樹狀分散式系統之工作分派

Efficient Task Assignment in Distributed Tree-structured System

計畫編號: 85-2213-E-126-001

執行期限：84/08/01 – 85/07/31

主持人:林耀鈴　靜宜大學資訊管理學系　副教授

協同主持人:杜宏毅　靜宜大學資訊管理學系　副教授

一、中文摘要（關鍵詞:分散式計算、並行計算、任務指派、最優任務調度、流水線處理、樹狀結構之計算機系統、演算法分析）

本計畫之目的就在探討如何將一樹狀結構之模組系統裡的每個任務指派到一樹狀結構之多計算機網路系統裡。給定二樹$T_t = (V_t, E_t)$與$T_p = (V_p, E_p)$，分別稱為任務樹(*task tree*)與處理機樹(*processor tree*).任務樹T_t是一個n節點之有根樹。一個任務指派使得其所有組件之最大加權負荷最小化，該任務指派就是所謂的最小任務指派(*minimum assignment*).給定一任務樹與一處理機樹，最佳任務指派問題(*minimum assignment problem*)就是在找出二者之間的最小任務指派。

樹狀任務指派問題在許多階層式任務/多處理機系統裡都經常出現。在樹狀結構之條件下，我們可以在多項式時間內找到最小任務指派。在更特殊的條件下，比如是$h = 1$(星狀結構)我們可以在$O(n \log\log n)$時間內找到最小任務指派；在$d = 1$(線性結構)時，我們可以在線性時間內找到最佳指派。

英文摘要(**Keywords**: Distributed computing, parallel computing, task assignment, optimal schedule, pipelining, tree-structured computer system, analysis of algorithms.)

The purpose of this project is to investigate methods of assigning tasks of a tree-structured, pipelined task system to a tree-structured network computers. Given two trees $T_t = (V_t, E_t)$ and $T_p = (V_p, E_p)$, called *task tree* and *processor tree*, respectively. Task tree T_t is a rooted tree with n vertices. Each node of T_t represents a module associated with an execution time (a positive integer) and each module communicates with its adjacent modules with a communication cost.

The tree tasks assignment problem naturally arises in many hierarchical task/multiprocessor systems. We can find optimal solutions in polynomial time. When $h = 1$, a star network, we can find the optimal solution in $O(n \log\log n)$ time, and we can find the solution in linear time when $d = 1$.

二、計劃的緣由與目的

Parallel or distributed computer system can have a performance-price ratio that is vastly superior to that of uniprocessors. However, it is usually difficult to develop efficient software that can extract this performance from a parallel/distributed system. In particular, the computation must be designed in such a way that it can be divided into modules which can be assigned to the various processors in the multicomputer [11]. One of the major problems is to allocate several tasks over the processors optimally. A number of studies have been reported in the literature. They are basically the graph-theoretical method [2, 14], the mathematical programming approach [8, 9], and the heuristic method [5, 12, 13]. Under certain constrains on the structure of the program and/or the multiprocessor system, this problem can indeed be solved in polynomial time [2, 3].

Several related research in this area is reported in [7, 1, 6]. For example, Megiddo [10], Cole [4], Becker and Perl [1] presented several polynomial time algorithms that partitions a rooted, weighted tree into q subtrees (by deleting $q-1$ edges) such that the weight of the maximum subtree is minimized (the min-max problem) or the weight of the minimum subtree is maximized (the max-min problem.)

However, the previous results on the tree q-partition problem are not practical in real applications since it is neither easy nor practical to reconnect the underlying hardware (network) connection of the multiprocessor system for each given tree-structured task system every time. In practice, we shall assume that the underlying hardware (network) connection is not changed frequently. Instead, for each tree-structured, pipelined task system, we will assign each task to a special processor without changing the the underlying hardware connection [2, 14].

The purpose of this project is to investigate methods of assigning tasks of a tree-structured, pipelined task system to a tree-structured network computers. An assignment minimizing the the weight of the heaviest of the components is a *minimum assignment*. The *minimum assignment problem* is to find, for given task tree and processor tree, a minimum assignment.

三、研究方法及成果

本計畫分成以下三個階段進行:

1. 規劃及整理各種不同的樹狀結構多計算機網路系統下之樹狀工作分派問題。
2. 針對不同種類的計算機網路結構發展有效且快速的演算法(algorithms)。
3. 將上述發展出的演算法製作成可執行的程式，並應用到實際的樹狀結構多計算機網路系統。

樹狀結構之模組系統裡的工作指派問題裡，一般而言，我們並不知道給定的Task tree之幾何結構(Topologic structure)；相對的，此多計算機網路系統之幾何(樹狀)結構通常已有固定硬體架構，不易輕易更改。因此，我們將對不同的多計算機網路系統(樹狀)結構發展不同的樹狀工作分派方法。給定二樹$T_t=(V_t,E_t)$與$T_p=(V_p,E_p)$，分別稱為任務樹(*task tree*)與處理機樹(*processor tree*)。任務樹T_t是一個n節點之有根樹。T_t上的每個節點代表一個模組並伴隨它所需之執行時間。每個模組並與其相鄰模組交互通訊並伴隨其間所需之通訊時間。概念上每個模組同時並行執行，模組之間以流水線處理方式彼此通訊。在本計畫中，我們將根據依T_t之幾何(樹狀)結構發展不同的樹狀工作分派方法。以下我們討論幾種有代表性的多計算機網路系統(樹狀)結構：

- 星狀結構(Star Structure)。
- 線性結構(Linear Structure)。
- 全樹結構(Complete Tree)。

在本計畫第一階段完成時，我們有系統的定義、規劃、整理各種不同的多計算機網路系統(樹狀)結構，以作為第二階段發展演算法的基礎。

本計畫第二階段的重點在於發展有效且快速的演算法。以下我們將依不同的多計算

機網路結構，分別說明本計畫採取的演算方式。

1. 星狀結構：
   對於具有星狀結構的多計算機網路來說，我們採取parametric search的方式來設計演算法。我們由leaves到root，一個subtree一個subtree考慮交由一個單一計算機執行，以達到尋找最佳解(optimal solution)。演算法的速度上需要$O(n \log\log n)$的時間。

2. 線性結構：
   對於具有線性結構的多計算機網路來說，我們以dynamic programming的方式來設計演算法。最後，在演算法的速度上我們找到一個linear-time的演算法。

3. 全樹結構（Complete Tree）：
   對於具有全樹結構的多計算機網路來說，我們綜合上述二種策略，使用parametric search及dynamic programming的方式來處理全樹狀結構的的問題。最後我們得到一個多項式時間的演算法。

最後，本計畫將前述第二階段所發展的演算法，製作成可執行的程式並應用到實際的樹狀結構多計算機網路系統。系統使用C程式語言編寫而成。

四、結論與討論

一般而言，最佳任務調度在計算上很困難的工作。在另一方面，若程式/計算機系統之連結方式為樹狀結構，最優任務調度這問題就可能在多項時間內找到答案。本計畫第二階段的重點即在於發展有效且快速的演算法，以解決不同多計算機網路結構之下的最佳化問題。我們將有系統，而且由簡入繁的發展三組演算法，以因應三種多計算機網路系統（樹狀）結構的需要。

本計畫針對前述的三種多計算機網路系統（樹狀）結構做有系統的定義、規劃、以及整理。規劃及整理各種不同的樹狀結構多計算機網路系統下之樹狀工作分派問題

隨著商業上平行分散式計算機系統之風行，以及便宜，易連結之大量多計算機系統逐漸普及，最優任務調度問題之解決也跟著日益重要。本計畫預計將前述第二階段所發展的演算法，製作成可執行的程式並應用到實際的樹狀結構多計算機網路系統：一群以樹狀結構網路連結起來的工作站（個人電腦）。當這套系統製作完成後，我們將使用這套系統去實作於一些收集來的任務樹之分派，並評估其效能，作為日後改進的依據。相信這套系統的完成，能為逐漸普及的大量多計算機系統舞臺之速度、效能之提昇，提供一份有價值的成果。

五、參考文獻

參考文獻

[1] R.I. Becker and Y. Perl. Shifting algorithms for tree partitioning with general weighting functions. *J. Algorithms*, 4:101–120, 1983.

[2] Shahid H. Bokhari. Partitioning problems in parallel, pipelined, and distributed computing. *IEEE Transactions on Computers*, 37(1):48–57, January 1988.

[3] Chi-Lok Chan and Gilbert Young. Scheduling algorithms for a chin-like task system. In *Proceedings of the 4th International Symposium on Algorithms and Computation – Lecture Notes in Computer Science 762*, pages 496–505. Springer-Verlag, December 1993.

[4] R. Cole. Slowing down sorting networks to obtain faster sorting algorithms. *J. ACM*, 30:852–865, 1983.

[5] K. Efe. Heuristic models of task assignment scheduling in distributed systems. *IEEE Transactions on Computers*, 15:50–56, 1982.

[6] G.N. Frederickson. Optimal algorithms for partitioning trees and locating p-centers in trees. Technical report, Purdue University, 1990.

[7] S. Kundu and J. Misra. A linear tree tree partitioning algorithm. *SIAM J. Computing*, 6:151–154, 1977.

[8] V.M. Lo. Heuristic algorithms for task assignment in distributed systems. *IEEE Transactions on Computers*, 37:1384–1397, 1988.

[9] P.Y. Ma, E.Y.S. Lee, and M. Tsuchiya. A task allocation model for distributed computing system. *IEEE Transactions on Computers*, 31:41–47, 1982.

[10] N. Megiddo. Applying parallel computation algorithms in the design of serial algorithms. *J. ACM*, 30:852–865, 1983.

[11] M.G. Norman and P. Thanisch. Models of machines and computation for mapping in multicomputers. *ACM Computing Surveys*, 25(3):263–302, September 1993.

[12] C.C. Shen and W.H. Tsai. A graph matching approach to optimal task assignment in distributed computing system using a minimax criterion. *IEEE Transactions on Computers*, 34:197–203, 1985.

[13] J.B. Sinclair. Efficient computation of optimal assignments for distributed tasks. *J. Parallel and Distribute Computing*, 4:342–362, 1987.

[14] H.S. Stone. Multiprocessor scheduling with the aid of networkflow algorithms. *IEEE Transactions on Software Engineering*, 3:85–93, 1977.