彩色编码技术的研究进展及应用 *)

刘云龙 王建新 陈建二
(中南大学信息科学与工程学院 长沙 410083)

摘 要 作为一种重要的参数化技术,彩色编码技术得到了越来越多的重视并在近年来取得了理论和应用上的一系列进展。本文首先介绍了彩色编码技术的基本思想和相关定义,并详细论述了随机式和确定式两种彩色编码技术。最后本文具体介绍和分析了彩色编码技术在路径查找、子图同构、matching 与 packing、(t,n)-环签名等问题上的应用,并探讨了彩色编码技术及其应用的进一步研究工作。

关键词 彩色编码,NP 难问题,复杂性理论,参数计算

The Evolution and Application of Color-coding

LIU Yun-Long WANG Jian-Xin CHEN Jian-Er
(School of Information Science and Engineering, Central South University, Changsha 410083)

Abstract As one of the most important FPT classification techniques, color-coding is attracting more and more research concerns and has achieved a series of breakthroughs both theoretically and pragmatically in recent years. This paper first introduces the essentials and related definitions of color-coding technique, and details the randomized and deterministic color-coding respectively. Then this paper makes a brief introduction to the application of color-coding in problems such as path-finding, subgraph isomorphism, matching&packing and (t,n)-ring signature. Moreover, this paper makes a discussion on color-coding and its future works.

Keywords Color-coding, NP-hard problem, Complexity theory, Parameterized computation

1 引言

计算难解性问题是计算机科学和应用中最根本和最重要的问题之一。目前被接受的方法主要有以下几种:近似算法、随机算法、启发式算法和平均情况分析。但是这些方法都有各种不同的局限性。本文将介绍一种近年新兴的一种算法思想——color-coding(彩色编码技术),阐述其设计思想的发展和应用。

Alon 等人通过对 k-PATH 问题的研究,于 1995 年首次提出了名为 color-coding(彩色编码)的算法技术[1]。k-PATH 问题可以描述为:给定一个图 G 和一个整数 k,要求在图 G 中找到一个包含 k 个节点的简单路径。k-PATH 是著名的 NP 难问题,最早提出的算法时间复杂度为 $O(2^k k!\ n^{O(1)})$[5,24]。文[1]指出 k-PATH 问题的难点在于对路径简单性的要求(即路径节点不重复),并阐明了解决该问题的本质正是在图 G 的所有的 n 个节点中选择 k 个节点构成一个子集。基于这个观察,文[1]首次提出了利用彩色编码技术求解 k-PATH 问题,并得到了时间复杂度为 $O(2^{O(k)} n^{O(1)})$ 的算法。该算法通过使用 k 种不同的颜色对图中的所有节点进行着色,并假定被查找的简单路径(即目标解)上的 k 个节点正好和 k 种颜色一一对应,从而将节点不重复的限制条件简单地转化到颜色不重复的限制条件,达到了简化问题的目的。

在彩色编码技术产生后,通过研究者们的努力,彩色编码技术本身及其应用的方法都得到了不断的改进。近年来,大量的文章对该技术都有所涉及。我们在此对其相关问题进行总结,希望人们能对此问题有更全面的了解,也有利于我们在该问题上有更加深入的研究。

本文第 2 节介绍了彩色编码技术的基本思想和相关定义;第 3 节将分类介绍其核心技术即着色方案构造的算法,特别重点分析其确定化技术;第 4 节介绍彩色编码技术在各种不同领域的应用;最后是关于彩色编码技术研究的结论和展望。

2 彩色编码的思想及其定义

在上面一节中,我们看到了彩色编码技术产生的背景,即 k-PATH 问题。从上面对 k-PATH 问题的描述中可以看到,k-PATH 问题是一个典型的子集元素选择问题。由此可以自然地推断,彩色编码这种用于求解 k-PATH 的技术也可以被应用到和 k-PATH 类似的子集选择问题上。所以,尽管 Alon 等人是将着色技术作为 k-PATH 问题的一种解决方案提出的,但它在降低问题复杂度上的显著优势使得它在解决涉及特定子集选择的问题上有着广泛的应用。然而,需要注意的是,彩色编码的技术本身并不直接解决目标问题,而是通过额外的限制条件有效地降低了目标问题的搜索空间。在求解某些问题时,我们需要枚举所有的元素组合情况,如同前面所指出的,这种穷举算法的时间复杂度将达到 C_n^k。而利用彩色编码技术后,一种着色可以覆盖大量不同的组合情况,从而可以有效减少搜索空间,降低了问题的规模。所以,彩色编码的

*)本课题得到国家自然科学基金重点项目“生物信息学中的相关组合理论和算法研究”(60433020)和新世纪优秀人才支持计划(NCET-05-0683)资助。**刘云龙** 硕士研究生,主要研究领域为生物计算、计算复杂性理论;**王建新** 博士,教授,博士生导师,主要研究领域为计算机算法、网络优化理论、生物信息学;**陈建二** 博士,长江学者特聘教授、博士生导师,主要研究领域为生物信息学、计算机理论、计算复杂性优化。

基本思想可以描述为：给定元素集合 U 和颜色集合 C，其中 $|U|=n$，$|C|=k$，将 U 中的每个元素都使用 C 中的一种颜色进行着色。通过假定原问题的目标解包含的 k 个元素正好被着色为 k 种不同的颜色，原问题将获得额外的着色约束条件，从而得到简化。

为了便于下面做进一步的说明，现在给出本文中对彩色编码的形式化描述。目前文献中对于彩色编码中使用的各种概念的描述形式还不统一，但本质是一致的。大部分文献倾向于以映射的观点来描述彩色编码的着色，即将着色表示为 $\{1,\cdots,n\}\rightarrow\{1,\cdots,k\}$[1]；还有部分文献采用划分的思想来描述着色，即将着色表示为对一个全集的 k-划分。本文则沿用文[18]中向量的描述方法。

定义 1[(n,k)着色]　给定元素全集 $U=\{e_1,e_2,\cdots,e_n\}$ 和颜色全集 $C=\{c_1,c_2,\cdots,c_k\}$，将集合 U 中的所有元素使用 C 中的任意颜色进行着色 $h_i=f(e_i)$（其中 $f(e)$ 表示对元素 e 赋予一种颜色），得到一个 n 元组 $H=\langle h_1,h_2,\cdots,h_n\rangle$，且 $\forall i$，$h_i\in C$，$\bigcup_i^n\{h_i\}=C$，即 U 中每个元素均可对应一种颜色，且颜色全集中每种颜色至少被使用过一次，则称该 n 元组为一个 (n,k) 着色。

注意到，定义 1 中将着色作为 n 元组考虑。尽管形式有异，但这个定义和其他文献中的映射是本质上等价的。所以，在本文的其他部分也使用 $f(e_i)$ 的形式表示着色的映射形式。

定义 2[覆盖]　给定一个 (n,k) 着色 $H=\langle h_1,h_2,\cdots,h_n\rangle$ 和一个子集 W，其中 W 是从 $U=\{e_1,e_2,\cdots,e_n\}$ 中任意选取 k 个元素所组成的子集。对于任意 i 和 j，若 $e_i,e_j\in W$ 且 $i\neq j$，有 $h_i\neq h_j$，则称着色 H 覆盖子集 W。

定义 3[(n,k)着色方案]　若 (n,k) 着色的集合满足：任取 U 的一个 k 元素子集 W，至少存在一个可以覆盖 W 的 (n,k) 着色，则称该集合为 (n,k) 着色方案，记做 $S(n,k)$。方案包含着色的数目称为着色方案的规模，记做 $|S(n,k)|$。

3　着色方案的构造

基于彩色编码的算法复杂度极大地依赖于需要枚举的着色数目。根据算法使用的着色集合特性，彩色编码可以分为确定式和随机式两种类型。其中，使用 (n,k) 着色方案的算法被称为确定式算法；否则，算法属于随机式彩色编码，即算法以一定概率得到正确的解。

3.1　随机式着色方案

在文[1]中，在首次提出彩色编码思想的同时，Alon 等人就详细介绍了随机式彩色编码在 k-PATH 中的应用。对于在无向图 G 中查找 k-PATH 的问题，正确解所包含的 k 个节点有 $k!$ 种可能性被正确着色，即每个节点的颜色不同；而 k 个节点被着色有 k^k 种不同的着色情况，则在均匀概率着色的情况下，正确着色的概率就是 $k!/(k^k)$。应用 Stirling 近似公式：

$$n!=\sqrt{2\pi n}(n/e)^n(1+\Theta(1/n))$$

可知一次着色正确的概率是 $O(e^{-k})$。

根据这个计算结果，在使用随机着色方法求解 k-PATH 的同时，Alon 等人也指出需要通过足够的重复次数来保证获得的解具有一定的准确率。在实际求解的过程中，通常着色次数取 $O(e^k)$。

3.2　确定式着色方案

在求解 k-PATH 问题过程中，尽管在 $O^*(5.44^k n^{O(1)})$ 的时间复杂度下随机式彩色编码算法可以得到较高的准确率，但在一些情况下，特别是对于结果的精度有较高要求的情况下（甚至是只有精确解可用），这样的概率解是不能令人满意的，甚至是不可用的，而由确定式算法产生的精确解则更为合适。

为了使着色确定化，Alon 等人在文[1]中指出：在给出一个从 $\{1,2,\cdots,n\}$ 映射到 $\{1,2,\cdots,k\}$ 的 k-完全散列函数族（k-perfect hash functions family）的前提下，可以得到一个确定式着色方案。

定义 4[完全散列函数]　有集合 $U=\{1,2,\cdots,n\}$ 和 $C=\{1,2,\cdots,k\}$，若对于集合 $W\subseteq U$ 中任意两个元素 i,j 均有 $f(i)\neq f(j)$，且 $f(i),f(j)\in C$，则称 f 是 W 的完全散列函数（perfect-hash function）。若对于任何 $W\subseteq U$，$|W|=k$，均有 $f\in F$ 是 W 的完全散列函数，则称 F 是 k-完全散列函数族。

以完全散列函数来构造确定式着色方案的方法均使用如下形式的散列函数形式：

$$g_{a,b,s}(x)=((ax+b)\bmod p_n)\bmod s$$

其中 p_n 是大于 n 的素数。而根据 Bertrand's Conjecture，对于任意的整数 n，都存在素数 p_n，满足 $n\leqslant p_n\leqslant 2n$。

使用完全散列函数来构造确定式着色方案的理论基础则来自于文[13]中 Fredman 等人对散列函数族的研究结果。Schmidt 和 Siegal 在 Fredman 等人的研究基础上，给出了一种 k-完全散列函数族的构造方法[26]。他们使用 $O(k)+2\log\log n$ bits 表示每个函数，从而散列函数族的规模为 $2O(k)\log^2 n$。具体的实现包括一个 $\log n$bit 的串、一个 $4k$bit 长的串 T_0、一个 $k$bit 长的串 T_1、一个 $4k$bit 长的串 T_2，以及一个 $3k$bit 长的串 K_0。这个散列函数是一个从 Z_n 到 Z_{3k} 的单射。在 $(3k,k)$ 着色方案已知的情况下即可得到 (n,k) 着色方案。

Moni Naor 随后就改进了这个算法。改进后的算法首先构造了 $\{1,2,\cdots,n\}$ 到 $\{1,2,\cdots,k^2\}$ 的 k^2 完全散列函数族；然后构造 $\{1,2,\cdots,k^2\}$ 到 $\{1,2,\cdots,k\}$ 的 k 完全散列函数族。算法的第二部分沿用了 Schmidt 的方法，而在第一部分使用了概率空间的方法来证明散列的有效性。复合两个函数族得到的结果规模为 $2O(k)\log n$[25]。

但需要注意的是，这些结果距离实用仍有相当大的距离。根据上面描述的文[26,13]中的构造方法，表示这样一个散列函数需要多于 $12k$ bits，枚举所有的散列函数需要至少 $2^{12k}>4000^k$，直接使用这种着色方案将使着色算法即使在 k 较小的时候也不太实用。

最近，关于基于完全散列函数的着色算法研究大幅改善了这个结果。通过重新构造散列函数和附加一定的限制条件，J. Chen 等人将完全散列函数分为三步完成：$Z_n\rightarrow Z_{k^2}\rightarrow Z_{k/4}\rightarrow Z_{c_j(c_j-1)}$，然后通过已知的 $(c_j(c_j-1),c_j)$ 着色方案，可以得到规模为 $O^*(6.1^k)$ 的着色方案[19]。在具体构造的过程中，该算法首先使用核心化方法将 (n,k) 着色问题在多项式时间内归约到 (k^2,k) 着色问题，然后将元素散列到 $k/4$ 个位置上，再利用另一组散列函数分别将每个位置的元素重新散列到 $c_j(c_j-1)$ 个位置上来解决冲突，最后使用已知的着色方案来构造总体的着色方案。在构造过程中，J. Chen 等人通过对散列函数的深入分析，有效地降低了所需的散列参数的数目，大幅改进了基于完全散列函数的构造算法。此外，文[19]还指出了在 $n\gg k$ 的情况下，着色方案规模的下界是 $\Omega(e^k)$。

4　彩色编码技术的应用

4.1　*k*-PATH 及其相关问题

作为彩色编码的第一种应用，k-PATH 在过去的 10 年中得到了不断的改进。在彩色编码的第一篇文献中，Alon 等人在图中使用着色技术和动态规划的方法将 k-PATH 的时间复杂度从 Bodlaender 等人算法的 $O(2^k k!\cdot V)$[5] 改进到了 $2^{O(k)}\cdot E\log V$；并将类似的 k-CYCLE 问题从 Monien 的 $O(k!\cdot VE)$ 改进到了 $O(2^{O(k)}\cdot V\omega\log V)$[1]。另外，值得注意的是，在 k-PATH 问题应用的彩色编码算法分为随机和确定两种类型，其中随机的 Monte Carlo 方法可以在 $O((2e)^k nm)$ 的时间内得到一定的正确率，而确定式方法直到文[19]才被改进到 $O(12.2^k nm)$。

通过应用彩色编码技术及动态规划方法的不断改进，k-PATH 问题在许多情况下的求解代价已经可以接受。在文[27]中，Jacob Scott 等人在酵母菌的蛋白质交互网络中应用彩色编码的方法查找蛋白质路径，并且在给定某些特定生物背景的限制条件下，算法可以在线性时间内运行。经过试验，文[27]中的算法可以在 1min 内查找到 8-PATH，2h 内查找到 10-PATH。Tomer Sholomi 等人进一步实现了这种技术，他们以基于彩色编码的路径查找技术为基础，实现了一个用于查找生物网络的框架，名为 Q-PATH。Q-PATH 已被实现为一个著名生物信息学软件 Cytoscape 的插件。在文[11]发表时，他们已经使用该工具在酵母菌中查找了 271 条路径。最近，Falk Hufführer 等人又使用了一种新的方法来使用彩色编码技术。除启发式算法之外，他们发现可以在 k-PATH 问题上使用超过 k 种颜色进行着色，并通过调整使用的实际颜色数目 k′在每次尝试所需的时间和尝试次数中取得折中。通过理论分析，使用 $k'=1.3k$ 种颜色，在错误率为 ε 的情况下，运行时间为 $O(|\ln\varepsilon| * 4.32^k m)$。实现后的算法可以在数秒内计算 13-PATH[16]。而在完成了图合并算法之后，这种方法在文[20]中也在多图路径查找中得到了应用。

k-PATH 的应用显然不限于生物信息学，很多现实世界的应用都可以被抽象为在给定图中查找指定长度或最长路径的问题。例如，交通网络就可以很容易地被抽象为图，而线路规划问题就是公众和铁路交通规划中的一个基础问题。交通线路规划需要寻找交通网络中能满足特定需求的线路，例如最小化建设费用，或是最小化旅行次数等等。在文[7,8]中 Borndörfer 就使用彩色编码技术来查找最大权重的 k-PATH，并将其应用于路径规划。

从这些列举的应用可以看出，只要对彩色编码和 k-PATH 问题的算法进行不大的改动，类似于加入节点的限制、附加的权重、子图的类型等等，这些技术就可以被应用到很多其它的问题上。也就是说彩色编码存在一系列潜在的应用，例如文[6]指出的，对限制带宽路径查找的改进。

4.2 子图同构问题

子图同构问题是模式匹配中的一个非常重要且常见的问题。许多重要的图论问题都可以归约到子图同构，例如哈密尔顿回路、团、匹配、周长、最短路径等。而子图同构也在很多实际问题中有重要应用，如分子结构比较、集成电路测试、微程序控制器优化等。一般化的子图同构问题可以形式化地定义为：给定图 G 和 Q，确定图 G 中是否存在子图 W 与 Q 同构。尽管一般化的子图同构问题仍然没有比 $O(n^{|H|})$ 更好的结果，但是在指出彩色编码在 k-PATH 问题上应用的同时，文[1]也指出当图 Q 为森林时，通过简单的修改，应用彩色编码到 Q 就可以得到 $2^{O(k)}\cdot E$(有向)和 $2^{O(k)}\cdot V$(无向)平均时间复杂度的算法。进一步，如果图 G 的树宽(图 G 在各种可能的树分解中的最小树宽)不大于 t，那么最高时间复杂度也可以被限制在 $2^{O(k)}\cdot V^{t+1}\log V$。

类似地，彩色编码和动态规划结合方法在文[4]中也被用于在生物网络分析中解决 Steiner Tree 问题。为了解决 Steiner Tree 问题，文[4]将问题的重点放在了解决树同构上，即在给定的图 G 中寻找和指定的树 T 同构的子图，并且达到权重最小。在对图 G 进行着色后，文中算法 WTCC 的大致工作方法如下：首先算法假定 T 中每个节点 n 都对应图中每个节点 v，然后算法记录所有以 v 为根与以 n 为根的子树同构且正确着色(即节点颜色两两不同)的子树所对应的子集。算法不断的迭代过程中以 $W[u,n,j,c]$ 为目标进行动态规划，其中 $u\in V_G$，$n\in V_T$，$0\leqslant j\leqslant d(n)$，$d(n)$ 表示 n 的孩子节点数目，而 c 表示一种着色。算法的总共开销，包括彩色编码和动态规划，共计时间复杂度为 $2^{O(k)}\log|V_G|O(|E_G|k)$。

4.3 Matching 和 Packing 问题

r-Dimensional Matching 问题(Karp 的 3D Matching 问题的泛化形式)可描述为：给定集合 $S\subseteq A=A_1\times\cdots A_r$，其中 $A_1,\cdots,A_r$ 是两两不相交的集合。确定 S 是否存在一个不小于 k 的匹配(Matching) P。而 r-Set Packing 问题(Karp 的 NPC 问题的扩展形式)可以描述为：给定有限集合 S 和 S 的一组子集，其中每个子集的大小均为 r，确定这组子集中是否有 k 个子集是两两不相交的。这个问题的最优化形式即最大化 Set Packing。这两个问题都是在实践和理论中基本的组合问题，也是重要的 NP 完全问题[14]，被广泛应用于任务调度和代码优化等领域。

文[12]系统地阐述了使用彩色编码技术和动态规划方法求解 r-D Matching、r-set packing、graph packing 和 graph edge packing 问题的方法，并将算法从 $O(n+k^{O(k)})$ 改进到 $O(n+2^{O(k)})$。为了将彩色编码有效地使用在 r-D Matching 上，文[12]先为问题构造了一个大小为 $O(k^r)$ 的核 K，并为核计算一个最大匹配 M，通过分析可知匹配 P 中最多只有 $(r-1)k$ 个元素不属于 M。基于这个发现，算法将彩色编码的元素全集置为 $U=\mathrm{val}(K)-\mathrm{val}(M)$，其中 val(K) 表示 K 的值构成的集合，而颜色集合置为 $\{1,\cdots,(r-1)k\}$。通过彩色编码和动态规划的方法，文[12]在 $O(n+2^{O(k)})$ 的时间内求解了 r-D Matching 问题。

通过在着色方案上的突破，使用动态规划求解 matching 和 packing 问题的时间复杂度也得到了改进。对于 3-D Matching 问题，文[19]中指出，对于给定的包含 n 个三元组的集合 S，如果存在包含 k 个互不相交三元组的子集 S_k 存在，并假设 S 的元素被 3k 种颜色着色且 S_k 正好有 3k 种颜色，那么动态规划算法可以在 $O(2^{3k}n)$ 的时间内完成并构造出 S_k，从而得到了总时间复杂度为 $O^*(12.8^{3k}n^2)$ 的算法。类似地，在 3-set packing 问题上也可以得到相同的时间复杂度，而在 Triangle Packing 问题上则得到时间复杂度为 $O^*(12.8^{3k}n^4)$ 的算法。

最近，在文[22]中 J. Chen 等人在深入分析 3-set packing 问题性质的基础上，通过运用彩色编码和动态规划技术，对 3-set packing 问题提出了一个时间复杂度为 $O*(4.61^{3k})$ 的算法。在更加一般化的 set packing 问题上，Ioannis Koutis 也在(k,t)-set packing 问题上使用了彩色编码的技术，构造了一个 $O(2^{O(t)}nN\log N)$ 的确定算法，改进了之前 $O((ck)^k n)$ 的结果[21]。

为了解决文[3]中基因重组和排序问题的应用，Luke 等

人将此类情况抽象为在无向图中查找最大数目的不相交环的问题,并在文[23]使用了皇冠分解技术(Crown Decomposition)和彩色编码的方法解决了 ETP(k-EDGE DISJOINT TRIANGLE PACKING)问题,得到了时间复杂度为 $O(2^{(9k/2)\log k+(9k/2)})$的算法。

4.4 (t,n)-环签名问题

假设有 t 个用户需要发布某个重要信息,并且如果在一个指定范围内所有用户中的至少 t 个用户能够担保信息的可靠性的话,人们就会相信信息的真实性。在这种应用背景下产生了一种叫(t,n)-环签名的技术。在这种技术作用下,被签名的信息包含所有 n 个用户的公钥和 t 个用户的私钥。这种加密技术被广泛应用于电子投票、数字彩票、电子信用卡等。在彩色编码技术产生后,Bresson 等人将彩色编码和传统(t,n)-环签名进行结合,并产生了 Ad-Hoc 环签名技术[9]。与传统(t,n)-环签名不同的是,在 Ad-Hoc 环签名中,存在一个 *Ad-Hoc* 组,即一个用户子集的列表,这些用户子集被称为可接受子集。而为了保持最大的匿名度,要求所有签名用户都属于至少一个可接受子集。为了达到这个要求,同时尽量降低算法开销,文中的算法考虑将用户环进行划分,并使得参与签名的用户在每个子环中都各存在正好一个,这种划分被称为"公平划分"(Fair partition)。可以看到,彩色编码技术正好可以满足这个要求,这里 Bresson 等人就使用了彩色编码的方法对环进行着色,并将着色解释为集合的划分方法。然后使用子环对消息进行签名,最后合并输出结果。值得指出的是,为了保证算法的有效性,算法只能采用确定的着色方案。

而文[17]同样采用了将着色解释为划分的方法来解决$(n-t)$-out-of-n 的签名问题,其中 t 代表的是不参加签名的用户而不是签名的。也就是说,不同于文[9]中的少数签名用户,这里的签名用户占到了多数。文[17]将用户环划分为 $t+1$ 个部分,并指出至少存在一个部分的用户都参与签名。但是为了保证不可否认性,划分应保证任何 $t+1$ 个用户在划分下位于不同的部分中。在为每种划分(着色)产生相应的签名后,算法合并这些签名,得到最后的结果。

4.5 其他参数化问题

描述复杂度问题(Descriptive Complexity Theory)是指给定一个有限结构 A 和符合某种逻辑 L 的一个句子 φ,确定 A 是否满足 φ,也被称为 L 的模式检查问题(Model checking problem)。模式检查被广泛使用在数据库查询中,而 Chandra 和 Merlin 发现在合取查询中的模式检查和同构问题是本质等价并且是 NP 完全的[10]。Martin 在这个问题上使用了彩色编码的技术:首先对 A 进行 k 着色,然后按照着色进行划分,并检查划分后的结构[15]。Martin 在文[15]中采用了 Monte-Carlo 式的随机方法,但他同时也指出了算法也可以用确定式着色来确定化。算法的时间复杂度为 $f(|\varphi|)||A||^c$,其中 f 是一个可计算的函数,而 $c>0$ 是一个常数。

计数问题(Counting Problem):文[2]中混合了使用 Karp-Luby 的近似计数技术和 Alon 的彩色编码技术来计算图 G 中与指定图 H 同构的子图数目。Arvind 在文[2]中应用的是随机化方法,在 $k^{O(k)}n^{O(1)}$ 的运行时间内得到了近似比 $1/k^{O(k)}$、错误率 $2^{-n^{O(1)}}$ 的结果。

结论与进一步研究 在最近的 10 年中,彩色编码技术经历了从产生到发展的过程,并取得了一些重要的进展。从文中我们可归纳出当前的彩色编码技术特点主要有两个方面:

①目前彩色编码技术在应用过程中在绝大部分情况下都分为两个部分:着色和求解。而求解的过程大多基于动态规划方法;

②目前彩色编码技术在理论方面主要被作为小参数理论中的参数化方法使用,而在实际应用过程中仍然是随机化方法占主导地位。确定式方法主要以完全散列函数为理论依据,但由于计算开销仍然较大,在应用上还存在一定的难度。

从文中我们还可以看到,尽管对彩色编码技术的研究取得了一些成果,然而彩色编码还是相当年轻的技术,不论是它本身还是它的应用都还存在很多有待解决的问题,主要包括以下几个方面:

(1)彩色编码技术应用问题的数学刻画;

(2)彩色编码技术中如何降低着色方案规模,以便进一步提高其使用效率;

(3)在实际问题中,如何保存和有效使用着色方案。

参 考 文 献

1 Alon N, Yuster R, Zwick U. Color-coding. Electronic Colloquium on Computational Complexity (ECCC), 1(009), 1994. Full paper appears in J ACM July 1995, 42(4): 844~856

2 Arvind V, Raman V. Approximation algorithms for some parameterized counting problems. In: Proc. 13th ISAAC, 2002. 2518: 453~464

3 Bafna V, Pevzner P A. Genome Rearrangements and Sorting by Reversals. SIAM J Comput, 1996, 25(2)

4 Betzler N. Steiner Tree Problems in the Analysis of Biological Networks. Diplomarbeit, Wilhelm-Schickard-Institut für Informatik, Universität Tübingen, 2006

5 Bodlaender H L. On linear time minor tests with depth-first search. J Algorithms, 1993, 14(1): 1~23

6 Borchert B, Reinhardt K. Searching Paths of Constant Bandwidth. LECTURE NOTES IN COMPUTER SCIENCE, 2006, 3831: 187

7 Borndörfer R. A Path-based Model for Line Planning in Public Transport. Konrad-Zuse-Zentrum fr Informationstechnik Berlin, 2005

8 Borndörfer R, Grötschel M, Pfetsch M E. Models for Line Planning in Public Transport. Konrad-Zuse-Zentrum fr Informationstechnik Berlin, 2004

9 Bresson E, Stern J, Szydlo M. Threshold ring signatures and applications to ad-hoc groups. Advances in Cryptology-CRYPTO, 2002. 18~22

10 Chandra A K, Merlin P M. Optimal implementation of conjunctive queries in relational data bases. In: Proceedings of the Ninth Annual ACM Symposium on Theory of Computing, 1977. 77~90

11 Shlomi T, Segal D, Ruppin E, et al. QPath: a method for querying pathways in a protein-protein interaction network. BMC Bioinformatics, 2006, 7(1): 199

12 Fellows M, Knauer C, Nishimura N, et al. Faster Fixed-Parameter Tractable Algorithms for Matching and Packing Problems. In: Proceedings of the European Symposium on Algorithms (ESA), 2004. 311~322

13 Fredman M L, Komlós J, Szemerédi E. Storing a sparse table with O(1) worst case access time. Journal of the ACM, July 1984, 31(3): 538~544

14 Garey M R, Johnson D S. Computers and Intractability: A Guide to the Theory of NP-Completeness. New York, Nork, NY, USA: Freeman WH & Co, 1979

(下转第 33 页)

个 IP 模块的通信代价。对于第 k ($0 \leqslant k \leqslant n-1$)个 IP 模块，它的通信代价 C_k 等于所有输入/输出的曼哈顿距离之和。即：

$$C_k=\sum_{i=0}^{O_k} d_{ki} \sum_{i=0}^{I_k} d_{ik} \tag{3}$$

Step6：映射调整。如果通信代价 C_k 很大，表明第 k 个 IP 模块需要跟许多个其他 IP 模块进行远距离通信，容易造成通信拥塞，使得整个 NoC 结构的负载很不平衡。因此为了减少 IP 模块之间的通信距离，降低通信功耗，这种情况应该避免。计算该模块对应的资源节点到其它所有资源节点之间的曼哈顿距离之和 d_k：

$$d_k=\sum_{i=0}^{n} d_{ki} \tag{4}$$

调节该 IP 模块的映射位置，使得 d_k 尽量小，即通信量大的模块尽量映射到 NoC 的中央资源节点上，然后转 Step3。重复上述过程直到所有 IP(虚)模块的映射位置都不能进一步优化为止。

4 实验结果

我们对 6 个视频处理器件进行了模拟实验：MPEG4 解码器(映射 14 个 IP 模块)，视频对象平面解码器 OPD(Video Object Plane Decoder，映射 16 个 IP 模块)，图像到图像的应用 PIP(Picture-In-Picture application，映射 8 个 IP 模块)，多窗口应用 MWA(Multi-Window Application，映射 14 个 IP 模块)，图像多窗口应用 MWAG(MWA with Graphics，映射 16 个 IP 模块)和双屏显 DSD(Dual Screen Display，映射 16 个 IP 模块)，其中后面的四个基准电路是高端视频应用。实验过程中系数 α 的值取 1.2。为了进行比较，我们还实现了文[8]中的分支定界算法 PBB(partial branch- and-bound algorithm)。程序源代码用 C++语言编写，在 Windows xp，P4 2.6GHz 环境下测试的最小通信功耗的比较结果如图 3 所示。从图 3 可以看出，本文提出的方法与 PBB 方法相比通信功耗能节省大约 7%。

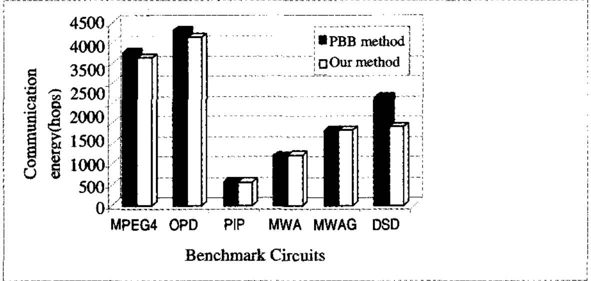

图 3 本文方法和 PBB 算法的通信功耗比较

算法的运行速度也是我们非常关心的问题。随着 NoC 尺寸的增大，映射算法的搜索速度在下降。利用本文提出的算法，PIP(映射到 3×3 的 NoC)的运行时间是 2 秒左右，而其余 5 个(映射到 4×4 的 NoC)的算法运行时间均在 10～12 秒之间。

结论 本文提出了一种新的不规则 IP 模块到二维 NoC 结构的映射方法。算法的基本思想是把大的 IP 模块看作尽可能少的虚模型，使得每一虚模型都能映射到 NoC 的一个资源节点上。如果 IP 模块的大小远小于资源节点的大小，则把尽可能多的类似 IP 模块看作一个虚模型，使得该虚模型能映射到一个资源节点上。通过计算输入/输出度，可以确定每个通信节点缓冲器的大小。根据最后计算的通信代价对初始的映射结果进行调整，从而可以避免通信拥塞，降低系统的功耗。本文在计算通信路径时采用的是确定性的 XY 路由算法，今后可进一步研究采用自适应路由算法计算 NoC 通信路径的方法。

参 考 文 献

1 周干民，高明伦. NoC 基础研究：[学位论文]. 合肥：合肥工业大学，2005

2 Guerrier P, Greiner A. A generic architecture for on-chip packet-switched interconnections. In: Proceedings of Design, Automation and test in Europe, Paris, France, 2000

3 Dally W J, Towles B. Route Packets, Not Wires. On-chip Interconnection Networks. In: Proceeding of Design Automation Conference, Las Vegas, 2001

4 Benini L, DeMicheli G. Powering networks on chip. In: Proceedings of the 14th International Symposium on System Synthesis, Montreal, Quebec, Canada 2001

5 Eisley N, Peh L S. High-level power analysis for on-chip networks. International conference on Compilers, architecture and synthesis for embedded systems, Washington DC, USA, 2004

6 Simunic T, Boyd S. Managing power consumption in networks on chips. In: Proceedings of Design, Automation and test in Europe, Paris France, 2002

7 Benini L, De G. Micheli. Networks on Chips: A New SoC Paradigm. IEEE Computer, 2002, 35(1): 70～78

8 Hu J C, Radu M. Energy- and performance-aware mapping for regular NoC architectures. IEEE Transaction on CAD of IC and Systems, 2005, 24(4): 551～562

(上接第 18 页)

15 Grohe M. Descriptive and parameterized complexity. In: Proceedings of the 13th International Workshop and 8th Annual Conference of the EACSL on Computer Science Logic, 1999. 14～31

16 Hüffner F, Wernicke S, Zichner T. Algorithm engineering for color-coding to facilitate signaling pathway detection. In: Proceedings of the 5th Asia-Pacific Bioinformatics Conference (APBC '07), Advances in Bioinformatics and Computational Biology. World Scientific, 2007(To appear)

17 Isshiki T, Tanaka K. An (n- t)-out-of-n Threshold Ring Signature Scheme. In: Information Security and Privacy, 10th Australasian Conference, ACISP, 2005, 2005. 4～6

18 Liu Y, Wang J, Chen J. Color-coding: n ≤ 2k. Manuscript, 2007

19 Sze Sing-Hoi, Chen Jianer, Lu Songjian, et al. Improved algorithms for path, matching, and packing problems. Manuscript, 2006

20 Kelley B P, Sharan R, Karp R M, et al. Conserved pathways within bacteria and yeast as revealed by global protein network alignment. Proceedings of the National Academy of Sciences, 2003, 100(20): 11394～11399

21 Koutis I. A faster parameterized algorithm for set packing. Information Processing Letters, 2005, 94(1): 7～9

22 Liu Yang, Lu Songjian, Chen Jianer, et al. Greedy localization and color-coding: Improved matching and packing algorithms. In: IWPEC, 84～95

23 Mathieson L, Prieto E, Shaw P. Packing Edge Disjoint Triangles: A Parameterized View. In: Proceedings of the International Workshop on Parameterized and Exact Computation, 2004

24 Monien B. How to find long paths efficiently. Annals of Discrete Mathematics1985, 25: 239～254

25 Naor J, Naor M. Small-bias probability spaces: efficient constructions and applications. New York, NY, USA: ACM Press, 1990

26 Schmidt J P, Siegel A. The spatial complexity of oblivious k-probe hash functions. SIAM J Comput, 1990, 19(5): 775～786

27 Scott J, Ideker T, Karp R M, et al. Efficient algorithms for detecting signaling pathways in protein interaction networks. In: RECOMB, 2005. 1～13