并行机互联网络拓扑结构描述语言——TOD*

李强国[1]，于　洋[2]，何　凯[2]，李　涛[2]，杨愚鲁[2]
(1. 太极计算机股份有限公司，北京 100083； 2. 南开大学 计算机科学与技术系，天津 300071)

摘　要：并行机仿真需要描述并行机互联网络拓扑结构。提出了通用的面向并行机静态互联网络拓扑结构的描述语言 TOD(TOpology Description)，为仿真系统提供了有力的拓扑结构描述工具。TOD 以图的运算为基础，通过基础运算、扩展运算和带条件的运算可以简便地描述主流的并行机拓扑结构，基本运算的使用保证了 TOD 能够描述任意结构。

关键词：互联网络；拓扑结构；图的运算
中图法分类号：TP312　　**文献标识码**：A　　**文章编号**：1001-3695(2006)03-0076-03

Topology Description Language for Interconnection Network in Parallel Computer: TOD

LI Qiang-guo[1]，YU Yang[2]，HE Kai[2]，LI Tao[2]，YANG Yu-lu[2]
(*1. Taiji Computer Co., Ltd., Beijing* 100083, *China*; *2. Dept. of Computer Science & Technology, Nankai University, Tianjin* 300071, *China*)

Abstract: Describing the topologies of various kinds is needed in a parallel computer simulator. A general topology description language: TOD(TOpology Description), which is used to describe the topology of the static interconnection network in parallel computers, is proposed in this paper. It becomes a right-hand tool for a parallel computer simulator. TOD is based upon the graph operations. It can flexibly and easily describe the prevalent topologies by using the base operations, the extended operations and the operations with conditions. The base operations ensure that TOD can describe any topology.

Key words: Interconnection Network; Topology; Graph Operation

并行机系统结构的研究需要构建目标系统的互联网络模型，作为并行机系统结构研究一项重要手段的仿真需要完成并行机互联网络拓扑结构的描述。大多数并行机仿真系统[1~3]只是建立指定点之间的连接，由用户根据具体结构分别描述，描述复杂。本文提出一种通用的描述并行机静态无向互联网络拓扑结构的语言 TOD。根据并行机互联网络的拓扑描述要求，TOD 语言使用图运算以及带条件的运算可以完成各种结构的描述。典型描述实例表明，TOD 语言的描述方法可以简便地构成当前常用的并行机拓扑结构。

1　互联网络的拓扑描述要求

通用并行机仿真系统的拓扑描述接口需要描述并行机正在与可能使用的各类互联网络的拓扑结构。经典结构 Hypercube 和 Mesh/Torus 具有单一的强规律性和中心对称的结构特点，是第一代和第二代并行机中使用的拓扑结构。由于自身的优势(Mesh，Torus 的低节点度和 Hypercube 的小直径)，它们至今仍被应用于一些并行机中。具有对经典结构继承性的拓扑结构是目前新型拓扑结构的一个发展方向：规律性的经典结构与其他结构的嵌套结构(如 CCC[4]，CCT Cube[5]) 具有多重规律；基于 Mesh/Torus 结构的变形根据其变形方式分别具有单一规律性(如 *k*-ary *n*-cube) 或复合规律性(如 SNT[6]，RDT[7])，这类结构继承了经典结构的中心对称特点；结合低节点度和低直径的、来源于拓扑学的结构(DeBruijn[8]，μ-star[9]) 是目前新型拓扑结构的另一个发展方向，这类结构具有明显的规律性；包括轴对称结构、无对称特点的结构在内的其他结构在未来的并行机研究中可能被采用。

据此本文提出了拓扑描述语言 TOD，并确定了该语言对并行机互联网络拓扑结构的描述范围：能够简便地描述经典结构和依据目前研究方向发展的拓扑结构，同时能够描述其他可能出现的结构。

2　TOD 语言

TOD 是一种通用的描述并行机静态无向互联网络拓扑结构的语言，它以图运算[10]为核心完成拓扑结构的描述。首先创建作为基础的点、线或者环，然后对点、线、环或其他指定的图进行所需的运算，直至生成预期拓扑结构。并、交、和、差、环和[10]等基本运算是 TOD 中运算的基础，保证了 TOD 能够描述各种拓扑结构；各种常用的积运算可以产生具有强规律性的图，其描述复杂度与拓扑结构的规模无关，方便于描述 Mesh，Torus，Hypercube 等经典拓扑结构；经典结构改造后得到的新拓扑结构，其规律性源于多种规律的嵌套或叠加，使用 TOD 定义的扩展积运算以及步长、有向、替换等扩展运算条件可以产生具有嵌套或叠加规律的图。

2.1　图运算

TOD 语言的核心是图运算，包括图的基本运算、常用积运

收稿日期：2005-03-11；修返日期：2005-06-08
基金项目：天津市自然科学基金资助项目(043600111)

算和扩展积运算。通过对指定的图进行运算, TOD 可以完成对各类拓扑结构的描述。

基本图运算(表 1) 保证了 TOD 通用于各类拓扑结构, 特别是非规律性结构的描述。对于两个无孤立点的图 $G1(V1, E1)$ 和 $G2(V2, E2)$, 可以进行如表 1 所示的基本图运算得到新的图 $G3(V3, E3)$。

图的积运算生成规律性强、对称的拓扑结构, 可以用来描述经典的拓扑结构。积运算一般需要两步进行: 生成顶点集和建立连接。在图的各种积运算中, 生成顶点集的方式是一样的。设有图 $G1(V1, E1)$ 和图 $G2(V2, E2)$, 且 $V1$, $V2$ 的任意两点不相邻, 设 $G3$ 为 $G1$ 和 $G2$ 积运算的结果图, $G3$ 的顶点集 $V3$ 为 $V1 \times V2$(参与运算的两个图的顶点集之间的笛卡尔积)。对于不同的积运算, 新的顶点集中的连接规则是不同的, 表 1 列出了常用的积运算的连接规则。

常用积运算能够描述具有规律性强、对称的拓扑结构, 描述复杂度与该结构规模无关, 但是对于经典结构的变形所具有的弱规律性难以描述。为此, TOD 减弱了常用积运算的连接条件, 定义了四个具有原子特性的扩展积运算, 更加灵活地生成各种不同的拓扑结构。扩展积运算的顶点集生成方式与常用积运算相同, 只是连接规则不同, 表 1 列出了 TOD 定义的扩展积运算连接规则。

表 1　TOD 运算

	名称	符号表示	运算说明
基本图运算	并	$G1 \cup G2$	$V3 = V1 \cup V2$, $E3 = E1 \cup E2$
	交	$G1 \cap G2$	$V3 = V1 \cap V2$, $E3 = E1 \cap E2$
	差	$G1 - G2$	$G3 = G1 - (G1 \cap G2)$
	环和	$G1 \quad G2$	$G3 = (G1 \cup G2) - (G1 \cap G2)$
	加	$G1 + G2$	$u \in V1$, $v \in V2$, $V3 = V1 \cup V2$, $E3 = E1 \cup E2 \cup E(uv)$
常用积运算	笛卡尔积	$G1 \square G2$	$(u1 = v1$ 且 $u2$ adj $v2)$ 或 $(u1$ adj $v1$ 且 $u2 = v2)$
	字典积	$G1[G2]$	$(u1$ adj $v1)$ 或 $(u1 = v1$ 且 $u2$ adj $v2)$
	绝对积	$G1 \cdot G2$	$(u1$ adj $v1$ 且 $u2$ adj $v2)$
	强积	$G1 \boxtimes G2$	$(u1 = v1$ 且 $u2$ adj $v2)$ 或 $(u2 = v2$ 且 $u1$ adj $v1)$ 或 $(u1$ adj $v1$ 且 $u2$ adj $v2)$
TOD 扩展积运算	左等积	$G1 <= G2$	$u1 = v1$
	右等积	$G1 => G2$	$u2 = v2$
	左邻积	$G1 <- G2$	$u1$ adj $v1$
	右邻积	$G1 -> G2$	$u2$ adj $v2$

2.2　扩展运算条件

图的积运算可以方便地描述具有单一规律的拓扑结构, 图的基本运算可以描述叠加了多种规律的拓扑结构, 但是难以描述具有嵌套规律的拓扑结构。为此 TOD 语言定义了步长、运算有向和替换三个扩展运算条件, 方便描述具有多规律嵌套型拓扑结构。

2.2.1　步长(Step)

步长是指位于同一条线上的、等距的、具有连接的点之间的距离。Step $= i(i = 1, 2, 3, \cdots)$ 表示位于一条线上距离为 i 的任意两点之间具有连接。

确定图中各点之间连接是图运算的一个关键步骤, 使用步长可以方便建立点之间规律性的连接。设定步长不是在原图中间隔选取一些点组成新图, 而是在原图的基础上建立点之间新的连接。图 1 中表示了 Step = 1(直线) 和 Step = 2(弧线) 时位于同一条线上的点之间的连接。

在并行机经典结构中, 将 Mesh/Torus 间隔一点或多点建立规律性连接, 是构造具有更小网络直径的拓扑结构的常用方法, 在 TOD 中可以通过设定步长值方便地描述这类改造结构。

2.2.2　替换

用图 G 替换另一个图 H 中的一个点, G 中的每一个点对应 H 中被替换点的一个连接。如果 H 中被替换点具有 $n(n \geq 2)$ 个连接, 那么应该用包含 n 个点的图 G 进行替换。作为一个整体, G 中的每个点增加一个连接替代 H 中被替换点的一个连接, 用于替换的图 G 可以是任意的, 如线、环等。结合两种拓扑类型的拓扑结构在并行机互联网络设计中是常见的, 替换对于描述这种类型的拓扑非常适合。存在替换的图进行积运算时, 需要在标准的积运算规则上进行调整, 如图 2 所示。

(1) 生成新图点集的运算规则。在标准的积运算中, $G1$ 中的节点 $A1$ 和 $G2$ 中的节点 $B1$ 在进行积运算时会生成一个新的点$(A1B1)$; 具有替换的图在运算时生成两个点 $A1$ 和 $B1$, 并且这两个点将组成线。

(2) 建立连接的运算规则。在对进行运算的图进行连接关系判断时, 替换点作为一个点确定连接关系; 在建立新图的连接时, 将替换点作为分离的点建立各自的连接。

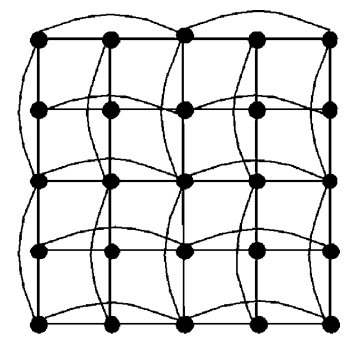


图 1　步长为 1, 2 的 Mesh

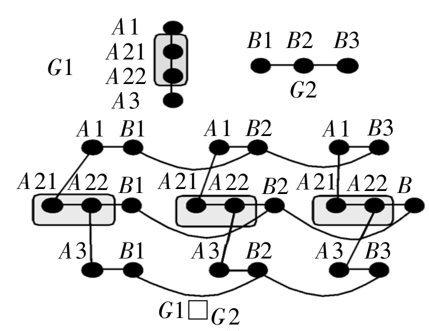
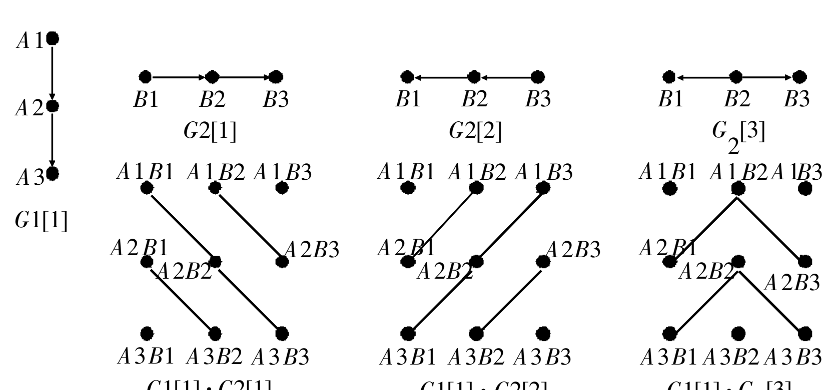


图 2　存在替换的图的笛卡尔积运算

2.2.3　运算有向

运算有向是指在进行积运算时将参加运算的图看作有向图, 其中的连接是单向的, 最终运算结果是无向图, 只在运算过程中存在方向。这一运算扩展条件的引进打破了图的积运算只能产生对称图的限制, 积运算的结果将不再具有对称性, 使 TOD 可以对具有非对称规律的拓扑结构进行描述。图 3 为无向图 $G1$, $G2$ 不同的运算方向条件下的绝对积运算结果。



其中: $G1[1]$表示无向图 $G1$ 的运算方向; $G2[1]$, $G2[2]$, $G_2[3]$分别表示无向图 $G2$ 的三种不同运算方向

图 3　图的有向绝对积运算

2.3　TOD 描述特点及语言编译器

在 TOD 中, 使用各类常用积运算可以简便地描述具有强规律性的经典结构和来自拓扑学的结构; 使用扩展积运算和带条件的积运算可以简便地描述具有弱规律性的 Mesh/Torus 变形结构; 积运算与基本运算的结合可以简便地描述具有嵌套规律的结构; 基本运算的使用保证可以描述任意的非规律性结构。

由于 TOD 的描述过程主要是运算过程, 因此描述复杂度由目标系统互联网络拓扑结构的规律性强度决定, 而与其规模

无关。

TOD 语言编译器采用解释执行的方式，使用对应于 TOD 语言各运算函数的函数库来支持语言中的所有运算，根据词法分析和语法语义分析的结果调用函数库中相应的函数进行运算，构成期望的拓扑结构。

3 TOD 语言描述方法及描述实例

TOD 语言可以方便地描述常用的并行机互联网络拓扑结构。同时对于基本图运算的使用，使得 TOD 可以描述各种非规律性拓扑结构，对于使用各种拓扑结构的互联网络具有描述的通用性。本节通过使用 TOD 描述一些具有代表性的拓扑结构，介绍 TOD 语言的使用方法。

3.1 直接应用各类积运算: *k*-ary *n*-cube

将 Torus 结构推广到多维，形成 *k*-ary *n*-cube，它是一般化了的 Torus 结构，由 *k* 的 *n* 次方个节点组成，每个点的编号由该点的 *n* 元 *k* 进制坐标向量表示。它是一个理论模型，可以描述环网(Torus) 推广到任意维后的拓扑结构。由于 *k*-ary *n*-cube 的强规律性，直接使用笛卡尔积进行递归运算即可得到该结构的描述。创建 *k*-ary *n*-cube 的 TOD 语言描述步骤如下:

(1) 创建包含 *k* 个点的无向环 *R*;

(2) 对 *n* 个无向环 *R* 进行笛卡尔积运算($R1 \square R2 \square \cdots \square Rn$) ，生成 *k*-ary *n*-cube。

3.2 应用基于扩展条件的积运算: CCC

CCC 是结合两种类型结构而成的典型拓扑结构，它将 *k* 维 Hypercube 的每一个节点用包含 *k* 个点的环代替，使得结构中每个点的度固定为 3。使用包含替换的积运算可以简洁地生成这种结构。创建 *k* 阶 CCC 的 TOD 语言描述步骤如下:

(1) 创建包含两个点的无向线 *L* 和包含 *k* 个点的无向环 *R*;

(2) 使用 *R* 替换 *L* 中的两个点，得到包含替换的图 *L*(*R*) ;

(3) 对包含替换的线 *L*(*R*) 和(*k* - 1) 个无向线 *L* 进行笛卡尔积运算($L(R) \square L1 \square L2 \square \cdots \square L(k-1)$) ，生成 *k* 阶 CCC。

3.3 组合/拆分多个积运算结果: RDT(2,4)

RDT[6] 是 Torus 结构的扩展，该结构由一个二维的 Torus 结构和在其上依照对角线方向递归构造的多个上层二维 Torus 叠加而成。这一结构具有理想的网络性能: 直径和平均距离远小于同尺寸的 Torus 结构，容易映射 Mesh/Torus 结构。

构造上层的 Torus 要设置其下一层 Torus 的步长。通过由步长限制的笛卡尔积依次生成各级 Torus，然后通过并运算结合不同尺寸的 Torus 构成 RDT。创建包含 $n \times n$ 个点的 RDT (2, 4) 的 TOD 语言描述步骤:

(1) 创建包含 *n* 个点的无向环 *R*;

(2) 对两个无向环进行笛卡尔积运算($R1 \square R2$) ，生成第 0 阶 Torus ($T0$) ;

(3) 将两个无向环步长设置为 2，进行绝对积运算($R1 \times R2$ |Step-$R1$ = 2, Step-$R2$ = 2) ，生成第 1 阶 Torus($T1$) ;

(4) 将两个无向环步长设置为 8，进行笛卡尔积运算($R1 \square R2$ | Step-$R1$ = 8, Step-$R2$ = 8) ，生成第 2 阶 Torus($T2$) ;

(5) 将两个无向环步长设置为 16，进行绝对积运算($R1 \times R2$ |Step-$R1$ = 16, Step-$R2$ = 16) ，生成第 3 阶 Torus($T3$) ;

(6) 将两个无向环步长设置为 64，进行笛卡尔积运算($R1 \square R2$ | Step-$R1$ = 64, Step-$R2$ = 64) ，生成第 4 阶 Torus($T4$) ;

(7) 对第 0 阶到第 4 阶的 Torus 进行并运算($T0 \cup T1 \cup T2 \cup T3 \cup T4$) ，生成包含 $n \times n$ 个点的 RDT(2, 4) 。

描述 RDT 的过程显示了利用 TOD 语言描述拓扑结构的一般思路，即将描述的目标拓扑拆分为几个规律的部分，根据各部分的规律，利用相应的图的积运算分别生成，然后用并、交等运算将各部分 "组合 "起来。

4 结论

在并行机系统结构研究中仿真是重要研究手段，通用拓扑描述接口为仿真工作提供了有力的工具。常用的互联网络拓扑结构大多具有全局或局部规律性。目前存在的大部分仿真系统中只提供简单的两点间的连接，由用户根据需要进行描述。

本文提出了一种用于描述静态互联网络的语言 TOD，使用图运算及其运算扩展条件实现对各种拓扑结构的描述。TOD 语言通过设置扩展条件使用常用积运算和扩展积运算，可以简洁地描述规律的拓扑图形。拓扑结构所具有的规律性越强，TOD 语言越方便描述，描述的复杂度与网络的规模无关。基本图运算的使用使 TOD 能够描述非规律性结构，对于任意拓扑结构具有通用性。

参考文献:

[1] elen Davis, Stephen R Goldschmidt, John Hennessy. Multiprocessor Simulation and Tracing Using Tango[C] . Proceedings of International Conference on Parallel Processing, 1991. II-99-II-107.

[2] Shibamura H, Kuga M, Sueyoshi T. INSIGHT: An Interconnection Network Simulator for Massively Parallel Computers[C] . IEEE Region 10 's 9th Annual International Conference, 1994. 77-81.

[3] Brewer E A, Dellarocas C N, *et al*. PROTEUS: A High Performance Parallel-architecture Simulator[R] . Technical Report MIT/LCS/TR-516, Massachusetts Institute of Technology, 1991.

[4] F P Preparata, J Vuillemin. The Cube-Connected Cycles: A Versatile Network for Parallel Computation[J] . Comm. ACM, 1981, 24(5) : 300-309.

[5] T Ishikawa. CCTcube: A Highly Parallel Processor Network Featuring Short Diameter and Few Links[J] . Transactions of the Institute of Electronics, Information and Communication Engineers, 1990, J73-0-I(6) .

[6] K Iwasaki, C Iseli, Y Sato. A Proposal of Uniform Network for VlSI Massively Parallel Computers[J] . Transactions of the Institute of Electronics, Information and Communication Engineers, 1992, J75-D-I (8) : 583-591.

[7] Y Yang, H Amano, H Shibamura, *et al*. Recursive Diagonal Torus: An Interconnection Network for Massively Parallel Computers[C] . Proc. the 5th IEEE Symposium on Parallel and Distributed Processing, 1993. 591-594.

[8] M R Samatham, D K Pladhan. The De Bruijn Multiprocessor Network: Versatile Parallel Processing and Sorting Network for VLSI[J] . IEEE Trans. Comput., 1989, 38(4) : 567-581.

[9] S B Akers, P Harel, B Krishnamurthy. The Star Graph: An Attractive Alternative to the *n*-cube[C] . ICPP '87, 1987. 393-400.

[10] Harary F. Graph Theory[M] . MA: Addison-Wesley, 1994.

作者简介:

李强国(1975-) ，男，硕士，主要研究方向为并行机系统结构; 于洋(1979-) ，女，博士研究生，主要研究方向为并行机系统结构、网格计算; 何凯(1973-) ，男，博士研究生，主要研究方向为网格计算; 李涛(1977-) ，男，博士研究生，主要研究方向为并行与分布计算、高性能网络; 杨愚鲁(1961-) ，男，教授，博导，主要研究领域为并行计算机系统结构、网格计算、可重构系统结构。