



计算机科学

COMPUTER SCIENCE

分层立方体网络在NoC线性阵列中的最优嵌入

过汝燕, 王岩, 樊建席, 樊卫北

引用本文

过汝燕, 王岩, 樊建席, 樊卫北. 分层立方体网络在NoC线性阵列中的最优嵌入[J]. 计算机科学, 2023, 50(4): 249-256.

GUO Ruyan, WANG Yan, FAN Jianxi, FAN Weibei. [Optimal Embedding of Hierarchical Cubic Networks into Linear Arrays of NoC](#) [J]. Computer Science, 2023, 50(4): 249-256.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[众核处理器研究技术综述和分析](#)

Summary and Analysis of Research on ManyCore Processor Technologies

计算机科学, 2022, 49(11A): 211000012-7. <https://doi.org/10.11896/jsjcx.211000012>

[基于图嵌入的代码相似性度量](#)

Code Similarity Measurement Based on Graph Embedding

计算机科学, 2022, 49(11A): 211000186-6. <https://doi.org/10.11896/jsjcx.211000186>

[蜻蜓网络上完全独立生成树的构造算法](#)

Construction Algorithm of Completely Independent Spanning Tree in Dragonfly Network

计算机科学, 2022, 49(11): 284-292. <https://doi.org/10.11896/jsjcx.211000037>

[语义增强的完全不平衡标签网络表示学习算法](#)

Semantic Information Enhanced Network Embedding with Completely Imbalanced Labels

计算机科学, 2022, 49(11): 109-116. <https://doi.org/10.11896/jsjcx.210900101>

[一种基于UIA接口的RPA系统设计方法](#)

Design and Implementation of RPA System Based on UIA Interface

计算机科学, 2022, 49(8): 225-229. <https://doi.org/10.11896/jsjcx.211100046>

分层立方体网络在 NoC 线性阵列中的最优嵌入

过汝燕¹ 王岩¹ 樊建席¹ 樊卫北²

1 苏州大学计算机科学与技术学院 江苏 苏州 215006

2 南京邮电大学计算机学院 南京 210003

(20205227015@stu.suda.edu.cn)

摘要 随着大数据时代的到来,大规模计算的需求使得人们对芯片性能的要求日益提高,片上网络(Network-on-Chip, NoC)作为芯片内部以网络通信为中心的互连结构,在通信的各个方面实现了良好的平衡。NoC 组件的物理布局及互连方式对芯片的总体性能(如信号延迟、电路成本等)有着很大的影响,因芯片面积有限,最小化连接组件的导线总长度,即最小化线长,被作为芯片设计的重点。分层立方体网络(Hierarchical Cubic Network, HCN)具有通信延迟低、可靠性和扩展性高等优点,而线性阵列是 NoC 常用的拓扑结构之一,将分层立方体网络移植到线性阵列上,就可以在线性阵列上模拟分层立方体网络的结构和算法。图嵌入是实现网络移植的关键技术。在图嵌入中,最小化导线总长度的目标可以通过求解具有最小线长的最优嵌入来达成。文中主要研究了分层立方体网络在线性阵列中的最优嵌入问题。首先,通过研究分层立方体网络的最优集,提出了分层立方体网络在线性阵列中的一种嵌入方案 *hel*, 并证明在嵌入方案 *hel* 下的线长相比其他嵌入方案下的线长是最小的,即 *hel* 为最优嵌入;然后给出了嵌入 *hel* 下线长的精确值以及一个时间复杂度为 $O(N)$ 的嵌入算法,其中 N 为 n 维分层立方体网络的顶点数且 $N=2^{2n}$;其次,还给出了分层立方体网络在 NoC 上的线性物理布局算法;最后,通过对比实验评估了嵌入 *hel* 的性能。

关键词: 片上网络;图嵌入;线长;分层立方体网络;线性阵列

中图法分类号 TP393

Optimal Embedding of Hierarchical Cubic Networks into Linear Arrays of NoC

GUO Ruyan¹, WANG Yan¹, FAN Jianxi¹ and FAN Weibei²

1 School of Computer Science and Technology, Soochow University, Suzhou, Jiangsu 215006, China

2 School of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Abstract With the advent of the era of big data, the demand of large-scale computing makes the requirements on the performance of chips constantly increasing. Network-on-Chip (NoC), as a network-communication-centered interconnection structure on chips, has achieved a good tradeoff in all aspects of communication. The physical layout and interconnection mode of NoC components have a significant impact on the overall performance of the chip (such as signal delay, circuit cost). Due to the limited area of the chip, minimizing the total length of wires connecting components, in other words, minimizing wirelength, is considered as the key of chip design. The hierarchical cubic network is an excellent interconnection network which has less communication delay, better reliability and greater scalability while the linear array is one of the common topologies of NoC. When the hierarchical cubic network is ported to the linear array, the structure and algorithm of hierarchical cubic network can be simulated on the linear array. Graph embedding is a key technology to realize network porting. In graph embedding, the goal of minimizing the total wire length can be achieved by finding the optimal embedding with minimum wirelength. This paper mainly studies the optimal embedding problem of hierarchical cubic networks into linear arrays with minimum wirelength. Firstly, by studying the optimal set of the hierarchical cubic network, an embedding scheme *hel* of hierarchical cubic networks into linear arrays is proposed, and it is proved that the wirelength under *hel* is minimum compared with other embedding schemes, that is, *hel* is an optimal embedding. Then, the exact value of the wirelength under *hel* and an embedding algorithm with $O(N)$ time complexity are given, where $N=2^{2n}$ is the number of vertices of the n -dimensional hierarchical cube network. Furthermore, an algorithm of physical linear layout in NoC

到稿日期:2022-01-04 返修日期:2022-05-16

基金项目:国家自然科学基金联合基金(U1905211);江苏高校优势学科建设工程资助项目

This work was supported by the Joint Fund of the National Natural Science Foundation of China (U1905211) and Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

通信作者:王岩(wangyanme@suda.edu.cn)

for hierarchical cubic networks is proposed. Finally, comparison experiments are conducted to evaluate the performance of embedding *hel*.

Keywords Network-on-Chip, Graph embedding, Wirelength, Hierarchical cubic networks, Linear arrays

1 引言

超大规模集成电路 (Very Large Scale Integration, VLSI) 技术可以将整个电子系统或其子系统集成到一块芯片上, 实现信息的收集、处理和存储等功能。芯片上电子系统组件间的互连方式很大程度上影响着该系统的通信性能以及成本消耗。随着电子系统组件数量的剧增, 组件间的互连方式成为制约芯片发展的重要因素, 片上网络 (NoC) 技术应运而生^[1]。NoC 在芯片上实现了以网络通信为中心的互连结构, 克服了传统电子系统的局限性, 在延迟、吞吐量和芯片面积消耗等方面实现了良好的平衡。

从拓扑角度来看, NoC 的结构可以用简单图 $G=(V, E)$ 来表示, 图中的顶点集 V 和边集 E 分别表示 NoC 上的物理组件及连接它们的物理导线。NoC 上组件的物理布局及其连接方式对芯片的总体性能 (如信号延迟、功耗、电路成本等) 有着很大的影响, 其中, 在芯片面积有限的情况下, 连接组件的导线总长度是制约芯片性能的关键因素。因此设计芯片时, 我们将重点关注最小化导线总长度问题, 即最小化线长 (Wirelength) 问题。

超立方体及其变型网络具有可扩展性好、可靠度高、通信时延低等良好的通信性能, 但它们的局限性在于, 在节点数量增加时需要改变基本的节点配置, 而且会导致网络的链路数量呈指数级增长, 进而降低了其对大型系统的普适性^[2]。分层互连网络可以利用并行应用的通信局部性来设计网络, 从而降低链路成本; 同时, 分层互连网络采用多层次网络和复制高级网络的技术, 拥有良好的通信和容错能力。Ghose 等提出的分层立方体网络 (HCN)^[3] 作为分层互连网络中的一种综合性质优良的网络, 与有着相同规模 (顶点数) 的超立方体相比, 其边数几乎是超立方体的一半, 直径也几乎是超立方体的 $2/3$ ^[4]。这意味着在最坏的情况下, 任意两个顶点之间的通信延迟减少了近 $1/3$ ^[5]。近年来, 不少研究都证明了分层立方体网络在容错和连通度方面的优良性能, 如 Bossard 等研究了分层立方体网络中一个源节点到多个目的节点之间的不相交路径^[6]; Liu 等研究了在 PMC 模型和 MM^* 模型下分层立方体网络的 g -额外条件诊断度^[7]; Li 等研究了在 PMC 诊断模型下分层立方体网络的条件诊断度^[8]; Zhao 等提出了分层立方体网络的广义 4-连通度^[9]。

分层立方体网络是一种性能优良的互连网络, 将其移植到 NoC 中, 我们就可以在 NoC 上模拟该网络的结构和算法, 从而使 NoC 也具有类似的通信模式。而上述的网络移植可以通过图嵌入来实现。图嵌入可以将一个网络 (客图) 嵌入另一个网络 (主图) 中, 从而实现客图的结构模拟和算法移植。因为芯片设计的重点是最小化线长, 所以研究互连网络在 NoC 上的嵌入时, 通常把线长作为衡量图嵌入性能的主要指标, 并且把具有最小线长的图嵌入称为最优嵌入。于是, 设计

芯片时的最小化导线总长度问题就转换成了求解最优嵌入问题。

NoC 中常用的拓扑结构有线性阵列 (Linear Arrays)、环 (Rings)、网格 (Grids)、环绕 (Torus) 等^[10-11], 其中线性阵列的嵌入问题是研究其他拓扑结构嵌入问题的基础。因此, 众多学者围绕上述嵌入问题展开了研究, 如 Manuel 等研究了超立方体在线性阵列和网格中的最优嵌入并给出了精确的最小线长^[12]; Arockiaraj 等首先研究了局部扭立方体在线性阵列中的最优嵌入^[13], 然后, Shalini 等以此为基础提出了局部扭立方体在网格中的最优嵌入算法^[14]; Fan 等解决了 3 元 n 维立方体^[15] 和交换立方体^[16] 在线性阵列和网格中的最优嵌入问题, 并且给出了具体的线性物理布局算法; Xia 等提出了增广立方体在网格中的最优嵌入算法^[17]。

由上述内容可知, 分层立方体网络在线性阵列中的最优嵌入问题目前还没有得到解决, 故本文针对该问题展开了研究。解决该问题的主要难点在于:

(1) 与超立方体及其变型网络相比, 分层立方体网络中边的连接方式更复杂, 这使得寻找最优嵌入的过程更困难。

(2) 在现有的研究中, 大多数文献求解得到的最优嵌入都为词典嵌入^[18], 即 $lex: V(G) \rightarrow V(H)$, 对于任意点 $u = u_{n-1}u_{n-2}\cdots u_0 \in V(G)$, 令 $lex(u) = \sum_{i=0}^{n-1} u_i 2^i$, 即 $lex(u)$ 为点 u 的十进制表示, 其中 $u_i \in \{0, 1\}$ 且 $0 \leq i \leq n-1$ 。但词典嵌入对分层立方体网络在线性阵列中的嵌入来说并不是最优嵌入。

我们通过研究分层立方体网络的最优集, 解决了分层立方体网络在线性阵列中的最优嵌入问题。本文的贡献如下:

(1) 研究了分层立方体网络的最优集。

(2) 提出了分层立方体网络在线性阵列中的嵌入方案 *hel*, 证明了嵌入 *hel* 为最优嵌入, 并且给出了该嵌入下最小线长的精确值。

(3) 给出了时间复杂度为 $O(N)$ 的嵌入算法以及 NoC 上的线性物理布局算法, 其中 N 表示一个 n 维分层立方体网络的顶点数且 $N = 2^{2n}$ 。

本文第 2 节给出了本文需要使用的图论相关术语和符号; 第 3 节介绍了解决最优嵌入问题的基础, 即最优集问题, 接着给出了分层立方体网络在线性阵列中的最优嵌入 *hel*、最小线长的精确值和相应的嵌入算法, 以及分层立方体网络在 NoC 上的线性物理布局算法; 第 4 节进行了对比实验, 实验结果表明本文提出的嵌入 *hel* 与其他两种嵌入方案相比具有更大的优势; 最后总结全文。

2 预备知识

2.1 术语和符号

一个简单无向图可以用 $G=(V(G), E(G))$ 来表示, 其中 $V(G)$ 是 G 的顶点集, $E(G)$ 是 G 的边集。给定两个简单无向图 $G_1=(V(G_1), E(G_1))$ 和 $G_2=(V(G_2), E(G_2))$, 如果

$V(G_2) \subseteq V(G_1)$ 并且 $E(G_2) \subseteq E(G_1)$, 那么称 G_2 是 G_1 的一个子图; 如果 $V' \subseteq V(G)$ 且 $E' = \{(u, v) \in E(G) \mid u, v \in V'\}$, 那么称 $G[V']$ 是 G 的导出子图。

给定两个简单图 $G = (V(G), E(G))$ 和 $H = (V(H), E(H))$, S 是 G 的一个顶点子集, 令 f 为一个一对一映射为 $f: V(G) \rightarrow V(H)$, 令 $T = \{x \in V(H) \mid f^{-1}(x) \in S\}$, 于是 $T = f(S)$ 且 $S = f^{-1}(T)$ 。对于图 G 和图 H , 如果存在两个一对一映射 $\Psi: V(G) \rightarrow V(H)$ 和 $\Phi: E(G) \rightarrow E(H)$, 对任意 $e \in (u, v) \in E(G)$, 有 $\Phi(e) = (\Psi(u), \Psi(v)) \in E(H)$, 那么称 G 同构于 H , 记作 $G \cong H$ 。

客图 G 在主图 H 中的嵌入定义如下^[12]: 1) f 是 $V(G) \rightarrow V(H)$ 的一对一映射; 2) f_p 是 $E(G) \rightarrow \{P_f(f(u), f(v))\}$ 的一对一映射, 其中 $P_f(f(u), f(v))$ 表示 H 中点 $f(u)$ 到点 $f(v)$ 的最短路径且 $(u, v) \in E(G)$ 。为了方便叙述, 我们把该嵌入简单地记作 f 。

接着, 我们给出边拥塞度和线长的定义。

定义 1^[12] 给定一个图 G 在图 H 中的嵌入 f , e 是 H 中的任意一条边, 边 e 的边拥塞度为 $EC_f(e) = |\{(u, v) \in E(G) \mid e \in E(P_f(f(u), f(v)))\}|$ 。

定义 2^[19] 给定一个图 G 在图 H 中的嵌入 f , 嵌入 f 下的线长为 $WL_f(G, H) = \sum_{(u,v) \in E(G)} d(f(u), f(v)) = \sum_{e \in E(H)} EC_f(e)$, 其中 $d(f(u), f(v))$ 表示路径 $P_f(f(u), f(v))$ 的长度。

进而引入图 G 在图 H 中的嵌入的最小线长, 其定义如下。

定义 3^[12] 图 G 在图 H 中的嵌入的最小线长为 $WL(G, H) = \min\{WL_f(G, H) \mid f \text{ 是一个图 } G \text{ 在图 } H \text{ 中的嵌入}\}$ 。

因此, 最优嵌入问题就是找到一个嵌入 f 使其线长满足 $WL_f(G, H) = WL(G, H)$ 。由于本文研究的是分层立方体网络在线性阵列中的最优嵌入问题, 我们首先介绍分层立方体网络的相关概念。

2.2 分层立方体网络

一个 n 维的分层立方体网络 \mathcal{H}_n 由 2^n 个 n 维超立方体组成, 这些超立方体也被称作簇。 \mathcal{H}_n 中顶点的总数为 $N = 2^{2n}$ 。点 $u \in V(\mathcal{H}_n)$ 可以被表示成一对 n 位的二进制序列, 即 $u = (u_{2n-1} \cdots u_n, u_{n-1} \cdots u_0)$, 其中 $u_{2n-1} \cdots u_n$ 代表点 u 的簇号, $u_{n-1} \cdots u_0$ 代表点 u 在簇中的点号, $u_i \in \{0, 1\}$ 且 $0 \leq i \leq 2n-1$ 。

定义 4^[3] \mathcal{H}_n 中两个点 $u = (x, y)$ 和 $v = (w, z)$ 相连当且仅当下列 3 个条件之一成立 (其中 \bar{x} 是 x 序列的二进制补码):

- (1) 如果 $x = w$, 那么 $H(y, z) = 1$;
- (2) 如果 $x \neq w$ 且 $x \neq y$, 那么 $x = z, y = w$;
- (3) 如果 $x \neq w$ 且 $x = y$, 那么 $w = z = \bar{x}$ 。

其中 $H(y, z)$ 表示序列 y 和序列 z 不同的位数。令 E_1, E_2 和 E_3 分别表示满足条件 (1), (2) 和 (3) 的边的集合。

根据上述定义可以得到以下结论: 如果两个簇的簇号是互补的, 那么它们之间有两条边, 否则就只有一条。

对于点 $u = (u_{2n-1} \cdots u_n, u_{n-1} \cdots u_0) \in V(\mathcal{H}_n)$, 令 $c(u) = \sum_{i=n}^{2n-1} u_i 2^i$ 表示点 u 所在簇的十进制簇号, $p(u) = \sum_{i=0}^{n-1} u_i 2^i$ 表示点

u 在簇中的十进制点号, $l(u) = \sum_{i=0}^{2n-1} u_i 2^i$ 是点 u 在 \mathcal{H}_n 中的十进制表示。我们把簇号为 $c(u)$ 的 n 维立方体记作 $Q_n^{c(u)}$, 并且称 $Q_n^{c(u)}$ 和 $Q_n^{2^n - 1 - c(u)}$ 是互补的。

图 1 给出了一个 2 维的分层立方体网络 \mathcal{H}_2 , 其中 E_1, E_2 和 E_3 中的边分别用粗线、实线和虚线来表示, 红色字体和黑色字体分别是点的十进制和二进制表示。

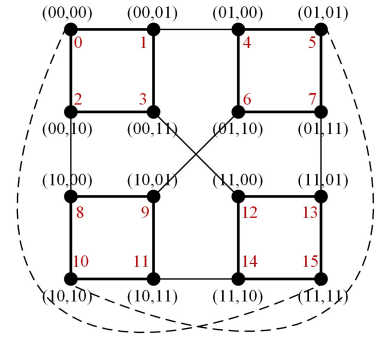


图 1 分层立方体网络 \mathcal{H}_2 (电子版为彩图)
Fig. 1 Hierarchical cubic network \mathcal{H}_2

3 分层立方体网络在线性阵列中的最优嵌入及线性物理布局

3.1 分层立方体网络的最优集

为了解决最优嵌入问题, 我们首先引入最优集问题和最大导出子图问题。

对于图 $G = (V(G), E(G))$, 在所有点数为 m 的导出子图中, 边数最多的导出子图被称作最大导出子图。令最大导出子图的边数为 $N_G(m) = \max\{|E(G[A])| \mid A \subseteq V(G) \text{ 且 } |A| = m\}$, 其中顶点集 A 满足 $A \subseteq V(G)$ 且 $|A| = m$ 。最大导出子图问题就是找到一个顶点集 $A \subseteq V(G)$ 满足 $|A| = m$ 和 $|E(G[A])| = N_G(m)$ 。

对于图 $G = (V(G), E(G))$, 在所有将 $V(G)$ 分成两部分 (A 和 $V(G) - A$) 且满足 $|A| = m$ 的边割中, 使得边割具有最少边数的顶点集 A 被称作最优集。令最优集对应边割的边数为 $\theta_G(m) = \min\{|[A, V(G) - A]| \mid A \subseteq V(G), |A| = m \text{ 且 } [A, V(G) - A] = \{(u, v) \in E(G) \mid u \in A, v \in V(G) - A\}\}$ 。最优集问题就是找到一个顶点集 $A \subseteq V(G)$ 满足 $|A| = m$ 以及 $|[A, V(G) - A]| = \theta_G(m)$ 。

一个正则图的最优集问题和最大导出子图问题本质是一样的^[16]。因为 \mathcal{H}_n 为 $(n+1)$ -正则图, 所以它的最大导出子图的顶点集也是它的最优集。

首先, 我们将在定义 5 和定义 6 中分别给出顶点集 $L_i'(k)$ 和 \mathcal{L}_i 的定义。

定义 5 在 \mathcal{H}_n 的一个簇 Q_n^k 中, 令有序点集 $L'(k)$ 为 $L'(k) = \{0 + k \cdot 2^n, 1 + k \cdot 2^n, \dots, 2^{n-1} - 1 + k \cdot 2^n, 2^n - 1 + k \cdot 2^n, 2^n - 2 + k \cdot 2^n, \dots, 2^{n-1} + 1 + k \cdot 2^n, 2^{n-1} + k \cdot 2^n\}$, 其中 $0 \leq k < 2^n$ 。 $L'(k)$ 的前 i 个点组成的点集记作 $L_i'(k)$, 其中 $i = 1, 2, \dots, 2^n$ 。

定义 6 对于 $0 \leq x < 2^n, 0 < y \leq 2^n$, 令 $i = x \cdot 2^n + y$ 。顶点集 \mathcal{L}_i 的定义如下:

$$\mathcal{L}_i = \begin{cases} \tilde{V}UL_y' \left(\frac{x}{2} \right), & x \bmod 2 = 0 \\ \tilde{V}UL' \left(\left\lfloor \frac{x}{2} \right\rfloor \right) UL_y' \left(2^n - 1 - \left\lfloor \frac{x}{2} \right\rfloor \right), & x \bmod 2 = 1 \end{cases}$$

其中 $\tilde{V} = \bigcup_{k=0}^{\lfloor \frac{x}{2} \rfloor - 1} L'(k) \cup L'(2^n - 1 - k)$ 。

接着,我们将在定理 1 中证明上述顶点集 \mathcal{L}_i 为分层立方体网络的最优集。为了证明定理 1,我们给出如下引理。

引理 1^[12] 设 Q_n 是一个 n 维的超立方体, $L_i = \{0, 1, \dots, i-1\}$ 是 Q_n 的一个最优集,其中 $i=1, 2, \dots, 2^n$ 。

引理 2 $L_i'(k)$ 是 Q_n^k 的一个最优集,其中 $i=1, 2, \dots, 2^n$ 并且 $0 \leq k < 2^n$ 。

证明:不失一般性,我们假设 $k=0$ 。于是有以下两种情况。

情况 1 $1 \leq i \leq 2^{n-1}$ 。很明显, $L_i'(0) = L_i = \{0, 1, \dots, i-1\}$, 因此 $L_i'(0)$ 是 Q_n^0 的一个最优集。

情况 2 $2^{n-1} < i \leq 2^n$ 。此时 $L_i'(0) = \{0, 1, \dots, 2^{n-1} - 1, 2^n - 1, \dots, 3 \times 2^{n-1} - i\}$ 。首先我们给出一个映射 $h: L_i'(0) \rightarrow L_i$, 对于 $L_i'(0)$ 中的任意点 u , 令 $h(u) = \begin{cases} 2^{n-1} - 1 - u, & u \in \{0, \dots, 2^{n-1} - 1\} \\ 3 \times 2^{n-1} - 1 - u, & u \in \{2^n - 1, \dots, 2^{n-1}\} \end{cases}$ 。设点 u 的二进制为 $(u_{2n-1} \dots u_n, u_{n-1} u_{n-2} \dots u_0)$, 以及点 $h(u)$ 的二进制为 $(u'_{2n-1} \dots u'_n, u'_{n-1} u'_{n-2} \dots u'_0)$, 其中 $u'_m = \begin{cases} 1 - u_m, & 0 \leq m < n-1 \\ u_m, & n-1 \leq m \leq 2n-1 \end{cases}$ 。接着我们证明对于任意两个点

$u, v \in L_i'(0)$, 若存在 $(u, v) \in E(Q_n^0[L_i'(0)])$, 则一定有 $(h(u), h(v)) \in E(Q_n[L_i])$ 。 $(u, v) \in E(Q_n^0[L_i'(0)])$ 表示点 u 和点 v 的二进制序列有且仅有一位不同。根据上述映射,可以看出点 $h(u)$ 和点 $h(v)$ 的二进制序列也是有且仅有一位不同,也就是说一定有 $(h(u), h(v)) \in E(Q_n[L_i])$ 。通过同构的定义,我们得知 $Q_n^0[L_i'(0)] \cong Q_n[L_i]$, 又根据引理 1, 可以证明 $L_i'(0)$ 是 Q_n^0 的一个最优集。

根据定义 4 可知 \mathcal{H}_n 中的任意 Q_n^s 和 Q_n^t 都满足 $Q_n^s \cong Q_n^t$, 其中 $0 \leq s < t < 2^n$ 。综上所述,引理 2 得证。

引理 3^[15] 在一个 n 维的超立方体 Q_n 中, $|E(Q_n[L_i])| + |E(Q_n[L_j])| + \min\{i, j\} \leq |E(Q_n[L_{i+j}])|$, 其中 $1 \leq i, j < 2^n$ 并且 $2 \leq i+j \leq 2^n$ 。

根据引理 3, 我们可证得如下引理 4 成立。

引理 4 $S_{k_1}(s)$ 是 Q_n^s 中任意一个具有 k_1 个点的点集, $S_{k_2}(t)$ 是 Q_n^t 中任意一个具有 k_2 个点的点集, 其中 $1 \leq k_1 \leq k_2 \leq 2^n, 0 \leq s, t < 2^n$, 并且 $S_{k_1}(s) \cap S_{k_2}(t) = \emptyset$, 则有 $|E(\mathcal{H}_n[S_{k_1}(s) \cup S_{k_2}(t)])| \leq |E(\mathcal{H}_n[\mathcal{L}_{k_1+k_2}])|$ 。

证明:设 $E(\mathcal{H}_n[A \cup B]) = \{(u, v) \in E(\mathcal{H}_n) \mid u \in A, v \in B\}$, 其中 $A, B \subset V(\mathcal{H}_n)$ 。令 $m_i = |E_i \cap E(\mathcal{H}_n[S_{k_1}(s) \cup S_{k_2}(t)])|$, 其中 $i=1, 2, 3$ 。由定义 4 可知, 两个簇之间最多有两条边, 故 $0 \leq m_2 + m_3 \leq 2$ 。不失一般性, 我们假设 $s=0$, 为获得最大导出子图, 令 $t=2^n-1-s$, 于是 $t=2^n-1$ 。设 $k=k_1+k_2$, 对于 $2 \leq k \leq 2^{n+1}$, 我们考虑下列两种情况。

情况 1 $2 \leq k \leq 2^n$ 。 $|E(\mathcal{H}_n[S_{k_1}(s) \cup S_{k_2}(t)])| = |E(\mathcal{H}_n[S_{k_1}(s)])| + |E(\mathcal{H}_n[S_{k_2}(t)])| + |E(\mathcal{H}_n[S_{k_1}(s) \wedge S_{k_2}(t)])| \leq |E(\mathcal{H}_n[L'_{k_1}(s)])| + |E(\mathcal{H}_n[L'_{k_2}(t)])| + \min\{k_1, k_2\} \leq |E(\mathcal{H}_n$

$[L'_{k_1+k_2}(s)])| = |E(\mathcal{H}_n[\mathcal{L}_{k_1+k_2}])|$ 。

情况 2 $2^n < k \leq 2^{n+1}$ 。 $|E(\mathcal{H}_n[S_{k_1}(s) \cup S_{k_2}(t)])| = |E(\mathcal{H}_n[S_{k_1}(s)])| + |E(\mathcal{H}_n[S_{k_2}(t)])| + |E(\mathcal{H}_n[S_{k_1}(s) \wedge S_{k_2}(t)])| \leq |E(\mathcal{H}_n[L'_{k_1}(s)])| + |E(\mathcal{H}_n[L'_{k_2}(t)])| + m_2 + m_3 \leq |E(\mathcal{H}_n[\mathcal{L}_{k_1+k_2}])|$ 。

综上所述,引理 4 成立。

根据引理 4, 在已证得 2 个簇的情况下, \mathcal{L}_i 为 \mathcal{H}_n 的最优集。使用类似的证明方法, 我们可以证得 k 个簇的情况下, \mathcal{L}_i 为 \mathcal{H}_n 的最优集, 其中 $2 < k \leq 2^n$, 于是可以得到如下定理。

定理 1 \mathcal{L}_i 是 \mathcal{H}_n 的一个最优集, 其中 $i=1, 2, \dots, 2^{2^n}$ 。

3.2 分层立方体网络的最优嵌入

在本节中, 我们将利用最优集来解决分层立方体网络在线性阵列中具有最小线长的最优嵌入问题。首先将介绍两个重要的引理。

引理 5^[12] 图 G 是一个 r -正则图, f 是图 G 在图 H 中的一种嵌入方案。 S 是 H 的一个边割, 去掉 S 后 H 将被分成两个部分 H_1 和 H_2 , 令 $G_1 = f^{-1}(H_1)$ 并且 $G_2 = f^{-1}(H_2)$ 。 S 若满足下列条件:

(1) 对任意边 $(u, v) \in E(G_1) \cup E(G_2)$, $P_f(f(u), f(v))$ 在 S 中没有边;

(2) 对任意边 $(u, v) \in E(G)$ 且 $u \in V(G_1), v \in V(G_2)$, $P_f(f(u), f(v))$ 在 S 中有且仅有一条边;

(3) $V(G_1)$ 是一个最优集。

则 $EC_f(S) = \sum_{e \in S} EC_f(e)$ 最小, 也就是说, 对于图 G 在图 H 中的任意一种嵌入方案 g 都满足 $EC_f(S) \leq EC_g(S)$ 。我们还可以得到 $EC_f(S) = r|V(G_1)| - 2|E(G_1)|$ 。

引理 6^[12] $f: G \rightarrow H$ 是图 G 到图 H 的一种嵌入方案, $\{S_1, S_2, \dots, S_p\}$ 是 $E(H)$ 的划分, 其中每个 S_i 都是 H 的一个边割。于是有 $WL_f(G, H) = \sum_{i=1}^p EC_f(S_i)$ 。

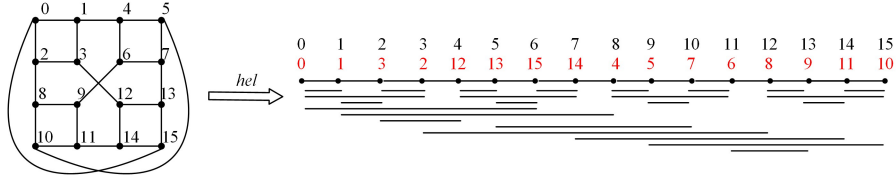
定义 7^[16] 对于任意一个整数 $n \geq 1$, 具有 n 个点的线性阵列可以被记作 P_n , 它的顶点集为 $V(P_n) = \{0, 1, \dots, n-1\}$, 边集为 $E(P_n) = \{(i, i+1) \mid 0 \leq i \leq n-2\}$ 。

然后, 在定义 8 中给出分层立方体网络在线性阵列中的一种嵌入方案 hel , 进而在引理 7 中证明该嵌入方案具有最小线长, 即该嵌入为最优嵌入。

定义 8 $hel: V(\mathcal{H}_n) \rightarrow V(P_{2^{2^n}})$ 是分层立方体网络 \mathcal{H}_n 在线性阵列 $P_{2^{2^n}}$ 中的一种嵌入方案。对于任意点 $u \in V(\mathcal{H}_n)$, 我们定义:

$$hel(u) = \begin{cases} 2^{n+1}c(u) + p(u), & 0 \leq c(u) < 2^{n-1}, \\ & 0 \leq p(u) < 2^{n-1} \\ 2^{n+1}c(u) - p(u) + 3 \cdot 2^{n-1} - 1, & 0 \leq c(u) < 2^{n-1}, \\ & 2^{n-1} \leq p(u) < 2^n \\ -2^{n+1}c(u) + p(u) + 2^{2n+1} - 2^n, & 2^{n-1} \leq c(u) < 2^n, \\ & 0 \leq p(u) < 2^{n-1} \\ -2^{n+1}c(u) - p(u) + 2^{2n+1} + 2^{n-1} - 1, & 2^{n-1} \leq c(u) < 2^n, \\ & 2^{n-1} \leq p(u) < 2^n \end{cases}$$

图 2 给出了 \mathcal{H}_2 在线性阵列 P_{2^4} 中的嵌入 hel , 其中黑色字体表示线性阵列中点的十进制表示, 红色字体表示分层立方体网络嵌入线性阵列后的点的十进制表示。

图2 分层立方体网络 \mathcal{H}_2 在线性阵列 P_{2^4} 中的嵌入(电子版为彩图)Fig. 2 Embedding of hierarchical cubic network \mathcal{H}_2 in linear array P_{2^4}

引理 7 嵌入 hel 下的线长是分层立方体网络 \mathcal{H}_n 嵌入到线性阵列 $P_{2^{2n}}$ 的最小线长。

证明:令 $G = \mathcal{H}_n$, $H = P_{2^{2n}}$, 边割 S_i 是线性阵列 $P_{2^{2n}}$ 的第 i 条边, 即 $S_i = \{(i-1, i)\}$, 其中 $1 \leq i < 2^{2n}$. S_i 将 H 分成两部分, 即 H_1 和 H_2 , $G_1 = hel^{-1}(H_1)$, $G_2 = hel^{-1}(H_2)$. 很明显, G_1 和 G_2 都满足引理 5 的条件(1)和(2). 又因为 $V(G_1) = \mathcal{L}_i$, 根据引理 5 和定理 1 我们可以得到 $EC_{hel}(S_i)$ 是最小的. 根据引理 6 可知, 引理 7 成立.

根据引理 7, 我们已经证得嵌入 hel 为分层立方体网络在线性阵列中的最优嵌入. 下面将根据引理 8, 在定理 2 中给出最小线长的精确值及证明过程.

引理 8^[12] 超立方体 Q_n 嵌入线性阵列 $P_{2^{2n}}$ 的最小线长为 $WL_f(Q_n, P_{2^{2n}}) = 2^{2n-1} - 2^{n-1}$.

定理 2 分层立方体网络 \mathcal{H}_n 在线性阵列 $P_{2^{2n}}$ 中的最小线长为 $WL_{hel}(\mathcal{H}_n, P_{2^{2n}}) = \frac{1}{3}(2^{4n-1} + 5 \times 2^{3n-2} - 2^{2n-1} + 2^n)$.

证明: 根据引理 7 和定义 2, 我们通过计算路径 $P_{hel}(hel(u), hel(v))$ 的总长度来计算该嵌入下的总线长, 其中 $(u, v) \in E(\mathcal{H}_n)$. 有如下 3 种情况.

情况 1 $(u, v) \in E_1$. 每个簇都是一个超立方体, 而 Q_n 可以被分成两个不相交的 Q_{n-1} , 根据引理 8, 我们得到此情况下的线长为 $\omega_1 = 2WL_{hel}(Q_{n-1}, P_{2^{2(n-1)}}) + (1+3+\dots+2^n-1) = 2^{2n-1} - 2^{n-1}$.

情况 2 $(u, v) \in E_2$. 不失一般性, 我们不妨假设 $c(u) < c(v)$.

情况 2.1 $0 \leq c(u) < c(v) < 2^{n-1}$. 满足该情况的路径中, 有 $2^{n-1}-1$ 条长度为 $2 \times 1 \times (2^n-1) + 1$ 的路径, $2^{n-1}-2$ 条长度为 $2 \times 2 \times (2^n-1) + 2$ 的路径, \dots , 以及一条长度为 $2 \times (2^{n-1}-1) \times (2^n-1) + 2^{n-1}-1$ 的路径. 因此, 该情况下的线长为 $\omega_2 = \sum_{j=1}^{2^{n-1}-1} [2j(2^n-1) + j](2^{n-1}-j)$.

情况 2.2 $0 \leq c(u) < 2^{n-1} < c(v) < 2^n$ 并且 $2c(u) \leq 2(2^n - c(v)) - 1$. 满足该情况的路径中, 有 2^{n-1} 条长度为 $[2^n - 1 - 2 \times (2^{n-1} - 1)](2^n - 1) - (2^{n-1} - 1)$ 的路径, $2^{n-1} - 1$ 条长度为 $[2^n - 1 - 2 \times (2^{n-1} - 2)](2^n - 1) - (2^{n-1} - 2)$ 的路径, \dots , 以及一条长度为 $(2^n - 1 - 2 \times 0)(2^n - 1) - 0$ 的路径. 因此, 该情况下的线长为 $\omega_3 = \sum_{j=1}^{2^{n-1}-1} [(2^n - 1 - 2 \times j)(2^n - 1) - j](j+1)$.

情况 2.3 $0 \leq c(u) < 2^{n-1} < c(v) < 2^n$ 并且 $2c(u) > 2(2^n - c(v)) - 1$. 满足该情况的路径中, 有 $2^{n-1}-1$ 条长度为 $2 \times 1 \times (2^n-1) - (2^{n-1}-1)$ 的路径, $2^{n-1}-2$ 条长度为 $2 \times 2 \times (2^n-1) - (2^{n-1}-2)$ 的路径, \dots , 以及一条长度为 $2 \times (2^{n-1}-1) \times (2^n-1) - 1$ 的路径. 因此, 该情况下的线长为 $\omega_4 =$

$$\sum_{j=1}^{2^{n-1}-1} [2j(2^n-1) - (2^{n-1}-j)](2^{n-1}-j).$$

情况 2.4 $2^{n-1} \leq c(u) < c(v) < 2^n$. 与情况 2.1 类似, 该情况下的线长为 $\omega_5 = \sum_{j=1}^{2^{n-1}-1} [2j(2^n-1) + j](2^{n-1}-j)$.

情况 3 $(u, v) \in E_3$. 满足该情况的路径中, 有 2^{n-1} 条长度为 $2^n + 2^{n-1}$ 的路径, 因此, 该情况下的线长为 $\omega_6 = 2^{n-1}(2^n + 2^{n-1})$.

最后, 我们可得分层立方体网络 \mathcal{H}_n 嵌入线性阵列 $P_{2^{2n}}$ 的最小线长为 $WL_{hel}(\mathcal{H}_n, P_{2^{2n}}) = \sum_{i=1}^6 \omega_i = \frac{1}{3}(2^{4n-1} + 5 \times 2^{3n-2} - 2^{2n-1} + 2^n)$, 定理 2 得证.

接着, 给出一个分层立方体网络在线性阵列中的最优嵌入算法, 即算法 1. 在算法 1 中, 为了更清晰地区分顶点, 我们用 u^i 来表示十进制为 i 的顶点 u . 对于每个点 $u \in V(\mathcal{H}_n)$, 首先将点 u 标记为它在 \mathcal{H}_n 中的十进制值, 然后将点 u 映射到线性阵列的点 $v = hel(u)$ 上; 最后得到最优嵌入 hel .

分层立方体网络 \mathcal{H}_n 在线性阵列 $P_{2^{2n}}$ 中的最优嵌入算法的时间复杂度为 $O(N)$, 其中 N 表示 \mathcal{H}_n 中顶点的数量且 $N = 2^{2n}$.

算法 1 \mathcal{H}_n 在 $P_{2^{2n}}$ 中的最优嵌入

输入: 分层立方体网络 \mathcal{H}_n 和线性阵列 $P_{2^{2n}}$

输出: \mathcal{H}_n 在 $P_{2^{2n}}$ 中的最优嵌入 hel

1. for $i=0$ to $2^{2n}-1$ do
2. $p \leftarrow i \% 2^n$;
3. $c \leftarrow (i-p)/2^n$;
4. /* 标记点 u^i , 其中 $u^i \in V(\mathcal{H}_n) *$ /
5. 将点 u^i 标记为 i ;
6. /* 将 \mathcal{H}_n 中的点 u^i 嵌入到 $P_{2^{2n}}$ 中的点 v^j , 其中 $v^j \in V(P_{2^{2n}}) *$ /
7. if $0 \leq c < 2^{n-1}$ then
8. if $0 \leq p < 2^{n-1}$ then
9. $j \leftarrow 2^{n+1}c + p$;
10. else
11. $j \leftarrow 2^{n+1}c - p + 3 \cdot 2^{n-1} - 1$;
12. if $2^{n-1} \leq c < 2^n$ then
13. if $0 \leq p < 2^{n-1}$ then
14. $j \leftarrow -2^{n+1}c + p + 2^{2n+1} - 2^n$;
15. else
16. $j \leftarrow -2^{n+1}c - p + 2^{2n+1} + 2^{n-1} - 1$;
17. 将点 v^j 标记为 j ;
18. 返回嵌入 hel .

3.3 分层立方体网络的线性物理布局

首先给出 NoC 物理布局中轨道的概念. 轨道指一条连续的水平或垂直的线, 将导线布局在上面可以使其不与其他导线重叠. 在线性物理布局中, 所有 \mathcal{H}_n 中的顶点都会被放置在同一个线性阵列上. 为了降低成本, 轨道数应尽可能少.

我们给出了分层立方体网络在 NoC 上的线性物理布局算法,即算法 2。在算法 2 中,首先根据文献[20]中提到的超立方体在线性阵列上的物理布局方法将 \mathcal{H}_n 中同一个簇内的点相互连接起来。对于不同簇之间的顶点,我们根据簇号的间隔 i 来对轨道上的导线进行划分,在第 i 类轨道上,将 j 号簇与 $i+j$ 号簇通过 E_2 中的边相连,同时必须满足此轨道上的导线不会与其他导线重叠,其中 $1 \leq i \leq 2^{2n} - 1$ 。最后用 E_3 中的边将每对互补的簇相连,并将这些导线放置在同一轨道上。根据上述步骤,我们可以保证放置在同一轨道上的导线是不重叠的,于是就获得了 \mathcal{H}_n 的线性物理布局。

图 3 给出了分层立方体网络 \mathcal{H}_2 的线性物理布局,框内的数字代表 \mathcal{H}_2 中的顶点标号,框上的数字代表线性阵列上的点的初始标号。

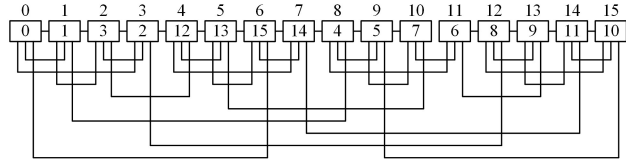


图 3 \mathcal{H}_2 的线性物理布局

Fig. 3 Linear physical layout of \mathcal{H}_2

根据文献[20]中的结论和算法 2,通过计算可以得到定理 3。

定理 3 算法 2 中 \mathcal{H}_n 的线性物理布局需要的轨道数为

$$t_{\text{num}}(\mathcal{H}_n) = 2^{2n-2} + 2^{n-1} + \left\lfloor \frac{2^{n+1}}{3} \right\rfloor + \left\lfloor \frac{n}{2} \right\rfloor.$$

算法 2 \mathcal{H}_n 的线性物理布局

输入: 分层立方体网络 \mathcal{H}_n 和线性阵列 $P_{2^{2n}}$

输出: \mathcal{H}_n 的线性物理布局 h

```

1. /* 将  $\mathcal{H}_n$  中同一个簇中的点相连 */
2. for  $i=0$  to  $2^{n-1}-1$  do
3.   连接点  $\text{hel}^{-1}(2i)$  和点  $\text{hel}^{-1}(2i+1)$ ;
4. if  $n\%2=1$  then
5.    $\text{start} \leftarrow 1$ ;
6.    $\text{flag} \leftarrow 1$ ;
7. else
8.    $\text{start} \leftarrow 2$ ;
9.    $\text{flag} \leftarrow 0$ ;
10. for  $k=0$  to 1 do
11.   for  $j=0$  to  $2^{n-2}-1$  do
12.     连接点  $\text{hel}^{-1}(4j+k)$  和点  $\text{hel}^{-1}(4j+k+2)$ ;
13. for  $i=\text{start}$  to  $n-1$  do
14.   if  $n\%2=\text{flag}$  then
15.     for  $j=0$  to  $2^{n-i-2}-1$  do
16.       连接点  $\text{hel}^{-1}(j \cdot 2^{i+2} + 2^i - 1)$  和点  $\text{hel}^{-1}(j \cdot 2^{i+2} + 2^{i+1} - 1)$ ;
17.   if  $n\%2=(\text{flag}+1)\%2$  then
18.     for  $k=0$  to  $2^i-1$  do
19.       if  $0 \leq k \leq 2^{i-1}-2$  then
20.         for  $j=0$  to  $2^{n-i-1}-1$  do
21.           连接点  $\text{hel}^{-1}(j \cdot 2^{i+1} + k)$  和点  $\text{hel}^{-1}(j \cdot 2^{i+1} + 2^i + k)$ ;
22.           连接点  $\text{hel}^{-1}(j \cdot 2^{i+1} + 2^i + k + 1)$  和点  $\text{hel}^{-1}(j \cdot 2^{i+1} + 3 \cdot 2^{i-1} + k + 1)$ ;
23.         if  $2^{i-1}-1 \leq k \leq 2^i-2^{i-1}$  then

```

```

24.           for  $j=0$  to  $2^{n-i-1}-1$  do
25.             连接点  $\text{hel}^{-1}(j \cdot 2^{i+1} + k)$  和点  $\text{hel}^{-1}(j \cdot 2^{i+1} + 2^i + k)$ ;
26.           if  $2^i-2^{i-1}+1 \leq k \leq 2^i-1$  then
27.             for  $j=0$  to  $2^{n-i-1}-1$  do
28.               连接点  $\text{hel}^{-1}(j \cdot 2^{i+1} - 2^{i-1} + k - 1)$  和点  $\text{hel}^{-1}(j \cdot 2^{i+1} + k - 1)$ ;
29.               连接点  $\text{hel}^{-1}(j \cdot 2^{i+1} + k)$  和点  $\text{hel}^{-1}(j \cdot 2^{i+1} + 2^i + k)$ ;
30. for  $k=0$  to  $2^{n-1}-1$  do
31.   if  $0 \leq k \leq 2^{n-1}-2^{n-2}$  then
32.     连接点  $\text{hel}^{-1}(k)$  和点  $\text{hel}^{-1}(2^n-k-1)$ ;
33.   else
34.     连接点  $\text{hel}^{-1}(k-1-2^{n-2})$  和点  $\text{hel}^{-1}(k-1)$ ;
35.     连接点  $\text{hel}^{-1}(k)$  和点  $\text{hel}^{-1}(2^n-k-1)$ ;
36.     连接点  $\text{hel}^{-1}(2^n-k)$  和点  $\text{hel}^{-1}(2^n-k+2^{n-2})$ ;
37. /* 将  $\mathcal{H}_n$  中不同簇中的点相连 */
38. for  $i=1$  to  $2^{2n}-1$  do
39.   for  $k=0$  to  $i$  do
40.      $j \leftarrow 0$ ;
41.     while  $j+k+i < 2^n$  do
42.       连接点  $u$  和点  $v$ , 其中  $c(u)=p(v)=j+k$ ,  $p(u)=c(v)=j+k+i$ ;
43.        $j \leftarrow j+i+1$ ;
44. for  $i=0$  to  $2^{2n}-1$  do
45.   连接点  $u$  和点  $v$ , 其中  $c(u)=p(v)=2i$ ,  $p(u)=c(v)=2i+1$ ;
46. 返回物理布局  $h$ 。

```

4 对比实验

本文通过 Python 环境来进行分层立方体网络嵌入线性阵列的对比实验,此处 Python 解释器的版本号是 3.9。

在互连网络中,链路的拥塞度过大会直接导致信息通信的高延迟,因此,我们将线长作为主要指标来衡量所有链路的拥塞情况。在实验中,我们首先给出了 3 种嵌入方案。一种是本文提出的嵌入 hel ,一种是随机嵌入 $rad: V(\mathcal{H}_n) \rightarrow \{0, 1, \dots, 2^{2n}-1\}$,另一种是词典嵌入 $lex: V(\mathcal{H}_n) \rightarrow \{0, 1, \dots, 2^{2n}-1\}$,其中 $lex(u)=u$ 。

在图 4 中,我们使用了 3 种嵌入方案将不同维度的 \mathcal{H}_n 嵌入其对应的线性阵列中,并且对比了它们的线长。对于相同数量的顶点而言,更大的线长意味着更大的布局成本和更长的消息队列延迟。很明显, rad 嵌入下的线长比 lex 和 hel 嵌入下的线长大多,究其原因,主要是嵌入 rad 下的点被随机映射,导致链路的拥塞度变得非常大,从而增加了嵌入的线长。

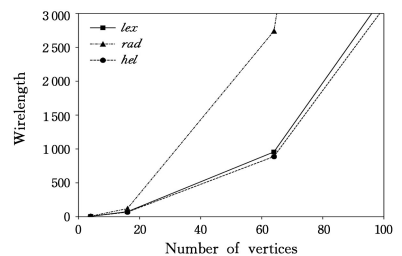


图 4 3 种嵌入方式下的线长

Fig. 4 Wirelengths of three embedding schemes

为了更清晰地表示嵌入 lex 和嵌入 hel 下线长的差异,表 1 列出了详细对比情况,其中 n 表示分层立方体网络 \mathcal{H}_n 的维度, N 表示 \mathcal{H}_n 中顶点的数量。从表 1 中可看出,当分层立方体网络的维度增加时,嵌入 lex 和嵌入 hel 下线长之差剧烈增长,并且嵌入 hel 始终比嵌入 lex 具有更小的线长。这就意味着当 \mathcal{H}_n 的顶点数量足够多时,使用本文提出的嵌入 hel 可以比嵌入 lex 节省更多的布线布局成本。

接着,我们将引入其他 4 个用于衡量嵌入质量的指标,即延展率(Dilation)、膨胀率(Expansion)、拥塞度(Congestion)和负载(Load)。给出图 G 在图 H 上的嵌入为 $f:G \rightarrow H$ 。嵌入 f 的延展率为 $dil(G, H, f) = \max\{d(f(u), f(v)) \mid (u, v) \in E(G)\}$,其中 $d(f(u), f(v))$ 表示点 $f(u)$ 和 $f(v)$ 之间的距离,这个指标常用于测量通信的有效性,更小的延展率意味着更短的通信延迟。嵌入 f 的膨胀率为 $exp(G, H, f) = |V(G)|/|V(H)|$,这个指标常用于测量处理器利用率,更大的膨胀率意味着更高的处理器利用率。嵌入 f 的拥塞度为 $cong(G, H, f) = \max\{EC_f(e) \mid e \in E(H)\}$,这个指标常用于测量信息排队延迟,更小的拥塞度意味着更短的排队延迟。嵌入 f 的负载为 $load(G, H, f) = \max\{load(v) \mid v = f(u), u \in V(G)\}$,其中 $load(v)$ 指图 G 映射到图 H 的 v 上的点数,这个指标常用于测量任务处理延迟。

表 1 嵌入 hel 和 lex 的线长之差

Table 1 Difference between wirelengths of embedding hel and lex

n	N	$WL_{lex} - WL_{hel}$
1	4	0
2	16	6
3	64	68
4	256	616
5	1024	5200
6	4096	42656
7	16384	345408
8	65536	2779776

表 2 列出了嵌入 hel 和 lex 下 4 个指标的对比结果,其中 n 表示分层立方体网络 \mathcal{H}_n 的维度, N 表示 \mathcal{H}_n 中顶点的数量。很明显,嵌入 hel 和 lex 具有相同的膨胀率和负载,且都为 1。当 $n > 1$ 时,嵌入 hel 下的延展率和拥塞度总是小于嵌入 lex 下的,也就是说,本文提出的嵌入 hel 比嵌入 lex 具有更短的通信延迟和排队延迟。为了更直观地反映差距,我们给出了对应的折线图,如图 5 所示,其中图 5(a)是嵌入 hel 和嵌入 lex 下的延展率对比,图 5(b)是嵌入 hel 和嵌入 lex 下的拥塞度对比,图 5(c)是嵌入 hel 和嵌入 lex 下的延展率和拥塞度之差。

表 2 嵌入 hel 和 lex 的线长之差

Table 2 Difference of embedding hel and lex with four metrics

n	N	Dilation		Congestion		Expansion		load	
		hel	lex	hel	lex	hel	lex	hel	lex
1	4	3	3	2	2	1	1	1	1
2	16	9	15	6	7	1	1	1	1
3	64	49	63	20	22	1	1	1	1
4	256	225	255	73	77	1	1	1	1
5	1024	961	1023	274	282	1	1	1	1
6	4096	3969	4095	1061	1077	1	1	1	1

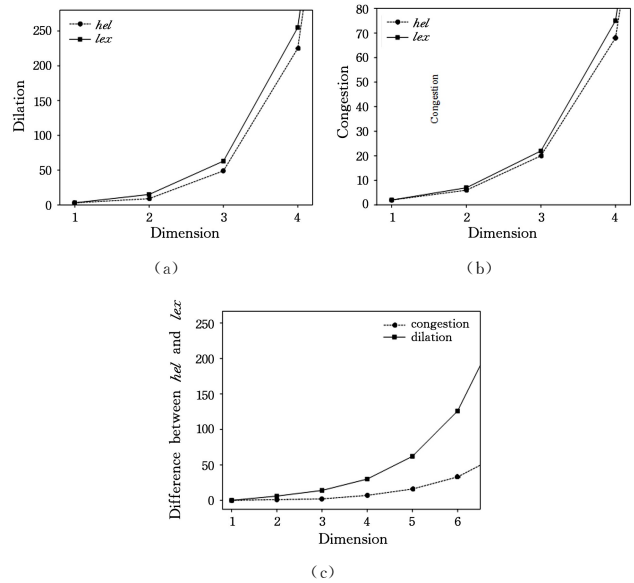


图 5 嵌入 hel 和嵌入 lex 下的延展率和拥塞度对比

Fig. 5 Comparison of dilation and congestion with embedding hel and lex

通过上述几组对比实验,我们可以得出如下结论:在分层立方体网络嵌入线性阵列的情况下,本文提出的嵌入方案 hel 比词典嵌入 lex 和随机嵌入 rad 具有更好的通信性能和更低的嵌入成本。

结束语 本文解决了分层立方体网络在 NoC 线性阵列中的最优嵌入问题。首先研究了分层立方体网络的最优集,在此基础上提出了 n 维分层立方体网络 \mathcal{H}_n 在线性阵列 P_{2^n} 中的一种嵌入方案 hel ,并证明了嵌入 hel 下的线长是最小的,即 hel 是最优嵌入。然后给出了嵌入 hel 下线长的精确值和嵌入算法。此外,还提出了分层立方体网络在 NoC 上的线性物理布局算法。最后进行了对比实验来评估本文提出的嵌入 hel 的性能和优势。未来,我们还可以研究分层立方体网络在其他互连网络中(如网格、环绕等)的嵌入。

参考文献

- [1] WANG X J, SHI F, ZHANG H. KLSAT: An application mapping algorithm based on Kernighan-Lin partition and simulated annealing for a specific WK-recursive NoC architecture[C]// International Conference on Network and Parallel Computing. Springer International Publishing, 2019: 31-42.
- [2] MOSTAFA A, TURKI F A. Topological properties of hierarchical interconnection networks: a review and comparison[J]. Journal of Electrical and Computer Engineering, 2011, 2011: 1-12.
- [3] GHOSE K, DESAI K R. Hierarchical cubic networks[J]. IEEE Transactions on Parallel and Distributed Systems, 1995, 6(4): 427-435.
- [4] ZHOU S M, SONG S L, YANG X Y, et al. On conditional fault tolerance and diagnosability of hierarchical cubic networks[J]. Theoretical Computer Science, 2016, 609: 421-433.
- [5] LIU Z, FAN J X, ZHOU J Y, et al. Fault-tolerant embedding of complete binary trees in locally twisted cubes[J]. Journal of Parallel and Distributed Computing, 2017, 101: 69-78.

- [6] BOSSARD A, KANEKO K. Node-to-set disjoint-path routing in hierarchical cubic networks [J]. *Computer Journal*, 2012, 55(12):1440-1446.
- [7] LIU H Q, ZHANG S Z, LI D. On g-extra conditional diagnosability of hierarchical cubic networks[J]. *Theoretical Computer Science*, 2019, 790:66-79.
- [8] LI X J, LIU M, YAN Z, et al. On conditional fault tolerance of hierarchical cubic networks[J]. *Theoretical Computer Science*, 2019, 761:1-6.
- [9] ZHAO S L, HAO R X, WU J. The generalized 4-connectivity of hierarchical cubic networks[J]. *Discrete Applied Mathematics*, 2021, 289:194-206.
- [10] FAN W B, FAN J X, LIN C K, et al. Embedding exchanged hypercubes into rings and ladders[C]// *International Conference on Algorithms and Architectures for Parallel Processing*. Springer International Publishing, 2018:3-17.
- [11] WANG W, QIAO L, TANG Z Z. Survey on the Networks-on-Chip interconnection topologies [J]. *Computer Science*, 2011, 38(10):1-5.
- [12] MANUEL P, RAJASINGH I, RAJAN B, et al. Exact wirelength of hypercubes on a grid [J]. *Discrete Applied Mathematics*, 2009, 157(7):1486-1495.
- [13] AROCKIARAJ M, ABRAHAM J, QUADRAS J, et al. Linear layout of locally twisted cubes[J]. *International Journal of Computer Mathematics*, 2017, 94(1):56-65.
- [14] SHALINI A J, ABRAHAM J, AROCKIARAJ M. A linear time algorithm for embedding locally twisted cube into grid network to optimize the layout[J]. *Discrete Applied Mathematics*, 2020, 286:10-18.
- [15] FAN W B, FAN J X, LIN C K, et al. Optimally embedding 3-ary n-cubes into grids[J]. *Journal of Computer Science and Technology*, 2019, 34(2):372-387.
- [16] FAN W B, FAN J X, LIN C K, et al. An efficient algorithm for embedding exchanged hypercubes into grids[J]. *Journal of Supercomputing*, 2019, 75(2):783-807.
- [17] XIA J J, WANG Y, FAN J X, et al. Embedding augmented cubes into grid networks for minimum wirelength[C]// *International Conference on Algorithms and Architectures for Parallel Processing*. Springer International Publishing, 2020:47-61.
- [18] BEZRUKOV S L, CHAVEZ J D, HARPER L H, et al. Embedding of hypercubes into grids[C]// *International Symposium on Mathematical Foundations of Computer Science*. Springer, 1998:693-701.
- [19] STALIN MARY R, RAJASINGH I. Embedding of complete graphs and cycles into grids with holes[J]. *Procedia Computer Science*, 2020, 172:115-121.
- [20] PATEL A, KUSALIK A J, MCCROSKY C. Area-efficient VLSI layouts for binary hypercubes[J]. *IEEE Transactions on Computers*, 2000, 49(2):160-169.



GUO Ruyan, born in 1998, postgraduate. Her main research interests include parallel and distributed systems and graph embedding.



WANG Yan, born in 1977, Ph.D, associate professor. Her main research interests include parallel and distributed systems, interconnection architectures, social networks, algorithms and graph theory.

(责任编辑:何杨)