV Controller and UAV with latency and packet loss.
UAV operation (Third VM)	Using Software-In-the-Loop (SITL) emulator that allows running the open-source UAVs' autopilot ArduCopter with the same behavior of a real UAV.

Fig 9. 實驗工具

準備好工具後作者在上面設定了如下圖 Fig 10. 的實驗參數。首先讓 UAV Controller 定義了一個飛行範圍 (為半徑 200m 的圓)，接著讓無人機以 10 (m/s) 的速度繞著邊界飛行，在飛行過程中，讓 UAV Controller 以 100 Hz 的頻率 (每 10ms 一次) 向無人機發送訊息 (修正無人機的飛行誤差)，總飛行的時間為 300 秒。此外，作者在 Network Emulator 上調整了各種不同的封包延遲 (0ms, 100ms, 200ms, 300ms, 400ms) 以及封包丟失率 (0%, 5%, 10%, 15%, 20%)。

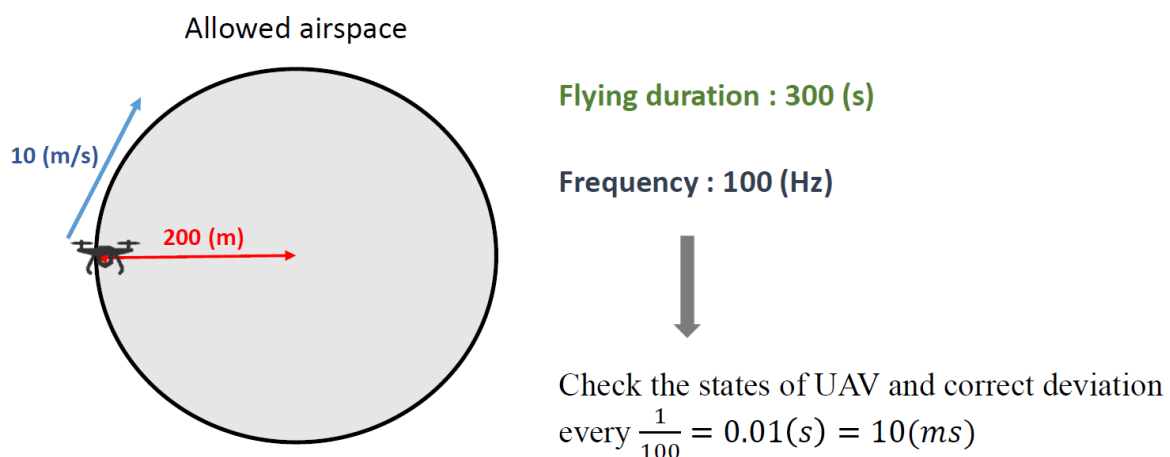


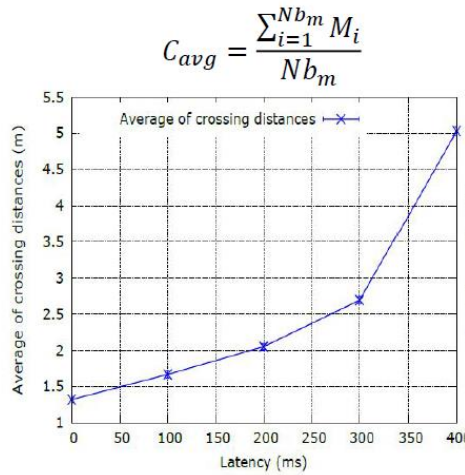
Fig 10. 無人機模擬飛行的實驗參數

□ 實驗結果及討論

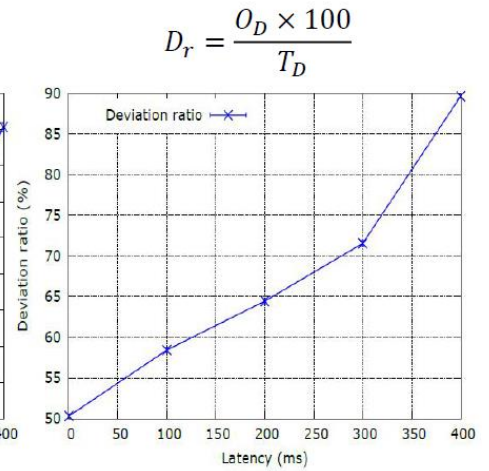
下方的 Fig 11. 與 Fig 12. 分別為封包傳輸延遲以及封包遺失率對無人機的影響。從實驗結果圖中你可以發現當延遲時間增加時，標準差也會隨之增加，而當延遲高達 400ms 時，標準差則大於 90%，這意味著無人機的飛行路徑幾乎都在邊界以外了。而 Fig 12. 中封包遺失率增加時，也與 Fig 11. 的趨勢相同，但我們也可從中發現封包延遲對無人機造成的影響會比封包丟失率的影響還大。而這個實驗結果也可以做為一個依據，讓我們以自己可以容忍的標準差 D_r 來設置自己的系統環境，同時也從中了解封包丟失率與封包傳輸延遲對無人機的控制有何種程度的影響。



(a) Desired (red) vs Traveled (yellow) trajectory in presence of high latency.



(b) Average of crossing distances.

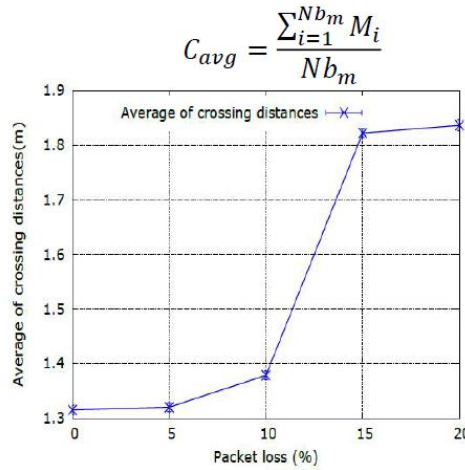


(c) Deviation ratio.

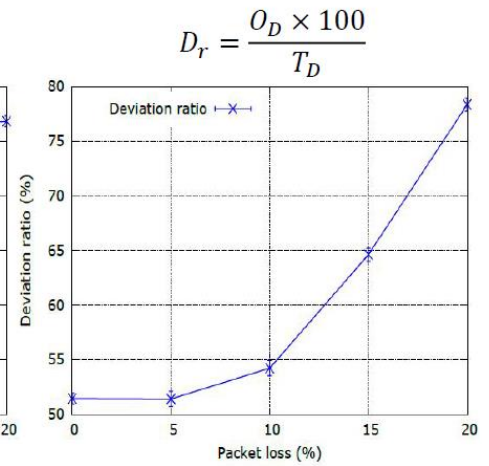
Fig 11. 封包傳輸延遲對無人機的影響



(a) Desired (red) vs Traveled (yellow) trajectory in presence of Packet loss.



(b) Average of crossing distances.



(c) Deviation ratio.

Fig 12. 封包丟失率對無人機的影響

III. PROPOSED SOLUTION

□ MEC-Based UAVs' Traffic Management Framework

根據上面的實驗結果，我們得知了封包延遲會對無人機的飛行控制產生劇烈的影響，為此作者提出了一種結合 MEC 與 UTM 的系統來確保無人機可以很高概率的待在自己的範圍內而不出界。下圖 Fig 13. 即為作者所提出的 MEC-Based UAVs' Traffic Management 架構。而這個架構我將會分成三個部份來解釋 (A : Cloud Domains, B : Core&Transport Network, C : MEC)。

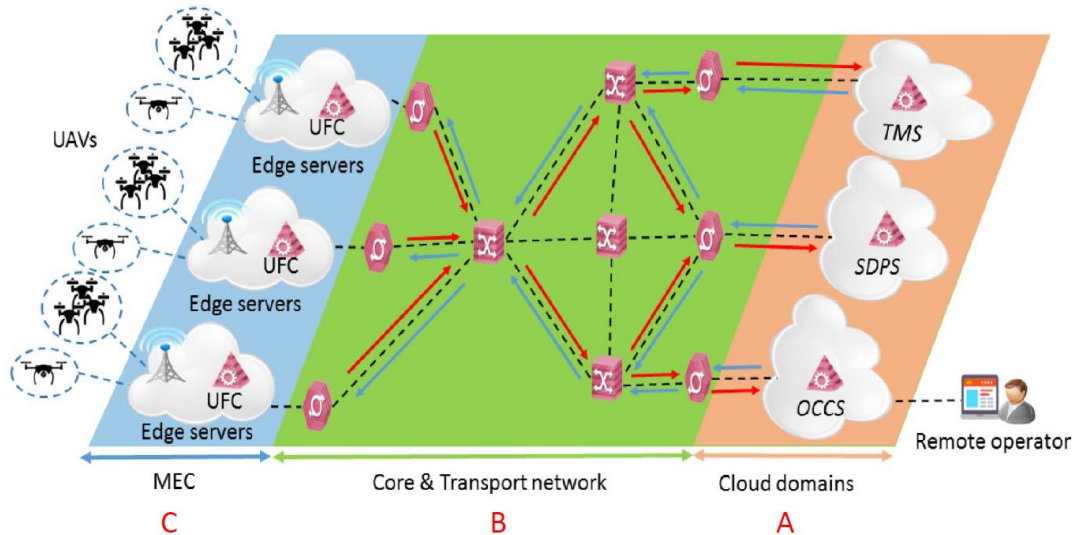


Fig 13. MEC-Based UAVs' Traffic Management Framework

□ Part A : Cloud Domains

Cloud Domain 負責託管了所有系統中的管理服務，以下又能細分成三種服務。

1. **UAVs' Traffic Management Service (UTMS) :** UTMS 擁有所有的無人機資訊，其中包含了無人機的位置、飛行計畫、飛行區域等。此外，無人機的可飛行區域也是由此服務來負責定義。
2. **Supplementary Data Provider Service (SDPS) :** SDPS 負責提供所有對無人機有益的訊息，以幫助規劃最適合的飛行路線，例如天氣、交通意外、交通阻塞等資訊。
3. **Operator Command and Control Service (OCCS) :** OCCS 負責提供使用者介面，以供客戶在上面對無人機進行遠程的即時監控或操縱。

□ Part B : Core and Transport Network

Core and Transport Network 指的就是 MEC 與 Cloud Domain 進行通訊的中介，一般指的便是骨幹網路以及電信核心網路，為了因應高可靠度以及低延遲的通訊 (URLLC)，因此在此部份需要有高的 QoS，而在當前架構中，一般都會在其中應用 SDN-enable Network 技術 (Fig 14.)。SDN 是一種新型網路架構，它利用了 OpenFlow protocol 將裝置的 Control plane 從 Data plane 中分離，改以軟體方式實作。該架構可使網路管理員在不更動硬體裝置的前提下，以中央控制方式用程式重新規劃網路，為控制網路流量提供了新方案，也為核心網路和應用創新提供了良好平台。

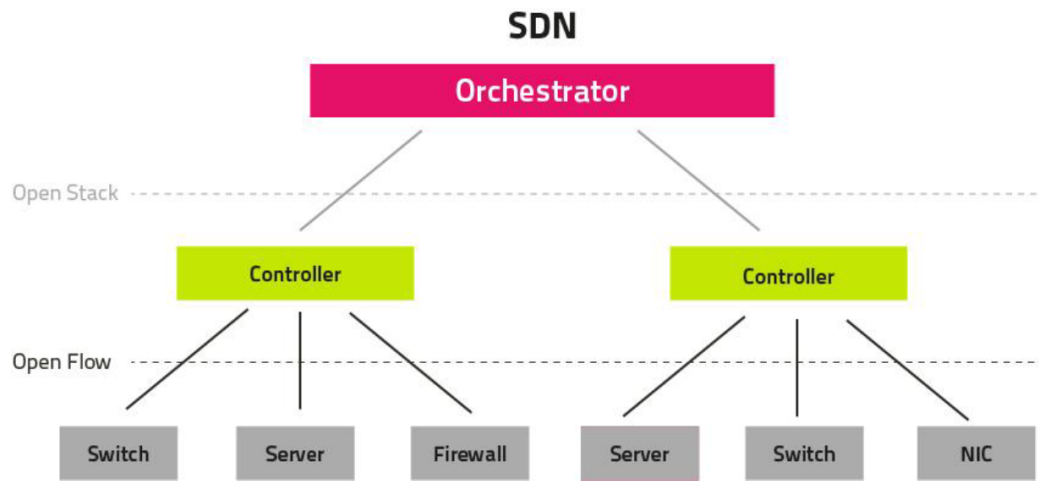


Fig 14. SDN-Enable Network

□ Part C : Mobile Edge Computing

在這個架構中，MEC Nodes 被置於基地台後方並用來 host 無人機飛行控制服務 (UFC) (Fig 15.)。UFC 是用來負責監視與控制所有連接到 MEC Nodes 的無人機，同時 UFC 也要負責收集所有由 Cloud Domains 中的服務所送來的訊息，例如天氣、飛行區域、飛行路徑等。

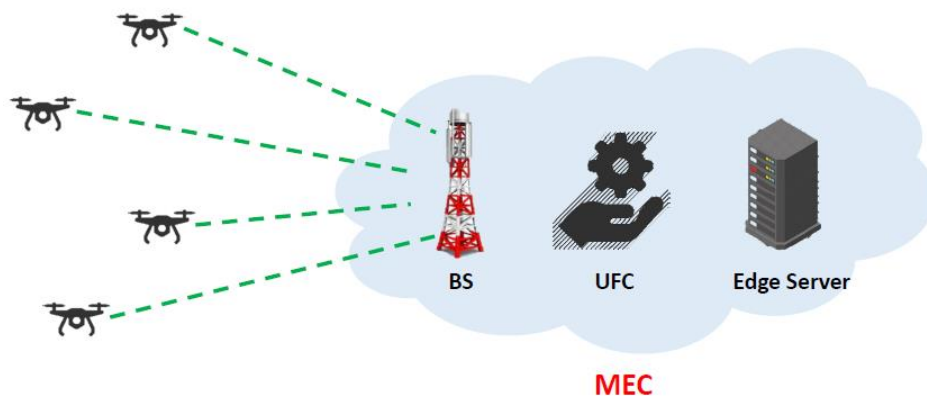


Fig 15. Mobile Edge Computing

□ 實驗工具與參數

在實作中，作者並非使用虛擬的 MEC Node，而是使用了一台真實的 Edge Server 來實現其系統，這台 Edge Server 名為 Intel Fog Reference Design (Fig 16.)。這台 Edge Server 並不會對外出售，這是 Intel 給出的一台 Edge Server 參照機，讓那些做 Edge Server 的廠商可以參考這台機子來做出屬於自己的 Edge Server。



Fig 16. Edge Server

下圖表 Fig 17. 是為實驗參數。其實驗參數與上面實驗中的設置大致相同。在實作時，作者用了不同的無人機數量來連上 Edge Server，並測時其不同數量時對無人機控制的效果為何。

Parameter	Description	Values
CPU cores	The number of physical CPU cores in the edge server.	4
RAM	Available RAM in the edge server.	32 GB
Throughput	Available throughput for the communication between the edge server and the UAVs.	100 Mbps
UAVs number	The number of UAVs connected to edge.	20, 40, 60, 80, 100
Speed	The speed of the UAVs	10 m/s
Duration	The duration of the flights and the simulations.	300 s
Radius	The radius of the allowed geofence.	200 m

Fig 17. 實驗參數

Fig 18. 的圖示說明不同數量的無人機同時連上 Edge Server 時，其無人機的平均出界距離變化趨勢，我們發現到當越多台無人機同時連到 Edge Server 時，每台無人機分配的 CPU 效能就會越少，進而導致對無人機的操控性減弱，幸運的是，因為有了 Edge Server 幫 Cloud 承接大量的工作負載，令封包延遲可以趨近於 0，因此也減低了其出界的標準差。以 Fig 18. 的結果為基準，我們便能依照自己可容忍的程度來分配無人機的數量抑或是增強 Edge Server 的效能。Fig 19. 則是在不同無人機數量時，在實驗過程中 Edge Server 總接收與總傳送 Byte 數。得到的結果為，每台無人機平均在 300 秒內產生了 3028Kb 的流量 (速度約 10Kbps)，而每台無人機平均在 300 秒內接收了 7152Kb 的流量 (速度約 24Kbps)。

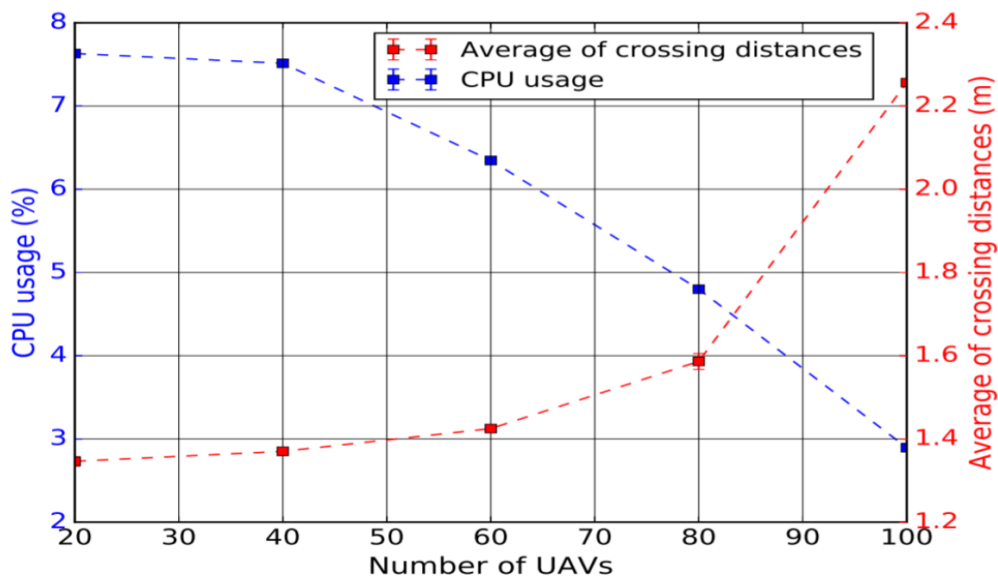


Fig 18. 不同無人機數量時的 C_{avg} 以及 CPU 使用率

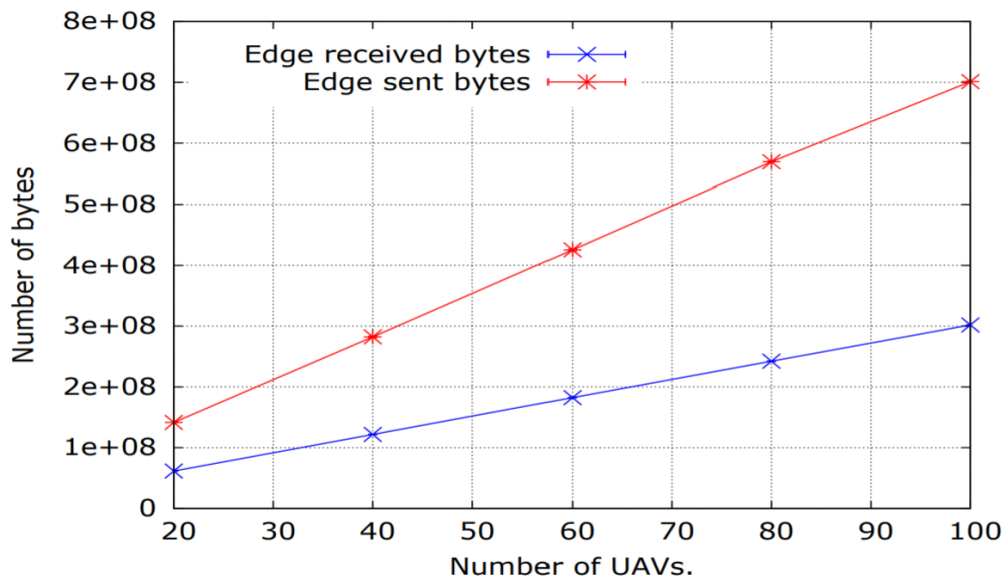


Fig 19. 不同無人機數量時 Edge Server 的傳誦與接收 Byte 數量

IV. DISCUSSION & CONCLUSION

我們統整一下無人機交通管理系統的演進流程 (Fig 20.)，從一開始傳統的 ATM，因為不適合用來處理高密度的無人機因而發產出了 UTM，但在 UTM 系統中存在著高延遲而導致對無人機的控制有劇烈的影響，因此作者提出了一種基於 MEC 的 UTM 系統，目的就是減輕 UTM 系統中的封包延遲問題，而在作者的 Framework 中，使用了 MEC 之後可以讓封包延遲低於 1ms，因此對無人機的控制取得很好的效果。

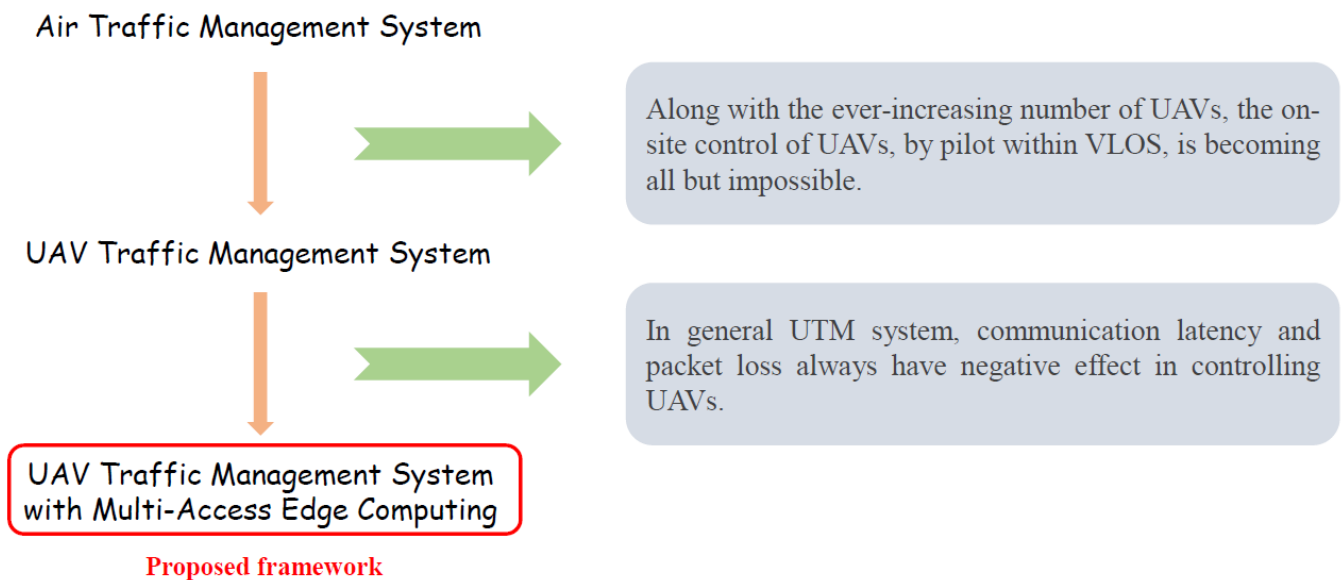


Fig 20. 無人機交通管理系統演進

下圖 Fig 21. 是一個簡單的系統比較表，主要比較的是傳統 ATM、UTM 以及作者提出的 MEC-based UTM。從圖表中可以看到，雖然 MEC-based UTM 系統同時有著可以處理高密度無人機以及低延遲的好處，但其缺點是若要在每個基地台後方都安裝高性能的 Edge Server 則是一筆不小的花費，因此需要按照各種領域的需求來抉擇要使用與否。但也由於其低延遲的優點，令其可應用的領域變得更加廣泛，一些需要低延遲的即時性操作將可以使用無人機並結合作者所提出的系統來達成。

	Air Traffic Management	UAV Traffic Management	UAV Traffic Management (MEC)
Application	That's not designed to handle high density of UAVs' traffic.	That's designed to handle high density of UAVs' traffic without high reliability.	That's designed to handle high density of UAVs' traffic with high reliability.
Latency and packet loss	With higher latency and packet loss.	With higher latency and packet loss.	Latency and packet loss would approach 0.
Cost	High (Labor cost)	Low	High (Deploy edge node)
Scope of application	Low	Middle	High

Fig 21. 無人機管理系統比較表

V. REFERENCE

- [1] (2018, February) Unmanned aerial vehicle (uav) market by system uav platforms, uav payloads – 2025 – marketsandmarkets. Market and Market. [Online]. Available: <https://www.marketsandmarkets.com/MarketReports/unmanned-aerial-vehicles-uav-market-662.html>
- [2] K. Sasaki, N. Suzuki, S. Makido, and A. Nakao, "Vehicle control system coordinated between cloud and mobile edge computing," in 2016 55th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), Sept 2016, pp. 1122-1127.
- [3] T. Jiang, J. Geller, D. Ni, J. Collura, "Unmanned aircraft system traffic management: Concept of operation and sys architecture," in 2017 International Journal of Transportation Science and Technology (IJTST).