

基于两级遗传算法的特定应用片上网络拓扑优化

赖国明¹ 林小拉²

(韩山师范学院计算机应用与技术系 潮州 521041)¹ (中山大学信息科学与技术学院 广州 510006)²

摘要 大规模片上系统面临着性能、同步、能耗等方面的通信问题,片上网络为复杂的片上系统通信问题提供了一种有效的解决方案。由于网络拓扑优化是一种 NP 难问题,针对片上系统主要面向特定应用的特点,提出一种基于两级遗传算法来求解特定应用片上网络的拓扑优化问题近似解。实验结果表明,相对于现有的三级遗传,两级遗传算法在最小能耗上平均有 1.1% 的改进,而且仿真运行时间大大减少,平均减少 97%。

关键词 特定应用片上网络,遗传算法,拓扑优化

中图分类号 TP393.03 **文献标识码** A

Application-specific Network-on-Chip Topology Optimization Based on Two-level Genetic Algorithm

LAI Guo-ming¹ LIN Xiao-la²

(Department of Computer Application and Technology, Hanshan Normal University, Chaozhou 521041, China)¹

(School of Information Science and Technology, Sun Yat-Sen University, Guangzhou 510006, China)²

Abstract Large scale system-on-chip is facing several communication problems, such as performance, synchronization and power dissipation. Network-on-chip provides a most promise solution to the communication challenges of complicated system-on-chip. Because network topology optimization is known to be an NP-hard problem, aiming at the characteristic that most system-on-chips are application-specific, this article proposed a two-level generic algorithm to solve the approximate solutions to the topology optimization problem. Experiments show that the proposed method can, on average, improve about 1.1% the minimal power consumption and 97% simulation time which is greatly reduced by comparing to the existing three-level generic algorithm.

Keywords Application-specific NoC, Genetic algorithm (GA), Topology optimization

1 引言

根据 2010 年国际半导体技术路线图(International Technology Roadmap for Semiconductors, ITRS)^[1] 的预测,到 2024 年,单个芯片上集成的晶体管数量将达到 2560 亿个。单个处理核无法完全使用到这些数量巨大的晶体管,为了充分利用晶体管,人们把它们组织成若干个处理元素(Processing Element, PE)或者知识产权核(Intellectual Property Cores, IPs)。在将来的高级嵌入式系统中,单个芯片上的多个 IP 核形成了一个完整而复杂的片上系统(System-on-Chips, SoCs)。由于处理核工作频率和计算速度的极大提高,片上系统的计算资源提供了强大的计算能力,同时,数据密集型应用的广泛使用,使得片上系统产生了巨大的通信需求。

传统上,片上系统的互联结构采用共享总线或者分段总线结构。但总线系统带宽利用率低,不具有良好的可扩展性,并受电磁干扰(EMI)、噪音、串音等深亚微米效应的影响,因此全局总线结构成为未来高端片上系统 SoC 性能提高的主要瓶颈,片上网络的提出为 SoC 片上通信问题提供了最为有

效的方案。

与标准的宏观数据网络相比,片上网络受到更多的资源和能耗限制。为了最小化 SoC 的实现成本,提高片上系统的性能,片上网络必需实现更小的面积和能耗开销,同时要把性能达到最优。可见,片上网络相关课题的研究对于高性能嵌入式片上系统和多核计算系统来说,具有极其重大的理论意义和实际应用价值。片上网络已经成为计算机及通信相关领域的研究热点^[1-16]。但是,大多数的研究都集中在通用的、基于规则拓扑的片上网络,只有数量相对较少的人从事本特定应用片上网络相关研究。规则拓扑的片上网络对于均匀的通信需求是高效的,然而,下一代高性能 SoC 主要针对特定应用或少数应用类,且具有很强的异构特性,各个 IP 核的计算能力和通信需求有很大的差异。SoC 的不同应用有着不同的数据通信模式,其应用有着明显的通信特征,NoC 系统结构只有根据不同应用的数据流特征进行定制,才能达到最好的性能、能耗和成本折衷。片上网络的拓扑结构与网络的路由、流量控制等方面紧密相关,极大地影响着片上网络的时延、功耗等性能指标^[2],片上网络的拓扑结构优化的研究是其它片上网络相关研究内容的前提和基础。如何利用应用通信需求

到稿日期:2012-04-05 返修日期:2012-07-25 本文受国家自然科学基金(60773199,61073055,60903215),广东省自然科学基金(10152104100004),韩山师范学院博士启动项目资助。

赖国明(1971—),男,博士,高级实验师,主要研究方向为网络与分布式计算、片上网络,E-mail:laigm@hstc.edu.cn。

特征来设计和优化面积,使功耗高效,并实现满足应用性能需求的特定应用片上网络拓扑结构是一个具有现实意义和应用价值的挑战性研究课题。研究人员提出了多种特定应用拓扑优化方法,如 ILP、三级 GA 等^[3-7],这些方法都有自己的优缺点。本文针对三级遗传算法,提出了一种基于两级遗传算法来优化特定应用片上网络的拓扑结构。相对于文献^[3,4],两级 GA 有着较好的优化效果和运行效率。

2 特定应用拓扑优化问题的描述

通常,SoC 复杂应用被划分为若干个功能相对独立、相互通信的任务。这些任务经过调度和映射在不同的 IP 中来执行,这些任务以数据传递方式实现相互的通信。这些应用子任务的通信特征是以应用通信特征图(Application Communication Characteristic Graph, ACCG)来表征的。网络拓扑 TG(N, R, L)是由网络组成元素(如处理节点,路由器和链路等)的数量互连方式决定的,包括网络中路由器及其互连方式和 IP 核与路由器间的连接。

特定应用拓扑优化问题可以形式化地描述如下:

• 假设给定:

①应用通信特征图 $ACCG=G(V, E)$,对于每个节点 $v_i \in V$,它的高度和宽度分别用 H_i 和 W_i 来表示。

②路由器特征参数,这里 η 表示路由器的端口数, Ω 表示端口所允许的最大带宽, P_i 和 P_o 分别表示单位带宽流过输入端口和输出端口时的单位带宽功耗。

③链路特征参数, L_{max} 表示单个时钟周期内信号能够传播的最大距离, P_l 表示单位带宽数据通过每毫米通信线路时的功耗。

④SoC 的系统级平面布局。

• 网络拓扑优化的目标是:求解网络拓扑 $TG(N, R, L)$,使得:

$$\min_{e \in E} \{ \sum_{e \in E} (\omega_r + \omega_l) \} = \min \{ \sum_{e \in E} (\sum_{r \in R} (P_r^i + P_r^o) * w(e) + \sum_{l \in L} (len(l) * P_l * w(e))) \}$$

式中, P_r^i 和 P_r^o 分别表示通信交通 e 穿过路由器 r 的输入输出端口时的功耗, $w(e)$ 是通信交通 e 的带宽, $len(l)$ 表示链路 l 的曼哈顿长度。

同时满足限制条件:

(i)对于每个应用通信交通 $e=(v_i, v_j)$,在拓扑图 TG 中存在一条路径 $p=\{(v_i, r_i), (r_i, r_k), \dots, (r_s, r_j), (r_j, v_j)\}$,同时满足带宽 $w(e_{ij})$ 和延迟 $\Theta(e_{ij})$ 的限制。

(ii)每个端口的带宽满足最大带宽 Ω 的要求。

特定应用片上网络拓扑优化问题是斯坦纳树(Steiner Tree)的一个变种^[8],它是一个 NP 难^[9]问题。对于 NP 难问题,目前没有线性时间的解决办法,遗传算法能在合理的时间找到问题有效的近似解^[10],它非常适合解这类 NP 难的问题。本文提出一种两级 GA 技术来解决特定应用片上网络的拓扑优化问题。

3 基于两级遗传算法的拓扑优化

这一节介绍基于两级遗传算法 GA 的拓扑综合技术,首先介绍染色体的数据结构表示方法,其次介绍其合法性条件,最后分别介绍两个级别的基因操作。

3.1 染色体串的数据结构及相关定义

在给出染色体数据结构前,先给出如下的定义。

定义 1(准全连接网络, quasi full connection) 如果网络中的每个路由器与所有其它路由器间(除了距离超过所允许的最大链路长度外),都有直接的物理链路进行连接,这样的网络称为准全连接的网络。换句话说,网络中的任何两个路由器之间都存在一条直接的物理链路,除非它们之间的曼哈顿距离大于单个时钟周期所允许的信号传播最大距离。

定义 2(准全连接矩阵) 准全连接网络的连接矩阵称为准全连接矩阵。

两级 GA 个体表示为一个两级的层次化结构。第一级是 NoC 体系结构级,其决定所有可能的最大数量路由器中哪些路由器被选来构成 NoC 体系结构。第二级是节点与路由器映射级,其染色体串表示节点与体系结构中的哪个路由相连接。NoC 的网络拓扑结构决定于网络中使用的路由器、节点与路由器的连接以及路由器与路由器的连接。而对于路由器与路由器的连接来说,让第一级个体选出来的所有路由器组成一个准全连接网络,然后再用最短路由算法来为 ACCG 中每个通信交通产生路由,最后,移除没有被使用的冗余链路,这样就确定了 NoC 的网络拓扑。两级 GA 技术的个体层次如图 1 所示。

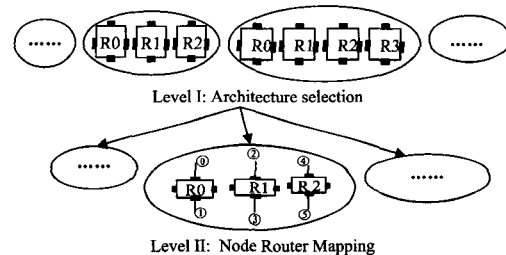


图 1 两级的个体表示

(1)NoC 体系结构级(也称为第一级)数据结构

两级 GA 技术把 NoC 体系结构级个体表示成一组常数个的二进制数组 $arstr[routermax]$,这里数值 $routermax$ 表示所有可能被使用的路由器总数。数组 $arstr$ 的第 i 个元素被随机地赋值为“0”或者“1”,表示第 i 个路由器是否被选择来构成 NoC 的体系结构,“1”表示有,“0”表示没有。变量 $i \in [0, routermax-1]$ 是一个整数。图 1 所示的第一级个体如图 2 第一级所示,其中 $arstr[4]=1$ 表示第 4 个路由器被选中,而 $arstr[5]=0$ 表示第 5 个路由器没有被选中。我们的 GA 技术保持常数 $arnum$ 个 $arstr$ 个体实例。这一级别需要一个整数数组 $routerid$ 来存储被选择的路由器在总的 $routermax$ 个路由器中的序号。

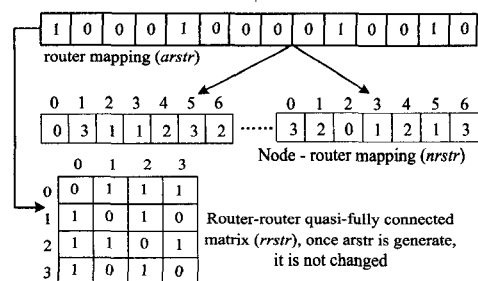


图 2 染色体数据结构表示

(2)节点/路由器映射级(也称为第二级)数据结构

在这一级,我们也使用染色体 $nrstr$ 来表示节点与路由器的映射。节点/路由器的映射存储在一个整数数组 $nrstr[nodenum]$ 中。这里变量 $nodenum$ 是应用通信特征图 ACCG 中节点的个数。数组元素 $nrstr[i]$ 的值表示节点 $v_i \in V$ 所映射的路由器号,如图 2 所示,节点 2 和 3 都被映射到路由器 1 上。

在这一级别,还需要一个辅助数据结构 $linknr$ 来存储每个路由器所连接的节点数。

(3) 其它辅助数据结构

除了上述两级数据结构外,为了表示路由器与路由器的连接,还需要辅助的第三级数据结构 $rrstr$,它是一个二维的二进制数组。由于我们的两级 GA 使用了准全互连的路由器与路由器的连接,一旦第一级的个体 $arstr$ 确定了,两级 GA 就调用 $initial_rrstr()$ 过程来产生对应的准全连接矩阵。若任两个路由器 i 和 j 的距离小于或等于最大允许的距离,则 $rrstr[i][j]$ 就被赋值为“1”,表示路由器 i 和 j 之间存在一条直接的物理链路,否则被赋值为“0”,表示之间由于违反最大允许长度而没有链路直接相连。 $rrstr$ 矩阵的阶数为其对应的 $arstr$ 个体中选择出来的路由器数,如图 2 中的连接矩阵的阶数为 4,因为其对应的 $arstr$ 中只有 4 个路由器被选中。矩阵 $rrstr$ 一旦被产生就保持不变,不参与交叉、变异等基因操作。如果第一级个体 $arstr$ 确定,其所对应的 $rrstr$ 就确定,这时只要节点/路由器的映射确定,整个 NoC 的拓扑结构就确定,那么,在满足带宽限制的条件下,就可以用 Dijkstra 最短路径算法来为 ACCG 中的每个通信交通产生静态路由。

在此,需要另外一个辅助数据结构:整数数组 $linkednum$ 来存储每个路由器的连接总数,包括节点/路由器连接和路由器/路由器连接。

3.2 个体表示的合法性条件

个体的数据结构表示必须满足一定的合法性条件,这一小节介绍两个级别的个体数据结构表示的合法性条件。

(1) 第一级的合法性条件

- 所选择的路由器总端口必须大于或等于节点数加上所选路由器数量减去 1 之差的两倍。也就是,如果有 n 个节点,所选的路由器有 r 个,每个路由器有 p 个端口,则有 $r \times p \geq n + 2(r-1)$ 。其原因是, r 个路由器要相互连接来形成一个强连通的网络至少需要用 $2(r-1)$ 个端口。

- 路由器的选择不能导致一个孤岛以保证网络的连通性。一个路由器成为孤岛是由于这个路由器因受最大线路长度的限制而不能与其它选出来的路由器相连。

(2) 第二级的合法性条件

- 一个节点必须映射给 NoC 结构中的一个路由器,而且只能映射给一个路由器。后者可以自动满足,我们使用路由器编号来作为 $nrstr$ 数组的下标索引。但是一个路由器可以连接到多个节点,也就是 $nrstr$ 数组中的元素可以重复。

- 节点到路由器连接线的长度必须小于或等于允许的最大线路长度以支持单一时钟周期的信号传播。

这里不限制路由器的连接数必须小于路由器的端口数。起初我们允许路由器的连接数大于端口数,而把个体中所有路由器的最大连接数包含在适应度函数里,如果第二级某个个体的最大连接数大于端口数,其适应度就比较低,这样在选择操作中该个体就优先被淘汰掉。

3.3 初始个体的产生和适应度函数

两级 GA 定制 NoC 拓扑的伪代码如图 3 所示,首先对两个级别的种群个体进行初始化,也就是产生 $arnum$ 个体系统结构级别个体和 $nrnum$ 个节点/路由器映射级别个体(第 1 行),在第 3-30 行,算法以概率 p_a 执行第一级的基因操作;以 p_n 的概率执行第二级的基因操作,并保留适应度最好的 $arnum$ 个第一级个体和 $nrnum$ 个第二级个体,不断迭代,直到满足停止条件,我们设置停止条件为最小的能耗保持连续 $|V|+|E|$ 次不变,即等于 ACCG 中节点数和通信交通数之和。从图 3 可知,算法的第一步就是种群初始化,然后计算各个个体的适应度值,这一小节首先介绍种群的初始化,然后介绍适应度函数。

```
GA_topology() {
1. initial_population();
2. finished=false;
3. while (! finished)
4. {
5.   set finished_nr=false;
6.   while (finished_nr)
7.   {
8.     nrcrossover( $p_n$ );
9.     retain best nrpopnum populations;
10.    nrmutation( $p_n$ );
11.    retain best nrpopnum populations
12.    if (satisfied exit criterion )
13.      finished_nr=true;
14.  }
15.  arcrossover( $p_a$ );
16.  retain best arpopnum populations
17.  set finished_nr=false;
18.  while (finished_nr)
19.  {
20.    nrcrossover( $p_n$ );
21.    retain best nrpopnum populations;
22.    nrmutation( $p_n$ );
23.    retain best nrpopnum populations
24.    if(satisfied exit criterion )
25.      finished_nr=true;
26.  }
27.  armutation( $p_a$ );
28.  retain best arpopnum populations
29.  if (satisfied exit criterion )
30.    finished=true; }
```

图 3 拓扑综合的 GA 伪代码

(1) 初始化种群

两级 GA 技术初始化产生 $arnum$ 个第一级个体 $arstr$ 和 $nrnum$ 个第二级个体 $nrstr$ 。一旦第一级个体产生了,GA 就调用 $initial_rrstr()$ 过程来产生对应的路由器/路由器的准全连接矩阵 $rrstr$,路由器/路由器的连接是一个二维对称的二进制矩阵。

首先,第一级种群 $arstr$ 是随机地赋值“1”或“0”给所有个体的 $arstr$ 数组元素来产生的,这一级的个体必须满足前面的第一级合法性条件。同时产生对应的第三级准全连接矩阵 $rrstr$,对于所有的路由器,如果路由器 i 与路由器 j ($i < j$) 之

间的距离小于或等于所允许的最大长度,则 $rrstr$ 矩阵的上三角元素 $rrstr[i][j]$ 被赋值为“1”,否则赋值为“0”,而 $rrstr$ 的下三角元素被赋值为其对应的上三角元素值;主对角线上的元素都赋值为“0”。其次,第二级的节点/路由器的映射数组 $nrstr$ 是通过为每一个节点 i 随机地分配给结构中的一个路由器来初始化的, $nrstr[i]$ 存储节点 i 所映射的路由器编号,第二级个体必须满足第二级的合法性条件。

(2) 适应度函数

GA 的选择操作根据适应度值保留最好的一些个体来产生下一代种群。两级 GA 技术的目标是 minimized NoC 的通信能耗和所使用的路由器资源。为了惩罚违反路由器端口数的限制的个体,适应度函数应该包含个体中所包含的最大连接数,因此个体的适应度函数定义为式(1):

$$fitness(solution) = \frac{1}{\alpha \times p + \beta \times r + \gamma \times l} \quad (1)$$

式中, p 是个体对所有通信交通的总能耗,个体的能耗是 AC-CG 中所有通信交通的能耗之和; r 是拓扑结构中所使用的路由器数, l 是拓扑中所有路由器的最大连接数,而 α, β 和 γ 分别是 p, r 和 l 的权值。

一旦个体确定了 NoC 的网络拓扑,即路由器被选择出来,就产生了节点/路由器映射和路由器/路由器映射,我们可以使用改进的 Dijkstra 最短路径算法来为每个通信交通产生静态路由,改进的 Dijkstra 算法可以满足 NoC 的最大带宽限制。节点与路由器距离和路由器之间距离是以它们的能耗来衡量的,由于节点和路由器可能的物理位置已经确定,我们就能够计算节点与路由器及路由器之间的曼哈顿距离,因此,一个通信交通 $e \in E$ 从路由器 r_1 流到路由器 r_2 所需的能耗可以由式(2)来计算:

$$\begin{cases} (P_i + P_o + dist(r_1, r_2) \times P_l) \times \omega(e), \\ rrstr[r_1][r_2] = 1, \text{且路由器 } r_1 r_2 \text{ 没有带宽违例} \\ \infty, \text{其它} \end{cases} \quad (2)$$

式中, $dist(m_1, r_2)$ 是路由器 r_1 和 r_2 之间的曼哈顿距离。类似地,一个通信交通 $e \in E$ 从节点 n_1 流到路由器 r_2 所需的能耗可以由式(3)来计算:

$$\begin{cases} (P_i + dist(n_1, r_2) \times P_l) \times \omega(e), \\ dist(n_1, r_2) \leq L_{max}, \text{且路由器 } r_2 \text{ 没有带宽违例} \\ \infty, \text{其它} \end{cases} \quad (3)$$

式中, $dist(m_1, r_2)$ 是节点 n_1 与路由器 r_2 之间的曼哈顿距离, L_{max} 为单时钟周期所允许的信号最大传播距离。

个体的能耗是完成所有通信交通时所有路由器和链路的能量消耗之和,也就是:

$$P = \sum_{e \in E} (\sum_{r \in R} E_r + \sum_{l \in L} E_l) \quad (4)$$

然而,对于某个个体,如果有某个通信交通没有从源到目标的合适路径,那么该个体的能耗就被设置为无穷大,这样,该个体就会被选择操作淘汰。

3.4 基因操作

本节分别介绍两个级别的交叉和变异操作,由于第三级采用准全连通网络,其连接矩阵不参与基因操作,而第一、二两级个体数据结构为一维数组,在此,我们选择了单点交叉操作。

(1) 交叉操作

• NoC 体系结构级交叉操作

这一级的交叉操作应用于 $arnum$ 个 NoC 体系结构级的每一个个体实例来产生新的 $arnum$ 子代个体 $arstr_cross$ 。其交叉操作如图 4 所示,两个随机选择的个体是从父代中选择出来的,一个随机的交叉点把每个个体分成 $A-B$ 和 $A'-B'$ 两部分,通过串接数组 $A-B'$ 和 $A'-B$ 来产生两个新的个体。如果新产生的个体不符合第一级的合法性条件,则交叉结果被拒绝,交叉过程重新选择两个个体来重复交叉过程。一旦交叉成功,就对两个新生成的个体产生对应的第二级个体。GA 技术复制尽可能多的第二级个体到子代个体中, A 和 A' 对应的第二级部分数组映射都复制到子代中,而 B' 和 B 对应的部分数组如果没有违反第二级合法性标准就都复制到子代中,如果违反合法性条件,就随机地映射到结构的一个符合合法性条件的路由器。

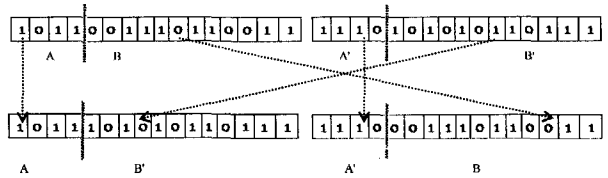


图 4 第一级交叉操作

• 节点/路由器映射级交叉操作

第二级的交叉操作应用于 $nrnum$ 个节点/路由器映射级的个体来产生新的 $nrnum$ 个子代个体 $nrstr_cross$ 。交叉操作随机地选择两个个体 $nrstr[m]$ 和 $nrstr[n]$ 来进行交叉产生两个子代的个体 $nrstr_cross[i]$ 和 $nrstr_cross[i+1]$, 被选择的两个个体必须属于同一个 NoC 体系结构级个体,由于这级交叉操作不影响第一级的个体,即第一级的路由器数量保持不变,其对应的第三级准全连接矩阵也不用改变,因此对应的第一和第三级的数组可以不做任何修改直接继承下来。交叉操作如图 5 所示,一个随机交叉点分别把数组 $nrstr[m]$ 和 $nrstr[n]$ 分成 $A-B$ 和 $A'-B'$, 然后把 A 和 B' 及 A' 和 B 串接成两个子代个体 $nrstr_cross[i]$ 和 $nrstr_cross[i+1]$ 。最后相应的每个路由器的节点连接数数组 $linknr$ 和辅助的路由器连接数数组 $linkednum$ 被重新计算。

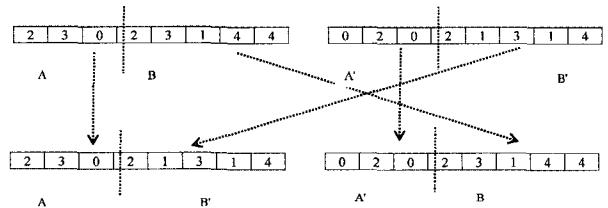


图 5 二级交叉操作

(2) 变异操作

在这一小节分别介绍两个级别的变异操作,变异操作从当前个体的染色体中随机地选择一个基因进行改变来产生新的个体。变异操作能够提高 GA 的局部搜索能力和保持种群的多样性。

• NoC 体系结构级的变异

这级的变异操作应用于当前 $arnum$ 个 NoC 体系结构级个体来产生 $arnum$ 个下一代变异个体 $arstr_mut$ 。由于路由器选择级个体是一个二进制数组,当进行变异时,变异操作随机地选择一个位置进行取反,也就是说,如果选择的是“0”,它被取反为“1”表明增加一个路由器到 NoC 的结构中来。这

时,对应的第三级全连接矩阵被重新生成,而对应的第二级个体由于未受到影响可以全部不做任何改变地继承下来。另一方面,如果所选择的位是“1”,它被取反为“0”表明从 NoC 结构中移除一个路由器。这种情况比较复杂,因为移除一个路由器可能导致违反总端口数的合法性条件,这时,GA 技术必须增加另外一个路由器。然后,GA 保留所有的没有受到影响的节点/路由器映射。由于移除一个路由器可能使得某个节点失连,因此把所有映射到被移除的节点改变为没有映射状态。在满足第二级合法性条件下,对所有第二级的个体,把没有映射的节点随机地映射到某个路由器。最后,把对应的第三级准全连接矩阵重新生成,并计算相应的辅助数组。

· 节点/路由器映射级的变异

这一级的变异操作应用于 n_{rnum} 个当前代个体来产生新的 n_{rnum} 个变异个体 $nrstr_mut$ 。变异操作随机地选择一个节点来改变其映射,它被映射给另外一个不同于原来映射的路由器,而其它节点/路由器的映射都被继承下来。如果变异操作导致违反第二级的合法性条件,那么该变异操作被拒绝,变异过程重复执行直到产生合法的个体。由于这级的变异操作没有改变 NoC 结构的路由器数量,一旦变异成功,变异操作就继承相应的第一级个体和第三级全连接矩阵,同时修改相应的路由器的节点连接数和连接总数两个数组。变异操作有一种特殊的情况,即 NoC 结构中仅有唯一的一个路由器,所有节点都连接在这个路由器上,这时变异操作直接返回。

4 实验结果及分析

两级 GA 技术以应用通信特征图 ACCG、片上系统 SoC 的平面布局信息 Floorplan 以及网络互连部件特征参数为输入。我们用 3 个多媒体应用标准程序来测试特定应用拓扑优化结果,把实验结果与现有的三级 GA 技术^[5,6]进行比较,以讨论两级 GA 技术的性能。

4.1 实验设置

(1)标准多媒体测试应用程序介绍

以 263dec/mp3dec,263enc/mp3dec 和 mp3enc/mp3dec^[11] 这 3 个组合的多媒体应用标准测试程序来评估两级 GA 技术的性能,这些多媒体应用的特征如表 1 所列。所有处理核的尺寸都是随机生成的,面积大约为 $1.5 \times 1.5 = 2.25\text{mm}^2$,其应用通信特征图 ACCG 请参照文献[11]。

表 1 多媒体标准应用程序

graph	Graph ID	Nodes	Edges
H. 263 DEC MP3 DEC	G1	14	15
H. 263 ENC MP3 DEC	G2	12	12
MP3 ENC MP3 DEC	G3	13	12

(2)GA 相关参数设置

GA 参数对算法运行时间有着很大的影响。从图 3 可以知道,两级遗传算法的时间复杂度为 $O(n^2)$ 。为了对搜索空间和算法运行时间进行有效地折衷,我们调节 GA 参数为 1024 个种群个体,它被分 32 个 NoC 体系结构级个体,对每个第一级个体,其对应的节点/路由器级的个体也是 32 个。第一级和第二级基因操作的概率设置为 0.1 和 0.5,算法终止的条件设置为最小能耗连续 $|V| + |E|$ 次迭代保持不变。

4.2 实验结果及讨论

两级 GA(2H-GA)采用两级的个体来表示 NoC 拓扑结构的路由器选择和节点与路由器映射,路由器之间采用准全

连接方式,而三级 GA(3H-GA)采用 3 个级别的个体来表示路由器选择、节点与路由器映射以及路由器与路由器映射。由于遗传算法总是给出问题的近似最优解,在 3H-GA 中,第三层路由器与路由器之间的连接有可能得到的就是近似优化的映射,而 2H-GA 对路由器之间的连接采用准全连接的网络,然后用最小路径算法从这些连接中选择出最小的路径来路由所有通信交通,这样在第三级的路由器与路由器间映射就是最优的路由器与路由器映射。因此,2H-GA 可以得到比 3H-GA 更优的解,即更小的最小能耗。同时,由于 3H-GA 的时间复杂度为 $O(n^3)$,而 2H-GA 的时间复杂度为 $O(n^2)$,因此,2H-GA 完成一次迭代的操作次数比较少,其运行时间就较短。对于 3 个多媒体应用,两种 GA 技术的最小能耗和运行时间对比如表 2 所列,相对 3H-GA,2H-GA 对最小能耗平均有 1.14% 的改进,其理由是 3H-GA 仅能得到近似优化的路由器/路由器映射,而 2H-GA 方法的路由器/路由器映射采用准全连接,继而用 Dijkstra 最短路径算法为通信交通得到最短的路由。虽然能耗改进不大,但是由于算法的时间复杂度从原来的 $O(n^3)$ 降低为 $O(n^2)$,其运行时间却得到了很大的减少,平均减少 97.88%。

表 2 两级 GA 与三级 GA 实验数据比较

Bench marks.	Minimum energy (μW)			Runtime (Sec)		
	3H-GA	2H-GA	Improv.	3H-GA	2H-GA	Improv.
G1	972.567	965.11	0.77%	615.1	12.5	97.97%
G2	27985.9	27884.3	0.36%	283.63	7.23	97.45%
G3	1610.5	1573.84	2.28%	526.3	9.44	98.21%
average			1.14%			97.88%

结束语 基于准全连通和准全连接矩阵的概念,本文提出了一种两级遗传算法来对特定应用片上网络的拓扑结构进行优化。二级 GA 对路由器之间的连接采用准全连接的网络,然后用最小路径算法从这些连接中选择出最小的路径来路由所有通信交通,这样在第三级的路由器与路由器间映射就是最优的路由器与路由器映射。因此,两级 GA 可以得到比三级 GA 更优的解,即更小的最小能耗。实验结果表明,对于典型的 7 个多媒体应用,相对于文献[5,6]中提出的三级遗传算法,两级遗传算法的通信能耗有一定程度的改善,大约减少了 1.1%。但是,由于两级遗传算法对个体的基因操作相对简单,因此,算法的运行时间大为减少,大约减少 97%。

参考文献

- [1] ITRS2010. International Technology Roadmap for Semiconductors Interconnect Technical Report of International Technology Roadmap for Semiconductors [R]. Semiconductor Industry Association, 2010
- [2] Hemani A, et al. Network on a Chip: An architecture for billion transistor era [C]// Proceeding of IEEE NorChip Conference. 2000:166-173
- [3] Lai G, Lin X. Floorplan-aware application-specific network-on-chip topology synthesis using genetic algorithm technique[J]. The Journal of Supercomputing, 2011(4)
- [4] Lai Guo-ming, et al. GA-Based Floorplan-Aware Topology Synthesis of Application-specific Network-on-chip [C]// Proceedings of 2010 IEEE International Conference on Intelligent Computing and Intelligent Systems. Xiamen, 2010, 2:554-558

(下转第 64 页)

的支持度函数矩阵偏差较大,通过支持度矩阵得到加权因子再进行加权融合会导致系统跟踪精度的降低;而基于全局状态估计得到的 TQMM 是对观测航迹误差大小的有效表征。观测精度高的点具有较大的 TQMM,相反,观测精度低的点将获得较小的 TQMM。无论传感器的观测精度如何变化,基于 TQMM 分配权值,可使精度高的观测航迹分配到较大的权值,而精度低的观测航迹得到较小的权值,因此,加权融合后的航迹更接近于真实航迹。

2)引入从融合中心到局部节点的反馈信息。将系统状态估计的一步预测反馈到局部节点,前一时刻的融合估计参与到该融合时刻的权值分配,突破了 DTFFC 算法仅利用局部状态估计分配权值的局限性,使权值分配更加合理,即使在传感器观测精度相差较大的情况下,也能实现高精度跟踪。

并且,历史权因子 α 对算法跟踪性能的影响很小,便于调试。随着传感器数目的增加,系统的跟踪性能逐渐提高,但传感器数目大于 4 个后,系统的融合精度并没有得到明显的改善。在工程应用中应该综合考虑跟踪性能和系统复杂度的关系,选择适当数目的传感器,使得既能获得较高的跟踪精度,又能使融合中心进行实时处理,尽可能地降低工程造价。

结束语 本文对加权航迹融合算法进行系统研究,提出了多模型航迹质量的概念,并引入了从融合中心到局部节点的反馈信息,给出了一种基于多模型航迹质量的融合算法。为了提高算法的跟踪性能和系统的稳定性,在系统中引入反馈机制。反馈环节势必会增加融合中心的计算量以及融合中心与局部节点间的通信量,但反馈过程仅仅是对全局状态的一步预测,计算量和通信数据量均较小,不会给系统增加过多的额外开销。因此,TFQMM 算法既有较高的跟踪精度,也不会影响系统的实时性。

参 考 文 献

[1] 费业泰. 误差理论与数据分析(第五版)[M]. 北京:机械工业出版社,2008:136-172
 [2] 吴顺君. 雷达信号处理和数据处理技术[M]. 北京:电子工业出

版社,2008:112-165

[3] Wu Sheng-li, Bi Ya-xin, Zeng Xiao-qin, et al. Assigning Appropriate Weights for the Linear Combination Data Fusion Method in Information Retrieval[J]. Information Processing and Management, 2009, 45: 413-426
 [4] Wu S, McClean S. Data Fusion with Correlation Weights[C]// Proceedings of the 27th European Conference on Information Retrieval. 2005: 275-286
 [5] Anand B, Anandaraj M, Kapil K, et al. Multi-Sensor Data Fusion Methods Using Sensor Data Compression and Estimated Weights[C]// IEEE International Conference on Signal Processing, Communications and Networking, Madras Institute of Technology. Anna University Chennai India, 2008, 250-254
 [6] 李伟, 何鹏举, 高社生. 多传感器加权信息融合算法研究[J]. 西北工业大学学报, 2010, 28(5): 674-679
 [7] 俞黎阳, 王能, 张卫. 无线传感器网络中基于神经网络的数据融合模型[J]. 计算机科学, 2008, 35(12): 43-47
 [8] 陈小惠, 万德钧, 王庆. 模糊逻辑在分布式多目标跟踪融合中的应用研究[J]. 东南大学学报: 自然科学版, 2003, 33(6): 754-757
 [9] 冉金和, 张玉. 基于航迹隶属度的分布式系统数据融合算法[J]. 信号处理, 2011, 27(2): 226-229
 [10] 马慧萍, 王永海, 张伟, 等. 一种实时目标跟踪航迹数据融合算法[J]. 计算机仿真, 2010, 27(11): 44-48
 [11] 全丽, 张伟, 张可. 一种滤波系数化的分布式航迹融合方法[J]. 计算机工程与应用, 2011, 47(36): 141-144
 [12] 全丽. 机载多传感器数据融合目标跟踪技术研究是实现[D]. 成都: 电子科技大学, 2012
 [13] Jeffery T. Track quality estimation for multiple-target tracking radars[C]// IEEE Radar Conference. 1989: 76-79
 [14] Tafti D, Sadati N. Novel Adaptive Kalman Filtering and Fuzzy Track Fusion Approach For Real Time Applications[C]// 3rd IEEE Conference on Industrial Electronics and Application. 2008: 120-125
 [15] 韩崇昭, 朱洪艳, 段战胜, 等. 多源信息融合[M]. 北京: 清华大学出版社, 2010: 368-390

(上接第 48 页)

[5] Leary G, et al. Design of Network-on-Chip architectures with a Genetic Algorithm-based technique [J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2009, 17(5): 674-687
 [6] Srinivasan K, et al. Linear Programming based techniques for synthesis of network-on-chip architectures [C]// Proceedings of the IEEE International Conference on Computer Design (ICCD'04). 2004: 422-429
 [7] Srinivasan K, et al. Linear Programming based techniques for synthesis of network-on-chip architectures [J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2006, 14(4): 407-420
 [8] Murali S, et al. Designing application-specific networks on chips with floorplan information [C]// Proceedings of the 2006 IEEE/ACM International Conference on Computer-aided Design. 2006: 355-362
 [9] Garey M R, Johnson D S. Computers and intractability: A guide to the theory of NP-completeness [M]. Freeman, 1979: 121-130
 [10] 邢文训, 谢金星. 现代优化计算方法(第 2 版)[M]. 北京: 清华大

学出版社, 2005: 1-147

[11] Hu J, Marculescu R. Energy-aware mapping for tile-based NoC architectures under performance constraints [C]// Proceedings of the ASP-DAC 2003 Design Automation Conference. 2003: 233-239
 [12] Lin J, Lin X. Power and latency efficient mechanism; a seamless bridge between buffered and bufferless routing in on-chip network[J]. The Journal of Supercomputing, 2012, 61(3): 1048-1067
 [13] Lin Jing, et al. Making-a-stop: A new bufferless routing algorithm for on-chip network [J]. J. Parallel Distrib. Comput., 2012, 72(4): 515-524
 [14] Tang Ming-hua, Lin Xiao-la. Injection Level Flow Control for Networks-on-Chip (NoC) [J]. J. Inf. Sci. Eng., 2011, 27(2): 527-544
 [15] Hu Su, Lin Xiao-la. A Symmetric Odd-Even Routing Model in Network-on-Chip[Z]. ACIS-ICIS. 2012: 457-462
 [16] Tang Ming-hua, Lin Xiao-la. Rqrt: Reduce Querying Routing Table for Mesh-Based Network-on-Chip[J]. Journal of Circuits, Systems, and Computers, 2011, 20(8): 1529-1545