

建構在無線網路上之矩陣相乘平行演算法之設計

A Parallel Mobile Computation Environment for Matrix Multiplication

游國忠 吳起智 賴建谷
真理大學數理科學研究所
yugj@email.au.edu.tw

摘要

隨著無線網路的快速發展，越來越多的計算裝置具備了行動計算的能力，在可預見的未來，大量行動計算的提供將成為一重要的研究領域。在目前較前瞻性的無線技術理論中，*OVSF-CDMA* 則具有通訊頻寬大、頻寬利用率高及不同通道碼間不會相互干擾的優點，這些優點使得 *OVSF-CDMA* 進行平行傳輸時的通訊效能最佳。平行計算可用來縮短複雜問題的計算時間，目前平行計算平臺大都建構在有線網路環境下，因為在有線網路環境存在著線路佈置不易、必須預先架構網路環境及網路拓撲變動不易等問題。因此在這篇論文中，我們探討如何在無線通訊環境下建構一平行計算架構，並以矩陣相乘的問題為例，提出以 Divide and Conquer 的概念為基礎之無線平行演算法之設計，將矩陣相乘問題切割成獨立的子矩陣相乘問題，以無線傳輸的方式，分派至多個行動裝置進行平行計算，計算完結果，再進行整合，以期能有效的縮短計算時間。我們在具有前瞻性的 *OVSF-CDMA* 技術上探討無線平行計算環境之技術。在 *OVSF-CDMA* 無線通訊技術方面，我們提出一動態頻寬分配與排程之演算法，使得平行計算演算法之效能可達到最佳化。經由實驗數據分析，我們提出的無線平行計算環境具有縮短計算時間、提升計算效能，提供彈性的網路環境等特性。

關鍵詞：無線網路、平行計算。

I. 簡介

近年來，行動通訊技術日益蓬勃發展，從最早第一代的類比式通訊系統，隨即發展出第二代將電信網路與網際網路(Internet)互相結合的 GSM 數位通訊系統，目前發展中的第三代行動通訊 3G(Third Generation)，更提供高達 2Mbps 的資料傳輸率。整體而言，這些行動通訊系統具有移動性，與傳統有線網路相比，不需花費線路佈置與規劃的成本，使用者不受有形線路的限制，可隨時隨地使用行動裝置。當這些行動通訊裝置具有計算能力，且使用者可以透過行動裝置連結網際網路，享受網際網路上的相關服務，我們可說這些裝置具有行動計算(Mobile Computing)的能力，而行動通訊技術的日益進步，也促使行動計算的相關應用領域快速發展。

行動計算應用的場景必須滿足移動性、即時

資料、以及立即運算的需求，如救災、戰場等都是行動計算常見的應用範圍。舉例來說，使用者旅遊時，可透過行動裝置連接網際網路，獲得道路交通資訊，然而最短路徑的道路可能塞車，此時，移動中的使用者，接收即時道路交通資訊，將道路中每條道路的通行速度作為參數，透過行動計算找出最短時間的路徑，這就是行動計算帶來的便利。越來越多的行動計算應用正不斷改變人們的生活。

目前行動計算的需求，主要是透過網際網路，將行動計算的需求，透過高速電腦或有線網路上的平行計算系統進行運算工作，然而行動通訊技術中，不管是 GSM、3G，皆因為無線網路的硬體限制，使得由 Server(Base Station)端到 Client 端的頻寬不對稱(由 Server 到 Client 的頻寬遠大於由 Client 到 Server 的頻寬)，這類特殊的通訊環境，當使用者欲進行的平行計算需求，需要傳輸大量的資料至 Server 時，將造成通訊傳輸的困境，Client 傳輸至 Server 將花費大量的傳輸時間，使得行動計算的效能大打折扣。另一方面，偏遠地區或是山區等沒有網際網路支援的地區，也無法透過網際網路提供行動計算的服務。舉例來說，許多考古隊、探險隊進行任務時，進入金字塔這類古老遺跡，因為遺跡的阻隔，網際網路的無線電波無法穿透這類建築，這些建築中常有防護設施，避免外人進入，若有需破解防護設施的緊急事件發生，短時間內進行圖形比對或密碼解碼等工作時，將需要一隨建即連網路下的行動計算環境來協助解決問題。

隨建即連網路將特定區域內的行動裝置建構為區域無線網路，在隨建即連網路內，行動裝置通訊範圍內其他裝置可建立連線，具有彈性的網路拓撲架構，不需與網際網路建立連線。因此，隨建即連網路可成為沒有網際網路支援地區、或需要傳輸大量資料至 Server 的行動計算場景的最佳解決方案，透過隨建即連網路中的行動裝置的協同運算，許多行動計算的需求皆可透過隨建即連網路完成。

目前平行計算相關研究大多建立在有線網路的環境，有線網路環境下的裝置，受線路的限制，無法提供使用者行動性的需求，況且當使用者位於山區或無法連上網際網路的地區，傳統行動計算的架構無法解決這種情況，然而，透過隨建即連網路可解決這類問題，在可見的未來，隨建即連網路將成為行動計算之研究的重要議題，因此，我們對於隨建即連網路的平行計算架構上進行研究。我們希望在隨建即連無線網路環境中，建構一平行計算平臺及演算法，使得大量計算如矩陣相乘等問題之計

算時間能大幅縮短。另一方面，藉由無線網路平行性之間進一步發展出更有效率之無線傳輸技術。

因為無線網路可以解決傳統有線網路佈線不易、硬性網路拓樸架構及網路裝置的移動性問題，因此我們採用具前瞻性技術的 OVSF-CDMA 作為無線通訊技術，OVSF-CDMA 具有頻寬利用率好及不同通道碼間不會相互干擾的優點，使得 OVSF-CDMA 進行平行傳輸時的通訊效能優異，透過平行處理可幫助使用者更有效率的計算複雜問題，所以我們希望可以提出一套運行在無線網路環境中的平行計算架構。

在我們提出的無線平行計算架構，我們將提出一套以 Divide and Conquer 為基礎之無線平行演算法，此平行計算架構可運用於多種可利用 Divide and Conquer 概念的運算，例如快速傅利葉轉換 (Fast Fourier Transform; FFT)、矩陣相乘等等。在本篇論文中，將以矩陣相乘的問題為例，將矩陣問題切割成獨立的數個小矩陣問題，透過階層式 recursive 切割問題，直到問題切割至合適的大小，最後階層式回報小問題的解答，有效縮短計算時間。有關 OVSF-CDMA 平臺上之研究，由發起平行計算請求的行動裝置，將參與平行計算裝置建構為一樹狀架構，發起平行計算請求的行動裝置為樹根，由樹根將問題進行切割，將切割後的獨立小問題傳輸到他的子結點，階層式的分派問題，整個分工架構為一完全樹的架構。

以下為本論文的章節架構，在第二章節中，我們將回顧在傳統有線網路上進行矩陣相乘運算的研究。第三章節中則介紹 OVSF-CDMA 通訊的原理與特性。在第四章節中，我們提出 OVSF-CDMA 的無線平行計算架構。第五章節中將以實驗數據來說明本論文所提出的方法之效能。最後一章節是結論，敘述本論文的結果及其未來的研究方向。

II. 相關研究

在本章節中，我們對於平行處理、無線平行計算、及以往矩陣相乘問題的平行計算相關研究作一番探討。

平行處理是一種利用多處理器系統，將複雜及龐大的數值運算工作，加以切割並分配到多個處理器上，藉由多個處理器同時進行計算工作，以降低計算所花費的時間。平行處理的議題已經發展許久，也帶動了許多相關技術的發展，例如在影像處理的應用中，常把影像視為矩陣型態的資料，並以矩陣的運算方式來達成影像處理的目的，當進行矩陣相乘等問題的平行處理運算時，可以藉由矩陣運算的平行化處理，縮短運算時間，因此矩陣相乘問題常常成為平行處理研究的重要議題。

Chih-Fang Wang 及 Sahni S. 提出一個 TIS-Mesh 環境下的矩陣相乘演算法[2]，這個矩陣相乘演算法，可使處理器間的資訊傳遞降到最低，

避免處理器間的資訊溝通降低矩陣相乘運作的系統效能，進而有效提高整體效能，不過這是一個有線通訊環境中的研究，對於日漸重要的裝置移動性問題並未解決，在我們提出的無線平行計算環境中可提供裝置隨時移動的特點。

無線通訊技術是近十年來相當受注目的新興研究議題，然而平行計算的研究上停留在有線通訊環境中，目前對於無線平行計算的研究甚少，Lih-Chyau Wu 及 Shing-Tsaan Huang 提出一個在無線環境中的平行計算架構[3]，該架構使用無線通訊技術取代分散式記憶體多處理器系統的高速網路技術，這個平行計算架構可以輕易的模擬 CRCW PRAM 的平行處理架構，並且提出一個演算法，對於找最大值與排序問題，可使用 N 個處理器協助平行運算工作，使得時間複雜度達到 $O(1)$ 。可是他們沒有對於通訊頻寬的分配與排程進行設計，因此通訊碰撞及頻寬利用率將無法達到最佳的效果。此外，Chao-Tsun Chang 等人提出一個在藍芽無線通訊環境下建構的 BlueCube 的平行計算架構[1]，其透過角色分配及角色互換的運作，建構一個類似 HyperCube 架構的無線平行計算環境-BlueCube，這個平行計算環境可減少平均的 Routing Length 及 Average Number of Path Served by Each Device，除此之外還提出一個分散式調整 Piconet 結構的協定，這個協定透過 Piconet 角色調整的方式，將發生流量擁塞的 Piconet 分解，然而這個平行計算架構並未針對實際平行計算問題作效能評估。基於上述方法的缺點，本篇論文所提出之方法將特別針對通訊頻寬的分配與排程進行設計，以期達到最佳的效能，並以矩陣相乘為例子對其做實際的效能評估。

III. 序言

在本章節中，我們將介紹無線平行計算環境及無線通訊背景知識，我們先介紹平行計算時的假設環境，接下來介紹 OVSF-CDMA 的特性及其所使用 OVSF Code Tree 產生通道碼的原理。

3.1 環境假設

本篇論文假設參與平行計算工作的所有的行動裝置皆可直接通訊，即任兩個裝置間可以直接通訊，且整個網路環境為同質網路環境，即每個行動裝置的計算能力相同，以每秒百萬指令 MIPS (Million Instructions Per Second) 為單位。

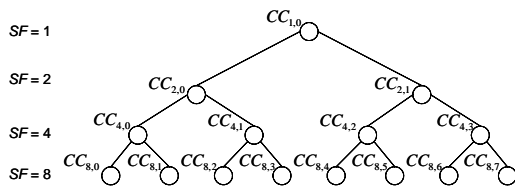
3.2 劃碼多工存取技術

在 CDMA 通訊機制中常見的是 WCDMA，WCDMA 採用 DS-SS (Direct Spread Code Division Multiple Access) 無線傳輸技術，每個傳輸通道都使用了彼此具有正交的通道碼，使得不同實體通道間，可以使用相同的頻率而不會互相干擾，

以下我們將介紹 WCDMA 常見的 OVFS-Code Tree 通道碼的通訊方式。

3.2.1 OVFS Code Tree

OVFS-CDMA 所使用的通道碼是由展頻係數所產生的,如圖(一)為 OVFS Code Tree 的通道碼,SF(Spreading Factor)為展頻係數, CC_{ij} 為通道碼, i 為展頻係數值, j 為在 i 展頻係數下的第 j 個通道碼,OVFS Code Tree 的通道碼具有下列特性,Tree 的每一階層通道碼具有相同的頻寬,上層通道碼的頻寬為下層通道碼頻寬的兩倍,此外,為了保持通道碼間的正交性,任一通道碼 CC_{ij} 需滿足兩個條件: (1) CC_{ij} 到根結點間沒有其他碼被配置。(2) CC_{ij} 的所有子樹的通道碼都沒被使用。



圖(一) : OVFS Code Tree 架構圖

IV. The Proposal Protocol

當進行平行計算的任務時,我們需要另行建立一個進行平行計算工作的網路環境,產生這個網路環境的協定我們稱為 Wireless Parallel Matrix Multiplication Protocol(WPMM),這個協定建立一個階層式的樹狀架構,這個階層式的樹狀架構融入 *Divide and Conquer* 的演算法概念,必須進行多對裝置間的平行傳輸, CDMA 對於平行傳輸既可動態調整通道碼又可使得不同頻道間的通訊不受干擾,使用 CDMA 可使平行傳輸的通訊效能提高,雖然目前的硬體花費較高,但是著眼於未來,CDMA 成為無線通訊的主流及其硬體花費也會降低。接下來我們對於使用到的符號進行定義。

4.1 符號定義

為了方便介紹本篇論文的詳細內容,我們將定義下列符號:

Definition: $A_{m,n}(i,j)$ 及 $B_{m,n}(i,j)$ (矩陣相乘的相乘矩陣 A 及矩陣 B)

矩陣相乘的基本精神為兩個矩陣相乘得到新的乘積矩陣,對於個別矩陣我們使用下列符號表示,一個矩陣包含行與列, A (or B)表示矩陣的識別標籤, $A(i,j)$ 表示 A 矩陣中第 i 列第 j 行的元素, m,n 表示該矩陣是由 m 列、 n 行所組成的矩陣,若矩陣的行數與列數相同,即 $A_{m,n}(i,j)$ 中 $m=n$,則使用 $A_m(i,j)$ 表示。

Definition: $C_{m,n}(i,j)$ (矩陣相乘的乘積矩陣 C)

矩陣相乘為兩個矩陣相乘得到新的乘積矩陣,對於 $A_{m,p}$ 與 $B_{p,n}$ 兩矩陣相乘後所得到的乘積矩陣 $C_{m,n}$,我們以下列符號表示

$$C_{m,n} = A_{m,p} \times B_{p,n}$$

其中, C 為乘積矩陣的標籤, $C_{m,n}$ 表示 C 矩陣是由 m 列、 n 行所組成的乘積矩陣。

Definition: $Dt(M)$ (M 矩陣傳輸所需的頻寬)

每個矩陣傳輸時所需的頻寬大小可依據該矩陣的維度與元素的資料大小來決定,我們以下列符號來表示 M 矩陣傳輸所需的頻寬大小:

$$Dt(M) = El \times m \times n$$

其中 M 為矩陣標籤, El 為 M 矩陣元素的資料量,例如矩陣元素的數值範圍在 -32768 至 32767 間,則 $El=2$ bytes, m 、 n 為 M 矩陣的維度。

Definition: $T^n_{i,j}$ (n 元樹的的樹狀架構)

我們提出的無線平行計算環境,所有參與平行計算工作的裝置,可組成一個階層式的樹狀拓撲架構,這個樹狀架構為 n 元樹的架構,每個裝置即為樹狀拓撲的結點,我們以 $T^n_{i,j}$ 來表示,其中 n 表示樹狀架構的分支數, i 表示樹狀結構中的第 i 層, $i \geq 1$, j 表示該階層中的裝置編號,依序由左向右。 $0 \leq j < n^i$,若省略 i,j 值, T^n 表示一 n 元樹的樹狀架構。

Definition: TL (階層式樹狀拓撲架構的層級)

我們提出的無線平行計算環境,將所有參與平行計算工作的裝置,組成一個階層式的樹狀拓撲架構 $T^n_{i,j}$, $T^n_{i,j}$ 的 i 表示該結點的層級,為了方便描述每個裝置的層級,我們以 TL 表示階層式樹狀拓撲架構的層級。

Definition: TL_{max} (階層式樹狀拓撲架構的最大層級)

我們提出的無線平行計算環境,將所有參與平行計算工作的裝置,組成一個階層式的樹狀拓撲架構,為了方便描述階層式的樹狀拓撲架構的最大層級,我們以 TL 表示階層式樹狀拓撲架構的最大層級,例如階層式的樹狀拓撲架構總共有三層,則 TL_{max} 為 3。

Definition: Dim_M (矩陣的行列數)

我們提出的無線平行計算演算法是以 divide and conquer 的概念來切割矩陣並使用快速矩陣相乘演算法,環境中使用的裝置是屬於同質 (homogeneous) 環境,使用四分法切割矩陣進行快速矩陣相乘運算,每個矩陣的行數與列數需相同,我們以 Dim_M 表示某矩陣的行列數, Dim_M 需滿足 $k=2^z$ 的條件, $z \in$ 整數。

Definition: N_d (參與平行計算工作的裝置數)

協助平行計算工作的裝置數越多，發起平行計算工作的裝置得到最後運算結果所花費的時間越少，我們以符號 N_d 表示所有參與平行計算工作的裝置數，其中 N_d 需滿足 $N_d=8^z, z \in \text{整數}$ 。

Definition: N_{max} (最大可參與平行計算工作的裝置數)

環境中可參與平行計算的裝置越多，建構階層式的樹狀架構時的樹高也可越高，平行計算的效能也越高，我們以符號 N_{max} 表示環境中最大可參與平行計算工作的裝置數。

Definition: EL (矩陣元素的資料空間)

每個進行矩陣相乘的矩陣的元素都有其數值範圍，我們以 EL 表示不同的矩陣元素的資料空間， $EL=1$ 表示使用 1 byte 儲存，數值範圍為 -128 ~ 127， $EL=2$ 表示使用 2 byte 儲存，數值範圍為 -32768 ~ 32767。

Definition: BW_{max} (通訊最大總頻寬)

通訊裝置受限於開放的頻帶大小，均有其最大頻寬的限制，我們以 BW_{max} 表示平行計算環境所使用的最大總頻寬。

Definition: $BW_{cc}(CC_{i,j})$ 通道碼通訊最大總頻寬)

通訊裝置受限於使用的通道碼所限制的頻寬大小，我們以 BW_{cc} 表示裝置使用的通道碼的最大頻寬為 BW_{cc} 。

Definition: D_p (行動裝置的計算能力)

參與平行計算的通訊裝置都有其特定的計算能力，我們以 D_p 表示裝置的計算能力，例如：一般的 PDA 裝置的 CPU 的運算能力為 100 MIPS，100 MIPS 處理 100M 個純量運算。

Definition: Ph (Scattering Phase 或 Gathering Phase 的子階段)

無線平行計算的裝置有如建構一個階層式的樹狀架構，無線平行計算的運作過程中的 Scattering、Gathering 階段，均包含階層式的樹狀架構的一些散佈矩陣資料或收集子矩陣資料的一些子階段，對於這些子階段我們使用符號 Ph 來表示， $Ph \leq TL_{max} - 1$ 。

Definition: Ph_{max} (最大 Scattering Phase 或 Gathering Phase 的子階段)

無線平行計算的裝置有如建構一個階層式的樹狀架構，階層式的樹狀樹高可視為散佈矩陣資料或收集子矩陣資料的一些子階段，因此子階段的總數量我們以表示 Ph_{max} ， $Ph_{max}=TL_{max}-1$ 。

4.2 矩陣相乘問題

矩陣乘法在許多科學領域上的應用相當廣泛，其主要原理為 $A_{m,p}$ 與 $B_{p,n}$ 矩陣相乘得到乘積矩陣 $C[m,n]$ ，我們以 $C_{m,n} = A_{m,p} \times B_{p,n}$ 表示，矩陣 A 的行數與矩陣 B 的列數必須相同，A 矩陣與 B 矩陣才可進行矩陣相乘運算，對於乘積矩陣的個別元素 $C_{i,j}$ 滿足 $C_{i,j} = \sum_k A_{i,k} \times B_{k,j}, 1 \leq k < \infty$ 。對

於矩陣相乘的運算，快速矩陣相乘演算法具有 *Divide and Conquer* 的概念，一般常見的快速矩陣相乘演算法為標準演算法，以下，我們將就標準演算法作概略的介紹。

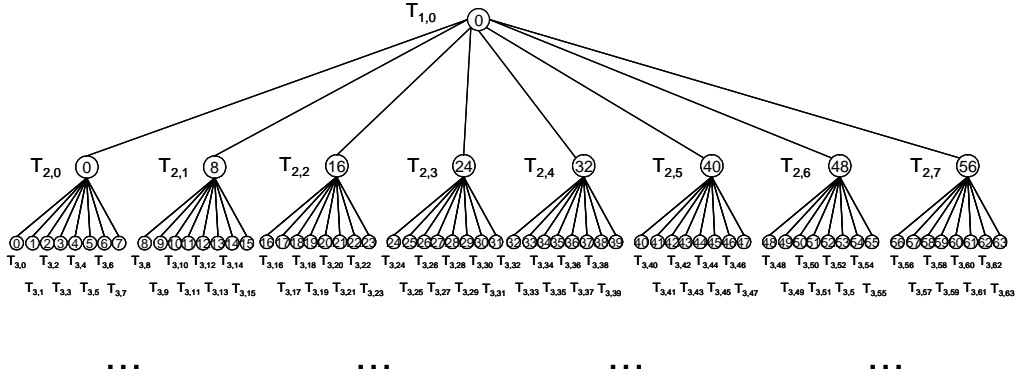
表(一)：標準演算法數學式

Pre-Additions	Recursive calls	Post-Additions
	$QP_1 = QA_{11} \times QB_{11}$	
	$QP_2 = QA_{12} \times QB_{21}$	
	$QP_3 = QA_{21} \times QB_{11}$	$QC_1 = QP_1 + QP_2$
	$QP_4 = QA_{22} \times QB_{21}$	$QC_2 = QP_3 + QP_4$
	$QP_5 = QA_{11} \times QB_{12}$	$QC_3 = QP_5 + QP_6$
	$QP_6 = QA_{12} \times QB_{22}$	$QC_4 = QP_7 + QP_8$
	$QP_7 = QA_{21} \times QB_{21}$	
	$QP_8 = QA_{22} \times QB_{22}$	

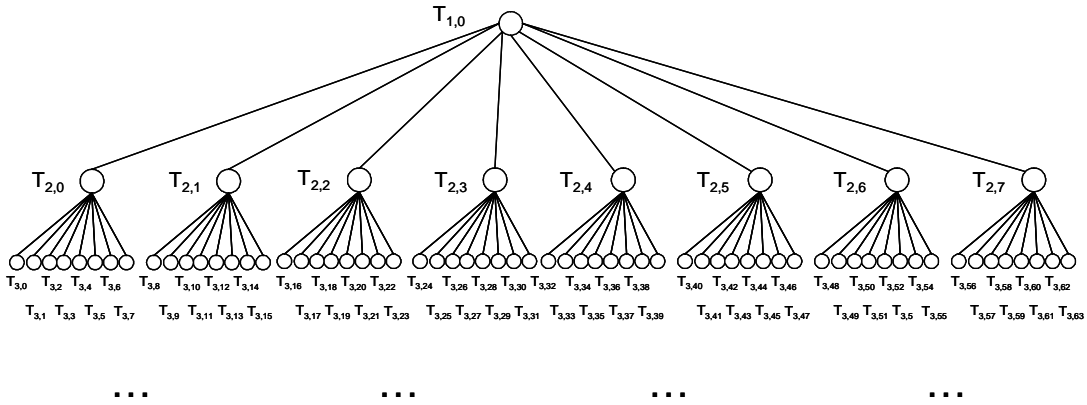
4.3 無線平行矩陣相乘演算法

對於複雜問題使用平行演算法可以縮短其等待時間，然而以往的平行計算研究都是在有線通訊的環境，在無線通訊的環境中使用平行演算法，往往裝置間的平行傳輸十分頻繁，因此建構平行傳輸的機制是無線平行演算法很重要的一個環節，而使用 OVSF-CDMA 的通訊機制可有較佳的平行傳輸效能，OVSF-CDMA 的通訊機制是透過彼此正交的通道碼的方式進行通訊，使用不同通道碼的通訊頻道，彼此不會相互干擾，這在通訊平行傳輸時，可解決通訊碰撞問題，為了更好的平行傳輸效能，我們採用 OVSF-CDMA 的通訊機製作為無線平行矩陣相乘演算法的通訊平臺。總結來說，我們提出的無線平行矩陣相乘演算法以 *Divide and Conquer* 的概念為核心，將複雜的問題分成問題切割 (Scattering Phase)、執行計算 (Computing Phase) 及合成問題 (Gathering Phase) 三個階段，以 OVSF-CDMA 為通訊平臺時，由於平行計算過程中傳輸時間是最主要的花費，而 OVSF Code Tree 切割的頻道碼數目為 2 的指數次方，對應到快速矩陣相乘演算法將原先複雜矩陣相乘問題變成八個較小的矩陣相乘問題，與 OVSF Code Tree 的通道碼的切割碼數為 2 的指數次方可有效整合、利用，提高頻寬利用率，我們採用標準的快速矩陣演算法進行 OVSF-CDMA 平臺下的平行計算演算法。

我們假設所有裝置都在通訊範圍內，任兩裝置可以直接通訊，且任一裝置皆知道所有通訊範圍內的其他裝置的 ID，一開始，要進行平行計算的



圖(二)：虛擬編號與階層式樹狀結構對照圖



圖(三)：平行計算階層式樹狀架構

裝置發起平行計算的請求，由於任一裝置知道所有通訊範圍內其他裝置的 ID，發起平行計算請求的裝置，視環境中裝置的計算能力、裝置個數及欲解決的矩陣問題複雜度，建構一個階層式的樹狀結構 T^S ，發起裝置為 $T_{1,0}$ ，並決定階層式樹狀架構的 TL_{max} ，此時所需的裝置數為 $8^{TL_{max}-1}$ ，將需參與平行計算的裝置，建立一實體 ID 與虛擬編號的對照表，稍後皆使用虛擬編號來表示通訊的裝置，對任一階層式樹狀結構裝置 $T_{i,j}$ ，其虛擬編號為 $j \times 8^{TL_{max}-i}$ ，如圖(二)所示 $TL_{max}=3$ ，則 $T_{2,3}=24$ ，裝置 $T_{2,3}$ 的虛擬編號為 24，其中，某些結點的虛擬編號相同，例如 $T_{1,0}$ 、 $T_{2,0}$ 、 $T_{3,0}$ ，表示該點為同一裝置且具有多種任務， $T_{1,0}$ 將矩陣資料散佈到 $TL=2$ 的裝置，下一階段 $T_{2,0}$ 將矩陣資料散佈到 $TL=3$ 的裝置，由於這些通訊不會互相衝突，所以我們都使用虛擬編號為 0 的裝置。建構出階層式的樹狀結構及其 TL_{max} 後，發起平行計算要求的裝置，將虛擬編號與實體 ID 對照表及樹狀結構資訊以 $CC_{1,0}$ 通道碼廣播給其他裝置，接下來的平行計算程式分為 Scattering Phase、Computing Phase、Gathering Phase。首先在 Scattering Phase，將矩陣相乘問題切割成八個子矩陣相乘問題，遞迴切割直到矩陣相乘問題的時間複雜度達到可接受的程度，接下來執行 Computing Phase，將這些時間複雜度達可接受範圍的矩陣相乘問題進行計算，最後在 Gathering Phase，將計算後的小矩陣相乘結果收集至發起平

行計算問題的裝置。Scattering Phase 又分為幾個小階段，總共有 $Ph_{max}=TL_{max}-1$ 個階段，Gathering Phase 也有相同的小階段收集資訊，如圖(四)所示。



圖(四)：OVSF-CDMA 平行計算階段圖

A. Scattering Phase

在 Scattering Phase 我們將參與平行計算工作的裝置視為一階層式的樹狀結構，發起平行計算工作的裝置為階層式樹狀結構的樹根裝置，對所有樹狀結構中的父結點而言，父結點將矩陣資料使用標準的快速矩陣相乘演算法，分割為八個子矩陣相乘問題，父結點為 $T_{i,j}$ 則子結點為

$$T_{i+1, j \times 8^{TL_{max}-i}} \sim T_{i+1, j \times 8^{TL_{max}-i+7}}$$

共八個裝置，整個 Scattering Phase 又分為多個小 phase， $Ph=1$ 時， $TL=1$ 的裝置將矩陣資料散佈到 $TL=2$ 的裝置， $PH=2$ 時， $TL=2$ 的裝置將矩陣資料散佈到 $TL=3$ 的裝置，依此類推，每個小階段需進行傳輸的裝置 $T_{i,j}$ 使用的通道碼為 $CC_{8^{i-1}, j}$ ，Scattering Phase 主要

是將矩陣相乘的矩陣資料以 Divide and Conquer 的方式分割出去，將矩陣資料散佈到其他裝置，散佈的方式可透過 OVSF-CDMA 的通訊裝置，假設通訊總頻寬為 BW_{max} ，通道碼為 $CC_{1,0}$ ，對於任一矩

陣相乘問題 $A_{m,k} \times B_{k,n}$ ，其維度為 Dim_A 及 Dim_B 且 $Dim_A=Dim_B$ ，在 $TL=1$ 的裝置為進行 $A_{m,k} \times B_{k,n}$ 矩陣運算的初始裝置，此時以快速矩陣相乘演算法將矩陣相乘問題進行切割，可將 $A_{m,k} \times B_{k,n}$ 變成如表(一)的 QP_1 至 QP_8 的八個矩陣相乘問題， QP_1 至 QP_8 的八個矩陣相乘問題的計算任務，由 $TL=2$ 的所有裝置來進行計算任務，即 $T_{2,0} \sim T_{2,7}$ 的八個裝置依序處理 QP_1 至 QP_8 ，依此類推，對於 $T_{i,j}$ 裝置將矩陣資料散佈出去，接受 $T_{i,j}$ 矩陣資料的結點為

$$T_{i+1, j \times 8^{TL_{max}-TL}} \sim T_{i+1, j \times 8^{TL_{max}-TL+7}}$$

其所需接收的資料如圖(五)所示， $t_1 \sim t_8$ 每個時段傳送 QA 或 QB 的部分矩陣資料， t_1 時段傳送 QA_{11} ， t_2 時段傳送 QA_{12} 依此類推，共需 PH_{max} 個階段將矩陣相乘問題遞迴分割出去，每個 Phase 需花費 $((Dim_A \div 2^{Ph-1})^2 \times 2 \times EL) \div (BW_{CC})$ 秒的傳輸時間，其中裝置 $T_{i,j}$ 使用的通道碼為 $CC_{8^{i-1}, j}$ ，其所處理的部分矩陣相乘問題 k 之維度 Dim_k 為

$$Dim_k = \frac{Dim_A}{2^{TL-1}}$$

以 $Dim_A=2048$ ， $TL_{max}=3$ ， $D_p=100M$ ， $BW_{max}=10MB$ ， $EL=2$ 為例，我們可以得到如圖(三)的階層式樹狀架構，此時所需的平行計算裝置數為 8^2 ，共需 2 個階段將矩陣問題分割出去，在 $Ph=1$ 的子階段， $TL=1$ 的所有裝置，為進行 $A_{m,k} \times B_{k,n}$ 矩陣運算的初始裝置，對應到圖(三)為 $T_{1,0}$ 結點，此時以快速矩陣相乘演算法進行分割，可將 $A_{m,k} \times B_{k,n}$ 變成如表(一)的 QP_1 至 QP_8 的八個矩陣相乘問題， QP_1 至 QP_8 的八個矩陣相乘問題的計算任務，由對應到圖(三) $TL=2$ 的所有裝置來執行任務，即 $T_{2,0} \sim T_{2,7}$ 的裝置進行處理，此時使用 $CC_{1,0}$ 通道碼，共需花費

$(Dim_k^2 \times 2 \times EL) \div (BW_{max} \times 1024^2) = 1.6 \text{ sec}$ ，在 $Ph=2$ 的子階段，對於 $T_{2,0} \sim T_{2,7}$ 的裝置再將矩陣相乘問題分散出去時使用的通道碼為 $CC_{8,0} \sim CC_{8,7}$ ，頻寬皆為 $1/8CC_{1,0}$ ，共需花費

$((Dim_k \div 2)^2 \times 2 \times EL) \div (BW_{CC} \times 1024^2) = 3.2 \text{ sec}$ 。在 Scattering Phase，不同子階段所花費的傳輸時間為 $((Dim_A \div 2^{Ph-1})^2 \times 2 \times EL) \div (BW_{CC})$ 秒。

B. Computing Phase

這個階段是階層式的樹狀架構的樹葉部分的裝置對於矩陣相乘問題進行計算，對於矩陣相乘運算而言，此時的矩陣維度 Dim_k 為原先矩陣維度的 $1/2^{TL_{max}-1}$ ，其進行純量加法或純量乘法的次數為 $2 \times Dim_A^3 - Dim_A^2$ ，裝置的計算能力為 D_p ，所需的時間為

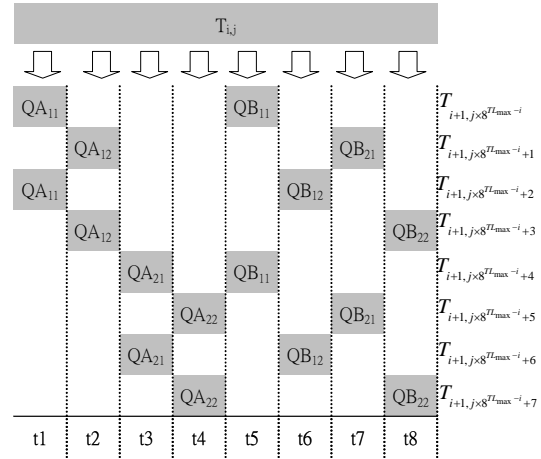
$$(2 \times (Dim_A / 2^{TL_{max}-1})^3 - Dim_A / 2^{TL_{max}-1}^2) \div D_p$$

以 Scattering Phase 的例子而言，直接由單一裝置運算 $A_{m,k} \times B_{k,n}$ 所需花費的時間為

$$(2 \times ((2048)^3 - 2048)) \div 100M = 163.84 \text{ 秒}$$

此時 Computing Phase 所需花費的時間為

$$(2 \times ((2048/2^2)^3 - (2048/2^2)^2)) \div 100M = 2.56 \text{ sec}$$



圖(五)：OVSF-CDMA Scattering Phase 父結點傳輸子矩陣資料給子結點示意圖

C. Gathering Phase

這個階段是將 Computing Phase 的小矩陣相乘運算結果匯整至初始發起平行計算工作的裝置的階段，對於任一 Computing Phase 的裝置 $T_{i,j}$ 必須將計算後的矩陣資料傳輸至其階層式樹狀架構的父結點為 $T_{i-1, \lfloor \frac{j}{8} \rfloor}$ 與 Scattering Phase 相同，總共

需進行 Ph_{max} 個小階段，每一小階段，父結點均接收所有子結點計算完的矩陣資料，對任一 $T_{i,j}$ 結點其 $TL=i$ ，其父結點總共需進行 7 次矩陣資料傳輸與四次矩陣加法運算，這七次矩陣運算的矩陣 k 之 $Dim_k = Dim_A / 2^{TL-1}$ ，使用的通道碼為

$CC_{8^{i-2}, j}$ ，其大小為 $BW_{max} / 8^{i-2}$ ，由於父結點與子結點有一為相同裝置，原先七次的傳輸可縮減為六次，這六次矩陣資料傳輸所需花費的時間為 $(6 \times Dim_k^2 \times EL) \div CC_{8^{i-2}, j}$ ，四次矩陣加法運算

所需花費的時間為 $(4 \times Dim_k^2) \div D_p$ ，假設 $Dim_A=2048$ ， $D_p=100 \text{ MIPS}$ ， $BW_{max}=10M$ ，以 $Ph=2$ 為例，即 $TL=3$ 的結點傳輸至 $TL=2$ 的父結點，傳輸所需花費的總時間為

$(6 \times 512^2 \times 2) \div (10M / 8) = 2.4 \text{ 秒}$ ，加法運算所花費的時間為 $(4 \times 1024^2) \div 100M = 0.04 \text{ 秒}$ ，以 $Ph=1$ 為例，即 $TL=2$ 的結點傳輸至 $TL=1$ 的父結點，傳輸所需花費的時間為

$(6 \times 1024^2 \times 2) \div (10M / 1) = 1.2 \text{ 秒}$ ，加法運算所花費的時間為 $(4 \times 1024^2) \div 100M = 0.04 \text{ 秒}$ 。

Scattering Procedure

- Step 1: 判斷是否收到進行 Scattering Phase 某個 Ph 小階段的訊息
- Step 2: 若收到進行 Scattering Phase 某個 Ph 小階段的訊息, 則 go to Step 3, 否則 go to Step 1。
- Step 3: 若目前裝置 T_{ij} 的 $i = Ph$, 則進行快速矩陣相乘的標準演算法, 使用 $CC_{8^{i-1}, j}$ 通道碼, 並依序將 $QA_{11} \sim QB_{22}$ 等八個部分矩陣資料傳送至子結點 $T_{i+1, j \times 8^{TL_{max}-TL}} \sim T_{i+1, j \times 8^{TL_{max}-TL+7}}$, go to Step 6。
- Step 4: 若目前裝置 T_{ij} 的 $i = Ph+1$, 則使用 $CC_{8^{i-2}, j}$ 通道碼準備接收矩陣資料, go to Step 5。
- Step 5: 判斷父結點傳輸的資料是否需要接收, 若需要才接收。
- Step 6: 判斷是否 $Ph < Ph_{max}$, 若 $Ph < Ph_{max}$ 則 go to Step 1, 否則結束 Scattering Procedure。

Computing Procedure

- Step 1: 判斷是否收到進行 Computing Phase 的訊息。
- Step 2: 若收到進行 Computing Phase 的訊息, 則 go to Step 3, 否則 go to Step 1。
- Step 3: 若目前裝置 T_{ij} 的 $i = TL_{max}$, 則進行矩陣相乘運算, 否則 go to Step 1。

Gathering Procedure

- Step 1: 判斷是否收到進行 Gathering Phase 某個 Ph 小階段的訊息。
- Step 2: 若收到進行 Gathering Phase 某個 Ph 小階段的訊息則 go to Step 3, 否則 go to Step 1。
- Step 3: 若目前裝置 T_{ij} 的 $i = TL_{max}-(Ph-1)$, 則將整合的矩陣資料以 $CC_{8^{i-2}, j}$ 通道碼傳送至 $T_{i, j}$, go to Step 6。
- Step 4: 若目前裝置 T_{ij} 的 $i = TL_{max}-(Ph-1)-1$, 則使用 $CC_{8^{i-2}, j}$ 通道碼準備接收矩陣資料, go to Step 5。
- Step 5: 判斷是否 $Ph < Ph_{max}$, 若 $Ph < Ph_{max}$ 則 go to Step 1, 否則結束 Gathering Procedure。

4.3.1 Enhanced OVSF 無線矩陣相乘平行演算法 (EWPMM)

我們原先提出的無線平行矩陣相乘演算法以 *Divide and Conquer* 的概念為核心, 將複雜的問題分成問題切割 (Scattering Phase)、執行計算 (Computing Phase) 及合成問題 (Gathering Phase) 三

個階段, 平行計算程式可分為 Scattering Phase、Computing Phase、Gathering Phase。然而在 Scattering Phase, 將矩陣相乘問題切割成八個子矩陣相乘問題, 遞迴切割直到矩陣相乘問題的時間複雜度達到可接受的程度, Scattering Phase 又分為幾個小階段, 總共有 $Ph_{max}=TL_{max}-1$ 個階段, 我們以階層式的樹狀結構表示參與平行計算工作的裝置, 由於 Scattering Phase 的散佈資料過程除了樹根裝置與樹葉裝置的其他裝置, 皆需接收部分矩陣資料再將接收的部分矩陣資料分割成更小的部分並散佈到子結點, 若是我們可以將樹根裝置的矩陣資料, 直接散佈到樹葉裝置上, 則可節省許多傳輸所花費的時間, 於是我們提出這個 Enhanced OVSF-CDMA 的無線平行矩陣相乘演算法, 改善其 Scattering Phase 的效能, 至於 Computing Phase 與 Gathering Phase 則與 OVSF-CDMA 的無線平行演算法相同。

Definition: $[i,j]PM_{n,m}$ (P 矩陣的部分矩陣 M)

為了縮短計算時間, 我們將矩陣資料散佈到各個裝置, 每各裝置將接收部分矩陣資料, 為了表示各個裝置接收的部分矩陣資料, 我們以 $[i,j]PM_{n,m}$ 表示, 其中 $[i,j]$ 表示在 M 矩陣中部分矩陣的起始位置, $PM_{n,m}$ 表示部分矩陣的維度, 因此 $[1,2]PA_{2,2}$ 表示, 在 A (如圖(六)(a)) 這個原始矩陣中的

$$\begin{bmatrix} A_{12} & A_{13} \\ A_{22} & A_{23} \end{bmatrix}$$

這個部分矩陣資料

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix} \quad \begin{bmatrix} B_{11} & B_{12} & B_{13} & B_{14} \\ B_{21} & B_{22} & B_{23} & B_{24} \\ B_{31} & B_{32} & B_{33} & B_{34} \\ B_{41} & B_{42} & B_{43} & B_{44} \end{bmatrix}$$

(a) 維度為 4x4 的 A 矩陣 (b) 維度為 4x4 的 B 矩陣

圖(六): 維度為 4x4 的矩陣元素

A. Scattering Phase

在 Scattering Phase, 發起平行計算請求的裝置, 視環境中裝置的計算能力、裝置個數及欲解決的矩陣問題複雜度, 建構一個階層式的樹狀結構 T^8 , 發起裝置為 $T_{1,0}$, 並決定階層式樹狀架構的 TL_{max} , 此時所需的裝置數為 $8^{TL_{max}-1}$, 並建立實體 ID 與虛擬編號的對照表, 假設通訊總頻寬為 BW_{max} , 通道碼為 $CC_{1,0}$, 對於任一矩陣相乘問題 $A_{m,k} \times B_{k,n}$, 其維度為 Dim_A 及 Dim_B 且 $Dim_A=Dim_B$, 我們的目標為從 $T_{1,0}$ 直接傳送矩陣資料至樹葉裝置, 為了達到這個目標, 我們透過如圖(七)的遞迴式, 可產生八個新的矩陣相乘問題, 及每個矩陣相乘問題所需包含的部分矩陣資料, $T_{1,0}$ 依序分派給 $T_{2,0} \sim T_{2,7}$, 同理 $T_{2,0} \sim T_{2,7}$ 的部分矩

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix} \times \begin{bmatrix} B_{11} & B_{12} & B_{13} & B_{14} \\ B_{21} & B_{22} & B_{23} & B_{24} \\ B_{31} & B_{32} & B_{33} & B_{34} \\ B_{41} & B_{42} & B_{43} & B_{44} \end{bmatrix} = \\
[0,0]PA_{2,2} \times [0,0]PB_{2,2} + [0,2]PA_{2,2} \times [2,0]PB_{2,2} + \\
[0,0]PA_{2,2} \times [2,0]PB_{2,2} + [0,2]PA_{2,2} \times [2,2]PB_{2,2} + \\
[2,0]PA_{2,2} \times [0,0]PB_{2,2} + [2,2]PA_{2,2} \times [2,0]PB_{2,2} + \\
[2,0]PA_{2,2} \times [0,2]PB_{2,2} + [2,2]PA_{2,2} \times [2,2]PB_{2,2} = \\
[0,0]PA_{1,1} \times [0,0]PB_{1,1} + [0,1]PA_{1,1} \times [1,0]PB_{1,1} + [0,0]PA_{1,1} \times [1,0]PB_{1,1} + [0,1]PA_{1,1} \times [1,1]PB_{1,1} + [1,0]PA_{1,1} \times [0,0]PB_{1,1} + [1,1]PA_{1,1} \times [1,0]PB_{1,1} + [1,0]PA_{1,1} \times [0,1]PB_{1,1} + [1,1]PA_{1,1} \times [1,1]PB_{1,1} + \\
[0,2]PA_{1,1} \times [2,0]PB_{1,1} + [0,3]PA_{1,1} \times [3,0]PB_{1,1} + [0,2]PA_{1,1} \times [3,0]PB_{1,1} + [0,3]PA_{1,1} \times [3,1]PB_{1,1} + [1,2]PA_{1,1} \times [2,0]PB_{1,1} + [1,3]PA_{1,1} \times [3,0]PB_{1,1} + [1,2]PA_{1,1} \times [2,1]PB_{1,1} + [1,3]PA_{1,1} \times [3,1]PB_{1,1} + \\
[0,0]PA_{1,1} \times [2,0]PB_{1,1} + [0,1]PA_{1,1} \times [3,0]PB_{1,1} + [0,0]PA_{1,1} \times [3,0]PB_{1,1} + [0,1]PA_{1,1} \times [3,1]PB_{1,1} + [1,0]PA_{1,1} \times [2,0]PB_{1,1} + [1,1]PA_{1,1} \times [3,0]PB_{1,1} + [1,0]PA_{1,1} \times [2,1]PB_{1,1} + [1,1]PA_{1,1} \times [3,1]PB_{1,1} + \\
[0,2]PA_{1,1} \times [2,2]PB_{1,1} + [0,3]PA_{1,1} \times [3,2]PB_{1,1} + [0,2]PA_{1,1} \times [3,2]PB_{1,1} + [0,3]PA_{1,1} \times [3,3]PB_{1,1} + [1,2]PA_{1,1} \times [2,2]PB_{1,1} + [1,3]PA_{1,1} \times [3,2]PB_{1,1} + [1,2]PA_{1,1} \times [2,3]PB_{1,1} + [1,3]PA_{1,1} \times [3,3]PB_{1,1} + \\
[2,0]PA_{1,1} \times [0,0]PB_{1,1} + [2,1]PA_{1,1} \times [1,0]PB_{1,1} + [2,0]PA_{1,1} \times [1,0]PB_{1,1} + [2,1]PA_{1,1} \times [1,1]PB_{1,1} + [3,0]PA_{1,1} \times [0,0]PB_{1,1} + [3,1]PA_{1,1} \times [1,0]PB_{1,1} + [3,0]PA_{1,1} \times [0,1]PB_{1,1} + [3,1]PA_{1,1} \times [1,1]PB_{1,1} + \\
[2,2]PA_{1,1} \times [2,0]PB_{1,1} + [2,3]PA_{1,1} \times [3,0]PB_{1,1} + [2,2]PA_{1,1} \times [3,0]PB_{1,1} + [2,3]PA_{1,1} \times [3,1]PB_{1,1} + [3,2]PA_{1,1} \times [2,0]PB_{1,1} + [3,3]PA_{1,1} \times [3,0]PB_{1,1} + [3,2]PA_{1,1} \times [2,1]PB_{1,1} + [3,3]PA_{1,1} \times [3,1]PB_{1,1} + \\
[2,0]PA_{1,1} \times [0,2]PB_{1,1} + [2,1]PA_{1,1} \times [1,2]PB_{1,1} + [2,0]PA_{1,1} \times [1,2]PB_{1,1} + [2,1]PA_{1,1} \times [1,3]PB_{1,1} + [3,0]PA_{1,1} \times [0,2]PB_{1,1} + [3,1]PA_{1,1} \times [1,2]PB_{1,1} + [3,0]PA_{1,1} \times [0,3]PB_{1,1} + [3,1]PA_{1,1} \times [1,3]PB_{1,1} + \\
[2,2]PA_{1,1} \times [2,2]PB_{1,1} + [2,3]PA_{1,1} \times [3,2]PB_{1,1} + [2,2]PA_{1,1} \times [3,2]PB_{1,1} + [2,3]PA_{1,1} \times [3,3]PB_{1,1} + [3,2]PA_{1,1} \times [2,2]PB_{1,1} + [3,3]PA_{1,1} \times [3,2]PB_{1,1} + [3,2]PA_{1,1} \times [2,3]PB_{1,1} + [3,3]PA_{1,1} \times [3,3]PB_{1,1}$$

圖(八)：以 Divide and Conquer 方式產生的矩陣相乘例子

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix} \times \begin{bmatrix} B_{11} & B_{12} & B_{13} & B_{14} \\ B_{21} & B_{22} & B_{23} & B_{24} \\ B_{31} & B_{32} & B_{33} & B_{34} \\ B_{41} & B_{42} & B_{43} & B_{44} \end{bmatrix} = \\
[0,0]PA_{2,2} \times [0,0]PB_{2,2} + [0,2]PA_{2,2} \times [2,0]PB_{2,2} + \\
[0,0]PA_{2,2} \times [2,0]PB_{2,2} + [0,2]PA_{2,2} \times [2,2]PB_{2,2} + \\
[2,0]PA_{2,2} \times [0,0]PB_{2,2} + [2,2]PA_{2,2} \times [2,0]PB_{2,2} + \\
[2,0]PA_{2,2} \times [0,2]PB_{2,2} + [2,2]PA_{2,2} \times [2,2]PB_{2,2}$$

圖(七)：以 Divide and Conquer 方式產生的矩陣相乘遞迴式

Scattering Procedure

Step 1：判斷是否收到進行 Scattering Phase 的訊息。

Step 2：若收到進行 Scattering Phase 的訊息，判斷 $T_{i,j}$ 是否為 $T_{1,0}$ ，若是則 go to Step 3，否則 go to Step 4。

Step 3：進行快速矩陣相乘的標準演算法使用 $CC_{1,0}$ 通道碼，並依序傳送部分矩陣資料，其 $Dim_k = \frac{Dim_A}{2^{TL_{max}-1}}$ ，go to Step 5。

Step 4：使用 $CC_{8^{i-2}}$ 通道碼準備接收矩陣資料，並判斷 $T_{1,0}$ 傳輸的資料是否需要接收，若需要才接收。

Step 5：結束 Scattering Procedure。

陣資料依然可透過圖(七)的式子分派給子結點，圖(八)為 $TL=3$ ， $Dim_A=4$ ，並使用圖(七)遞迴式所產生的例子，因此，在 Scattering Phase 開始時，可產生一虛擬編號、實際 ID 及包含的矩陣資料的表格， $T_{1,0}$ 裝置可以 $CC_{1,0}$ 通道碼傳送部分矩陣資料 k ，部分矩陣 k 的

$$Dim_k = \frac{Dim_A}{2^{TL_{max}-1}}$$

傳送部分矩陣 k 所需花費的時間為 $(Dim_k^2 \times 2 \times EL) \div (BW_{max} \times 1024 \times 1024)$ ，共需傳送 $8^{TL_{max}-1}$ 個部分矩陣，傳送需花費的總時間 $(Dim_A^2 \times 2 \times EL) \div (BW_{max} \times 1024 \times 1024)$ 。以 $Dim_A=2048$ 、 $BW_{max}=10M$ 、 $EL=2$ 為例，所花費的總時間為 $(2048^2 \times 2 \times 2) \div (10 \times 1024 \times 1024) = 1.6$ 秒。與 OVSF-CDMA 平行矩陣相乘演算法所花費的時間 $1.6+3.2=4.8$ 秒相比，減少了 3.2 秒。

V. 實驗與結果

我們將欲參與平行計算工作的裝置，建構成一階層式的樹狀平行計算架構，以矩陣相乘問題為例，評估我們提出的 WPMM 演算法的效能。本論文主要是在 OVSF-CDMA 的無線通訊環境下，On-Demand 建立階層式的樹狀平行計算架構，並觀察使用我們提出的 Divide and Conquer 矩陣相乘演算法後，經 WPMM 通訊協定所帶來的績效。在 OVSF-CDMA 的環境中，所有裝置皆使用 $C_{1,0}$ 的通道碼作為溝通頻道，來建構階層式的樹狀平行計算架構。在我們的實驗中，皆使用快速矩陣相乘演算法，來進行單一電腦矩陣相乘問題的計算，以 OVSF 無線通訊協定(OVSF)及加強的 OVSF 無線通訊協定(E-OVSF)等通訊協定，對平行計算架構的效能進行分析。

5.1 模擬環境

我們模擬環境所需要的參數如下：

1. 矩陣維度：2048 x 2048
2. 矩陣元素資料所需的儲存空間：2 bytes
3. 行動裝置個數：64
4. 行動裝置的計算能力：100 MIPS
5. 無線通訊的頻寬：79 MB

根據以上的模擬環境，我們針對不同的目標來評估 WPMM 通訊協定所帶來的績效。本論文所提出的協定除了帶給平行計算裝置移動能力，提升

平行計算架構在傳輸時的通訊頻寬利用率，並解決多對行動裝置平行傳輸的碰撞問題，我們就以下的事項來說明 WPMM 所帶來的績效：

實驗(一) 行動裝置個數與平行計算過程的總花費時間

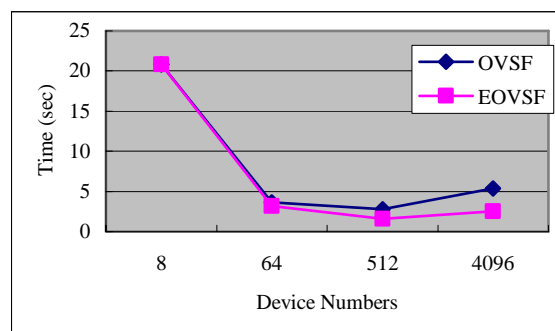
OVSF-CDMA 可根據使用的通道碼來動態決定其通訊頻寬，由於無線平行計算環境主要的目標是針對計算能力較差的行動裝置，因此我們將行動裝置的計算能力定為 100 MIPS，藉此觀察在行動裝置計算能力不高的情況下，裝置個數的增加對於平行計算過程的總花費時間所產生的影響。理論上，當行動裝置計算能力不改變，行動裝置的個數增加，將使得平行計算過程的總花費時間大幅減少，然而由圖(九)的模擬結果觀察，OVSF 及 E-OVSF 在裝置個數由 512 變成 4096 時，總花費時間不降反升，但是圖(十)的模擬結果顯示，行動裝置個數增加平行計算過程的總花費時間將逐漸減少，會發生這種情況，主要是因為行動裝置個數增多時，平行計算所建構的階層式樹狀架構的樹高增高，樹葉部分的裝置所需進行計算的矩陣相乘問題的矩陣維度縮小，而主要進行矩陣計算工作的行動裝置為樹狀架構的樹葉部分的所有裝置，所以平行計算過程中的計算時間將減少，另一方面，由於樹高增加，通訊所需的代碼中繼點增多，傳輸所需的時間相對增加，總結來說，當通訊頻寬較高時，傳輸所增加的時間較計算所增加的時間少，平行計算過程的總花費時間也會減少；當通訊頻寬較低時，傳輸所增加的時間較計算所增加的時間多，平行計算過程的總花費時間將會增多。綜合以上觀察，我們得到一個結論，當行動裝置個數增多的同時，傳輸頻寬也必須相對提高，才能得到較佳的平行計算的效能。

實驗(二) 無線通訊頻寬與平行計算過程的總花費時間

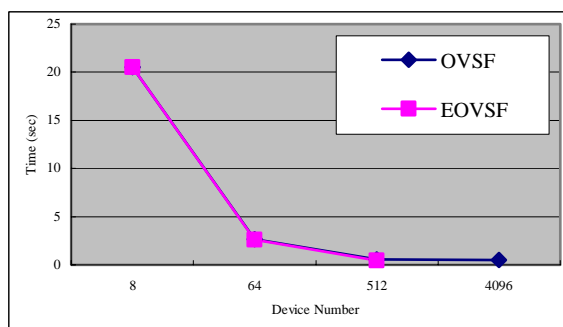
由圖(十一)可觀察出，當無線通訊頻寬增加時，平行計算過程中的計算時間並不會增加，相對的傳輸頻寬的增大，傳輸所花費的時間將減少，因此，平行計算過程的總花費時間將會降低，由於 OVSF 及 E-OVSF 可動態的調整通訊頻寬，充分使用通訊頻寬，使其平行計算過程的傳輸時間與計算時間的比率很低，當無線通訊頻寬增加，其平行計算過程的資料傳輸時間皆將降低。

實驗(三) 矩陣維度與平行計算過程的總花費時間

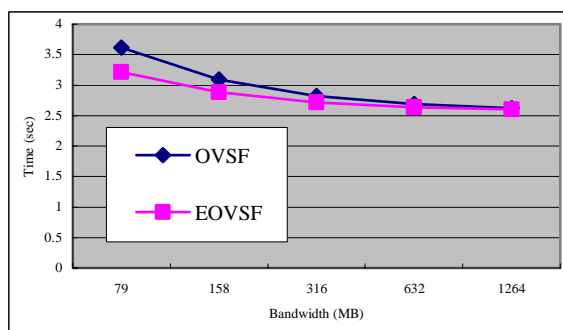
事實上，當矩陣維度增加時，傳輸時間相對也會增加，由圖(十二)的模擬結果來看，矩陣維度增加時，其平行計算過程的總花費時間將增加，這是因為矩陣維度增加時，其平行計算過程的計算時間與資料傳輸時間都將增加，增加的幅度約為 2^3 倍，不論是 OVSF 或 E-OVSF 皆是如此，因此，我們可以觀察到在平行計算的過程中，當矩陣的維度增加，平行計算過程的總花費時間將增加約 2^3 倍。



圖(九)：在通訊頻寬為 79MB 時，行動裝置個數對於平行計算過程的總花費時間所造成的影響



圖(十)：在通訊頻寬為 790MB 時，行動裝置個數對於平行計算過程的總花費時間所造成的影響



圖(十一) 無線通訊頻寬對於平行計算過程的總花費時間所造成的影響

實驗(四) 行動裝置的計算能力與平行計算過程的總花費時間

當只有行動裝置的計算能力增加時，傳輸工作所花費的時間並不會改變，計算工作所花費的時間卻相對減少，承圖(十三)的模擬結果，單一電腦當行動裝置計算能力以 k 倍增加時，其平行計算過程的總花費時間將以 $1/k$ 倍減少，對於 OVSF 或 E-OVSF 的平行計算架構而言，行動計算裝置的計算能力對於平行計算過程的總花費時間的影響並不明顯，由此我們可以觀察出在平行計算的過程中，行動裝置的計算能力增加，對於平行計算過程的總花費時間影響並不大，因為平行計算過程中，傳輸時間的花費比重遠較行動計算裝置的計算能力高，對於我們提出的平行計算協定來說，行動裝置的計算能力增加越大，計算所花費的時間將少量減少。

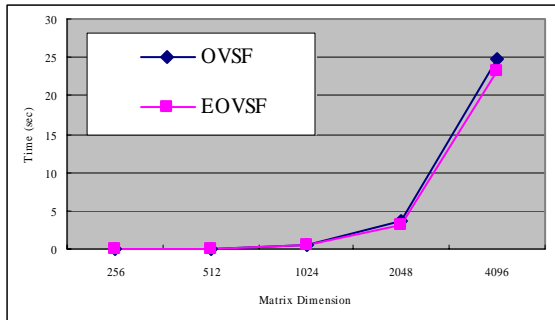


圖 (十二)：矩陣維度對於平行計算過程的總花費時間所造成的影響

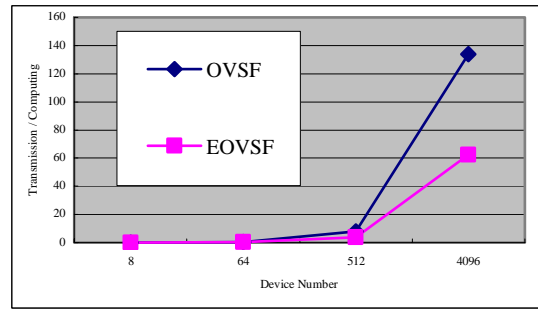


圖 (十四)：行動裝置個數對於平行計算過程的傳輸時間與計算時間的比率所造成的影響

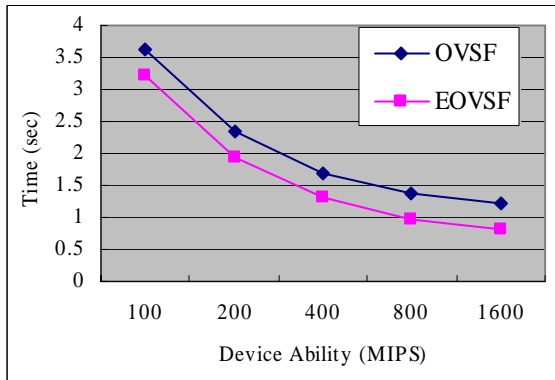


圖 (十三)：行動裝置的計算能力對於平行計算過程的總花費時間所造成的影響

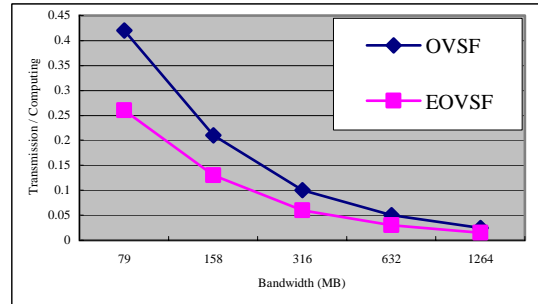


圖 (十五)：無線通訊頻寬對於平行計算過程的傳輸時間與計算時間的比率所造成的影響

實驗(五) 行動裝置個數與平行計算過程的傳輸時間與計算時間的比率

在行動裝置計算能力不高的情況下，裝置個數的增加對於平行計算過程的傳輸時間與計算時間的比率將產生影響。由圖 (十四)顯示，當行動裝置計算能力不改變，行動裝置的個數增加，將使得平行計算過程中的傳輸時間與計算時間的比率大幅增加，會發生這種情況，主要是因為行動裝置個數增多時，平行計算所建構的階層式樹狀架構的樹高增高，使得樹葉部分的裝置所需進行計算的矩陣相乘問題的矩陣維度縮小，而主要進行矩陣計算工作的行動裝置為樹狀架構的樹葉部分的所有裝置，所以平行計算過程中的計算時間將減少，另一方面，由於樹高增加，通訊所需的代傳中繼點增多，傳輸所需的時間相對增加。

實驗(六) 無線通訊頻寬與平行計算過程的傳輸時間與計算時間的比率

由圖 (十五)可觀察出，當無線通訊頻寬增加時，平行計算過程中的計算時間並不會增加，相對的傳輸頻寬的增大，傳輸所花費的時間將減少，因此，平行計算過程的傳輸時間與計算時間的比率將會降低，由於 OVSF 及 E-OVSF 可動態的調整通訊頻寬，充分使用通訊頻寬，使其平行計算過程的傳輸時間與計算時間的比率很低。當無線通訊頻寬增加，其平行計算過程的資料傳輸時間皆將降低。

實驗(七) 矩陣維度與平行計算過程的傳輸時間與計算時間的比率

事實上，當矩陣維度增加時，傳輸時間相對也會增加，由圖 (十六)的模擬結果來看，矩陣維度增加時，其平行計算過程的傳輸時間與計算時間的比率卻降低，這表示矩陣維度增加時，其平行計算過程的計算時間也增加，且增加的幅度約為傳輸時間的兩倍，不論是 OVSF 或 E-OVSF 皆是如此，因此，我們可以觀察到在平行計算的過程中，當矩陣的維度增加，計算所增加的時間花費約為傳輸所增加的時間的兩倍，對於我們提出的平行計算協定來說，矩陣維度越大，傳輸所花費的時間的比重將相對降低。

實驗(八) 行動裝置的計算能力與平行計算過程的傳輸時間與計算時間的比率

當只有行動裝置的計算能力增加時，傳輸工作所花費的時間並不會改變，計算工作所花費的時間卻相對減少，承圖 (十七)的模擬結果，行動裝置的計算能力增加時，其平行計算過程的傳輸時間與計算時間的比率卻增加為兩倍左右，這表示行動裝置的計算能力增加時，其平行計算過程的計算時間將減少，而傳輸工作所花的時間並未增減。不論是 OVSF 或 E-OVSF 皆是如此，因此，我們可以觀察到在平行計算的過程中，當行動裝置的計算能力增加，並不影響其資料傳輸工作，對於我們提出的平行計算協定來說，行動裝置的計算能力增加越大，計算所花費的時間的比重將相對降低。

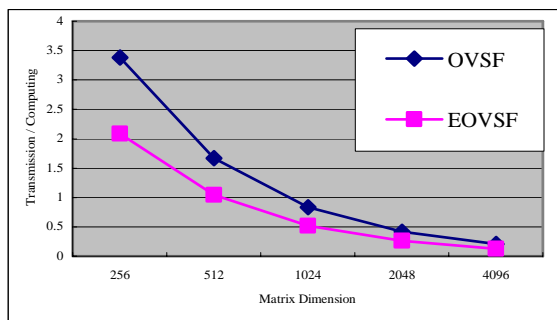


圖 (十六): 矩陣維度對於平行計算過程的傳輸時間與計算時間的比率所造成的影響

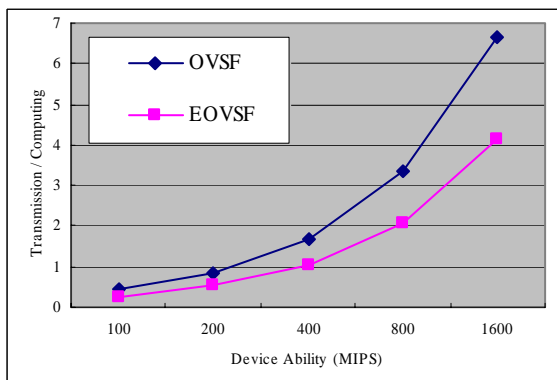


圖 (十七): 行動裝置的計算能力對於平行計算過程的傳輸時間與計算時間的比率所造成的影響

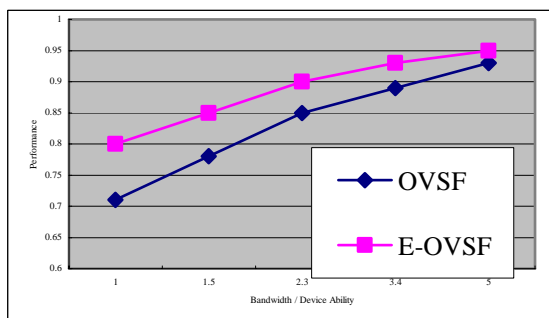


圖 (十八): 平行計算效能與無線通訊頻寬對行動裝置計算能力的比值所造成的影響

實驗(九) 平行計算效能與無線通訊頻寬對行動裝置計算能力的比值

隨著電子與通訊技術的不斷提升，行動裝置的計算能力及無線通訊頻寬也不斷提升，著眼於未來平行計算的環境，無線通訊頻寬對行動裝置計算能力的比值對於我們提出的 WPMM 協定的影響是非常重要的，因此，我們將無線通訊頻寬對行動裝置計算能力的比值作為參數，觀察當行動裝置的計算能力以 100MIPS、150MIPS... 的速度提升，此時無線通訊頻寬對行動裝置計算能力的比值若以 1.5 的倍數提升，此時的 WPMM 效能的變化為何，對於 WPMM 的效能我們以下列函式進行評估：

$$Performance = \frac{Time_{para} \times Dn}{Time_{singal}}$$

承圖 (十八) 的結果，我們可以發現當無線通訊的頻寬不斷增加，而行動裝置的計算能力提升有限的情況下，無線通訊頻寬對行動裝置計算能力的比值越高，則我們提出的 WPMM 的平行計算效能提升非常顯著，WPMM 在未來的無線通訊技術的不斷提升下，將可提供無線平行計算一個有效率的無線平行計算環境。

VI. 結論與未來工作

本論文對於 OVSF-CDMA 的環境，提出無線平行計算協定 WPMM。在 OVSF-CDMA 的環境中，我們提出的協定可分兩部分，一是 OWPMM 協定，透過 OVSF Code Tree 通道碼的動態頻寬分配與排程，最佳化網路頻寬利用率，另一是 EWPMM 協定，透過 Scattering Phase 直接將矩陣資料散佈給其他行動裝置，有效的縮短 Scattering Phase 的傳輸時間。實驗數據結果顯示，OWPMM 協定與 EWPMM 協定，在縮短傳輸時間、提高頻寬利用率，皆具有極高的效能表現。未來我們的研究方向將進行多平行任務的平行計算架構之研究與 multi hop 無線平行計算環境之研究。

VII. Acknowledgement

作者特別感謝國科會以計畫編號 NSC 94-2213-E-156-001- 及 NCS 94-2524-S-156- 001- 提供對本論文研究部份財務上之支持。

Reference :

- [1]. C. T. Chang, C. Y. Chang, and J. P. Sheu, "BlueCube: Constructing a Hypercube Parallel Computing and Communication Environment over Bluetooth Radio System", International Conference on Parallel Processing, Kaohsiung, Taiwan, October 06 - 09, 2003.
- [2]. C. F. Wang, S. Sahni, "Matrix multiplication on the OTIS-Mesh optoelectronic computer", IEEE journal on Computers, Volume: 50, pp. 635 - 646, July 2001.
- [3]. L. C. Wu, S. T. Huang, "A new parallel computation model on synchronous wireless interconnection", IEEE Region 10 Conference on Computer, Communication, Control and Power Engineering (TENCON), Issue: 0, 19-21 Oct. 1993.