

行政院國家科學委員會補助專題研究計畫成果報告

均勻相依演算法映射至格狀互連處理機陣列之通道相衝的檢查 Checking Link Conflicts in the Mapping of Uniform Dependence Algorithms into Grid-Connected Systolic Arrays

計畫類別：個別型計畫 整合型計畫

計畫編號：NSC 90-2213-E-009-134-

執行期間：90年8月1日至91年7月31日

計畫主持人：蔡中川

共同主持人：

本成果報告包括以下應繳交之附件：

- 赴國外出差或研習心得報告一份
- 赴大陸地區出差或研習心得報告一份
- 出席國際學術會議心得報告及發表之論文各一份
- 國際合作研究計畫國外研究報告書一份

執行單位：國立交通大學資訊工程系

中華民國91年10月 日

行政院國家科學委員會專題研究計畫成果報告

均勻相依演算法映射至格狀互連處理機陣列之通道相衝的檢查

Checking Link Conflicts in the Mapping of Uniform Dependence Algorithms into Grid-Connected Systolic Arrays

計畫編號：NSC 90-2213-E-009-134-

執行期限：90 年 8 月 1 日至 91 年 7 月 31 日

主持人：蔡中川 交通大學資訊工程系

計畫參與人員：黃為霖、張博揚、邱威傑 交通大學資訊工程系

一、中文摘要

通道相衝之檢查為設計格狀互連處理機陣列上一項無法避免的必要程序。過去已有許多學者提出通道相衝檢查之方法，但由於假設的陣列模式之差異，這些方法大部份皆不可應用於檢查格狀互連處理機陣列上的通道相衝。在本計劃中，我們深入研究格狀互連處理機陣列上的通道相衝問題，並提出了適用的通道相衝檢查方法。我們的方法較過去方法更有效率且適用於更一般之處理機陣列模式。我們也進一步研究了一些特殊情況，並在這些情況下改進了我們的方法之效率。此外，我們詳細探討了我們的方法與過去方法之間的關係，並證明過去方法為我們的方法之特例。

關鍵詞：格狀互連處理機陣列、通道相衝、空時映射、均勻相依演算法

Abstract

Checking link conflicts is a necessary process for the design of grid-connected systolic arrays. Although many methods for checking link conflicts have been designed in the literature, they are not suitable for grid-connected systolic arrays due to the difference of the underlying processor array models. In this project, we study the problem of checking link conflicts on grid-connected systolic arrays and propose a checking method for the problem. Our checking method is more efficient and general than the

previous method. Also, for some special cases, the performance of our method is further improved. In addition, our method is compared with the previous method. We show that the previous method is a special case of our method.

Keywords: grid-connected systolic array, link conflict, space-time mapping, uniform dependence algorithm

二、緣由與目的

韻律陣列(systolic array)為一類特殊的平行機器，其由許多排列規律的處理機(processor)互相鄰近連接而成[6]。由於其在資料通道連接上的局部性與規律性，韻律陣列已被大量用於許多演算法的 VLSI 實作上 [7]。格狀互連處理機陣列(grid-connected systolic array)，簡稱格狀陣列，為一類只允許處理機以正交資料通道互連之特殊韻律陣列。較一般韻律陣列而言，格狀陣列在資料通道連接上更為簡單且規則，因此更適於使用 VLSI 實作[8]。

可在韻律陣列上執行之演算法，通常可以均勻相依演算法(uniform dependence algorithm)來表示；此類演算法由凸形計算定義域(index set)與有限數量之常數相依向量(dependence vector)所組成[15,14]，其在科學計算與信號處理領域中相當常見，如矩陣相乘、LU 分解等問題之演算法皆屬此類[7]。

在韻律陣列的設計上，一種常用的方法是空時映射(space-time mapping)法，其觀

念為利用一空時映射矩陣將一 n 維均勻相依演算法直接映射成一 k 維韻律陣列 [7,11,12]。在映射的過程中，為了確保陣列的正確性，我們必須檢查是否發生通道相衝(link conflict)；所謂通道相衝意指有兩不同資料元素(token)同時進出同一資料通道。過去已有許多關於通道相衝檢查方法之研究，如文獻[1,2,5,9,10,16,17]中已提出各種的通道相衝檢查方法。然而，過去這些方法皆限制了陣列中一資料流(data stream)只能往單一方向流動，因而它們皆無法應用於檢查格狀陣列上的通道相衝，因為格狀陣列允許一資料流可有多個不同的流向。

格狀陣列上的通道相衝問題最早是由 Lee[8]所提出。在[8]中，Lee 推導出格狀陣列發生通道相衝的充要條件，並直接應用此條件於通道相衝之檢查。然而此條件必須對計算定義域內 $O(N^{2n})$ 對計算點(index point)檢查一資料流內最多 k 個不同流向上的通道相衝，其中 N 為計算定義域之最大寬度，故此法具有相當高之時間複雜度。此外，Lee 假設了格狀陣列中之資料通道皆只能連接鄰近處理機，因而其方法無法應用於允許非鄰近處理機互連之格狀陣列上。

有關格狀陣列上的通道相衝問題之研究甚少，據我們所知，上述結果為目前文獻中唯一發表之結果，因此我們認為此問題仍有相當大的研究空間。在本計劃中，我們提出一更有效率的通道相衝檢查方法來解決格狀陣列上的通道相衝問題。我們的方法不但能適用於 Lee 所假設的格狀陣列，也適用於允許非鄰近處理機互連之格狀陣列。另外，針對一些特殊的情況，我們提出了效率更佳的檢查方法。此外，我們亦探討了我們的方法與 Lee 的方法之間的關係。

三、結果與討論

本節簡要描述本計劃之研究成果，至於更詳細之內容，我們將另外撰寫論文發表。

3.1 格狀陣列模式

我們將[8]中所述之格狀陣列模式加以擴充延伸，使其成為一允許非鄰近處理機互連之新陣列模式。新陣列模式定義處理機定義域(processor space)為包含 $\{S\vec{j} | \vec{j} \in \mathcal{J}\}$ 之容納盒(bounding box)，其中 S 為空間映射矩陣， \mathcal{J} 為均勻相依演算法之計算定義域。我們以 $\vec{l}_1, \vec{l}_2, \dots, \vec{l}_k$ 表示第 i 資料流之資料元素在格狀陣列中會依序經過的正交資料通道，我們允許 $\gcd(\vec{l}_{ij}) \geq 1, 1 \leq j \leq k$ ，以表示非鄰近處理機的互連。

在空時映射下，每一由處理機 \vec{p} 計算出之資料元素須傳到處理機 $\vec{p} + S\vec{d}_i$ ，其中 \vec{d}_i 為對應第 i 資料流之相依向量，且 $S\vec{d}_i$ 與 $\vec{l}_1, \vec{l}_2, \dots, \vec{l}_k$ 需滿足

$$S\vec{d}_i = (\vec{l}_1, \vec{l}_2, \dots, \vec{l}_k) \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_k \end{pmatrix}$$

之關係，其中 u_1, u_2, \dots, u_k 為非負整數。上述資料元素在陣列中之流動方式為：先由處理機 \vec{p} 經由資料通道 \vec{l}_1 依序通過處理機 $\vec{p} + \vec{l}_1, \vec{p} + 2\vec{l}_1, \dots, \vec{p} + u_1\vec{l}_1$ ，再經由資料通道 \vec{l}_2 依序通過處理機 $\vec{p} + u_1\vec{l}_1 + \vec{l}_2, \vec{p} + u_1\vec{l}_1 + 2\vec{l}_2, \dots, \vec{p} + u_1\vec{l}_1 + u_2\vec{l}_2$ ，以此類推，最後經由資料通道 \vec{l}_k 流至處理機 $\vec{p} + \sum_{j=1}^k u_j \vec{l}_j (= \vec{p} + S\vec{d}_i)$ 。上述非負整數 u_1, u_2, \dots, u_k 可視為[2]中所提出之連結數(linking number)的一個推廣。

3.2 通道相衝發生之充要條件

我們在[2]中提出了虛擬計算點(virtual node)之觀念來解決非格狀陣列的通道相衝問題。在本計劃中，我們另外提出相依向量分解(dependence decomposition)之觀念將虛擬計算點進一步加以延伸，使其可應用於格狀陣列中。

定義相依向量 \vec{d}_i 之一相依向量分解為一組整數向量 $\vec{d}_{i1}, \vec{d}_{i2}, \dots, \vec{d}_{ik}$ ，滿足 $\vec{d}_i = \vec{d}_{i1} + \vec{d}_{i2} + \dots + \vec{d}_{ik}$ 及

$$\begin{pmatrix} H \\ S \end{pmatrix} (\vec{d}_{i1}, \vec{d}_{i2}, \dots, \vec{d}_{ik}) = \begin{pmatrix} u_{i1}r_{i1}, u_{i2}r_{i2}, \dots, u_{ik}r_{ik} \\ u_{i1}\bar{l}_{i1}, u_{i2}\bar{l}_{i2}, \dots, u_{ik}\bar{l}_{ik} \end{pmatrix},$$

其中 H 為時間映射向量， r_{ij} 為 \bar{l}_{ij} 上之暫存器數加 1。使用相依向量分解，對應第 i 資料流之虛擬計算點可定義為下列形式之分數點：

$$vn_i(\vec{j}, z, c, \beta) = \vec{j} + z\vec{d}_i + \sum_{j=1}^{c-1} \vec{d}_{ij} + \frac{\beta}{u_{ic}} \vec{d}_{ic},$$

其中 $\vec{j} \in J$ ， $z, c, \beta \in \mathbb{Z}$ ， $1 \leq c \leq k$ ， $0 \leq \beta < u_{ic}$ 。經由仔細之研究，我們證明了上述虛擬計算點與格狀陣列上的通道相衝有下列關係：

定理 1 空時映射 H 與 S 引起格狀陣列上之通道相衝，若且唯若存在兩有效(valid)虛擬計算點 $\vec{r} = vn_i(\vec{j}_1, z_1, c, \beta_1)$ 與 $\vec{s} = vn_i(\vec{j}_2, z_2, c, \beta_2)$ 滿足 $\vec{j}_1 \neq \vec{j}_2 + a\vec{d}_i, \forall a \in \mathbb{Z}$ ，且使得 $H\vec{r} = H\vec{s}$ 與 $S\vec{r} = S\vec{s}$ 。

當所有資料元素限制只能由處理機定義域邊界進出格狀陣列時，定理 1 可進一步改寫如下：

定理 2 令向量 \vec{L} 與 \vec{U} 表示處理機定義域在各座標軸上之座標下界(lower bound)與座標上界(upper bound)。則 H 與 S 引起通道相衝，若且唯若存在兩虛擬計算點 $\vec{r} = vn_i(\vec{j}_1, z_1, c, \beta_1)$ 與 $\vec{s} = vn_i(\vec{j}_2, z_2, c, \beta_2)$ 使得

1. $\vec{j}_1 \neq \vec{j}_2 + a\vec{d}_i, \forall a \in \mathbb{Z}$,
2. $H\vec{r} = H\vec{s}$ 且 $S\vec{r} = S\vec{s}$,
3. $\vec{L} \leq S\vec{r} \leq \vec{U}$ 。

3.3 通道相衝之檢查

根據上述結果，我們從三方面發展不同的通道相衝檢查方法，且每一方法皆較 Lee 的方法更有效率。

首先，相似於[2]之推導，我們利用 Hermite normal form[4,13]與定理 2 證明了檢查格狀陣列上之通道相衝等價於檢查 k 個 $n > k + 1$ 維且寬度為 $O(2M)$ 之凸形多面體(convex polytope)內的 $O((2M)^{n-k+1})$ 個整數點。再進一步使用[2]中之列舉方法，則我們最多只要列舉共 $O(k(2M)^{n-k})$ 個整數點，便可檢查出格狀陣列上的通道相衝。

接著，除上述列舉法外，我們也證明了通道相衝之檢查亦等價於求解 kn 個具有相同變數與限制式數目的整數規劃問題(integer linear programming problem)。令 s 為表示每一整數規劃問題所需位元數，則使用 Kannan[3]提出之整數規劃方法，我們可在 $O(kns(n-k+1)^{9(n-k+1)/2})$ 運算時間內檢查出是否發生通道相衝。當 N 甚大於 n 時，此整數規劃方法較上述列舉法有更佳之效率。

最後，我們更進一步改進了 $k = n > 1$ 與 $k = n > 2$ 之特殊情況(special cases)下的通道相衝檢查方法。我們證明了當 $k = n > 1$ 與 $k = n > 2$ 時，通道相衝之檢查分別等價於求解 $\sum_{j=1}^{n-1} (u_{ij} - 1)$ 與 $\sum_{j=1}^{n-2} u_{ij}$ 個單變數整數規劃問題。由於單變數整數規劃問題可於線性時間(linear time)內求解，故對此兩特殊情況，我們可在更短的時間內檢查出是否發生通道相衝。

3.4 與 Lee 的方法之關係

除了上述之研究成果外，我們也研究了我們的方法與 Lee 的方法之間的關係，進而證明 Lee 於[8]中提出的通道相衝檢查條件皆為我們的通道相衝檢查條件之特例。

四、計劃成果自評

我們已達到本計劃之預期目標。我們已提出了適用於格狀陣列且更有效率之通道相衝檢查方法，也證明了新方法之正確性與複雜度，並針對一些特殊情況進一步改進了新方法之執行效率。我們也比較了

我們的方法與過去方法之間的關係，並證明了過去方法為我們的方法之特例。

我們相信本計劃之研究結果將有助於各種不同陣列模式上之通道相衝問題的瞭解及高效率之通道相衝檢查方法的設計與實現。

五、參考文獻

- [1] K. N. Ganapathy and B. W. Wah. Optimal synthesis of algorithm-specific lower-dimensional processor arrays. *IEEE Trans. Parallel and Distri. Sys.*, 7(3):274–287, 1996.
- [2] W. L. Huang. *A virtual node approach to checking link conflicts in the mapping of dependence graphs into processor arrays*. Master thesis, National Chiao Tung University, Taiwan, R.O.C., 2001.
- [3] R. Kannan. Minkowski's Convex Body Theorem and Integer Programming. *Mathematics of Operation Research*, 12(3):pp. 415–440, 1987.
- [4] R. Kannan and A. Bachem. Polynomial algorithms for computing the smith and hermite normal forms of an integer matrix. *SIAM Journal on Computing*, 8(4):499–507, 1979.
- [5] J. Y. Ke and J. C. Tsay. An approach to checking link conflicts in the mapping of uniform dependence algorithms into lower dimensional processor arrays. *IEEE Trans. Comput.*, 48(7):732–737, July 1999.
- [6] H. T. Kung. Why systolic architectures? *Computer*, 15(1):37–46, 1982.
- [7] S. Y. Kung. *VLSI Array Processor*. Prentice-Hall Int., Englewood Cliffs, NJ, 1988.
- [8] P. Z. Lee. Mapping nested loop algorithms into grid-connected systolic arrays without data collisions in the data links. *ISCA International Journal of Computers and Their Applications*, 2(2):72–85, 1995.
- [9] P. Z. Lee and Z. M. Kedem. Synthesizing linear array algorithms from nested for loop algorithms. *IEEE Trans. Comput.*, C-37(12):1578–1598, December 1988.
- [10] P. Z. Lee and Z. M. Kedem. Mapping nested loop algorithms into multidimensional systolic arrays. *IEEE Trans. Parallel and Distri. Sys.*, 1(1):64–76, January 1990.
- [11] D. I. Moldovan. On the design of algorithms for VLSI arrays. *Proc. of the IEEE*, 71(1):113–120, January 1983.
- [12] S. K. Rao. *Regular iterative algorithms and their implementations on processor arrays*. Ph.D. dissertation, Stanford University, 1985.
- [13] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, 1986.
- [14] W. Shang and J. A. B. Fortes. On time mapping of uniform dependence algorithms into lower dimensional processor arrays. *IEEE Trans. Parallel and Distri. Sys.*, 3(3):350–363, May 1992.
- [15] W. Shang and J. A. B. Fortes. Time optimal linear schedules for algorithms with uniform dependencies. *IEEE Trans. Comput.*, 40(6):723–742, Jun 1991.
- [16] J. Xue. A unified approach to checking data link and computational conflicts in the design of algorithm-specific processor arrays. Technical Report 94-100, Dep. Mathematics, Statistics and Computing Science, The University of New England, Australia, 1994.
- [17] J. Xue. Closed-form mapping conditions for the synthesis of linear processor arrays. *Journal of VLSI Signal Processing*, (10):181–199, 1995.