

基于匈牙利算法评估路由算法中网络负载的方法

张方爽, 段新明

(天津工业大学计算机科学与软件学院, 天津 300387)

摘要: 最坏情况的吞吐率是衡量路由算法性能的重要因素之一。负载最重的地方是最坏情况吞吐率的体现, 因此最坏情况的吞吐率在路由算法中很关键。在此基础上本文提出了通过利用匈牙利算法来评估网络负载的方法并且通过实验仿真进行比较。将匈牙利算法和穷举法运用到Oblivious路由中的OITURN、VAL等算法中进行比较。实验结果表明运用该方法与利用传统的穷举法相比, 可以大大减少计算量、降低时间复杂度, 实验结果证明了方法的可行性和有效性。

关键词: 匈牙利算法; 最坏情况吞吐率; Oblivious路由; 穷举法

中图分类号: TP393 **文献标识码:** A

Methodology for Evaluating Network Load in Routing Algorithm Based on Hungarian Algorithm

ZHANG Fangshuang, DUAN Xinming

(School of Computer Science and Software, Tianjin University of Technology, Tianjin 300387, China)

Abstract: The worst-case throughput traffic is one of the important factors that evaluate the performance of the routing algorithm. The heaviest load is the demonstration of the worst-case throughput traffic, so the worst-case throughput traffic is critical in the routing algorithm. Based on this, this paper proposes a method to evaluate network load by using Hungarian algorithm and compares them through simulation experiments. Hungarian algorithm and exhaustive method are applied to OITURN, VAL and other algorithms in Oblivious routing for comparison. The experimental results show that compared with the traditional exhaustive method, the proposed method can greatly reduce the computational complexity and the time complexity. It also proves the feasibility and effectiveness of the method.

Keywords: Hungarian algorithm; worst-case throughput; oblivious routing; exhaustive method

1 引言(Introduction)

路由算法是决定网络性能重要因素之一, 最坏情况的吞吐率是衡量路由算法的重要指标。通过随机选择路由路径网络传输过程中故障节点绕道路由问题, 以及自适应网络选择路由的路径问题, 负载最重的地方是最坏情况吞吐率的体现, 因此最坏情况的吞吐率在路由算法中很关键^[1,2]。本文通过利用匈牙利算法的思想计算网络吞吐率的负载。利用Oblivious路由功能的线性, 找到最坏情况的流量模式被视为二分图的最大权重匹配。使用这种结构, 问题通常在多项式时间内解决, 快速产生确切的最坏情况结果。使用最坏情况来确定特定系统的最坏情况吞吐。William J. Dally等人提出将匈牙利算法运用到Oblivious路由中的DOR和ROMM中评估吞吐率^[3], 在此基础上本文引入扩展到Oblivious路由的OITURN等算法中^[1,4]。

2 相关知识(Related information)

匈牙利算法是由匈牙利数学家Edmonds于1965年提出, 因而得名。匈牙利算法是基于Hall定理中充分性证明的思想, 它是二分图匹配最常见的算法, 该算法的核心就是寻找增广路径, 它是一种用增广路径求二分图最大匹配的算法。用于为任意网络拓扑上的路由功能找到确切的最坏情况模式。

网络负载对性能有很大的影响。一般而言, 对于给定目的节点的分布, 网络负载对网络平均消息延迟的影响比其他设计参数的影响大很多, 因此需要合理的选择这些参数。而且, 吞吐率主要受通信模式也就是目的节点的影响。因此, 计算网络负载时非常重要。

由于网络通道不饱和, 可以利用通道负载的线性相关。线性意味着特定通道上的负载仅仅是每对源节点到目的节点引起的负载的总和。实际上这个可以将最坏情况模式搜索限

制为置换矩阵。然后，通过将所有置换矩阵表示为单个二分图内的匹配，并且用源节点到目的节点通道负载的边进行加权，则最大权重匹配产生特定通道及其相应负载的确定最坏情况的置换排列。最后，在网络中所有通道的集合上重复以找到最差情况下，最大权重匹配的理想吞吐量^[1,3]。

Oblivious路由算法是一种在进行路由决策时，不考虑网络的状态，具有较高的灵活性。评估Oblivious路由算法的最坏情况吞吐率的关键是利用算法通道负载线性的特性。也就是说，只要网络的通道不饱和，特定通道c上的负载就是每对源节点到目的节点在流量模式中的所有负载的总和^[3]：

$$\gamma_c(\pi, \Lambda) = \sum_{i,j} \lambda_{i,j} \gamma_c(\pi)_{i,j}$$

其中，流量矩阵(Λ):任意双随机矩阵，其中入口 $\lambda_{i,j}$ 表示从源i到目的j的通道流量；遗忘路由算法(π):一种路由算法；通道负载($\gamma_c(\pi, \Lambda)$):流量矩阵 Λ 和路由选择函数 π 在每个周期信道c的分组数量。

定理1：对于任何Oblivious的路由算法，置换矩阵总能实现理想的最坏情况吞吐量。在论文^[3]中已得到了证明。

在穷举法中找到最坏情况的负载，就要考虑所有的置换排列。时间复杂度为 $O(N!)$ ，是一个NP难问题，计算量很大。为此文本利用匈牙利算法计算最坏情况的吞吐量。大大降低了时间复杂度，提高算法的效率^[2]。

3 算法模式(Algorithm mode)

3.1 穷举法算法模式

- (1)初始化数组；
- (2)利用递归找出所有情况的排列；
- (3)找到最后的结果。

3.2 匈牙利算法的基本模式

由于任何特定的置换，使用Oblivious路由算法的线性特性，二分图可用于表示单个通道上的负载。二分图匹配的最大流算法的核心算法是找增广路径(augment path)其基本模式如下：

- (1)初始时最大匹配为空；
- (2)while找得到增广路径。
do把增广路径加入到最大匹配中。

引理1：如果从一个点A出发，没有找到增广路径，那么无论再从别的点出发找到多少增广路径来改变现在的匹配，从A出发都永远找不到增广路径。

匈牙利算法和最大流算法很相似。不同之处在于增广路径就有它一定的特殊性，虽然根本上是最大流算法，但是它不需要建构网络模型。二分图最大流的核心算法中找增广路径的基本模式及相关定理，引出匈牙利算法基本模式，其基本模式如下：

- (1)初始时最大匹配为空；
- (2)for二分图左半边的每个点i；
- (3)do从点i出发寻找增广路径。如果找到，则把它取反(即增加了总的匹配数)。

如果二分图的左半边一共有n个点，那么最多找n条增

广路径。如果二分图中共有m条边，那么每找一条增广路径(DFS或BFS)时最多把所有边遍历一遍，所花时间也就是m。所以总的复杂度大概是 $O(n*m)$ 。

KM(Kuhn - Munkras)算法用来解决最大权重匹配问题的算法。其基本原理：KM算法是通过给每个顶点一个标号(叫做顶标)来把求最大权重匹配的问题转化为求完备匹配的问题。设顶点Xi的顶标为A[i], 顶点Yj的顶标为B[j], 顶点Xi与Yj之间的边权为w[i, j]。在算法执行过程中的任一时刻,对于任一条边(i, j), $A[i]+B[j] \geq w[i, j]$ 始终成立。

KM算法基本模式：

- (1)初始化可行顶标的值；
- (2)用匈牙利算法寻找完备匹配；
- (3)若未找到完备匹配则修改可行顶标值；
- (4)重复(2)(3)直到找到相等子图完备匹配为止。

4 网络吞吐量评估(Network throughput evaluation)

使用Oblivious路由算法，对于任何特定的置换排列，二分图可用于表示单个通道上的负载。如图1所示，第一组N个节点用于表示数据的源节点，第二组N个节点表示数据的目的节点。在每对源节点和目标节点之间添加边，从而获得总共 N^2 个边^[3]。排列矩阵与二分图的完美匹配之间存在一对一对应的关系。需要注意的是，二分图的结构与底层互连网络的拓扑结构无关。

源节点s到目的节点d的每条边进行加权，使得负载量给一个特定的通道c，即 $\gamma_c(\pi)_{s,d}$ 。这些权重由置换引起的负载量仅为为其相应二分图匹配中边权重的总和^[3-5]如图1所示。对于源到目的对(s,d)，用穷举法列举出路由算法生成的从s到d的所有路径，以及包含通道c的每条路径计算边缘权重，是NP问题，时间复杂度是 $O(N!)$ 。

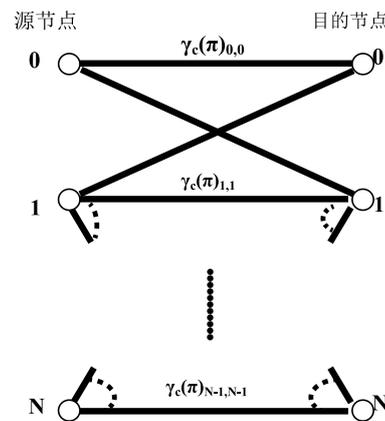


图1 二分图匹配

Fig.1 Bipartite graph matching

根据二分图的构造，可以找到该图的最大权重匹配。从匹配和排列之间的对应关系来看，找到一个最大权重匹配就相当于评估： $\gamma_{c,max}(\pi) = \max_{P \in P} \gamma_c(\pi, P)$ 其中P是所有置换矩阵的集合。通过在所有通道上重复该操作，可以确定理想的最差情况吞吐量为^[3]：

$$\Theta_{ideal,wc}(\pi) = \min_{c \in C} [b_c / \gamma_{c,max}(\pi)]$$

最大权重匹配算法存在。穷举法的时间复杂度 $O(N!)$ ，而

利用匈牙利算法计算的时间复杂度 $O(N^3)$ 。穷举法和匈牙利算法在效率上的差别，可以看出，利用匈牙利算法求吞吐率算法的性能是相当好的。

5 仿真与分析(Simulation and analysis)

在Mesh网络中，对于Oblivious路由算法论文^[3]中对DOR和ROMM算法进行了仿真实验，本文在此基础上扩展了Oblivious中的OITURN，以及VAL^[3,6]算法中。与DOR相比，源节点到目的节点之间的所有流量都集中在单一路径上。ROMM更均匀地将源节点到目的节点流量分布在更多数量的通道上，ROMM选取节点时比较灵活。ROMM和VAL与DOR路由算法不同。ROMM和VAL路由算法是将路由路径分为两个阶段，第一阶段消息从源节点传输到中间节点，在第二阶段消息在由中间节点到达目的地，它们的区别在于选取中间节点方法不同。OITURN路由算法^[1]具有非常结构简单，其随机使用XY或YX路由算法。OITURN和VAL路由算法都具有良好的最坏情况网络的吞吐率。基于Oblivious的几种通讯模式^[1]，进行了仿真实验的比较，如表1所示。

表1 仿真的通讯模式

Tab.1 Communication modes of simulation

最坏情况	最坏情况通讯矩阵导致的最差网络吞吐率
转置	与节点(x,y)传输至(y,x)
互补	有节点(x,y)传输至(k-x-1,k-x-1)
DOR_WC	由节点(x,y)传输至(k-y-1,k-x-1)
平均情况	对随机产生的通讯矩阵网络吞吐率求平均值

当网络规模比较低的情况下利用传统的穷举法和匈牙利算法进行实验仿真得到结果是一样的。如表2所示，采用 4×4 网络规模进行实验，理论和实验结果证明了利用穷举法和匈牙利算法得到结果是一样的。但是当网络的规模越大时穷举法复杂度成指数级增长，速度很慢，所以无法验证。

表2 当网络是 4×4 实验结果

Tab.2 Network throughput when 4×4

4×4 网络	DOR	ROMM	OITURN	VAL
转置	0.313	0.479	0.625	0.5
互补	0.625	0.652	0.625	0.5
DOR_WC	0.313	0.479	0.625	0.5
平均情况	0.465	0.640	0.550	0.5
最坏情况	0.312	0.459	0.500	0.5

实验采用以1000为周期的前提下，利用穷举法和匈牙利算法在不同的网络模式时间效率的比值 $I = \frac{\text{穷举法运行时间}}{\text{匈牙利算法运行时间}}$ (运行时间单位: ms)。实验以网络模式为 2×2 、 3×3 ，以及 4×4 进行比较，如表3结果所示。

表3 穷举法与匈牙利算法时间效率的比值

Tab.3 Exhaustive and Hungarian algorithm time efficiency ratio

网络规模	穷举法/匈牙利法比值
2×2 网络	0.96
3×3 网络	1.32×10^3
4×4 网络	3.60×10^6

表3的结果分析可知，网络模式越大时，传统的穷举法与匈牙利算法的时间效率比值成指数级别增长。从实验结果可知很明显利用匈牙利算法极大的改善了运行时间的效率。采用穷举法的时间复杂度是 $O(N!)$ ，而采用匈牙利算法的时间复杂度是 $O(N^3)$ 。当随着网络规模增加时传统的穷举法复杂度会越来越高，速度很慢。利用匈牙利算法评估网络吞吐率可以有效的改善计算的运行速度、提高效率。

6 结论(Conclusion)

利用匈牙利算法我们可以在多项式时间内找到Oblivious的路由算法的最坏情况吞吐量，这使得最坏情况的分析变得易于处理。二分图构造来分析Oblivious路由算法是评估最坏情况下网络吞吐率的一个方法。对比穷举法，计算最坏情况下的吞吐率时间效率得到很大改善；在时间复杂度上，穷举法的时间复杂度是 $O(N!)$ ，而利用匈牙利算法的时间复杂度是 $O(N^3)$ 。实验通过利用Oblivious路由算法中OITURN等算法，以及运行时间的对比，证明了利用匈牙利算法评估网络吞吐率的可行性和有效性。

参考文献(References)

- [1] Seo D, Ali A. Near-Optimal Worst-Case Throughput Routing for Two-Dimensional Mesh Networks[C]. International Symposium on Computer Architecture, Proceedings IEEE, 2005, 33(2): 432-443.
- [2] 张立, 戴一奇. 随机Oblivious路由算法中随机性与时间代价的研究[J]. 计算机学报, 1996, 19(5): 388-397.
- [3] Towles B, Dally WJ. Worst-case traffic for oblivious routing functions[C]. Fourteenth ACM Symposium on Parallel Algorithms and Architectures, ACM, 2002, 1(1): 1-8.
- [4] Sullivan, Herbert. A large scale, homogeneous, fully distributed parallel machine, I. [C]. Proc Fourth Symposium on Computer Architecture ACM, 1977, 5(7): 105-117.
- [5] J. Upadhyay, V. Varavithya, P. Mohapatra. A Traffic Balanced Adaptive Wormhole Routing Scheme for Two-Dimensional Meshes[J]. Transactions on Computers, 1997, 46(2): 190-197.
- [6] Rajasekaran S, Overholt R. Constant queue routing on a mesh[J]. Journal of Parallel & Distributed Computing, 1991, 15(15): 160-166.

作者简介:

张方爽(1993-), 女, 硕士生. 研究领域: 片上网络技术.
段新明(1970-), 男, 博士, 副教授. 研究领域: 互联网络技术, 片上网络技术, 容错路由.