

行政院國家科學委員會專題研究計畫成果報告

在命題邏輯推理方法中去除多餘性的研究與應用

計畫編號：NSC 89-2218-E-002 -037

執行期限：88年8月1日至90年7月31日

主持人：項潔 國立台灣大學資訊工程學研究所

一、中文摘要

本計劃之目的為用命題邏輯 (propositional logic) 的推理方法解決數學中的 open problems。探討的問題包括 lattice point problems、complete mapping problems 和一些 quasi group 相關的問題。

有限數學中的問題理論上來說均可用命題邏輯來表示，但其複雜度通常高到無法控制，而且需要使用窮舉法來作搜尋的工作，所以如何在窮舉法當中消除搜尋中的不必要部分 (redundancy) 是一個相當重要的問題。根據我們的研究發現，在問題轉換中常常因為本身的對稱性而產生 redundancy。針對此一發現，我們發展兩套去除因對稱性而產生的 redundancy 方法。一個是 symmetry-cutting strategies，另一個是 partition strategies。

在實驗結果裡，我們使用這兩種方法已成功證明了幾個 lattice point problems 和 complete mapping problems 中的 open problems，並將之應用於 latin squares 及 n-queen problems 等問題。

關鍵詞：命題邏輯、自動推論、多餘性、對稱性

Abstract

Automated deduction had made tremendous stride in recent years. Significant advances in new methods for equational logic and model checking have lead to major

successes such as the proof of the Tarski Conjecture of Robbins Algebra, verification and correction of hardware and geometry theorem provers that are far better than human being. Another area which has also made significant success but attracted less attention is on using automated deduction to solve open problems. In particular those in finite groups, have been solves using automated deduction methods.

We propose two strategies to solve these open problems. One is symmetry-cutting strategies, and the other is partition strategies. We successful use these strategies to solve lattice point problems, complete mapping problems.

Keywords: propositional logic, automated deduction, symmetry, redundancy, open problems, lattice point problems, complete mapping problems.

二、緣由與目的

在有限數學中有許多問題均超過百年歷史，但是由於沒有好的數學工具而無法解決。隨著電腦速度與能力的迅速進步，這些數學問題的解決也露出了曙光。對於這些數學問題，我們對其中兩大類特別感興趣，一是存在性的問題 (existence problems)，另一個是計數的問題 (enumeration problems)，其中計數問題比存在性問題更為複雜。科學家運用 forward checking, Davis-Putnam procedure, Boolean ring method, OBDD, simulated annealing, hill climbing 等來解決存在性問題。而運用

closed formula , recurrence relation , generating function , approximation 來解決計數性問題。這些技巧已廣泛地運用在各個領域裡，但有更多的有趣且重要的數學問題無法運用上述技術來獲得解決，最終得靠窮舉法(exhaustive enumeration)來解決它的存在性及計數問題，像是 Ramsey number , lattice point problems , complete mapping problems , n-queen problems 等。

我們將研究重心放在窮舉法上來解決這些數學問題，首先，我們需要的是一個表達方式來表達這些數學問題。基本上，所有的有限數學均可以用命題邏輯來表現，但有些時候我們也會用等價的，直覺的 CSP (Constraint Satisfaction Problems) 表達方式來表達一個數學問題。對於一個表達式來說，我們會關心兩個問題，一個是解的存在性問題，另一個是解的數量問題。運用窮舉法來解決數學問題，其最主要的困難在於搜尋空間(search space)太大。所以如何能有效降低搜尋空間是這方法中最重要的課題。我們利用 symmetry-cutting strategies 和 partition strategies 來降低搜尋空間，進而加速計算。

三、結果與討論

一個 constraint satisfaction problem 是探討 $CSP(V,D,C)$ 解的狀況，其中 V 是一群變數集合， D 是變數的值域， C 是一些變數上的限制，解是從 V 對到 D 的一個函數 I ，使得 I 可以滿足 C 的限制。對於一個 CSP 來說，我們一樣得探討該問題到底有沒有解 (existence problem)，該問題到底有多少解 (enumeration problem)。雖然我們可以將很多問題用 CSP 很直覺地描述出來，但是如果透過全盤解決 CSP 問題，進而解決這些問題卻是不大可能的，因為 CSP 的 existence problem 是一個 NP complete 的問題，而 enumeration problem 更是是一個比 NP complete 更難的 #P complete 問題。所以現有的方法都是利用技巧加快計算或是利用該數學

問題本身的一些性質來去除 redundancy，而都沒有 polynomial time 的保證。

解決 CSP 的一個基本方法就是窮舉法，窮舉法的架構通常是用 Assignment Tree Traversal，也就是說每一個 internal node 都代表著一個 partial assignment，而每一個 leaf node 則代表著一個或一群 assignments。所有的 leaf nodes 正代表著所有可能的 assignments。在這個 Assignment Tree Traversal 架構上，各個學者在 traversal node 上應用不同的化約技巧來降低展開的 node 數目。例如 forward checking , Davis-Putnam procedure , Boolean ring method 等。

我們提出的 Symmetry-Cutting strategies 也是屬於這類的技巧，我們首先對該數學問題進行對稱性的了解，挖掘以下的性質： I 是 $CSP(V,D,C)$ 的一個解則 $T(I)$ 也是 $CSP(V,D,C)$ 的一個解，反之亦然。其中我們稱 T 是一個對稱運算子，結合所有我們找到的運算子，我們可以構成一個運算子群，進而在解集上作一個分割，而有一個 equivalent classes。而後，我們在 assignment space 上架一個 total ordering，並將之延伸成為 partial assignment space 上的一個 partial ordering。其次，我們將運算子的定義域從 assignments 延伸到 partial assignments。最後，在每一個 equivalent class 上，我們選最小的 assignment 當成這個 class 的代表，這意思是說當我們在 traversal 的過程中，對於同一個 class 我們只需探索其 class 中最小的一個。在一個 partial assignment p 的 inner node 上其 symmetry-cutting rule 如下：如果存在一個運算子 T ，使得 $T(p)$ 比 p 還小的話則 backtrack。在這個方法我們可以運用於解決存在性問題，但是如果我們要運用它來解決計數問題時，我們需要在發現每一個 equivalent class 的最小解時將其整個 class 的個數算出來並加到總和中。

窮舉法的另一個基本架構則是

Assignment Diagram Maintenance , 這種架構以 OBDD 為代表 (Ordered Binary decision diagram) , 它是將一個 formula 的所有解用一個 OBDD 儲存起來, 在計算的過程中, 從小的 sub-formulas 合併成 target formula , 最終的 OBDD 則代表著該問題的所有解答。如果一個 formula 沒有解, 則它的 OBDD 會是空的。我們發展窮舉法的第三個架構是 Divide and Conquer, 我們稱它是 partition strategies。

在我們針對計數性問題在做研究時, 我們發現一些舊有的技術並不適用, 例如在 Davis-Putnam 的 pure-literal rule 裡, 因為我們要計算所有的解, 所以我們不能直接將 pure-literal 直接設為 true, 我們也得考量 false 的狀況, 如此一來, 就等同於沒有 pure-literal rule 了。symmetry-cutting 是設計來解決高對稱性的存在性問題, 在修改到應用計數問題時, 我們得要花時間去計算同 class 理解的個數, 因為每一個 equivalent class 可能會有不同的 size。

目前對於 enumeration problem, 我們提出 partition strategies 可以有效解決某些的 CSP enumeration problem。首先我將變數集 V 切成兩部分 U 跟 L , 也就會將一個 assignment I 切成兩的 partial assignment I_U 跟 I_L , 然後對該數學問題進行了解及挖掘如下定理: 對於所有的 assignment I , I_U 滿足 P_i , 則 I_L 滿足 Q_i , 反之亦然。 其中 P_i 是 U 上的 constraint 而 Q_i 是 L 上的 constraint。這也多少有點對稱的味道, 運用這樣的定理, 我們發展 partition strategies 來利用這樣的性質。

Partition strategies 是利用 Divide and Conquer 當成它的基礎架構來解決 CSP (V,D,C) 。首先它將變數集 V 分成兩部分 U 和 L , 限制集分成 C, C_U, C_L , 和 C_{UL} 。一個 Localized Partition 是 (P,Q) , 其中 $P=\{P_i\}$ 是 D^U 上的一個分割, $Q=\{Q_i\}$ 是 D^L 上的一個分割, $\{P_i \wedge Q_i\}$ 則是解集合上的一個分割。然後對於每一個 i 我們去

解決 $(U,D,C_U \wedge P_i)$ 和 $(L,D,C_L \wedge Q_i)$ 兩個子問題, 而後合併兩的子問題的解答合成 (V,D,C) 的一個 assignment, 最後檢查合併後的 assignment 是否滿足 C_{UL} 來決定它是否為解。透過如此特殊的 partition 限制, 我們可以減少 merge 的時間, 因為對於一個從 $(U,D,C \wedge P_i)$ 找出來的 partial assignment 我們不需要去合併從 $(L,D,C \wedge Q_i)$ 找出來的 partial assignments。

一個 Free Partition (P,Q) 是一個 Localized Partition, 而且 $C_U \wedge C_L$ imply C_{UL} 。則對於 $(U,D,C_U \wedge P_i)$ 和 $(L,D,C_L \wedge Q_i)$ 兩個子問題的解答合併後即為 (V,D,C) 的一個解答, 故我們不需檢查合併後的 assignment 是否滿足 C_{UL} , 而只要將兩個子問題的解答數乘起來加到 (V,D,C) 的解個數中。透過 free partition 限制, 我們可以更加速 merge 的時間, 但是並不是每個數學問題都能找到如此佳的定理。

Partition Method 可以遞迴呼叫, 也可以平行計算, 還可應用一些基本技巧來加速計算如 forward checking, dynamic programming, clause verification 等。而且因為有些數學問題在分割後, 分割區的大小都差不多, 則我們也可以在程式未跑完之前估計其所需的時間。

四、計劃成果自評

我們使用 Symmetry-Cutting strategies 成功地計算格子點問題 $n(5,2)=17$, $n(7,2)=25$, $n(5,3)=37$, 而運用 partition strategies 已成功計算出 Z_{17}, Z_{19}, Z_{21} 和 Z_{23} 的 complete mapping 數目及所有個數小於 24 的 group 的 complete mappings 及 strong complete mappings 的個數。但更重要的是我們提出了兩個重要的方法可以運用在其他問題之上。兩個方法都呈現著如下的觀念: 如果我們能夠對一個數學問題越加了解, 也就是挖掘越多有關該問題的定理, 我們也就越能解決它, 而 symmetry-cutting strategies 和 partition strategies 正是代表著兩類定理的具體應用

方式。另外一個值得談的是傳統上不管是解決存在性問題或是計數問題，大部分的學者都是運用 Assignment Tree Traversal 的方法，很少有人運用 Divide and Conquer 來解決計數問題 (enumeration problem)，而我們提出的 partition strategies 正是開啟運用 Divide and Conquer 來解決計數問題。

五、參考文獻

- [1] S. A. Cook, *The complexity of theorem-proving procedures*, in Conference Record of Third Annual ACM Symposium on Theory of Computing, Shaker Heights, Ohio, 3-5 May 1971, pp.151-158.
- [2] M. Davis and H. Putnam, *A computing procedure for quantification theory*, J. Assoc. Comput. Mach., 7(1960), pp.201-215.
- [3] P. Erdős, P. M. Gruber, and J. Hammer, *Lattice points*, Longman Scientific & Technical, England, 1989.
- [4] G. S. Huang, *Search reduction techniques and applications to problems in combinatorics*, PhD thesis, National Taiwan University, 1999.
- [5] H. B. Mann, *The construction of orthogonal Latin squares*, *Ann. Math. Statistics*, 13(1942), pp.418-423
- [6] S. Rivin, I. Vardi, and P. Zimmermann, *The n-queens problem*, vol. 101 of *Amer. Math. Monthly*, 1994, pp. 629-639.
- [7] Y. P. Shieh, *Partition strategies for #P-complete problems with application to enumeration combinatorics*, PhD thesis, National Taiwan University, 2001.