



(12) 发明专利

(10) 授权公告号 CN 114363241 B

(45) 授权公告日 2023. 02. 28

(21) 申请号 202210015846.7

(22) 申请日 2019.05.15

(65) 同一申请的已公布的文献号
申请公布号 CN 114363241 A

(43) 申请公布日 2022.04.15

(62) 分案原申请数据
201910408017.3 2019.05.15

(73) 专利权人 清华大学
地址 100084 北京市海淀区清华园北京
100084-82信箱

(72) 发明人 向东

(74) 专利代理机构 北京方圆嘉禾知识产权代理
有限公司 11385
专利代理师 王月松

(51) Int.Cl.

H04L 45/12 (2022.01)

H04L 45/00 (2022.01)

H04L 49/90 (2022.01)

H04L 49/00 (2022.01)

(56) 对比文件

CN 107959643 A, 2018.04.24

CN 104539536 A, 2015.04.22

CN 102170402 A, 2011.08.31

CN 109561034 A, 2019.04.02

US 2016344618 A1, 2016.11.24

审查员 张建军

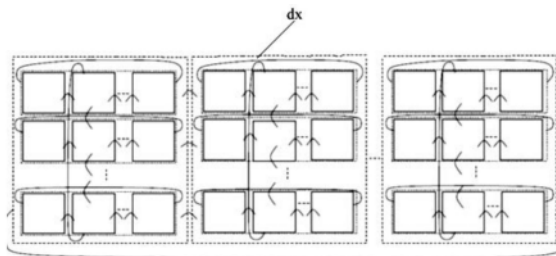
权利要求书3页 说明书11页 附图5页

(54) 发明名称

高维度Torus网络架构及自适应路由方法

(57) 摘要

本发明提供一种高维度Torus网络架构及自适应路由方法,高维度Torus网络架构包括:由低维到高维依次排列的多单元维度;多单元维度包括 k_n 个计算单元, k_n 个计算单元的节点坐标分别为 k_0, k_1, \dots, k_{n-1} ; k_n 个计算单元按照 $0, 1, \dots, k_n - 1$ 顺次连接形成闭环; k_n 为大于或等于2的正整数,多单元维度在进行路由选择时,所传输的数据包的当前节点坐标与目标节点坐标不相等时,对应 $i=0$ 到 $n-1$:若 $d_i - c_i > k_i/2$ 或 $-k_i/2 \leq d_i - c_i < 0$,则向第 i 维的负方向选择下一计算单元作为路由节点;若 $0 < d_i - c_i < k_i/2$ 或 $d_i - c_i < -k_i/2$,则沿着第 i 维的正方向选择下一计算单元作为路由节点,提高了在Torus网络中路由方法的效率,并提高了等分带宽。



1. 一种高维度Torus网络架构,其特征在于,所述高维度Torus网络架构包括:

由低维到高维依次排列的多单元维度;其中,多单元维度包括 k_n 个计算单元, k_n 个计算单元的节点坐标分别为 k_0, k_1, \dots, k_{n-1} ; k_n 个计算单元按照 $0, 1, \dots, k_i - 1$ 顺次连接形成闭环; k_n 为大于或等于2的正整数, i 代表维度的数量;

所述多单元维度在进行路由选择时,所传输的数据包的当前节点坐标 c_i 与目标节点坐标 d_i 不相等时,对应 $i=0$ 到 $n-1$:

若 $d_i - c_i > k_i/2$ 或 $-k_i/2 \leq d_i - c_i < 0$,则向第 i 维的负方向选择下一计算单元作为路由节点;

若 $0 < d_i - c_i < k_i/2$ 或 $d_i - c_i < -k_i/2$,则沿着第 i 维的正方向选择下一计算单元作为路由节点;

其中,所述下一计算单元选择路由节点时满足流控技术设置的预设安全要求;预设安全要求为:若所述下一计算单元包括 f 个空闲虚拟通道以及 s 个安全虚拟通道,则:若 $f > 1$,或 $f = 1$ 且 $s \geq 1$,或 $f = 1$ 且 $s = 0$ 且从当前计算单元到下一计算单元遵循 p^* -cube方法,则所述下一计算单元作为路由节点;

p^* -cube路由方法是针对超立方网络的一种部分适应行路由方法,对于目标节点 $(d_0, d_1, \dots, d_{n-1})$ 和当前节点 $(c_0, c_1, \dots, c_{n-1})$ 均只相差一跳时,采用 p^* -cube路由方法;

将当前节点和目标节点间标号不同的维度划分成两个子集 U_1 及 U_2 ,初始 U_1 和 U_2 设置为空集 Φ ;对所有维度 $i=0$ 到 $i=n-1$,如果 $c_i \neq d_i$,且该维度的下一跳步是 ch_i+ ,则 $U_1 \leftarrow U_1 \cup \{i\}$;否则,该维度的下一跳跳步是 ch_i- , $U_2 \leftarrow U_2 \cup \{i\}$;

p^* -cube路由方法如下:从当前节点的坐标 $C(c_0, c_1, \dots, c_{n-1})$ 到目的节点坐标 $D(d_0, d_1, \dots, d_{n-1})$ 的数据包首先适应性地跨越 U_2 中的所有维度,当 U_2 所有维度走完,适应性的选择 U_1 中的维度传输数据包。

2. 根据权利要求1所述的高维度Torus网络架构,其特征在于,所述高维度Torus网络架构的维度大于等于2。

3. 一种自适应路由方法,应用于如权利要求1至2中任一项所述的高维度Torus网络架构,其特征在于,所述自适应路由方法包括:

步骤一,在 n 维度Torus网络架构中,设数据包路由时,当前节点的坐标为 $C(c_0, c_1, \dots, c_i, \dots, c_{n-1})$,目的节点坐标为 $D(d_0, d_1, \dots, d_i, \dots, d_{n-1})$,空闲快速缓存buffer的数目 f 为 $(f_0+, f_0-, f_1+, f_1-, \dots, f_i+, f_i-, \dots, f_{n-1}+, f_{n-1}-)$,安全数据包数目 S 为 $(s_0+, s_0-, s_1+, s_1-, \dots, s_i+, s_i-, \dots, s_{n-1}+, s_{n-1}-)$;

其中, c_i 表示消息当前所在节点在第 i 维上的坐标, d_i 表示消息目的节点在第 i 维上的坐标, f_i+, f_i- 分别表示当前节点 C 在第 i 维正、负方向上的相邻节点的输入端口中空闲虚拟通道的数目, s_i+, s_i- 分别表示当前节点 C 在第 i 维正方向、负方向上的相邻节点的输入端口中安全虚拟通道的数目;

步骤二,如果数据包当前所在节点坐标 C 与目的节点坐标 D 相等,则数据包已经到达目的节点,路由结束;

步骤三,如果数据包当前所在节点坐标 C 与目的节点坐标 D 不相等,则从维度 $i=0$ 到维度 $i=n-1$,依次执行以下操作:

若维度 i 的计算节点数为3,若 $d_i - c_i < 0$ 且 $|d_i - c_i| = 1$,则 $CH = CH \cup \{ch_i-\}$; CH 为数据包的可选输出通道集合,初始值为空集 Φ ;若 $d_i - c_i < 0$ 且 $|d_i - c_i| = 2$,则 $CH = CH \cup \{ch_i+\}$;若 $d_i - c_i >$

0且 $|d_i - c_i| = 1$, 则 $CH = CH \cup \{ch_i +\}$; 若 $d_i - c_i > 0$ 且 $|d_i - c_i| = 2$, 则 $CH = CH \cup \{ch_i -\}$;

若维度 i 的计算节点数为4, 若 $d_i - c_i < 0$ 且 $|d_i - c_i| \leq 2$, 则 $CH = CH \cup \{ch_i -\}$; 若 $d_i - c_i < 0$ 且 $|d_i - c_i| > 2$, 则 $CH = CH \cup \{ch_i +\}$; 若 $d_i - c_i > 0$ 且 $|d_i - c_i| \leq 2$, 则 $CH = CH \cup \{ch_i +\}$; 若 $d_i - c_i > 0$ 且 $|d_i - c_i| > 2$, 则 $CH = CH \cup \{ch_i -\}$;

若维度 i 的计算节点大于4, 若 $d_i - c_i > k_i/2$ 或 $-k_i/2 \leq d_i - c_i < 0$, 且 $f_i - , s_i -$ 满足预设动态数据包状态驱动的流程技术的条件, 则提供数据包沿着当前点 C 到其第 i 维负方向上的跳步 $ch_i -$ 前进, $CH = CH \cup \{ch_i -\}$;

若 $0 < d_i - c_i \leq k_i/2$ 或者 $d_i - c_i < -k_i/2$, 且 $f_i + , s_i +$ 满足预设动态数据包状态驱动的流程技术的条件, 则提供数据包沿着当前点 C 到其第 i 维正方向上的跳步 $ch_i +$ 前进, $CH = CH \cup \{ch_i +\}$;

步骤四, 从集合 CH 中自适应地选择一条通道, 作为最终的输出通道 ch ;

在目标节点 $(d_0, d_1, \dots, d_{n-1})$ 和当前节点 $(c_0, c_1, \dots, c_{n-1})$ 均只相差一跳时, 采用 p^* -cube路由, 具体包括:

将当前节点和目标节点间标号不同的维度划分成两个子集 U_1 及 U_2 , 初始 U_1 和 U_2 设置为空集 Φ ; 对所有维度 $i = 0$ 到 $i = n-1$, 如果 $c_i \neq d_i$, 且该维度的下一跳步是 $ch_i +$, 则 $U_1 \leftarrow U_1 \cup \{i\}$; 否则, 该维度的下一跳跳步是 $ch_i -$, $U_2 \leftarrow U_2 \cup \{i\}$;

p^* -cube路由方法如下: 从当前节点的坐标 $C(c_0, c_1, \dots, c_{n-1})$ 到目的节点坐标 $D(d_0, d_1, \dots, d_{n-1})$ 的数据包首先适应性地跨越 U_2 中的所有维度, 当 U_2 所有维度走完, 适应性的选择 U_1 中的维度传输数据包。

4. 根据权利要求3所述的自适应路由方法, 其特征在于, 所述自适应路由方法还包括: 为数据包提供到达目的节点的最短路径。

5. 根据权利要求4所述的自适应路由方法, 其特征在于, 所述为数据包提供到达目的节点的最短路径, 具体包括:

步骤S1: 定义安全数据包和非安全数据包, 如果数据包满足以下条件之一则为安全数据包:

若数据包的下一跳步是通过环绕通道, 该维度放置的计算节点数大于4, 且环绕通道所在的维度是数据包从当前点到目的节点所需要通过环绕通道中的最低维, 对于下一跳步上的节点, 该数据包是安全数据包;

若数据包从当前节点到达目的节点不再需要通过任何维度该维度放置的计算节点数大于4的环绕通道, 并且数据包下一跳步遵循 p^* -cube方法, 则对于下一跳步上的节点, 该数据包是安全数据包;

否则该数据包为非安全数据包;

步骤S2: 在虚拟直通VCT交换技术下, 任意节点路由器的输入端口包含两个缓冲区, 每个缓冲区至少完整存储一个数据包, 且两个缓冲区不分类;

设 f 表示该输入端口中空闲虚拟通道数量, s 表示该输入端口中安全虚拟通道资源的数量; 缓冲区的状态包括空闲、安全和非安全, 空闲虚拟通道表示缓冲区中未存储任何数据包; 安全虚拟通道表示缓冲区存储着安全数据包, 或者该缓冲区资源原始存储着非安全数据包, 该非安全数据包已经申请到了下一跳节点上的缓冲区资源; 非安全虚拟通道表示缓冲区存储着非安全数据包, 且该非安全数据包还未申请到下一跳节点上的缓冲区资源;

步骤S3: 基于步骤S2提供的缓冲区状态信息, 设下一跳路由器节点有 f 个空闲虚拟通

道, s 个安全虚拟通道, 有以下流控机制:

若 $f > 1$, 则数据包可以前进下一个节点;

若 $f = 1$ 且 $s \geq 1$, 则数据包可以前进下一个节点;

若 $f = 1$ 且 $s = 0$, 若从当前点到下一节点遵循 p^* -cube 方法, 则该数据包可以前进下一个节点;

其他情况下, 数据包不能前进到下一个节点。

高维度Torus网络架构及自适应路由方法

[0001] 本申请是名为《高维度Torus网络架构及自适应路由方法》的专利申请的分案申请,原申请的申请日为2019年05月15日,申请号为201910408017.3。

技术领域

[0002] 本发明涉及分布式计算技术领域,特别是涉及一种高维度Torus网络架构及自适应路由方法。

背景技术

[0003] 互连网络(Interconnection Networks)是一种总线技术,其突破了由于电气限制而达到的性能极限,成为解决现代数字系统的系统级通信问题的通用方案。互连网络是影响整个多处理器系统的性能和扩展性的重要模块,目前,如何在现有技术基础上,为互连网络的处理节点之间的提供低延迟、高效率的通信成为互连网络亟待解决的关键问题。

[0004] 互连网络主要由三个要素来描述:拓扑结构、交换机制和路由方法。高性能互连是搭建超级计算机的核心技术。高性能互连主要从两方向发展:(1)低度网络,每个计算节点(或处理器)的度数较低,主要有低维度Torus网络;如Cray Titan的Gemini互连,节点度数为8;IBM Blue/Gene L及P,节点度数为6;(2)高阶网络(High-Radix),主要有胖树,Dragonfly网络等。其中,低度网络的特点是:每个物理通道的吞吐率高,但每个节点的交换机设计简单且网络直径高;高阶网络的特点是网络的直径低,但交换机设计复杂且每个物理通道的吞吐率相对较低。

[0005] Torus网络是一种互连网络,其为完全对称的拓扑结构,具有很多优良特性,比如所有节点度相同,结构简单,路径多以及可扩展性好等,因此被广泛应用于商用系统中。Torus网络中的路由方法性能仍然有待提高;而且等分带宽偏低。

发明内容

[0006] 本发明的目的是提供一种高维度Torus网络架构及自适应路由方法,可提高系统性能,解决现有技术中在Torus网络中的路由方法普遍存在效率低,等分带宽低的问题。

[0007] 为实现上述目的,本发明提供了如下技术方案:

[0008] 一种高维度Torus网络架构,所述高维度Torus网络架构包括:

[0009] 由低维到高维依次排列的多单元维度;其中,多单元维度包括 k_n 个计算单元, k_n 个计算单元的节点坐标分别为 k_0, k_1, \dots, k_{n-1} ; k_n 个计算单元按照 $0, 1, \dots, k_i - 1$ 顺次连接形成闭环; k_n 为大于或等于2的正整数, i 代表维度的数量;

[0010] 所述多单元维度在进行路由选择时,所传输的数据包的当前节点坐标 c_i 与目标节点坐标 d_i 不相等时,对应 $i = 0$ 到 $n - 1$:

[0011] 若 $d_i - c_i > k_i / 2$ 或 $-k_i / 2 \leq d_i - c_i < 0$,则向第 i 维的负方向选择下一计算单元作为路由节点;

[0012] 若 $0 < d_i - c_i < k_i / 2$ 或 $d_i - c_i < -k_i / 2$,则沿着第 i 维的正方向选择下一计算单元作为路由

节点；

[0013] 其中,所述下一计算单元选择路由节点时满足流控技术设置的预设安全要求;预设安全要求为:若所述下一计算单元包括 f 个空闲虚拟通道以及 s 个安全虚拟通道,则:若 $f > 1$,或 $f = 1$ 且 $s \geq 1$,或 $f = 1$ 且 $s = 0$ 且从当前计算单元到下一计算单元遵循 p^* -cube方法,则所述下一计算单元作为路由节点。

[0014] 可选地,所述高维度Torus网络架构的维度大于等于2。

[0015] 为实现上述目的,本发明还提供了如下技术方案:

[0016] 一种自适应路由方法,应用于上述高维度Torus网络架构,所述自适应路由方法包括:

[0017] 步骤一,在 n 维度Torus网络架构中,设数据包路由时,当前节点的坐标为 $C(c_0, c_1, \dots, c_i, \dots, c_{n-1})$,目的节点坐标为 $D(d_0, d_1, \dots, d_i, \dots, d_{n-1})$,空闲快速缓存buffer的数目 f 为 $(f_0^+, f_0^-, f_1^+, f_1^-, \dots, f_i^+, f_i^-, \dots, f_{n-1}^+, f_{n-1}^-)$,安全数据包数目 S 为 $(s_0^+, s_0^-, s_1^+, s_1^-, \dots, s_i^+, s_i^-, \dots, s_{n-1}^+, s_{n-1}^-)$;

[0018] 其中, c_i 表示消息当前所在节点在第 i 维上的坐标, d_i 表示消息目的节点在第 i 维上的坐标, f_i^+, f_i^- 分别表示当前节点 C 在第 i 维正、负方向上的相邻节点的输入端口中空闲虚拟通道的数目, s_i^+, s_i^- 分别表示当前节点 C 在第 i 维正方向、负方向上的相邻节点的输入端口中安全虚拟通道的数目;

[0019] 步骤二,如果数据包当前所在节点坐标 C 与目的节点坐标 D 相等,则数据包已经到达目的节点,路由结束;

[0020] 步骤三,如果数据包当前所在节点坐标 C 与目的节点坐标 D 不相等,则从维度 $i = 0$ 到维度 $i = n - 1$,依次执行以下操作:

[0021] 若维度 i 的计算节点数为3,若 $d_i - c_i < 0$ 且 $|d_i - c_i| = 1$,则 $CH = CH \cup \{ch_i^-\}$; CH 为数据包的可选输出通道集合,初始值为空集 Φ ;若 $d_i - c_i < 0$ 且 $|d_i - c_i| = 2$,则 $CH = CH \cup \{ch_i^+\}$;若 $d_i - c_i > 0$ 且 $|d_i - c_i| = 1$,则 $CH = CH \cup \{ch_i^+\}$;若 $d_i - c_i > 0$ 且 $|d_i - c_i| = 2$,则 $CH = CH \cup \{ch_i^-\}$;

[0022] 若维度 i 的计算节点数为4,若 $d_i - c_i < 0$ 且 $|d_i - c_i| \leq 2$,则 $CH = CH \cup \{ch_i^-\}$;若 $d_i - c_i < 0$ 且 $|d_i - c_i| > 2$,则 $CH = CH \cup \{ch_i^+\}$;若 $d_i - c_i > 0$ 且 $|d_i - c_i| \leq 2$,则 $CH = CH \cup \{ch_i^+\}$;若 $d_i - c_i > 0$ 且 $|d_i - c_i| > 2$,则 $CH = CH \cup \{ch_i^-\}$;

[0023] 若维度 i 的计算节点大于4,若 $d_i - c_i > k_i/2$ 或 $-k_i/2 \leq d_i - c_i < 0$,且 f_i^-, s_i^- 满足预设动态数据包状态驱动的流控技术的条件,则提供数据包沿着当前点 C 到其第 i 维负方向上的跳步 ch_i^- 前进, $CH = CH \cup \{ch_i^-\}$;

[0024] 若 $0 < d_i - c_i \leq k_i/2$ 或者 $d_i - c_i < -k_i/2$,且 f_i^+, s_i^+ 满足预设动态数据包状态驱动的流控技术的条件,则提供数据包沿着当前点 C 到其第 i 维正方向上的跳步 ch_i^+ 前进, $CH = CH \cup \{ch_i^+\}$;

[0025] 步骤四,从集合 CH 中自适应地选择一条通道,作为最终的输出通道 ch 。

[0026] 可选地,所述自适应路由方法还包括:为数据包提供到达目的节点的最短路径。

[0027] 可选地,所述为数据包提供到达目的节点的最短路径,具体包括:

[0028] 步骤S1:定义安全数据包和非安全数据包,如果数据包满足以下条件之一则为安全数据包:

[0029] 若数据包的下一跳步是通过环绕通道,该维度放置的计算节点数大于4,且环绕通

道所在的维度是数据包从当前点到目的节点所需要通过环绕通道中的最低维,对于下一跳步上的节点,该数据包是安全数据包;

[0030] 若数据包从当前节点到达目的节点不再需要通过任何维度该维度放置的计算节点数大于4的环绕通道,并且数据包下一跳步遵循 p^* -cube方法,则对于下一跳步上的节点,该数据包是安全数据包;

[0031] 否则该数据包为非安全数据包;

[0032] 步骤S2:在虚拟直通VCT交换技术下,任意节点路由器的输入端口包含两个缓冲区,每个缓冲区至少完整存储一个数据包,且两个缓冲区不分类;

[0033] 设 f 表示该输入端口中空闲虚拟通道数量, s 表示该输入端口中安全虚拟通道资源的数量;缓冲区的状态包括空闲、安全和非安全,空闲虚拟通道表示缓冲区中未存储任何数据包;安全虚拟通道表示缓冲区存储着安全数据包,或者该缓冲区资源原始存储着非安全数据包,该非安全数据包已经申请到了下一跳节点上的缓冲区资源;非安全虚拟通道表示缓冲区存储着非安全数据包,且该非安全数据包还未申请到下一跳节点上的缓冲区资源;

[0034] 步骤S3:基于步骤S2提供的缓冲区状态信息,设下一跳路由器节点有 f 个空闲虚拟通道, s 个安全虚拟通道,有以下流控机制:

[0035] 若 $f > 1$,则数据包可以前进下一个节点;

[0036] 若 $f = 1$ 且 $s \geq 1$,则数据包可以前进下一个节点;

[0037] 若 $f = 1$ 且 $s = 0$,若从当前点到下一节点遵循 p^* -cube方法,则该数据包可以前进下一个节点;

[0038] 其他情况下,数据包不能前进到下一个节点。

[0039] 可选地,在目标节点 d_i 和当前节点 c_i 只相差一跳时,采用 p^* -cube路由,具体包括:

[0040] 将当前节点 $(c_0, c_1, \dots, c_i, \dots, c_{n-1})$ 和目标节点 $(d_0, d_1, \dots, d_i, \dots, d_{n-1})$ 标号不同的维度划分成两个子集 U_1 及 U_2 ,初始 U_1 和 U_2 设置为空集 Φ ;对所有维度 $i = 0$ 到维度 $i = n - 1$,若 $c_i \neq d_i$,且该维度的下一跳步是 ch_{i+} ,则 $U_1 \leftarrow U_1 \cup \{i\}$;否则,该维度的下一跳跳步是 ch_{i-} , $U_2 \leftarrow U_2 \cup \{i\}$;

[0041] 从当前节点的坐标 $C(c_0, c_1, \dots, c_{n-1})$ 到目的节点坐标 $D(d_0, d_1, \dots, d_{n-1})$ 的数据包,跨越 U_2 中的所有维度,当 U_2 所有维度走完后,适应性地选择 U_1 中的维度传输数据包。

[0042] 根据本发明提供的具体实施例,本发明公开了以下技术效果:本发明提供的高维度Torus网络架构及自适应路由方法,高维度Torus网络架构包括多个多单元维度,多单元维度包括 k_n 个计算单元,且每个计算单元在进行路由选择时采用适应性路由的方法,基于最短路径原则选择最优路径,考虑当前网络的计算单元数量,通过改变传输方向,选择最短路径。所设计高维度torus网络可有效地提高网络的等分带宽以及数据传输效率,有利于网络性能的提高。

附图说明

[0043] 为了更清楚地说明本发明实施例或现有技术中的技术方案,下面将对实施例中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

- [0044] 图1为本发明高维度Torus网络架构的示意图；
- [0045] 图2为本发明实施例提供的一个 $3 \times 3 \times 3 \times 3 \times 3$ 的5维torus网络的示意图；
- [0046] 图3本发明实施例提供的从 $3 \times 3 \times 3 \times 3 \times 3$ 网络中提取的5维超立方子网络的示意图；
- [0047] 图4为本发明自适应路由方法的流程图；
- [0048] 图5为本发明实施例提供的实现基于动态数据包状态信息的交换机示意图；
- [0049] 图6-1为本发明实施例的示例提供的含偶数个计算节点的、沿着高维度torus网络某一指定维度划分时的等分带宽示意图；
- [0050] 图6-2为本发明实施例的示例提供的含奇数个计算节点的、沿着高维度torus网络某一指定维度划分时的等分带宽示意图；
- [0051] 图7-1、图7-2分别为本发明实施例提供的沿着最高两个维度的等分带宽计算示意图；
- [0052] 图7-3为本发明实施例的示例提供的沿着d0-d7维度的等分带宽计算示意图。

具体实施方式

[0053] 下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0054] 本发明的目的是提供一种高维度Torus网络架构及自适应路由方法,有效提高了网络的等分带宽以及数据传输效率,有利于网络性能的提高。

[0055] 为使本发明的上述目的、特征和优点能够更加明显易懂,下面结合附图和具体实施方式对本发明作进一步详细的说明。

[0056] 图1以及图2示出了本发明实施例提供的一种Torus网络架构的示意图。

[0057] 图1中每个dx代表一个维度的网络架构。

[0058] 如图2所示,本发明高维度Torus网络架构包括:由低维到高维依次排列的多单元维度;其中,多单元维度包括 k_n 个计算单元, k_n 个计算单元的节点坐标分别为 k_0, k_1, \dots, k_{n-1} ; k_n 个计算单元按照 $0, 1, \dots, k_n-1$ 顺次连接形成闭环; k_n 为大于或等于2的正整数, i 代表维度的数量。具体地,所述高维度Torus网络架构的维度大于等于2。本发明实施例中,按照计算单元 $0 \rightarrow$ 计算单元 $1 \rightarrow \dots \rightarrow$ 计算单元 k_n-1 顺次连接。

[0059] 作为一种具体地实施方式,所述高维度Torus网络架构包括:由低维到高维依次排列的N个三单元维度,其中,N为大于或等于7的正整数;所述三单元维度包括三个计算单元,三个所述计算单元的节点坐标分别为 $0, 1, 2$;三个计算单元按照 $0 \rightarrow 1 \rightarrow 2$ 顺次连接形成闭环。

[0060] 其中,高维度Torus网络架构在进行数据包传输时,需要进行路由选择,而等分带宽是衡量一个网络优劣的重要参数。如果不同维度布置的计算节点数目不同时,相同网络沿着不同维度的等分带宽会不一样;具体地,沿着放置较多计算节点的维度传输,则等分带宽较小;沿着放置较少计算节点的维度均分传输,则等分带宽较大。

[0061] 路由方法决定了每个消息或分组的数据包在网络中传输的路径,将数据包正确无

误的发送到目的节点。通常情况下,消息或分组在任意节点对之间总是提供相同的路径,而与网络状态无关,方法简单,但路径通常不是最优路径。对于三单元维度,为了提高等分带宽,应选择最短路径进行数据包传输,具体地,所述三单元维度在进行路由选择时,且所传输的数据包的当前节点坐标 c_i 与目标节点坐标 d_i 不相等,即数据包未传输到目标节点时,存在以下几种情况:

[0062] 若 $d_i - c_i = -1$,则沿着该维度负方向传输数据包,即向所述第一方向的反方向选择下一计算单元作为路由节点。

[0063] 其中,如图2所示, $d_i - c_i = 1$ 的场景包括两种组合,组合(1) $d_i = 0$ 且 $c_i = 1$,或组合(2) $d_i = 1$ 且 $c_i = 2$ 。对于组合(1) $d_i = 0$ 且 $c_i = 1$ 时,此时数据包的传输方向为 $1 \rightarrow 0$,最短路径应为向所述第一方向的反方向传输,选择下一计算单元作为路由节点。对于组合(2) $d_i = 1$ 且 $c_i = 2$,此时数据包的传输方向为 $2 \rightarrow 1$,最短路径应为向所述第一方向的反方向传输,选择下一计算单元作为路由节点。

[0064] $d_i - c_i = 2$ 的场景包括一种组合,即组合(3) $d_i = 2$ 且 $c_i = 0$,此时数据包的传输方向为 $0 \rightarrow 2$,最短路径应为向所述第一方向的反方向传输,选择下一计算单元作为路由节点。

[0065] 若 $d_i - c_i = 1$ 或 $d_i - c_i = -2$,则向所述第一方向选择下一计算单元作为路由节点。具体地, $d_i - c_i = 1$ 的场景包括两种组合(4) $d_i = 1$ 且 $c_i = 0$,或组合(5) $d_i = 2$ 且 $c_i = 1$;组合(4) $d_i = 1$ 且 $c_i = 0$ 时,此时数据包的传输方向为 $0 \rightarrow 1$,最短路径应为向所述第一方向传输,选择下一计算单元作为路由节点,组合(5)与组合(4)情况相同,本发明实施例在此不再赘述。

[0066] $d_i - c_i = -2$ 的场景包括组合(6) $d_i = 0$ 且 $c_i = 2$,此时数据包的传输方向为 $2 \rightarrow 0$,最短路径应为向所述第一方向传输,选择下一计算单元作为路由节点。

[0067] 作为一种具体地实施方式,高维度Torus网络架构包括至少三个三单元维度,三单元维度包括三个计算单元,且每个计算单元在进行路由选择时采用适应性路由的方法,基于最短路径原则选择最优路径,考虑当前网络的计算单元数量,通过改变传输方向,选择最短路径,所设计高维度torus网络可有效地提高网络的等分带宽以及数据传输效率,有利于网络性能的提高。本发明实施例解决了现有技术中,在Torus网络中的路由方法普遍存在效率低,等分带宽低的问题。

[0068] 可选地,高维度Torus网络架构还包括:由低维到高维依次排列的N个三单元维度,其中,N为大于或等于7的正整数。其中,当高维度Torus网络架构为10维时,N为7,此时 $d_0 - d_9$ 均配置3个计算单元,上述网络架构的直径为10;系统总计算节点数达到 $3^{10} = 59049$ 。当高维度Torus网络架构为11维时,N为8,此时 $d_0 - d_{10}$ 均配置3个计算单元,上述网络架构的直径为11;系统总计算节点数达到 $3^{11} = 177147$ 。当高维度Torus网络架构为12维时,N为8,此时 $d_0 - d_{11}$ 均配置3个计算单元,上述网络架构的直径为12;系统总计算节点数达到 $3^{12} = 531441$ 。

[0069] 作为本发明一种实施方式,高维度Torus网络架构还包括:由低维到高维依次排列的一个所述三单元维度、三个k单元维度以及一个四单元维度;其中,所述k单元维度包括k个计算单元,k为大于或等于5的正整数,即 d_4 也为三单元维度,而 d_5, d_6, d_7 为k单元维度, d_8 为四单元维度,该高维度Torus网络架构为8维网络;当k等于5时,该高维度Torus网络架构的直径仅为12,此时系统采用该配置的总体计算节点数目为 $(3^3) * 3 * (5^3) * 4 = 40500$,超过E级计算机的计算能力。

[0070] 作为本发明一种实施方式,高维度Torus网络架构还包括:由低维到高维依次排列

的四个所述三单元维度、两个k单元维度；其中，所述k单元维度包括k个计算单元，k为大于或等于5的正整数，即d0-d7均配置三个计算单元，而d8、d9均配置k单元维度，该高维度Torus网络架构为8维网络；当k等于5时，该高维度Torus网络架构的直径仅为12，此时系统采用该配置的总体计算节点数目为 $(3^7) * (5^2) * 4 = 164025$ 。

[0071] 可选地，四单元维度在进行路由选择时，执行如下自适应路由方法：若所传输的数据包的当前节点坐标 c_i 与目标节点坐标 d_i 不相等，则

[0072] (1) 若 $d_i - c_i < 0$ 且 $|d_i - c_i| \leq 2$ 或 $d_i - c_i = 3$ 则向所述第一方向的反方向选择下一计算单元作为路由节点。其中， $d_i - c_i < 0$ 且 $|d_i - c_i| \leq 2$ 的情况，即沿第一方向上， c_i 在 d_i 的前方，此时可能出现的组合包括： $d_i = 0$ 且 $c_i = 1$ ； $d_i = 0$ 且 $c_i = 2$ ； $d_i = 1$ 且 $c_i = 2$ ； $d_i = 1$ 且 $c_i = 3$ ； $d_i = 2$ 且 $c_i = 3$ ； $d_i = 2$ 且 $c_i = 4$ ； $d_i = 3$ 且 $c_i = 4$ ；以上几种情况中 c_i 应原路返回至 d_i 点，即向所述第一方向的反方向选择下一计算单元作为路由节点。 $d_i - c_i = 3$ 的情况，包括 $d_i = 3$ 且 $c_i = 0$ ，此时应向所述第一方向的反方向选择下一计算单元作为路由节点。

[0073] (2) 若 $d_i - c_i = 1$ 或 2 则向所述第一方向选择下一计算单元作为路由节点。中， $d_i - c_i = 1$ 或 2 情况，即沿第一方向上， d_i 在 c_i 的前方； $d_i - c_i = 1$ 时可能出现的组合包括： $d_i = 1$ 且 $c_i = 0$ ； $d_i = 2$ 且 $c_i = 1$ ； $d_i = 3$ 且 $c_i = 2$ ； $d_i - c_i = 2$ 时可能出现的组合包括： $d_i = 2$ 且 $c_i = 0$ ； $d_i = 3$ 且 $c_i = 1$ ；以上几种情况中向所述第一方向选择下一计算单元作为路由节点。

[0074] 本发明上述实施例中，多单元维度在进行路由选择时，执行如下自适应路由方法：所传输的数据包的当前节点坐标 c_i 与目标节点坐标 d_i 不相等时，对应 $i = 0$ 到 $n-1$ ：

[0075] (1) 若 $d_i - c_i > k_i/2$ 或 $-k_i/2 \leq d_i - c_i < 0$ ，则向第 i 维的负方向选择下一计算单元作为路由节点。其中， $d_i - c_i > k_i/2$ 时，即沿正方向上， d_i 在 c_i 的前方，且在正方向上二者之间距离的节点数目是大于 $k_i/2$ 的，则在反方向上二者之间距离的节点数目是小于 $k_i/2$ 的，因此向反方向传输数据包为最短路径； $-k_i/2 \leq d_i - c_i < 0$ 时，同理，本发明实施例在此不再赘述。

[0076] (2) 若 $0 < d_i - c_i < k_i/2$ 或 $d_i - c_i < -k_i/2$ ，则沿着第 i 维的正方向选择下一计算单元作为路由节点。其中， $d_i - c_i < k_i/2$ 时，即沿正方向上， d_i 在 c_i 的前方，且在正方向上二者之间距离的节点数目是小于 $k_i/2$ 的，则在反方向上二者之间距离的节点数目是大于 $k_i/2$ 的，因此向正方向传输数据包为最短路径； $d_i - c_i < -k_i/2$ 时，同理，本发明实施例在此不再赘述。

[0077] 其中，所述下一计算单元选择路由节点时满足流控技术设置的buffer要求。

[0078] 具体地，预设安全要求为：

[0079] 若所述下一计算单元包括 f 个空闲虚拟通道以及 s 个安全虚拟通道，则：

[0080] 若 $f > 1$ ，或 $f = 1$ 且 $s \geq 1$ ，或 $f = 1$ 且 $s = 0$ 且从当前计算单元到下一计算单元遵循p*-cube方法，则所述下一计算单元可作为路由节点。

[0081] 关于预设安全要求，首先定义安全数据包和非安全数据包，如果数据包满足以下条件之一则为安全数据包：

[0082] 1) 如果数据包的下一跳步是通过单元维度内部的环境通道，该维度放置的计算单元数大于4，且环境通道所在的维度是数据包从当前节点到目的节点所需要通过环境通道中的最低维，对于下一跳步上的节点，该数据包是安全数据包。

[0083] 2) 如果数据包从当前节点到达目的节点不再需要通过计算节点数大于4维度的环境通道，并且数据包下一跳步遵循高维度torus网络中的无死锁自适应路由方法，比如p*-cube路由方法，那么对于下一跳步上的节点，该数据包是安全数据包。

[0084] 如果数据包对以上两个条件都不满足,那么该数据包为非安全数据包。

[0085] 此外,关于虚拟通道,在虚拟直通(Virtual-Cut Through,VCT)交换技术下,任意节点路由器的输入端口均包含两条虚拟通道,每条虚拟通道包含一个缓冲区,每个缓冲区可以完整存储一个数据包,两个缓冲区不分类。设 f 表示该输入端口中空闲虚拟通道数量, s 表示该输入端口中安全虚拟通道资源的数量。缓冲区可以分为空闲、安全和非安全三种状态,空闲虚拟通道表示缓冲区中未存储任何数据包。若空闲虚拟通道数量大于1,则所述下一计算单元可作为路由节点。

[0086] 具体地,安全虚拟通道表示缓冲区存储着安全数据包,或者该缓冲区资源原始存储着非安全数据包,但该非安全数据包已经申请到了下一跳节点上的缓冲区资源;非安全虚拟通道表示缓冲区存储着非安全数据包,且该非安全数据包还未申请到下一跳节点上的缓冲区资源。若空闲虚拟通道数量等于1且安全虚拟通道数量 $s \geq 1$,则所述下一计算单元可作为路由节点。

[0087] 每条虚拟通道都对有一个对应的比特位最后位(Last Token,LT),当一个数据包的头微片到达某条虚拟通道的时候,该虚拟通道的LT位设置为1,另外一条虚拟通道的LT位设置为0。LT用于保证数据包的按序传输,如果输入端口中的两条虚拟通道,存储的数据包头微片具有相同的目的节点,那么之前(LT为0对应的那个虚拟通道)的数据包头微片进入虚拟通道分配阶段。否则,两个数据包都进入虚拟通道分配阶段。此外,在每个输入端口的虚拟通道都有一个SAFE比特位,用于表示占用该虚拟通道的数据包是否为安全数据包,该比特位在虚拟通道接收到数据包头微片的时候进行更新。

[0088] 当头微片传输到下游节点的时候,SAFE比特位将向上游节点传输更新信息。当上游节点接收到所述更新信息的时候,上游节点的路由器将更新其计数器中所计数的安全虚拟通道的数目SAFE,且FREE参数也进行相应的更新。当存储在非安全的虚拟通道的非安全数据包申请到了下游路由节点的虚拟通道时,该非安全虚拟通道将动态变化为安全虚拟通道,相应的SAFE更新信息将传递给上游路由节点,用于上游节点更新SAFE域。

[0089] 此外,若 $f=1$ 且 $s=0$,且从当前计算单元到下一计算单元遵循 p^* -cube方法,则所述下一计算单元可作为路由节点。

[0090] 关于 p^* -cube路由方法, p^* -cube路由方法是针对超立方网络的一种部分适应行路由方法,对于目标点 $(d_0, d_1, \dots, d_{n-1})$ 和当前点 $(c_0, c_1, \dots, c_{n-1})$ 均只相差一跳时,可采用 p^* -cube路由方法。

[0091] 将当前节点和目标节点间标号不同的维度划分成两个子集 U_1 及 U_2 ,初始 U_1 和 U_2 设置为空集 Φ ;对所有维度 $i=0$ 到 $i=n-1$,如果 $c_i \neq d_i$,且该维度的下一跳步是 ch_i+ ,则 $U_1 \leftarrow U_1 \cup \{i\}$;否则,该维度的下一跳跳步是 ch_i- , $U_2 \leftarrow U_2 \cup \{i\}$ 。

[0092] p^* -cube路由方法如下:从 $(c_0, c_1, \dots, c_{n-1})$ 到 $(d_0, d_1, \dots, d_{n-1})$ 的数据包首先适应性地跨越 U_2 中的所有维度,当 U_2 所有维度走完,适应性的选择 U_1 中的维度传输数据包。

[0093] 通过以 p^* -cube路由算法为基准路由策略及基于动态数据包状态信息的流控技术构建高位torus网络完全适应性路由算法。

[0094] 作为示例,参见图3,图3为路由器结构内部的示意图,即实现基于动态数据包状态信息的交换机。与传统路由器结构相比,在路由器输出端口需要增加FREE域和SAFE域;FREE域表示下游节点输入端口空闲虚拟通道数量,SAFE域表示下游节点的输入端口缓冲区中存

储安全数据包的数量。数据包在前进的时候利用这两个参数,按照上述预设安全要求可以有效避免任一输入端口的两条虚拟通道同时存储非安全数据包,即若 $f > 1$,那么数据包可以前进下一个节点;若 $f = 1$ 且 $s \geq 1$,那么数据包可以前进下一个节点;若 $f = 1$ 且 $s = 0$,如果数据包对下一个节点是安全数据包,那么该数据包可以前进下一个节点;其他情况下,数据包不能前进道下一个节点。

[0095] 可选地,本发明上述实施例中,所述高维度Torus网络架构在选择下一计算单元作为路由节点后,输出可选通道集合;

[0096] 所述可选通道集合中包括当前计算单元的可选通道以及所述下一计算单元的可选通道。

[0097] 其中,如果向第一方向(正方向)传输,选择下一计算单元作为路由节点, $CH_i = CH_i \cup \{ch_{i+}\}$, ch_i 为当前计算单元的可选通道集合, c_{i+} 为沿第一方向的下一计算单元的可选通道集合, ch_i 为可选通道集合。

[0098] 如果向所述第一方向的反方向传输,选择下一计算单元作为路由节点, $CH_i = CH_i \cup \{c_{i-}\}$, CH_i 为当前计算单元的可选通道集合, c_{i-} 为沿第一方向的反方向的下一计算单元的可选通道集合, CH_i 为可选通道集合。

[0099] 本发明实施例涉及高维torus网络的互连架构,适用于VCT或者缓存虫孔(BufferedWorm Hole,BWH)交换技术,通过网络布局使得计算节点数达到40500时,网络的直径仅为12;当计算节点数达到164025时,网络直径仍然只为12,此时系统的总体计算能力约为E级计算机的4.5倍;节点数达到177147时,网络直径仅为11,达到4.8E级计算机的计算能力;节点数达到531441时,网络直径仅为12,达到14.4倍E级计算机的计算能力。

[0100] 以上介绍了本发明实施例提供的高维度Torus网络架构,下面将结合附图介绍本发明实施例提供的自适应路由方法。

[0101] 如图4所示,本发明的自适应路由方法应用于上述高维度Torus网络架构,所述自适应路由方法包括:

[0102] 步骤401,在n维度Torus网络架构中,设数据包路由时,当前节点的坐标为 $C(c_0, c_1, \dots, c_i, \dots, c_{n-1})$,目的节点坐标为 $D(d_0, d_1, \dots, d_i, \dots, d_{n-1})$,空闲快速缓存buffer的数目 f 为 $(f_0^+, f_0^-, f_1^+, f_1^-, \dots, f_i^+, f_i^-, \dots, f_{n-1}^+, f_{n-1}^-)$,安全数据包数目 S 为 $(s_0^+, s_0^-, s_1^+, s_1^-, \dots, s_i^+, s_i^-, \dots, s_{n-1}^+, s_{n-1}^-)$;

[0103] 其中, c_i 表示消息当前所在节点在第i维上的坐标, d_i 表示消息目的节点在第i维上的坐标, f_i^+, f_i^- 分别表示当前节点C在第i维正、负方向上的相邻节点的输入端口中空闲虚拟通道的数目, s_i^+, s_i^- 分别表示当前节点C在第i维正方向、负方向上的相邻节点的输入端口中安全虚拟通道的数目。

[0104] 步骤402,如果数据包当前所在节点坐标C与目的节点坐标D相等,即 $C = D$,则数据包已经到达目的节点,路由结束。

[0105] 步骤403,如果数据包当前所在节点坐标C与目的节点坐标D不相等,则从维度 $i = 0$ 到维度 $i = n - 1$,依次执行以下操作:

[0106] 若维度i的计算节点数为3,若 $d_i - c_i < 0$ 且 $|d_i - c_i| = 1$,则 $CH = CH \cup \{ch_{i-}\}$;CH为数据包的可选输出通道集合,初始值为空集 Φ ;若 $d_i - c_i < 0$ 且 $|d_i - c_i| = 2$,则 $CH = CH \cup \{ch_{i+}\}$;若 $d_i - c_i > 0$ 且 $|d_i - c_i| = 1$,则 $CH = CH \cup \{ch_{i+}\}$;若 $d_i - c_i > 0$ 且 $|d_i - c_i| = 2$,则 $CH = CH \cup \{ch_{i-}\}$;

[0107] 若维度*i*的计算节点数为4,若 $d_i - c_i < 0$ 且 $|d_i - c_i| \leq 2$,则 $CH = CH \cup \{ch_i^-\}$;若 $d_i - c_i < 0$ 且 $|d_i - c_i| > 2$,则 $CH = CH \cup \{ch_i^+\}$;若 $d_i - c_i > 0$ 且 $|d_i - c_i| \leq 2$,则 $CH = CH \cup \{ch_i^+\}$;若 $d_i - c_i > 0$ 且 $|d_i - c_i| > 2$,则 $CH = CH \cup \{ch_i^-\}$;

[0108] 若维度*i*的计算节点大于4,若 $d_i - c_i > k_i/2$ 或 $-k_i/2 \leq d_i - c_i < 0$,且 f_i^-, s_i^- 满足预设动态数据包状态驱动的流控技术的条件,则提供数据包沿着当前点C到其第*i*维负方向上的跳步 ch_i^- 前进, $CH = CH \cup \{ch_i^-\}$;

[0109] 若 $0 < d_i - c_i \leq k_i/2$ 或者 $d_i - c_i < -k_i/2$,且 f_i^+, s_i^+ 满足预设动态数据包状态驱动的流控技术的条件,则提供数据包沿着当前点C到其第*i*维正方向上的跳步 ch_i^+ 前进, $CH = CH \cup \{ch_i^+\}$ 。

[0110] 步骤404,经过所述步骤403的操作,得到数据包可选的输出通道的集合CH,从集合CH中自适应地选择一条通道作为最终的输出通道ch。

[0111] 进一步地,所述自适应路由方法还包括:为数据包提供到达目的节点的最短路径。具体包括:

[0112] 步骤S1:定义安全数据包和非安全数据包,如果数据包满足以下条件之一则为安全数据包:

[0113] 若数据包的下一跳步是通过环绕通道,该维度放置的计算节点数大于4,且环绕通道所在的维度是数据包从当前点到目的节点所需要通过环绕通道中的最低维,对于下一跳步上的节点,该数据包是安全数据包。

[0114] 若数据包从当前节点到达目的节点不再需要通过任何维度该维度放置的计算节点数大于4的环绕通道,并且数据包下一跳步遵循高维度torus网络中的无死锁自适应路由方法(如p*-cube路由),则对于下一跳步上的节点,该数据包是安全数据包。

[0115] 若数据包对以上两个条件都不满足,则该数据包为非安全数据包;

[0116] 步骤S2:在虚拟直通VCT交换技术下,任意节点路由器的输入端口包含两个缓冲区,每个缓冲区至少完整存储一个数据包,两个缓冲区不分类。

[0117] 设f表示该输入端口中空闲虚拟通道数量,s表示该输入端口中安全虚拟通道资源的数量;缓冲区可以分为空闲、安全和非安全三种状态;空闲虚拟通道表示缓冲区中未存储任何数据包;安全虚拟通道表示缓冲区存储着安全数据包,或者该缓冲区资源原始存储着非安全数据包,该非安全数据包已经申请到了下一跳节点上的缓冲区资源;非安全虚拟通道表示缓冲区存储着非安全数据包,且该非安全数据包还未申请到下一跳节点上的缓冲区资源。

[0118] 步骤S3:基于步骤S2提供的缓冲区状态信息,设下一跳路由器节点有f个空闲虚拟通道,s个安全虚拟通道,有以下流控机制:

[0119] 若 $f > 1$,则数据包可以前进下一个节点。

[0120] 若 $f = 1$ 且 $s \geq 1$,则数据包可以前进下一个节点。

[0121] 若 $f = 1$ 且 $s = 0$,若从当前点到下一节点遵循p*-cube方法,则该数据包可以前进下一个节点。

[0122] 其他情况下,数据包不能前进到下一个节点。

[0123] 安全虚拟通道可以是本来就存储一个安全数据包,也可以由非安全虚拟通道状态转换而来,只要当非安全的数据包申请到下一跳的缓冲区资源即可实现,这也是基于动态

数据包状态的流控技术的核心。

[0124] 可选地,目标节点 d_i 和当前节点 c_i 只相差一跳时,采用 p^* -cube路由,具体包括:

[0125] 将当前节点 $(c_0, c_1, \dots, c_i, \dots, c_{n-1})$ 和目标节点 $(d_0, d_1, \dots, d_i, \dots, d_{n-1})$ 标号不同的维度划分成两个子集 U_1 及 U_2 ,初始 U_1 和 U_2 设置为空集 Φ ;对所有维度 $i=0$ 到维度 $i=n-1$,若 $c_i \neq d_i$,且该维度的下一跳步是 ch_{i+} ,则 $U_1 \leftarrow U_1 \cup \{d_i\}$;否则,该维度的下一跳跳步是 ch_{i-} , $U_2 \leftarrow U_2 \cup \{d_i\}$;

[0126] 从当前节点的坐标 $C(c_0, c_1, \dots, c_{n-1})$ 到目的节点坐标 $D(d_0, d_1, \dots, d_{n-1})$ 的数据包,跨越 U_2 中的所有维度,当 U_2 所有维度走完,适应性选择 U_1 中的维度传输数据包。

[0127] 本发明上述实施例中,高维度Torus网络架构在进行路由选择时,每个计算单元在进行路由选择时采用适应性路由的方法,基于最短路径原则选择最优路径,考虑当前网络的计算单元数量,通过改变传输方向,选择最短路径,所设计高维度torus网络可有效地提高网络的等分带宽以及数据传输效率,有利于网络性能的提高。本发明实施例解决了现有技术中,在Torus网络中的路由方法普遍存在效率低,等分带宽低的问题。

[0128] 作为示例,图5示出了一个从一个3-ary 5-立方提取出来的五维超立方子立方。根据从该子立方提取出来的安全信息可以引导可靠通讯。

[0129] 一个网络的等分带宽是该网络优劣的重要评判标准。

[0130] 图6-1至图6-2所示表示沿着高维度torus网络某一指定维度划分时的等分带宽计算。

[0131] 图6-1表示当该维度含偶数个计算节点的情形。对于一个 $k_0 \times k_1 \times \dots \times k_{i-1} \times k_i \times k_{i+1} \times \dots \times k_{n-1}$ 网络,沿着第 i 维的等分带宽可计算如下: $4k_0 k_1 \dots k_{i-1} k_{i+1} \dots k_{n-1}$ 。

[0132] 图6-2所示为第 i 维含奇数个计算节点的情形。这种情况必须多层次的计算等分带宽,假设 $k_0 \leq k_1 \leq k_2 \leq \dots \leq k_{n-2} \leq k_{n-1}$,则该 n 维torus网络的等分带宽可计算如下:

[0133] $BB = 2k_0 k_1 \dots k_{i-1} k_{i+1} \dots k_{n-2} + 2k_0 k_1 \dots k_{i-1} k_{i+1} \dots k_{n-3} + \dots + 2k_0 k_1 + 2k_0 + 2$

[0134] 图7-1至图7-3分别示出了计算 $3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 5 \times 5$ 网络的等分带宽。图7-1,图7-2分别表示沿着最高两个维度的等分带宽计算。

[0135] 图7-3为沿着 d_0 - d_7 维度的等分带宽计算。

[0136] 图3示出了本发明对应的交换机实现细节。

[0137] 判断一个网络的优劣有很多特性参数:节点度数,直径,及等分带宽。如表1所示, $3 \times 3 \times 3 \times 3 \times 5 \times 5 \times 5 \times 4$ 网络的等分带宽高于IBMBLue/Gene Q采用的 $16 \times 16 \times 16 \times 12 \times 2$ 网络k-Computer, $2 \times 3 \times 2 \times 16 \times 16 \times 32$,及IBM Blue/Gene P, $64 \times 48 \times 32$ 的等分带宽,并且大幅降低了网络直径。

[0138] 表1:

configurations (架构)	networks (网络)	bisection (等分带宽)	processors (处理机数目)	diameter (直径)	radix (路由器度数)
$16 \times 16 \times 16 \times 12$ $\times 2$	Blue/Gene Q	24576	98304	31	9
$2 \times 3 \times 2 \times 16 \times 32$	k-Computer	12288	98304	35	10
$16 \times 32 \times 48$	Blue/Gene Q	6144	131072	72	6
$3 \times 3 \times 3 \times 5 \times 5 \times 5 \times 4$	proposed	36530	81000	12	16
$3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 5 \times 5$	proposed	85920	164025	12	20
$3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3$	proposed	17146	177147	11	22
$3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3$	proposed	472392	531441	12	24
(28, 392)	dragonfly	75832	153664	3	56

[0139] 高维度torus网络 $3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 5 \times 5$ 的等分带宽优于dragonfly网络(28,392) (即每个路由器组含28个路由器,共含392个路由器组),而且每个dragonfly网络中路由器的度数为56,而 $3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 3 \times 5 \times 5$ 网络中的节点度数为20度,因而每条物理通道的带宽近似为(28,392) dragonfly网络的3倍。

[0141] 以上所描述的装置实施例仅仅是示意性的,其中所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部模块来实现本实施例方案的目的。本领域普通技术人员在不付出创造性的劳动的情况下,即可以理解并实施。

[0142] 通过以上的实施方式的描述,本领域的技术人员可以清楚地了解到各实施方式可借助软件加必需的通用硬件平台的方式来实现,当然也可以通过硬件。基于这样的理解,上述技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来,该计算机软件产品可以存储在计算机可读存储介质中,如ROM/RAM、磁碟、光盘等,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)执行各个实施例或者实施例的某些部分所述的方法。

[0143] 本文中应用了具体个例对本发明的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本发明的方法及其核心思想;同时,对于本领域的一般技术人员,依据本发明的思想,在具体实施方式及应用范围上均会有改变之处。综上所述,本说明书内容不应理解为对本发明的限制。

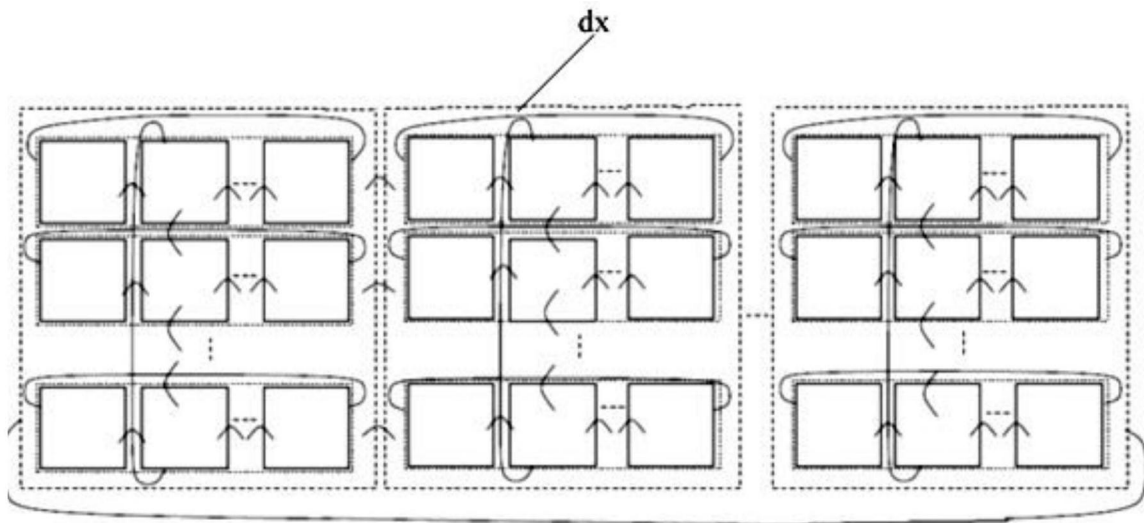


图1

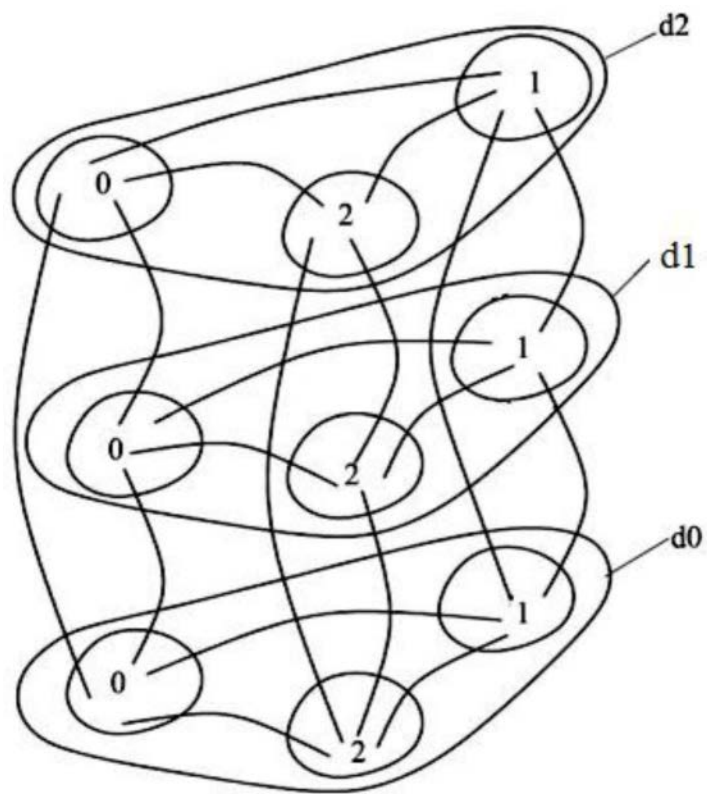


图2

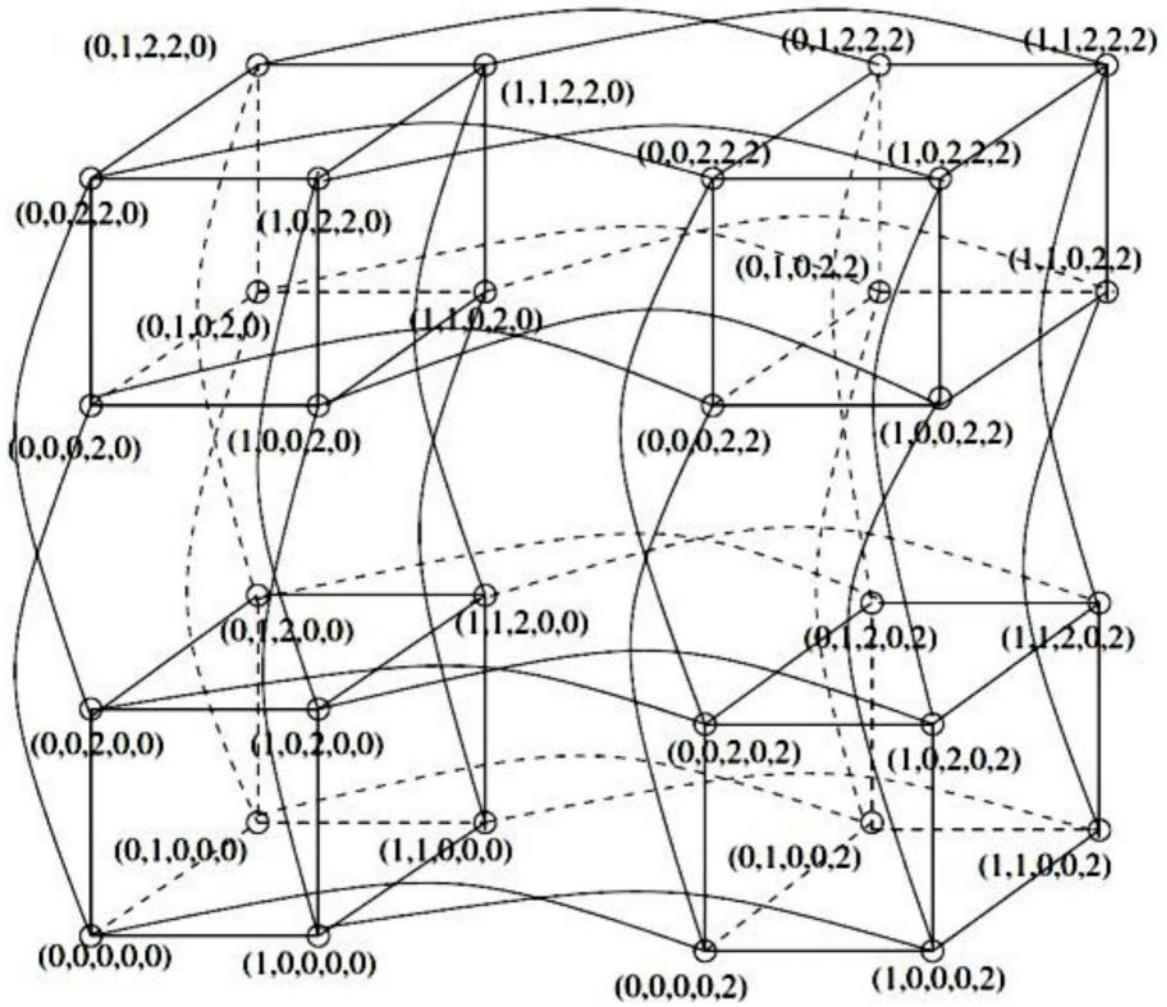


图3

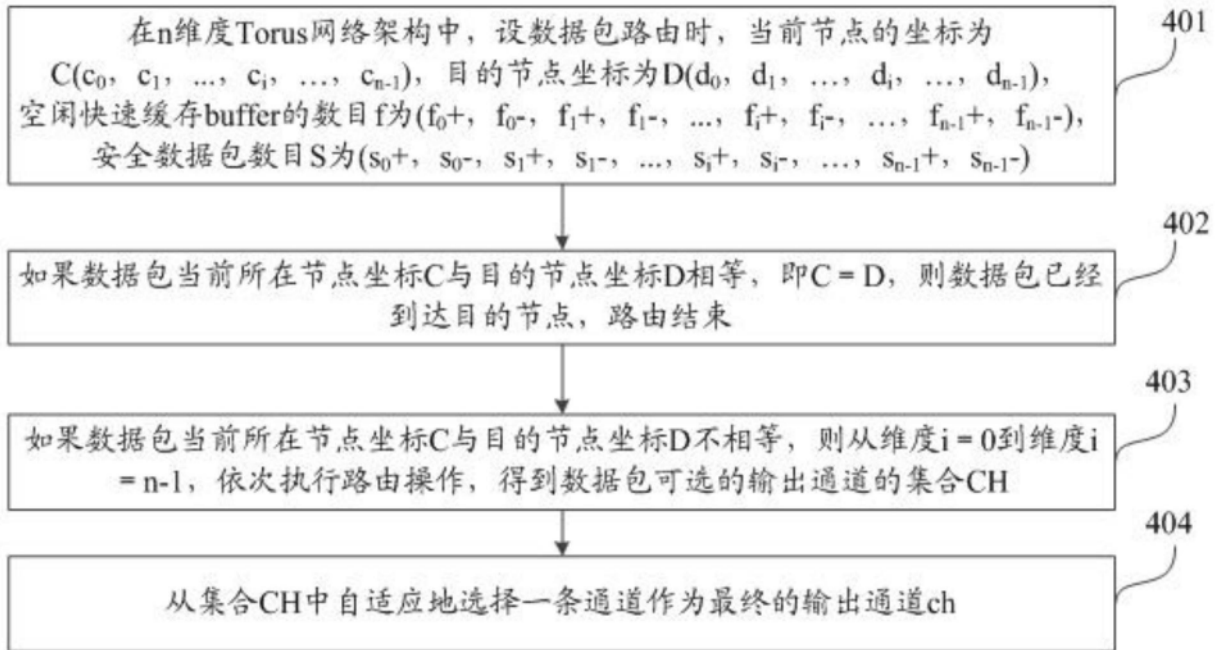


图4

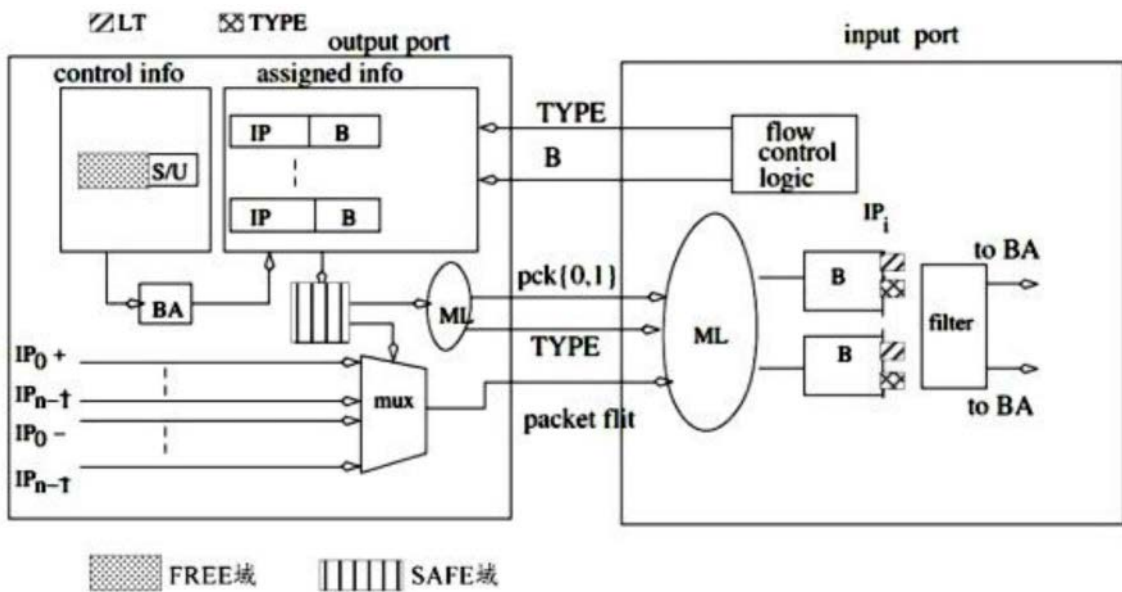


图5

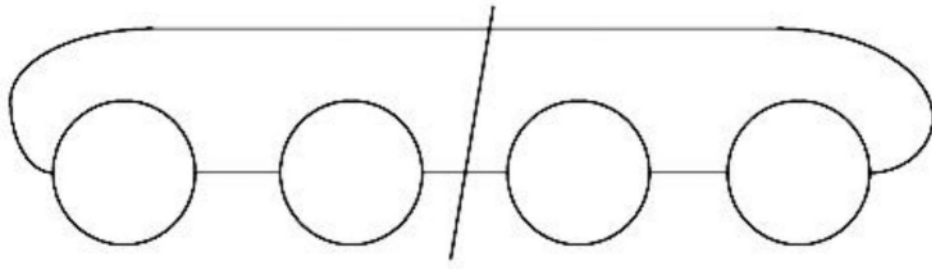


图6-1

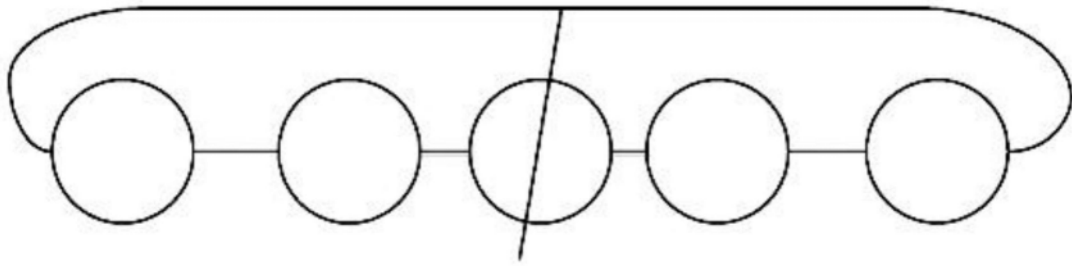


图6-2

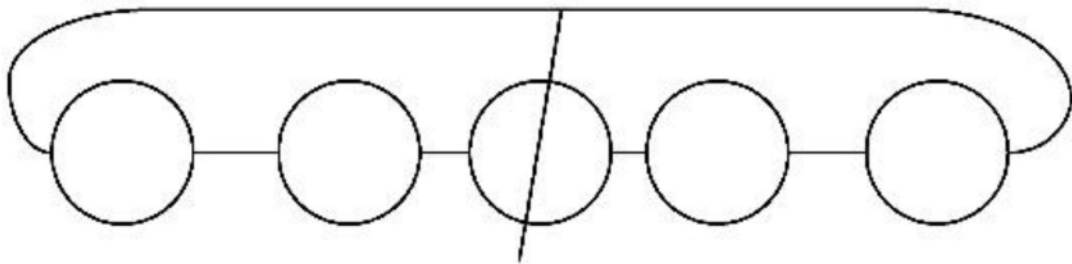


图7-1

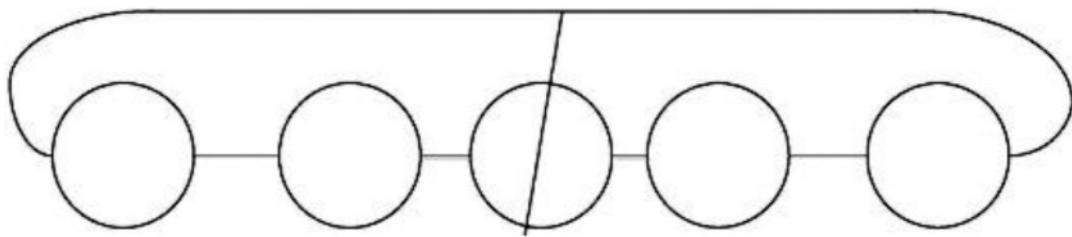


图7-2

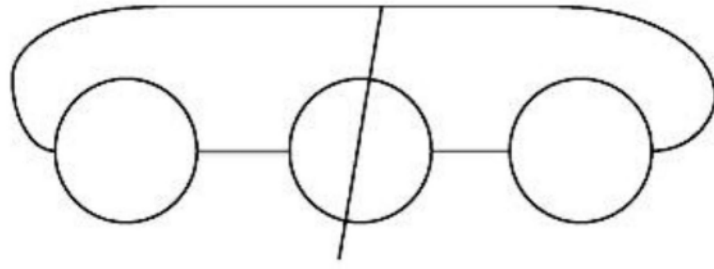


图7-3