超立方体网络中路由生成算法的子立方分裂方法*)

汪 洋 余少华

(华中科技大学计算机科学与技术学院 武汉430074) (武汉邮电科学研究院 武汉430074)

摘 要 超立方网络拓扑是并行计算中最重要的多处理机互连结构之一,目前对它的研究热点主要集中在当超立方的网络互连结构中具有失效结点时,如何快速计算其上点到点的并行路由。本文提出利用子立方来分解整个空间,使得原来的超立方网络互连结构可以非常规整地分解成若干个子立方,因而若以子立方为路由目的,可以显著简化路由表,从而降低并行计算中的寻路开销。进一步,这种技术应用在具有局部失效结点的超立方网络时,同样能够压缩路由表,避免从整体上计算点到点的最佳路由,将这种计算分布在超立方中的多数正常结点独立完成。这种寻路方式为高度对称的网络互连拓扑中路由的生成提供了新的思路。

关键词 超立方,子立方,路由,分布式,容错

A Sub-Cube Splitting Method in Generating Routing Table in Hypercubes

WANG Yang YU Shao-Hua

(College of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074)

(Wuhan Research Institute of Post and Telecommunications, Wuhan 430074)

Abstract Hypercube is an important type of interconnection topology in parallel computing. Recent researches focus on the computing of parallel routes of arbitrary end to end with the tolerance of faulty nodes in the hypercube. The description of this problem is: given the source and the destination, how to compute the many feasible routes between the ends? In this paper, we try to decompose the hypercube to its sub-cubes, and in this manner, the whole hypercube can be partitioned into sub-cubes with perfect regularity. Therefore, if the routing table of each single node in the hypercube aggregates its routing entries with their destinations to individual nodes into entries with their destinations to sub-cubes, the routing table can be conspicuously shortened, thus the overall routing cost is reduced in hypercube for parallel computing. Furthermore, we employ this method in the scenario where there are locally distributed faulty nodes, and likewise the routing table is considerably simplified. By avoiding computing the optimal routes from end to end, the individual normal nodes have their routing tables computed distributively. At the final part of this paper, we discuss the scenario where the method presented in this paper does not cover, and the possible approach that could generalize our work to it. This routing method brings forth a fresh idea in highly symmetric interconnection networks.

Keywords Hypercube, Sub-cube, Routing, Distributed, Fault tolerance

1 引言

超立方体是一种高度并行,容错能力极强的,具有递归结构的网络拓扑。许多其它类型的拓扑,如环、线性向量、网格等在多数应用中都可以嵌入超立方[7],从而使得基于那些网络拓扑的并行算法可以仅做不多的修改即可以运行在超立方上。由于超立方的这些优势,在并行计算领域,它被广泛用于构建多处理机系统(MPS)组成的高性能计算平台[10],因而对基于超立方的路由研究一直没有间断[2,3,5,7,8]。根据问题的由简到繁,可以分为以下三类:(1)结点到结点的并行路由;(2)结点到结点集的并行路由;(3)结点集之间的并行最佳路由[6]。在常规情形,这三个问题都已得到较满意的结果,目前对这些问题的研究主要在超立方里具有大量失效结点的情况下进行,这方面已作了大量工作[3,6,8~10]。这些工作讨论了在给定源和目的集合以及失效结点集合时,如何从整体上计算从源到目的绕过失效结点的并行路由。与上述工作不同,本文主要解决超立方体网络拓扑在具有失效结点的情况下,单个结点(主机)上的路由表生成算法。我们的工作没有过多地讨论如何从整体上计算路由的生成,因为已有的工作已对之进行了较充分的讨论,而且在实际应用中,类似源路由选路的应用并不多见。我们更多地讨论就超立方中单个结点而言,如何在获得失效结点通告的条件下快速地更新其路由表,从而使超立方上的通信有快速的恢复能力。

2 基本概念和记法

n-维超立方体网络 H_n 由 2^n 个结点构成,每一个结点由一个 n 位二进制串惟一标识,H_n 中两个结点是相邻的当且仅当两个用 n 位二进制串标识的结点恰有一位不同。对于给定的 n-维超立方体网络 H_n,将其上结点记为 $v(a_1a_2\cdots a_n)$,其中 $a_i\in\{0,1\}$ $i=1,\cdots,n$ 因此,H_n 实际上是一个无向图 $H_n=G(V,E)$。超立方 H_n 中两点 v_1,v_2 的距离用标识这两点的二进制串的海明距离表示,记为 $h(v_1,v_2)$,直观上,它表示连接这两个结点所需最少的边的条数。H_n 中的一个 k-维子立方是 H_n 的子图:对于 H_n 中那些 n 位二进制串标识的结点,将其中指定的 $n-k$ 位取值固定,而让其余 k 位取遍所有可能值,这样得到的 2^k 个结点以及连接这些结点的边称为 H_n 上的一个 k-维子立方。显然,为了描述一个 k-维子立方,需要这个 k-维子立方中的任何一个结点 v_0 以及一个 n 维0-1向量 V_m,这

*)本文得到国家"863"项目新型城域网关键技术及实验系统和高性能以太网交换机核心芯片开发(项目编号分别为2003AA121110和2003AA1Z1180)资助。汪 洋 博士研究生,主要研究方向为光城域网。余少华 博士,教授,博士生导师。

个向量正好有 k 个取值为1的分量，我们称该向量为此子立方的掩码向量，掩码向量中那些取值为1的分量称为该子立方的 k 个特征位。为此我们可以将 k-维子立方 H_k 记为 $H_{k/n}=H_{k/n}(v_0^{(k)},V_m^{(k)})$，在不会发生混淆的情况下，可以简单地记为 $H_k=H_k(v_0,V_m)$，该子立方体的维数 k 也记为 $Dim(H_k)$。注意到 v_0 可以是 H_k 中的任意结点，我们一般选取在所有 k 个特征位上全为1的结点，下文中若非特别说明都假设如此。H_n 中所有子立方的集合记作 $\boldsymbol{H}$，$k=n$ 时，子立方即 H_n 本身，$k=0$时，子立方退化成一个结点。异于 H_n 本身的子空间称为真子立方。$k=0$和 $k=n$ 时的子立方称为“平凡的”，在本文其余部分的讨论中，若非特别声明，当我们说子空间 $H_k\in\boldsymbol{H}$ 时，均默认 H_k 不是平凡的子立方。

H_n 中的一条路径是由若干个互异的相邻结点构成的一个序列，记为 $P=s_1s_2\cdots s_l$，其中 s_i 彼此互异，$h(s_i,s_{i+1})=1$ $i=1,2,\cdots,l-1$这样一条路径的长度为 $l-1$，其中 s_1,s_l 分别称为路径的源结点和目的结点。进一步，我们定义结点到子立方的距离：

定义1 H_n 中子立方 H_i 之外的一结点 v_0到 H_i 的距离 $d(v_0,H_i)=\min\{h(v_0,v)\mid v\in H_i\}$其中 h 表示海明距离，下同。

关于结点和子立方之间的距离，有以下重要性质：

性质1 设子立方 H_i 与其外一结点 v_0的距离为 d，而 $v_i\in H_i$ 使得 $h(v_0,v_i)=d$，则 H_i 中任何异于 v_i 的结点 v_i' 满足 $h(v_0,v_i')>d$

证明：设 H_i 的掩码向量为 V_m，则$\forall$ $v\in H_i$，$h(v_0,v)=h(v_0\wedge V_m,v\wedge V_m)+h(v_0\wedge\overline{V}_m,v\wedge\overline{V}_m)$其中 $\overline{V}_m$ 为 V_m 逐位取反得到的补向量。

上式中 $h(v_0\wedge\overline{V}_m,v\wedge\overline{V}_m)$是常量，选取 v，使得 v 在 s 个特征位上的取值与 v_0一致，这样选取的 v 使得 $h(v_0\wedge V_m,v\wedge V_m)=0$，如此 $h(v_0,v)$达到最小值 $h(v_0\wedge\overline{V}_m,v\wedge\overline{V}_m)=d$，因此，它正好就是题设中所说的 v_i。不难看出，v 取值为 H_i 中的任何其它结点 v_i' 均使得 $h(v_0\wedge V_m,v\wedge V_m)>0$，从而 $h(v_0,v_i')>d$。证毕

以上证明实际上给出子立方 H_i 外一结点 v_0到 H_i 距离的算法，其中 H_i 内与 v_0达到最短距离的那个的惟一结点称为 v_0在 H_i 上的投影，记为 $P_{H_i}(v_0)$。

进一步，我们定义两个子立方之间距离的概念：

定义2 设 H_i 和 H_t 是 n-维超立方体网络 H_n 中的两个子立方，$H_i\cap H_t=\phi$，则

$d=\min\{h(v_i,v_t)\mid v_i\in H_i,v_t\in H_t\}$ 称为 H_i 和 H_t 之间的距离，记为 $d(H_i,H_t)$

距离概念的引入使得我们在考察 H_n 中结点集时能够表述它们之间的疏远程度。关于两个交集非空的子立方，我们有以下重要的性质：

性质2 设 $H_i,H_t\in\boldsymbol{H}$，其掩码向量分别为 V_i 和 V_t，$H_i\cap H_t\neq\phi$，则 $H_i\cap H_t\in\boldsymbol{H}$ 且 $\mathrm{Dim}(H_i\cap H_t)=h(V_i\wedge V_t,0)$。

该性质不难证明。

借助子立方体及其距离，H_n 中每个结点的路由表不再需要以任意结点为目的来计算路由，而是目标子立方为目标路由，这正是我们在第3节中讨论的内容。

3 结点路由表的生成

H_n 中结点到结点之间的路由在绝大多数情况下都不是惟一的，这即使在 $n=2$都是显而易见的：如由 $A(00)$，$B(01)$，$C(11)$，$D(10)$构成的超立方 H_2，由 A 到 C 的可选路径有两条：$A\to B\to C$ 和 $A\to D\to C$。如果每个结点都维护到其他所有结点的所有可能路由，当 n 较大时，其路由表的尺寸将十分庞大，从而对报文的转发带来相当大的开销。为此，我们将结点到结点的选路分解为两步：设 $v_1,v_2\in H_n$，则 v_1到 v_2的路径可以分解为1）v_1到 v_2所在超立方的路径和2）v_2所在超立方内到 v_2的路径。显然，v_2可以属于多个子立方，如何规定 v_2应该属于哪个子立方似乎难以确定。但是，我们可以不考虑 v_2，而以源结点 v_1为中心考虑问题，具体而言：将 H_n 中除 v_1以外的诸结点分割成一系列交集为空的子立方，为此，我们给出以下性质：

性质3 设 v_0为 n-维超立方体网络 H_n 中一给定的结点，则 H_n 可以分裂成 n 个子立方，$H_0,H_1,\cdots,H_{n-1}$，满足 $H_i\cap H_j=\phi$，$H_i\cap v_0=\phi$，$d(v_0,H_i)=1$

这里 $i,j=0,1,\cdots,n-1$ $i\neq j$

证明：（数学归纳法）当 $n=1,2$时显然成立，若 n 直到 k 时结论都成立，当 $n=k+1$时：

记 $v_0=v_0(a_1a_2\cdots a_{k+1})$，记 $H_k=\{(a_1v_2v_3\cdots v_{k+1})\mid v_i\in\{0,1\},i=2,3,\cdots k+1\}$。则 H_k 是 H_{k+1}中的一个 k-维子立方体，$v_0\in H_k$ 且 $H_{k+1}\backslash H_k=\{(\bar{a}_1v_2v_3\cdots v_{k+1})\mid v_i\in\{0,1\},i=2,3,\cdots,k+1\}$显然 H_{k+1}/H_k 同样是 H_{k+1}中的一个 k-维子立方体。将归纳假设中关于 k 时结论成立应用于 H_k 上，注意结论成立时的海明距离是 H_k 中 k-维的海明距离：它只比较 H_k 中诸串中后 k 位中相异的位数，但这个距离可以等值、一意地延拓为 H_{k+1}中 $k+1$-维海明距离。显然，

$H_k\cap(H_{k+1}/H_k)=\phi$，$d(v_0,H_{k+1}\backslash H_k)=1$

故 n 在 $k+1$时同样成立，结合归纳假设，原结论恒成立。证毕

H_n 在按如上方式分裂成 n 个子立方后，每个子立方与 v_0的距离都为1，根据性质1，也就是说，v_0在子立方 H_i 上的投影 $p_{H_i}(v_0)$满足 $d(v_0,P_{H_i}(v_i))=1$，如此，结点 v_0在子立方的意义下的路由表仅包括 n 个表项：

目的子立方	下一跳地址	跳数
H_i	$P_{H_i}(v_0)$	1

对于目标结点，根据其属于哪一个子立方，可以迅速查到交给下一跳的地址。一般地，设 H_n 中 $v_0=v_0(a_1a_2\cdots a_n)$，则 $H_{n-1}=H_{n-1}((\bar{a}_1\cdots1),(01\cdots1))$；$H_{n-2}=H_{n-2}((a_1\bar{a}_21\cdots1),(001\cdots1))$；$H_{n-3}=H_{n-3}((a_1a_2\bar{a}_31\cdots1),(0001\cdots1))$；$\cdots$；$H_0=\{(a_1a_2\cdots a_{n-1}\bar{a}_n)\}$。按照第2节中的记法，若记 $H_i=H_i(v_i,V_i)$，则判断 H_n 中任一个结点 v 是否属于 H_i 的算法为：$v\wedge\overline{V}_i=v_i\wedge\overline{V}_i\Leftrightarrow v\in H_i$ $i=0,1,\cdots,n-1$需要略加说明的是 $i=0$时，$v_0=v_0(a_1a_2\cdots a_{n-1}\bar{a}_n)$，$V_0$为0向量。此外，$v_0$对于要转发的报文，首先判断该目的结点是否是自己的邻接结点，仅在目的结点不是邻接结点时才查询路由表，否则直接转发。

以上讨论了一个无错的超立方体 H_n 上各个结点的路由表生成算法，下面我们针对 H_n 具有失效结点的情况进一步讨论。

4 失效结点存在时路由表的生成

n-维超立方体网络 H_n 由于其很强的对称性，使得它具有较好的容错能力。事实上，H_n 的容错能力是 $n-1$，即是说至多 $n-1$个结点失效的情形下，H_n 都能保持连通。不难看出，当 n 个结点失效而使 H_n 失去连通性仅对应于这 n 个失效结点全部都是某一个结点的邻接结点这样的极端情况。失效结点的发现一般采用定期探测的方式来完成，为了将我们

的注意力集中在要讨论的关键问题，本文假定存在一种方式使得 H_n 中失效能迅速被其他结点发现，这种假设不失实际意义。

从直观上看，对于一组失效结点，它们对 H_n 中路由的影响程度取决于这组失效结点的分布情况，即它们所"分散"的范围以及它们之间位置的疏密程度，需要一种方法来刻画这种状态，为此我们引入结点集张成的子立方的概念：

定理1 设 S 为 H_n 中的结点集和，记 $E(S)=\bigcap\{H_t|H_t\in H, S\subseteq H_t\}$，则 $E(S)\in H$ 定理1是性质2的直接推论，不证。

定义3 如定理1中所示的 $E(S)$ 称为 S 所张成的子立方。

直观地说，$E(S)$ 为包含 S 的"最小"子立方。设 F 为 H_n 中的失效结点集和，F 张成的子立方 $E(F)$ 将为我们研究路由表的生成提供方便，为此给出如下引理和定理：

引理1 设 $H_{n-1}\in H$，则 $H_n\backslash H_{n-1}\in H$，且 $\forall\ v\in H_{n-1}, d(v,H_n\backslash H_{n-1})=1$

该引理不困难，不证。

定理2 设 $H_t\in H$，则 $H_n\backslash H_t$ 可以分裂成一系列子空间 $\{H_\alpha\}_{\alpha\in\Omega}$，其中 Ω 是一个有限的指标集合。这样的分裂满足：$H_n\backslash H_t\in\bigcup_{\alpha\in\Omega}H_\alpha, H_\alpha\cap H_\beta=\phi, d(H_\alpha,H_\beta)=1, d(H_t,H_\alpha)=1$，其中 $\alpha,\beta\in\Omega,\alpha\neq\beta$。

该定理的证明可以采用数学归纳法，但是其证明比较冗长，我们在此也略去不证，直接讨论这个定理在我们寻路算法中的应用。

H_n 中给出一个失效结点集 F，记 $H_F=E(F)$，则根据 H_F 的大小可以分为两种情况：(Ⅰ) H_F 是 H_n 的真子立方；(Ⅱ) H_F 就是 H_n 本身。我们假定失效结点局限在一个不大的范围，仅讨论情况(Ⅰ)：此时 H_F 外结点的选路若目的在 H_F 外，则可以利用第3节中提出的按目标子立方为选路依据；若目的在 H_F 内，则选路需要计算一个中间环节。

考虑 $H_F\in H$，根据定理2，H_F 之外存在一系列的 $\{H_\alpha\}_{\alpha\in\Omega}$，$\{H_\alpha\}_{\alpha\in\Omega}$ 中每个结点都为正常结点，对于一个 H_F 之外一个正常结点 v_0，v_0 必然属于某个 H_{α_0}，$\alpha_0\in\Omega$。对于 v_0 而言，对 H_F 之外正常结点的选路可以以子立方为目的，从而形成如下所示的 v_0 的路由表第一部分，共有 $|\Omega|-1$ 个表项：

目的子立方	下一跳地址	跳数
$H_\alpha\ \alpha\in\Omega\ \alpha\neq\alpha_0$	$P_{H_\alpha}(v_0)$	1

v_0 对于 H_{α_0} 内部的结点的选路，是在一个无错的子立方内部的选路问题，可以采用第3节中提出的子立方分裂技术完成，因此形成 v_0 的路由表第二部分，共有 $\mathrm{Dim}(H_{\alpha_0})$ 个表项：

目的子立方	下一跳地址	跳数
$H_{\alpha_0}^{(i)}$	$P_{H_{\alpha_0}^{(i)}}(v_0)$	1

v_0 的路由表第三部分是对 H_F 中那些正常结点的选路，图1显示了选路途径：

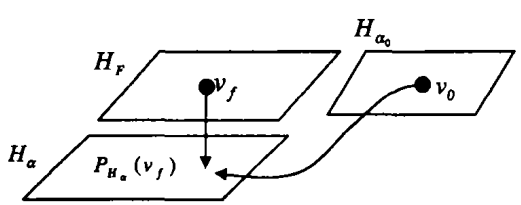

图1 对 H_F 中正常结点选路的抽象表示

设 v_f 为 H_F 中任何一个正常结点，在异于 H_F 子立方的方向任取一个与 v_f 海明距离为1的结点 v_f'，则 v_f' 必落在某个 $H_{\alpha(v_f)}(\alpha(v_f)\in\Omega)$ 中，事实上，因为 H_F 与子立方簇 $\{H_\alpha\}_{\alpha\in\Omega}$ 中的任何一个的距离都是1，结合性质1和定理2，不难看出，v_f' 即为 $P_{H_\alpha}(v_f)$，因此目的结点为 v_f 的路由将交给为 v_f'，再针对 v_f' 所在的 $H_{\alpha(v_f)}$ 寻路。如此，我们得到 v_0 的路由表第三部分，共 $|H_F|-|F|$ 个表项：

目的结点	所在子立方	下一跳地址	跳数
v_f	$H_{\alpha(v_f)}$	$H_{\alpha(v_f)}$	$h(v_0,v_f)$

同第三节中所述在无错超立方中选路一样，v_0 对于要转发的报文，首先判断该目的结点是否是自己的邻接结点，仅在目的结点不是邻接结点时才查询路由表，否则直接转发。

到此为止，对于局部出现失效结点的 H_n，利用子立方分裂的办法，可以充分利用各个结点分布计算的能力，对路由表做尽量的简化，减少了总体寻路开销。

结束语 本文主要讨论了在 n-维超立方体网络 H_n 利用子立方分裂技术简化各个结点的路由表的算法，如果在 H_n 中含有的错误结点在其张成的子立方的意义下局限于 H_n 的某个真子立方，我们提出的算法可以显著简化各个结点的路由表。今后的工作主要围绕当失效结点分散在 H_n 中各处时，如何利用子立方分裂技术进行局部寻路。我们首先想到的采用分治技术(Divide and Conquer)，在子立方中采用比较成熟的算法，然后将相关的子立方接合起来寻路，这方面还有比较深入的工作待完成。

参考文献

1 陈国良. 并行计算. 北京：高等教育出版社，1999

2 刘长河，童明生，范天佑. 超立方网络上的平行路径. 计算机学报，1999，22(2)：120～125

3 王国军，陈松桥，陈建二. 具有大量错误结点的超立方体网络中高效路由算法的设计与讨论. 计算机学报，2001，24(9)：909～916

4 童明生，刘长河，范天佑. 一般化超立方网络的容错寻径算法. 计算机学报，1998，21(12)：1074～1083

5 Lee T C, Hayes J P. A Fault-Tolerant Communication Scheme for Hypercube Computers. IEEE Transactions on Computers, 1992, 41 (10): 1242～1256

6 Gu Qian-Ping, Peng Shietung. Node-to-Set and Set-to-Set Cluster Fault Tolerant Routing in Hypercubes. Parallel Computing, 1998, 24: 1245-1261

7 Saad Y, Shultz M H. Topological properties of hypercubes. IEEE Transactions on Computers, 1988, C-37(7): 867～872

8 Wu Jie. Adaptive Fault Tolerant Routing in Cube based Multicomputers Using Safety Vectors. IEEE Transactions on Parallel and Distributed Systems, 1998, 9 (4): 321～334

9 Chiu G M, Wu S P. A Fault Tolerant Routing Strategy in Hypercube Multicomputers. IEEE Transactions on Computers, 1996, 45 (2): 143～155

10 Abdol-Hossein Esfahanian. Generalize Measures of Fault Tolerance with Application to N-Cube Networks. IEEE Transactions on Computers, 1989, 38(11)

11 Culler D E, et al. Parallel Computer Architecture: A hardware/software approach San Francisco: Morgan Kaufmann Publishers, 1999

12 Quinn M J. Parallel Computing: Theory and Practice, Second Edition New York: McGraw-Hill, 1994