一种在元素与颜色规模相近时的有效着色算法及其应用

王建新 刘云龙 陈建二

(中南大学信息科学与工程学院 长沙 410083)

摘 要 着色算法(color-coding)是求解 NP 难问题的重要手段之一.而在应用着色算法时,着色算法所产生的着色方案的规模极大地影响着问题的求解性能,故构造一个尽可能小的着色方案是着色算法所寻求的目标.目前存在的着色算法均基于完全散列函数,并要求元素数目 n 远大于颜色数目 k,且 k 比较小,这个限制条件使得这些着色算法在一些实际情况下无法应用.该文主要研究在元素与颜色规模相近时($n\leq 2k$)的有效着色算法,并着重分析在 $n\leq 2k$ 情况下着色算法的性能.该文提出了一种基于划分思想的着色方案构造算法 PBCC,证明了由 PBCC 产生的着色方案确实可以覆盖到所有的子集,并具体给出了可应用于(l,d)-(20,16) Motif 查找问题的 403 种着色的构造方法.文章进一步分析了 PBCC 产生的着色方案规模,并证明了在 $n\leq 2k$ 且 $n-k\geq 2$ 的情况下,任何着色算法所产生的着色方案的规模$|S(n,k)|$都不小于$\binom{\lceil n/2\rceil}{n-k}+\left[\binom{n}{n-k}-\binom{\lceil n/2\rceil}{n-k}2^{n-k}\right]/(2^{n-k}-2)$.此外,文中也采用了渐进分析技术,证明了 PBCC 算法生成着色方案规模为 $O(e^{2Rootof(e^{x}-e^{\mu x}+1)(n-k)})$,在 $n=2k$ 的情况下结果是$O(2.62^{n-k})$;同时,文中也证明了 $n\leq 2k$ 情况下着色方案规模的下界为 2^{n-k}.

关键词 着色;着色算法;NP 难问题;渐进分析;界限分析

中图法分类号 TP301

An Effective Coloring Algorithm for Close Relationship Between the Scales of Element Set and Color Set and Its Application

WANG Jian-Xin LIU Yun-Long CHEN Jian-Er

(*School of Information Science and Engineering, Central South University, Changsha* 410083)

Abstract Color-coding is one of the most important solutions to those problems proved to be NP-Hard. The performance of those color-coding based algorithms principally depends on the scale of the coloring schemes generated by coloring algorithms they deploy. Therefore, the ultimate objective of coloring algorithm is to provide a coloring scheme as small as possible. All existing coloring algorithms are mainly based on perfect hash functions, requiring of the number of elements n far greater than the number of colors k, and k relatively small. This paper mainly focuses on the study of effective algorithms under the circumstance where the size of element set and color set are close ($n\leq 2k$) and the analysis of performance of coloring algorithm while $n\leq 2k$. This paper proposes a novel partition based algorithm PBCC, and proves that the coloring scheme generated by PBCC can cover all the subsets. A detailed method for constructing a (20,

收稿日期:2006-08-24;最终修改稿收到日期:2007-09-06.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)、国家自然科学基金(60773111)、湖南省杰出青年基金(06JJ10009)、新世纪优秀人才支持计划(NCET-05-0683)和国家教育部创新团队资助计划(IRT0661)资助.**王建新**,男,1969 年生,博士,教授,博士生导师,研究领域为计算机算法、网络优化理论和生物信息学.E-mail:jxwang@mail.csu.edu.cn.**刘云龙**,男,1983 年生,硕士,研究方向为计算机算法.**陈建二**,男,1954 年生,博士,教授,博士生导师,主要研究领域为计算机理论、计算复杂性及优化、生物信息学和计算机网络优化算法.

16)-coloring scheme with size 403 which can be applied for (l,d)-(20,16) motif finding problem is also provided. Furthermore, this paper analyzes the scale of coloring schemes generated by PBCC, and proves that while $n\leq 2k$ and $n-k\geq 2$, no algorithm can create a coloring scheme with its size $|S(n,k)|$ less than $\binom{\lceil n/2\rceil}{n-k}+\left[\binom{n}{n-k}-\binom{\lceil n/2\rceil}{n-k}2^{n-k}\right]/(2^{n-k}-2)$. At last, using asymptotic analysis, it is proved that coloring scheme generated by PBCC is of size $O(e^{2Rootof(e^x-e^{(\mu x)}+1)(n-k)})$, and it is $O(2.62^{n-k})$ when $n=2k$. Also, it is showen that any coloring scheme with $n\leq 2k$ will be larger than 2^{n-k}.

Keywords　coloring; coloring algorithm; NP-hard; asymptotic analysis; bound analysis

1 引　言

在计算科学的研究领域中，有相当一部分 NP 难的问题可以被表述成为在一个元素全集中进行子集元素选择的问题，例如 k-PATH. 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 $O(2^k k!n^{O(1)})$[1-2].

文献[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 的所有节点中选择一个包含 k 个节点的子集. 基于这个观察，文献[3]首次提出利用着色算法(color-coding)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 个节点正好与 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.

需要注意的是，着色算法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，其算法复杂度将达到 $O\binom{n}{k}$，而利用着色算法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. 着色的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.

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 NP 难问题上有着广泛的应用. 例如，目前着色的思想已经被应用到蛋白质调控网络中的路径查找[4]、子图匹配[5]及 Motif 查找[6]中.

基于着色的算法复杂度极大地依赖于需要枚举的着色方法数目，根据枚举着色的方法，着色可以分为随机着色和确定式着色. Alon 等人在文献[2]中提出以着色思想解决 k-PATH 问题时，在着色过程中使用了简单的随机着色方法，并通过足够的重复次数来保证获得的解具有一定的准确率. 由于一次着色满足要求的概率为 e^{-k}，所以在 $O^*(5.44^k n^{O(1)})$ 的时间复杂度下随机算法可以得到比较高的准确率. 但是值得注意的问题是，虽然随机算法可以提供一定精度的结果，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，这样的概率解是不能令人满意的，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 等人在文献[2]中指出：使用完全散列函数(perfect-hash function，无冲突的散列函数)，可以在 $O^*(2^{O(k)})$ 构造出一个确定式着色方案. 但需要注意的是，该结果距离实用仍有比较大的距离. 根据文献[7-8]中的理论，表示任一散列函数需要多于 12 Kbits，枚举所有的散列函数需要至少 $2^{12k}>4000^k$，这说明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，Chen 等将这一结果改进到 $O^*(7.63^k n)$[9]，最近他们又得到了 $O^*(6.1^k n)$ 的结果[10].

现有的这些 color-coding 的确定化方法[8-10]主要依赖于完全散列函数理论和小参数理论，其应用背景和假设前提都要求 n 远大于 k，且 k 比较小. 而

在相当一部分实际情况下,这种限制条件是不合适的.例如,Motif 查找问题就是生物计算中一个重要的问题,而其中序列条数 $K=20$ 的 (l,d)-(20,16) 问题是目前生物学家十分关注的 Motif 查找问题,其 $n=20$,而 $k=16$.在这个问题和其类似问题上,其参数(对应着色问题中的元素和颜色规模)之间差距不会太大,而这些问题上同样有应用 color-coding 的空间.在这些情况中,文献[8-10]中的着色构造算法都会导致方案规模过大而不可用.针对这一问题,本文主要研究元素和颜色规模相近($n\leq 2k$)情况下的着色算法,并对着色规模进行了理论分析和证明,提出了一种应用于 $n\leq 2k$ 情况下的基于划分的算法 PBCC(Partition-Based Color-Coding).

2 PBCC 着色算法

在 $n\leq 2k$ 情况下,尤其是 k 比较大时,目前关于确定式着色的最好结果 $O^*(6.1^k n)$ 也无法实际应用.例如:针对 (l,d)-(20,16) 的 Motif 查找问题,应用文献[10]的着色算法产生的着色方案大小为 354,213,844,620,169,979,207,614,464≈3.54214E+26.很显然,由于 $n\gg k$ 限制条件被打破,基于完全散列函数的着色算法的时间复杂度实际上等价于 $O^*(c^{O(n)}n)$,因此基于完全散列函数的算法在这种情况下是完全不适用的.

为了便于描述,给出本文所需定义如下:

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.如果 C 中某种颜色 c_i 在 H 中仅出现一次,则称该颜色是专有色;否则称该颜色为复用色.

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,即 $H\xrightarrow{c}W$.

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,记作 $S(n,k)$.方案包含着色的数目称为着色方案的规模,记作 $|S(n,k)|$.

首先分析一下着色和其能覆盖的子集之间的联系.对于任意的 (n,k)着色,若某元素 e_i 的所着颜色 c_i 是专有色,则该 (n,k)着色能覆盖的所有子集 W 必定含有该元素 c_i;若某颜色 c_j 是复用色,则该 (n,k)着色能覆盖的任意子集 W 中有且仅有一个着色为 c_j 的元素.对于一个 (n,k)着色问题,其着色中至多有 $n-k$ 种颜色为复用色.为了减小 $|S(n,k)|$,使每个着色能尽量多覆盖一些子集,即着色应尽量平均.因此,在 $n\leq 2k$ 情况下,PBCC 将考虑正好有 $n-k$ 种颜色被复用的情况,而这 $n-k$ 种颜色正好各自被复用一次.这个前提下,PBCC 将通过分析枚举这 $n-k$ 种复用色在着色中的分布情况来确定 (n,k)着色方案.

对一个大小为 2 的子集进行着色时,子集中的两个元素要么着相同颜色,要么着不同颜色.这种简洁的特性是大小超过 2 的子集所不能提供的,故 PBCC 着色算法尽量将 U 划分为大小不超过 2 的一系列子集,这些子集被称为块,其中大小为 1 的块称为单元素块,大小为 2 的块称为双元素块.设集合 U 的元素个数为 n,算法 PBCC 将其划分为 $\lceil n/2\rceil$ 个块,其中每个块的大小均不超过 2.划分后的所有块中最多存在一个元素个数为 1 的块,即 $\forall i,|\{B_i||B_i|=1\}|\leq 1$;对于任意块 B_i,其元素个数一定不大于 2,即 $\forall i,0<|B_i|\leq 2$.

下面我们首先讨论一些简单情况下的着色方案,并进一步给出在 $n\leq 2k$ 情况下 (n,k)着色方案构造方法.

2.1 简单情况下的着色方案构造方法

本节我们主要讨论 $k=n$,$k=n-1$,$k=n-2$ 和 $k=n-3$ 几种简单情况下 (n,k)着色方案的构造方法.

(1) $k=n$

在这种情况下,只需将集合 C 中的颜色和集合 U 中元素一一对应就可以得到一个规模为 1 的着色方案,即 $|S(n,n)|=1$.

(2) $k=n-1,n\geq 2$

由于 $n-k=1$,则只存在一种复用色.首先将集合 U 划分为 $\lceil n/2\rceil$ 个块,着色方案可以通过每次选择一个块 B_i,将 B_i 中的两个元素着相同的颜色,在下文中将使用同色块代表这种块中元素着色相同的块;相应的,如果块中元素着色不同,则称为异色块.

然后将剩余的元素和颜色进行任意的一一对应即可以最小着色方案规模完成覆盖，而当选择的块中只包含一个元素时，算法将通过调整块划分的方法来使得该块包含两个元素，得到着色方案规模 $|S(n,n-1)|=\lceil n/2\rceil$.

(3) $k=n-2, n\geq 4$

由于 $n-k=2$，存在 2 种复用色. 首先将集合 U 划分为 $\lceil n/2\rceil$ 个块 B_i，并根据 W 和 B_i 重合情况将 $\binom{n}{k}$ 个子集 W 分为两类：

(a) $|\{B_i||B_i-W|=1\}|=2$，即有且仅有一个元素不属于 W 的块有 2 个. 覆盖此类子集时，着色需使用 2 个同色块. 首先从所有块中选择 2 个块，当选择的块中有单元素块时，则从其它块中移出元素使得选择的 2 个块都为双元素块. 用两种不同的颜色分别将这 2 个块中的元素着色，每个块中颜色相同，然后将剩余元素和剩余颜色进行任意的一一对应.

(b) $|\{B_i||B_i-W|=1\}|=0$，即不存在有且仅有一个元素不属于 W 的块. 覆盖此类子集时，着色无需使用同色块. 因此，着色时可以将块作为单一元素考虑，并相应地减少所需颜色数目，该情况就会被等价地转化为一个 $k'=n'-1$ 的情况，其中 $n'=\lceil n/2\rceil$，而 $k=n-1$ 时的解法是已知的. 所以在得到 $k'=n'-1$ 的着色后，若 $(n',n'-1)$ 着色中颜色 c_i' 所赋予的元素中有对应 $(n,n-2)$ 着色问题中双元素块的，则所有赋予 c_i' 的元素所对应的双元素块着色为 $\langle c_{i1},c_{i2}\rangle$，而赋予 c_i' 的元素所对应的单元素块则通过从赋予专有色的元素所对应块中移出元素补入成为双元素块后着色为 $\langle c_{i1},c_{i2}\rangle$.

计算得到 $|S(n,n-2)|=\binom{\lceil n/2\rceil}{2}+\left\lceil\frac{n}{4}\right\rceil$.

(4) $k=n-3, n\geq 6$

由于 $n-k=3$，存在 3 种复用色，首先将集合 U 划分为 $\lceil n/2\rceil$ 个块 B_i，并根据 W 和 B_i 重合情况将 $\binom{n}{k}$ 子集 W 分为两类：

(a) $|\{B_i||B_i-W|=1\}|=3$，即有且仅有一个元素不属于 W 的块有 3 个，覆盖此类子集时，着色需使用 3 个同色块. 构造方法和 $k=n-2$ 时情况(a)完全一致.

(b) $|\{B_i||B_i-W|=1\}|=1$，即有且仅有一个元素不属于 W 的块有 1 个，覆盖此类子集时，着色还需使用 1 个同色块. 首先从所有块中选择 1 个块，当选择的块为单元素块时，则从其它块中移出元素使得这个块都为双元素块，用 1 种颜色将这个块中的元素着相同的颜色，然后排除这个块及其使用的颜色，就只需考虑不使用同色块的情况了，参照 $k=n-2$ 情况(b)即可解决这种情况.

同样，计算得知 $|S(n,n-3)|=\binom{\lceil n/2\rceil}{3}+\left\lceil\frac{n}{2}\right\rceil\left\lceil\frac{\lceil n/2\rceil-1}{2}\right\rceil$.

从这 4 种简单情况的着色方案构造方法可以看出：对于任何一个满足 $n\leq 2k$ 的着色问题，都可以将其元素块 B_i 的着色分为同色和异色两种类型. 对于前者而言，可以通过枚举其分布情况解决；而后者可以通过递归到更小规模的问题上来解决. 这样，着色算法就可以被推广到 $n\leq 2k$ 的一般情况.

2.2　在 $n\leq 2k$ 情况下的着色方案构造方法

在具体描述 PBCC 算法之前，首先介绍关于调整单元素块的具体方法. 当 $|U|\%2=1$，并有算法要求处理双元素块却得到 B_i 有 $|B_i|=1$ 时，需要对该 B_i 进行调整，使得调整后所有需要处理的块 B_i 都包含有两个元素.

如图 1 所示，在需要调整的块集合 A 中存在确实需要调整的块时，BlockAdjust 通过从 A 之外的块集合中选择任一块，并从该块中移出一个元素补入需要调整的块中以保证集合 A 中所有的块都为双元素块.

算法 BlockAdjust：调整元素块 *BlockAdjust*(F,A).

输入：元素块全集 F，需要调整的块集合 A
输出：调整好的块集合 A，其中每个块的大小均为 2
if 存在 $g\in A$ 且 $|g|=1$ then
　{ 任选 $g'\in F-A$，再任选元素 $e\in g'$
　　使 $g\leftarrow g+\{e\}$，$g'\leftarrow g'-\{e\}$}

图 1　调整单元素块算法 BlockAdjust

如图 2 所示，除 $k=n$ 的简单情况外，PBCC 算法在 $n\leq 2k$ 的限制条件下使用 $n-k$ 种复用色覆盖所有子集. PBCC 算法首先将元素集合 U 划分为 $\lceil n/2\rceil$ 个子集. 继而，算法枚举块中着色需要的同色块数目，并枚举相应数量的同色块在所有块中的分布，同时求解一个子着色问题. 该子着色问题中包含等于原着色问题中非同色块数目的元素，并相应地包含原有颜色去除同色块使用颜色后一半(向上取整)数目的颜色. 然后根据枚举的同色块分布和子着色方案中着色的组合，将枚举中同色块位置选择的块中元素着为同色，如果选择中包含了单元素块，则通过单元素块调整算法将块调整为双元素块再着色，

并将这些使用过的颜色从可用颜色集中去除.

对于子着色方案中的每种着色 p,分别处理子着色方案中使用的颜色 c_i'.若在 p 中,c_i'所赋予的元素对应的块中有双元素块,则将这些块都用一个包含原着色问题中可用的两种颜色的集合(称为颜色组)着为异色块,如果这些块中也包含单元素块,则通过单元素块调整算法将所有同色块分布中的块调整为双元素块再着色,并将这些使用过的颜色从可用颜色集中去除.若 c_i'所赋予的元素对应的块中只有单元素块,则将该块用仅包含一种可用颜色的颜色组着色,并将这种颜色从可用颜色集中去除.

算法 PBCC(Partition-Based Color-Coding):$PBCC(n,k)$

输入:元素全集 U,可用的颜色集合 C 且 $n\leq 2k$
输出:一个着色方案 $S(n,k)$
1. $S(n,k)\leftarrow\varnothing$;
2. if $n=k$ then
生成着色 H,其中元素和颜色一一对应. return $S(n,k)\leftarrow\{H\}$;
3. 将 U 尽量平均地划分为$\lceil n/2\rceil$个子集,记这些子集为 B_i,所有 B_i的集合为 F;
4. for $i\leftarrow 0$ to $\lfloor(n-k)/2\rfloor$ do
{产生所有在$\lceil n/2\rceil$个位置中选择的 $n-k-2i$ 个位置的列表 L;
$S(k-n/2+2i,\ k-n/2+i)\leftarrow PBCC(k-n/2+2i,\ k-n/2+i)$;
for each 在 L 中的条目 e do
for each 在 $S(k-n/2+2i,\ k-n/2+i)$中的着色 q do
{初始化 C 为所有可用颜色的集合;
记 e 包含位置对应块的集合为 A,$BlockAdjust(F,A)$;
对 A 中每个块 B_i,$\forall c_i\in C$将块的元素着为 c_i,$C\leftarrow C-\{c_i\}$;
将$(k-n/2+2i,\ k-n/2+i)$子着色问题中元素一一映射到 $F-A$ 中的块,记这个映射关系为 $g(e_i')\rightarrow B_j$;
记着色 q 中所有着色为c_i'的元素e_i'对应的块组成的集合为 $F(c_i')$,若有任意 $g(e_i')\in F(c_i')$为双元素块,
$BlockAdjust(F-A,F(c_i'))$.
$\forall c_i,c_j\in C$,$C\leftarrow C-\{c_i,c_j\}$,并将 $g(e'i)$着色为$\{c_i,c_j\}$.若 $|F(c_i')|=1$且 $g(e_i')\in F(c_i')$为单元素块,$\forall c_i\in C$,$C\leftarrow C-\{c_i\}$,并将 $g(e_i')$着色为$\{c_i\}$;
记上面得到的着色为 H,$S(n,k)\leftarrow S(n,k)+\{H\}$;}
}
return $S(n,k)$;

图 2 算法 PBCC

另外,注意到由于 $n\leq 2k$,则 $k-n/2+2i\leq 2(k-n/2+i)$,所以在递归过程中并不会产生算法 PBCC 无法处理的情况.

2.3 PBCC 在(20,16)-Motif 查找中的应用

如上文提到的,在 Motif 查找问题中,(20,16)的着色问题是一个实际重要的问题,对这个问题的优化同时就是对 Motif 查找问题的改进.下面本文将探讨通过 PBCC 获得一个优化的(20,16)着色方案的方法.通过分析,可知 PBCC 将产生如下着色方案:

(1)在 $i=0$ 时,算法从划分的 10 个块中枚举 4 个块的位置,并计算子着色方案 $S(6,6)$.对于每种枚举出的块位置组合和 $S(6,6)$中的每一种着色($S(6,6)$只含有一种着色),算法使用 4 种颜色将这 4 个位置上的块着为同色块.对于 $S(6,6)$中的着色,它的 6 种颜色正好和它的 6 种元素一一对应,而 6 个元素相应的块也都为双元素块,故对于 $S(6,6)$中的每个颜色,算法分配 2 种颜色对该颜色对应元素相应块进行着色.可以看到其实际效果将使尚未使用的 12 种颜色和尚未着色的 12 个元素一一对应.在枚举完所有 4 个位置的组合情况和 $S(6,6)$中的着色之间的组合后,算法得到$\binom{10}{4}\times|S(6,6)|=210$ 条着色;

(2)在 $i=1$ 时,算法从划分的 10 个块中枚举 2 个块的位置,并计算子着色方案 $S(8,7)$.对于每种枚举出的块位置组合和 $S(8,7)$中的每一种着色,算法使用 2 种颜色将这 2 个位置上的块着为同色块.对于 $S(8,7)$中的着色,其元素相应的块均为双元素块,故对于 $S(8,7)$中的每个颜色,算法分配 2 种颜色对该颜色对应元素相应块进行着色.同样可以看到,由于 $S(8,7)$中有一种颜色为复用色,该颜色所赋予的元素相应的 2 个块将使用同样的颜色组合进行着色,而剩余尚未使用的颜色和尚未着色的元素还是一一对应.在枚举完所有 2 个位置的组合情况和 $S(8,7)$中的着色之间的组合后,算法得到$\binom{10}{2}\times|S(8,7)|=180$ 条着色;

(3)在 $i=2$ 时,算法不枚举任何块的位置,但继续计算子着色方案 $S(10,8)$.对于 $S(10,8)$中的每一种着色,其元素相应的块均为双元素块,故对于 $S(10,8)$中的每个颜色,算法分配 2 种颜色对该颜色对应元素相应块进行着色.在这种情况下,由于 $S(10,8)$中有 2 种颜色为复用色,这 2 种颜色所赋予的元素分别对应的 2 个块将分别使用同样的颜色组合进行着色,而剩余尚未使用的颜色和尚未着色的元素还是一一对应.在枚举完 $S(10,8)$中的着色后,算法得到$\binom{10}{0}\times|S(10,8)|=13$ 条着色;

综上可知,PBCC 为(20,16)着色问题构造的着色方案规模为 $210+180+13=403$.

3 PBCC 算法的分析

3.1 PBCC 算法的正确性分析

定理 1. 算法 PBCC 所求得的着色方案 $S(n,k)$能够覆盖 U 的所有 k 元素子集,即在 U 中任取大小为 k 的子集,至少存在一个着色 $p\in S(n,k)$能够

覆盖该子集.

证明.　设有全集 U，$|U|=n$，任取 $W\subseteq U$，$|W|=k$，并将 U 划分成 $\lceil n/2\rceil$ 个块，则 U 中有 $n-k$ 个元素不属于 W. 任取块 B_i，B_i 中不属于 W 的元素数目只有 0，1，2 这 3 种可能，所以正好 1 个元素不属于 W 的块数目与不属于 W 的元素数目 $n-k$ 之差必为偶数. 假设在 $\lceil n/2\rceil$ 个块中有 $n-k-2i$ 个块有正好 1 个元素不属于 W，由于 $0\leq n-k-2i\leq\lceil (n-k)/2\rceil$，可知 i 的取值范围为 0 到 $\lfloor (n-k)/2\rfloor$. 这 $n-k-2i$ 个块的分布必会被组合 $\binom{\lceil n/2\rceil}{n-k-2i}$ 所枚举. 经 PBCC 着色算法处理后，这些块中内部的元素必定颜色相同，而处于不同块中的元素必定颜色相异，即 W 在这 $n-k-2i$ 个块中的元素均有互不相同的颜色. 所以这 $n-k-2i$ 个块属于 W 的元素的着色必各不相同. 若这些块中存在单元素块，则表示这个单元素块中的元素不属于 W，其着色对问题并无影响.

在剩余的 $\lceil n/2\rceil-(n-k-2i)=k-n/2+2i$ 个块中，不属于 W 的元素数目只有 0，2 这 2 种情况，即块的元素要么全部属于 W，要么全部不属于 W. 而且元素全部属于 W 的块数目为 $\left\lceil\frac{k-(n-k-2i)}{2}\right\rceil=k-n/2+i$. 假设所有的子方案都完成覆盖，且 W 中剩余元素均处于某 $k-n/2+2i$ 个块中. 子方案 $S(k-n/2+2i,k-n/2+i)$ 中必存在一种着色覆盖这 $k-n/2+i$ 个块对应元素的子集. 经 PBCC 着色算法处理后，这些块中所有的元素在该着色生成的着色中颜色也将各不相同. 且注意到，PBCC 算法在处理同色块分布和子方案着色两步中使用的颜色并不重复. 综合以上讨论，可知 W 中元素着色互不相同. 那么对于 U 的任意子集 W，PBCC 均能产生一种着色使其中元素着色互不相同，即 PBCC 所得着色方案 $S(n,k)$ 能够覆盖 U 的所有 k 元素子集. 证毕.

3.2　PBCC 算法产生着色方案的规模 $|S(n,k)|$ 分析

为了得到着色方案 $S(n,k)$，PBCC 算法枚举了 $n-k-2i$ 个同色块的分布，并将异色块的分布递归到子问题 $S(k-n/2+2i,k-n/2+i)$ 中，通过上面简单的分析，可以看到，在 $n\leq 2k$ 时，算法构造的着色方案规模为

$$S(n,k)=\sum_{i=0}^{i\leq\lfloor (n-k)/2\rfloor}\binom{\lceil n/2\rceil}{n-k-2i}S(k-\lfloor n/2\rfloor+2i,k-\lfloor n/2\rfloor+i)\qquad(1)$$

式(1)给出了算法 PBCC 产生着色方案规模的递归计算公式，其具体计算结果如图 3 所示. 图 3 给出了在不同 n 和 $n\leq 2k$ 的条件下 PBCC 算法产生着色方案的规模.

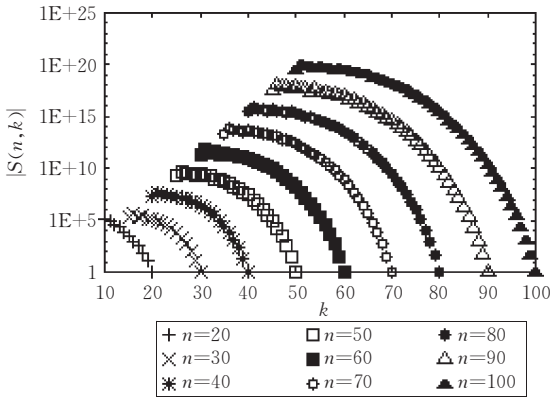

图 3　在 $n\leq 2k$ 下由 PBCC 产生的 $|S(n,k)|$

从图 3 可以看出，在 $n\leq 2k$ 的限制条件下 $|S(n,k)|$ 基本上是随着 $n-k$ 的增大而增长的，但是由于在某些特殊情况下，着色方案的分布更加容易均匀一些，在算法的体现上使得 $|S(n,k)|$ 的大小并不是 $n-k$ 的严格递增数. 特别是，在 $n-k=k$ 的时候，$|S(n,k)|$ 并没有达到最大，函数的峰值实际上是出现在 $n=2k$ 的附近.

图 4 给出了文献[7]中基于完全散列函数的算法 PH 和本文基于划分的算法 PBCC 在 $n\leq 2k$ 下的实际规模的比较. 从图 4 中可见，由于算法基本思想的差异，所产生的着色方案规模变化规律出现了很大的差异. PH 算法的着色方案规模基本呈递增趋势，而算法 PBCC 所构造的着色方案规模是递减的. 从图 4 中可以看出由完全散列函数构造的着色方案都明显比由 PBCC 所构造的规模大得多，在 $n=20$，$k=10$ 时，PH 算法的着色方案规模为 5.17114E+19，而算法 PBCC 所构造的着色方案规模仅为

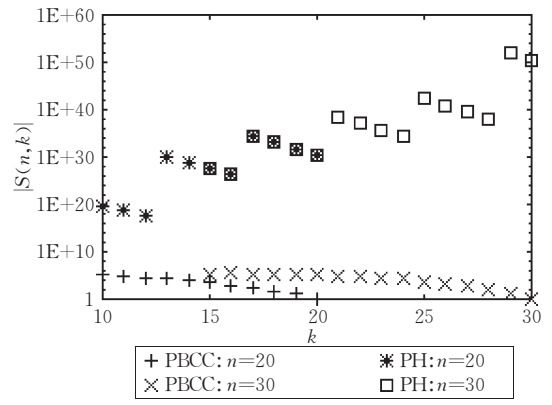

图 4　基于完全散列函数的算法 PH 和算法 PBCC 在 $n\leq 2k$ 下的实际规模

157096,其重要原因就是基于完全散列函数着色构造法并不能适应 $n\leq 2k$ 的限制条件.

3.3 PBCC 算法着色方案规模 $|S(n,k)|$ 的渐进界分析

注意到图 3 和图 4 中 PBCC 算法产生的实际着色方案的规模都有随着 k 增大而递减的趋势,显然继续沿用原有以 k 为小参数的思想,继续假设 $|S(n,k)|=O^*(c^k)$ 是不合理的.

那么,作为 $n-k$ 的近似递增函数(可以从图 3 中看到 $|S(n,k)|$ 并非 $n-k$ 的严格递增函数),假设 $|S(n,k)|$ 满足 $O^*(c^{n-k})$ 的形式更为合理.但是,同时再注意到式(1)和图 4 中 PBCC 算法的曲线在对数坐标系中并非近似的直线,且其曲率也受到参数 n 的影响,因此,直接假设规模上界是一个常数底的指数函数也是不尽合理的.

所以,综合上述所有的观察结果,下面假设 PBCC 的着色方案规模 $|S(n,k)|$ 满足 $O^*(f(k/n)^{n-k})$ 的形式,即 $|S(n,k)|$ 同时是 k/n 和 $n-k$ 的函数,且 k/n 影响复杂度的底数而 $n-k$ 则为指数.

定理 2. PBCC 算法生成的着色方案规模 $|S(n,k)|=O(\mathrm{e}^{2Rootof(\mathrm{e}^x-\mathrm{e}^{\mu x}+1)(n-k)})$,其中 $\mu=3-2k/n$,且 $\mu\in[1,2]$.

证明. 参见式(1),并假设 $|S(n,k)|\leq c\,\mathrm{e}^{m(n-k)}$,其中 c 为正常数,且 m 是待定参数.

为了分析方便,证明过程中公式中的取整操作均被忽略.

$$
\begin{aligned}
|S(n,k)| &= \sum_{i=0}^{i\leq(n-k)/2}\binom{n/2}{n-k-2i}\cdot \\
&\quad |S(k-n/2+2i,k-n/2+i)| \\
&\leq \sum_{i=0}^{i\leq(n-k)/2}\binom{n/2}{n-k-2i}c\cdot \mathrm{e}^{mi} \\
&= \sum_{i=0}^{i\leq(n-k)/2}\binom{n/2}{k-n/2+2i}c\cdot \mathrm{e}^{mi} \\
&= c\binom{n/2}{k-n/2}\mathrm{e}^0+c\binom{n/2}{k-n/2+2}\mathrm{e}^m+\cdots+ \\
&\quad c\binom{n/2}{k-n/2+2i}\mathrm{e}^{mi}+\cdots+c\binom{n/2}{n/2}\mathrm{e}^{(n-k)m/2} \\
&= c\,\mathrm{e}^{-\frac{m}{2}(k-n/2)}\left(\binom{n/2}{k-n/2}\mathrm{e}^{\frac{m}{2}(k-n/2)}+\right. \\
&\quad \binom{n/2}{k-n/2+2}\mathrm{e}^{\frac{m}{2}(k-n/2+2)}+\cdots+ \\
&\quad \left.\binom{n/2}{n/2}\mathrm{e}^{\frac{m}{2}\cdot\frac{n}{2}}\right) \qquad (2)
\end{aligned}
$$

注意到式(2)与二项式展开式的类似性,考虑通过二项式对其进行归约.现在,参考一下二项式 $(1+\mathrm{e}^{m/2})^{n/2}$ 的展开形式.

$$
\begin{aligned}
(1+\mathrm{e}^{m/2})^{n/2} &= \binom{n/2}{0}\mathrm{e}^{0\cdot\frac{m}{2}}+\binom{n/2}{1}\mathrm{e}^{1\cdot\frac{m}{2}}+\cdots+ \\
&\quad \binom{n/2}{n/2}\mathrm{e}^{\frac{n}{2}\cdot\frac{m}{2}} \qquad (3)
\end{aligned}
$$

对比式(2)和式(3),很容易看到,

$$S(n,k)\leq c\mathrm{e}^{-m(k-n/2)/2}(1+\mathrm{e}^{m/2})^{n/2} \qquad (4)$$

那么,如果有 $c\mathrm{e}^{-m(k-n/2)/2}(1+\mathrm{e}^{m/2})^{n/2}\leq c\mathrm{e}^{m(n-k)}$ 成立,前面的假设就是正确的.现在考虑这个条件如何成立.

$$
\begin{aligned}
c\mathrm{e}^{-m(k-n/2)/2}(1+\mathrm{e}^{m/2})^{n/2} &\leq c\mathrm{e}^{m(n-k)}(1+\mathrm{e}^{m/2})^{n/2} \\
&\leq \mathrm{e}^{m(3n/4-k/2)} \qquad (5)
\end{aligned}
$$

令 $y=1+\mathrm{e}^{m/2}$,则 $m=2\ln(y-1)$.那么应用式(5),有

$$
\begin{aligned}
n/2\ln y &\leq 2(3n/4-k/2)\ln(y-1)\ln y \\
&\leq 2(3-2k/n)\ln(y-1) \qquad (6)
\end{aligned}
$$

令 $\mu=3-2k/n$,其中 $\mu\in[1,2]$,那么

$$y\geq \mathrm{e}^{Rootof(\mathrm{e}^x-\mathrm{e}^{\mu x}+1)}+1 \qquad (7)$$

$$m\geq 2Rootof(\mathrm{e}^x-\mathrm{e}^{\mu x}+1) \qquad (8)$$

图 5 中展示了 μ 和 x 之间的函数关系,即

$$|S(n,k)|=O(\mathrm{e}^{2Rootof(\mathrm{e}^x-\mathrm{e}^{\mu x}+1)(n-k)}) \qquad (9)$$

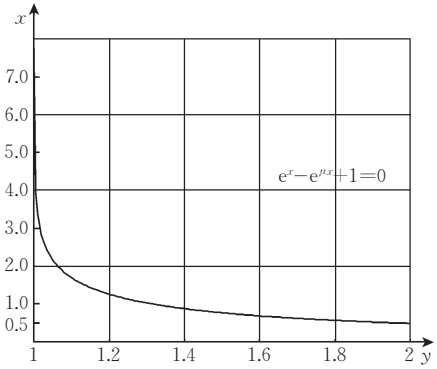

图 5 隐函数 $\mathrm{e}^x-\mathrm{e}^{\mu x}+1=0$

综合上述所有推导过程,可知 PBCC 算法生成的着色方案规模为

$$|S(n,k)|=O(\mathrm{e}^{2Rootof(\mathrm{e}^x-\mathrm{e}^{\mu x}+1)(n-k)}).$$ 证毕.

推论 1. 当 $n=2k$ 时,PBCC 算法可以生成规模为 $O(2.62^{n-k})$ 的着色方案.

证明. 由定理 2 的结论 $|S(n,k)|=O(\mathrm{e}^{2Rootof(\mathrm{e}^x-\mathrm{e}^{\mu x}+1)(n-k)})$ 可以看到,当 $n=2k$,则 $\mu=2$,解 $\mathrm{e}^x-\mathrm{e}^{\mu x}+1=0$ 可以得到 $m=0.96$.也就是说,前面的假设 $|S(n,k)|=O(\mathrm{e}^{0.961(n-k)})=O(2.62^{(n-k)})$ 在 $n=2k$ 的时候成立. 证毕.

进一步，如果 $k/n \in [0.5, 0.75]$，m 可以取值 1.53，那么 $|S(n,k)| = O(4.62^{n-k})$；如果 $k/n \in [0.5, 0.8]$，那么 $S(n,k) = O(5.73^{n-k})$.

推论 1 声明在 $n=2k$ 时，PBCC 算法生成的着色方案规模为 $O(2.62^{n-k})$. 虽然 $2.62 < e$，但是由于 $n \geq k$ 条件在这种情况下不再成立，因此这个结论与文献[7]中着色方案规模下界为 $O(e^k)$ 的结论是不相互矛盾的. 本文将在 4.2 节中给出这个问题的进一步解释.

4　$n \leq 2k$ 情况下的着色方案规模的下界分析

4.1　严格下界分析

在文献[7]中 Chen 等人通过对着色最大化覆盖的方法证明了在 n 远大于 k 时，确定式着色方案规模的下界为 e^k. 本文将分析在 $n \leq 2k$ 的情况下着色方案规模的下界，这为在该情况下设计着色算法提供了理论指导. 首先我们给出在任何情况下关于着色规模的下界的一个引理.

引理 1.　任何规模小于 $L(n,k)$ 的着色集合均不可能为一个 (n,k) 着色方案. 其中

$$L(n,k) = \frac{\binom{n}{k}}{\lceil n/k \rceil^{n\%k} \lfloor n/k \rfloor^{k-n\%k}} \tag{10}$$

证明.　一个大小为 n 的全集 U 的 k 元素子集个数为 $\binom{n}{k}$，为使得着色方案的规模尽可能小，我们必须使得方案中每种着色可以独立覆盖的子集尽可能多. 设某着色中为第 k 种颜色的元素数为 C_k，则该方案可覆盖的子集数为 $\prod_k C_k$，其中

$$\sum_k C_k = n, \text{ 其中 } C_k \text{ 为自然数} \tag{11}$$

所以我们可以由 Cauchy 不等式推得

$$\prod_k C_k \leq (n/k)^k, \text{ 当所有 } C_k \text{ 均为 } n/k \text{ 时，等号成立} \tag{12}$$

注意到限制条件 C_k 是一个自然数，所以当 n 不能整除 k 的时候，式(12)中的等号无法成立. 在此情况下，只能尽可能地平均分配着色数目，使 $\prod_k C_k$ 达到最大，即有 $n\%k$ 种颜色各自对 $\lceil n/k \rceil$ 个元素着色，剩余的 $k-n\%k$ 种颜色则各自对 $\lfloor n/k \rfloor$ 个元素着色. 一个这样的着色可以覆盖到最多的子集，即

$$Q(n,k) = \max \prod_k C_k = \lceil n/k \rceil^{n\%k} \lfloor n/k \rfloor^{k-n\%k} \tag{13}$$

一个大小为 n 的全集 U 的 k 元素子集个数为 $\binom{n}{k}$，若要覆盖所有的 $\binom{n}{k}$ 个子集，则最少需要 $\frac{\binom{n}{k}}{Q(n,k)}$ 种着色.　证毕.

从这里很容易看出，要使得着色方案规模尽可能地逼近下界，就要求着色方案中的所有着色尽可能地独立覆盖最多的子集. 要使得着色覆盖的子集尽可能地多，需使得着色中每种颜色的使用次数尽量平均. 然而这种简单的策略只能保证该着色本身覆盖最多的子集而不保证它是独立覆盖这些子集的，也就是说其它着色很可能也对该着色覆盖的子集进行了重复覆盖，但这种重复覆盖几乎是不可避免的——除了极少数特殊情况. 即使在 $k=2$ 这种简单情况下，重复覆盖也会使得实际的着色方案比下界高得多.

为便于分析，给出相关定义如下.

定义 4(最大覆盖着色).　给定元素全集 U 和 (n,k) 着色 H，若不存在任何 (n,k) 着色 H'，使得 H' 可以覆盖的子集数目超过 H 可以覆盖的子集数，则称 H 为最大覆盖着色.

定义 5(独立覆盖).　给定着色方案 $S(n,k)$，属于 $S(n,k)$ 的着色 H 及子集 W. 若 $H \xrightarrow{c} W$，且不存在任何 $H' \in S(n,k)$，使得 $H' \xrightarrow{c} W$，则称 H 独立覆盖 W. 相应的，如果存在这样的 H'，则称 H 和 H' 存在重复覆盖.

引理 2.　在 $n \leq 2k$ 的情况下，如果有最大覆盖着色 H，使得 $n-k$ 种复用色正好出现在 $n-k$ 个块中，那么必存在一个被 H 覆盖子集 W 满足这 $n-k$ 个块各有一个元素不属于 W.

证明.　首先产生一个包含全集的所有元素的子集 W；对于 $n-k$ 个包含复用色的块，如果块为同色块，则从 W 中去除块中任意一个元素，并继续处理下一个块；若块为异色块，设颜色为 $\{c_i, c_j\}$，同样从 W 中去除块中任意一个元素，设这个元素的着色为 c_i，则寻找包含着色 c_j 的块，这个块同样也是异色块，设着色为 $\{c_j, c_k\}$，然后从这个块中从 W 中去除块中着色为 c_j 的元素，再寻找包含着色 c_k 的块，直到寻找到的块中包含 c_i 再处理其它块. 按照这个步骤，从 W 中分别去除 $n-k$ 个块上一个元素，且每种复

用色所赋值的元素都被去除了一个,则显然 W 中元素着色互不相同,有 $H \xrightarrow{c} W$. 而且,由于处理过程中可以去除块中任意一个元素,所以容易看到在 $n>k$ 时至少存在 2 个这样的子集 W. 证毕.

引理 3. 在 $n\leq 2k$ 的情况下,任何着色方案 $S(n,k)$ 中不存在重复覆盖的最大覆盖着色数目不大于 $\binom{\lceil n/2\rceil}{n-k}$.

证明. 设已存在一个最大覆盖着色 H 和一种划分,且在这种划分下,H 有 $n-k$ 个同色块. 任取一种新的最大覆盖着色 H',将有以下情况:

(1) H' 含有复用色的块也正好和 H 相同,那么这些块中的元素所着颜色都为复用色. 由引理 2 可知,这种 H' 和 H 之间一定存在重复覆盖.

(2) H' 含有复用色的块为 $n-k$ 个,但至少存在一个块与 H 不同. 在这种情况下,H' 不会与 H 重复覆盖.

(3) 如果加入的 H' 包含复用色的块多于 $n-k$ 个而且与 H 不存在重复覆盖,就可以去除着色 H'.

由此可知,为了在不存在重复覆盖的条件下,最大化能加入着色方案的最大覆盖着色数目,要求每次加入方案的着色都仅包含 $n-k$ 个含有复用色的块. 这种着色的数目等同于在 $\lceil n/2\rceil$ 个块中选择 $n-k$ 个块的方法,在超出这个数目以后,加入其它最大覆盖着色将不可避免地和方案中原有的最大覆盖着色产生重复覆盖. 由此可知在 $n\leq 2k$ 的情况下,任何着色方案 $S(n,k)$ 中不存在重复覆盖的最大覆盖着色数目不大于 $\binom{\lceil n/2\rceil}{n-k}$. 证毕.

下面我们进一步分析在 $n\leq 2k$ 的情况下着色方案的下界. 由于 $k=n-1$ 的情况下,PBCC 可以得到最小规模着色方案,下面仅对 $n-k\geq 2$ 的情况给出更严格的着色方案规模下界.

定理 3. 在 $n\leq 2k$ 且 $n-k\geq 2$ 的情况下,$|S(n,k)|$ 不小于 $\binom{\lceil n/2\rceil}{n-k}+\dfrac{\binom{n}{n-k}-\binom{\lceil n/2\rceil}{n-k}2^{n-k}}{2^{n-k}-2}$.

证明. 从引理 3 可以看到,在 $n\leq 2k$ 的情况下,在能够加入的不存在重复覆盖的最大覆盖着色只有 $\binom{\lceil n/2\rceil}{n-k}$ 种. 又由引理 2 可知,在方案中已经存在这 $\binom{\lceil n/2\rceil}{n-k}$ 个最大覆盖着色之后,其余任何最大覆盖着色 H 都至少和方案中的着色有 2 个子集被重复覆盖,则 H 能独立覆盖的子集最多为 $2^{n-k}-2$ 种. 此外非最大覆盖着色能覆盖的子集数目最多为 $3\times 2^{n-k-2}$ 种. 在 $n-k\geq 2$ 的情况下,前者覆盖的子集数目较多. 在这种情况下,可以构造的着色方案规模至少为 $\binom{\lceil n/2\rceil}{n-k}+\dfrac{\binom{n}{n-k}-\binom{\lceil n/2\rceil}{n-k}2^{n-k}}{2^{n-k}-2}$.

如果着色方案中不存在重复覆盖的最大覆盖着色数目小于 $\binom{\lceil n/2\rceil}{n-k}$,设这个数目为 T. 假设有着色方案仅含有这 T 个着色,则若新加入的着色 H 能独立覆盖的子集数目不大于 $2^{n-k}-2$ 种,那么这种情况下将无法得到比 $\binom{n/2}{n-k}+\dfrac{\binom{n}{n-k}-\binom{\lceil n/2\rceil}{n-k}2^{n-k}}{2^{n-k}-2}$ 更小的着色方案. 而由假设条件,H 也不能独立覆盖 2^{n-k} 个子集. 那么,如果着色 H 可以独立覆盖的子集数目为 2^{n-k-1},则 H 是一个最大覆盖着色,且它正好和一个最大覆盖着色 H' 发生重复覆盖. 为了满足这个条件,H 含有复用色的块数目一定大于 $n-k$. 而且由于 H 不和除 H' 以外其它任何着色产生重复覆盖,所以如果着色方案中不包含 H,就可以继续加入不产生重复覆盖的最大覆盖着色,并获得比加入 H 产生的着色方案规模更小的着色方案. 证毕.

由于定理 3 分析的下界中说明了着色方案中的着色不可能都是不存在重复覆盖的最大覆盖着色,所以这个下界比引理 3 中给出的下界更加严格. 应用定理 3 到(20,16)-Motif 查找中,可以看到要构造一个(20,16)着色问题的着色方案至少需要 317 种着色.

4.2 渐进下界分析

尽管文献[7]已经为着色方案规模 $|S(n,k)|$ 给出了 e^k 的渐近下界,但是注意到第 3.7 节中得到的结论:当 $n=2k$ 时,PBCC 算法产生的着色方案规模为 $O(2.62^{n-k})=O(2.62^k)<O(e^k)$. 这个结论看起来是有问题的,但是问题其实在 e^k 渐近下界的成立条件上. 当 $(n-k)\to\infty$ 时,$|S(n,k)|\geq e^k$. 而在这一节中,本文将着重分析 $n\leq 2k$ 情况下,着色方案规模的渐近下界.

本文使用与文献[10]中计算 $|S(n,k)|$ 下界类似的方法,但是消除 k 的小参数限制条件.

定理 4. 给定 $k/n=\lambda$,任意 (n,k) 着色方案的规模为 $\Omega((1/(1-\lambda)^{n-k})$.

证明.　已知$|S(n,k)|\geq\frac{\binom{n}{k}}{(n/k)^k}$，使用 Stirling 近似公式，得到

$$\binom{n}{k}=\Omega(n^n/(\sqrt{k}(n-k)^{n-k}k^k)) \tag{14}$$

因此，

$$|S(n,k)|\geq n^{n-k}/(\sqrt{k}(n-k)^{n-k}) \tag{15}$$

如果 $k/n=\lambda$，任何着色方案的规模都满足

$$|S(n,k)|=\Omega(1/(1-\lambda)^{n-k}) \tag{16}$$

这就是说，无论构造任何着色方案构造算法，它产生的着色方案规模至少为 $\Omega(1/(1-\lambda)^{n-k})$.

特别的，如果 $k/n=0.5$，$|S(n,k)|=\Omega(2^{n-k})$；如果 $k/n=0.75$，$|S(n,k)|=\Omega(4^{n-k})$；而 $k/n=0.8$，$|S(n,k)|=\Omega(5^{n-k})$. 换言之，在 $n\leq 2k$ 的情况下，构造一个能产生着色方案规模为 $O(c^{n-k})$（c 为常数）的着色方案构造算法是不可能的，这个结果也从一个侧面解释了式(9)没有得到常数底的原因.

5　结　论

目前存在的着色算法均基于完全散列函数，并要求 n 远大于 k，且 k 比较小. 这个限制条件使得这些着色算法在一些现实应用中并不实用. 本文探讨了在 $n\leq 2k$ 情况下的着色算法，这种算法适用于类似于 Motif 查找的各种应用. 本文提出了一种基于划分思想的着色算法 PBCC，证明了由 PBCC 产生的着色方案确实可以覆盖到所有的子集，并可以用于着色算法的确定化. 文章进一步分析了 PBCC 产生的着色方案规模，并给出了在 $n\leq 2k$ 情况下更严格的着色方案规模的下界.

参　考　文　献

[1] Bodlaender H L. On linear time minor tests with depth-first search. Journal of Algorithms, 1993, 14(1):1-23

[2] Monien B. How to find long paths efficiently. Annals of Discrete Mathematics, 1985, 25: 239-254

[3] Alon N, Yuster R, Zwick U. Color-coding. Journal of the ACM, 1995, 42(4): 844-856

[4] Scott J, Ideker T, Karp R M et al. Efficient algorithms for detecting signaling pathways in protein interaction networks. Journal of Computational Biology, 2005, 13(2):133-144

[5] Chen J, Lu S, Sze S H et al. On subgraph matching problems in biological interaction networks. Department of Computer Science, Texas A&M University, 2005

[6] Wang Jian-Xin, Huang Yuan-Nan, Chen Jian-Er. A novel motif finding algorithm based on color coding technique. Journal of Software, 2007, 18(6): 1298-1307(in Chinese)
(王建新，黄元南，陈建二. 一种基于彩色编码的基序发现算法. 软件学报, 2007, 18(6): 1298-1307)

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

[8] Schmidt J P, Siegel A. The spatial complexity of oblivious k-probe Hash functions. SIAM Journal of Computing, 1990, 19(5): 775-786

[9] Chen J, Lu S, Sze S H et al. Color-coding revised. Department of Computer Science, Texas A&M University, 2005

[10] Chen J, Lu S, Sze S H et al. Improved algorithms for path, matching, and packing problems//Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 07). New Orleans, Louisiana, 2007: 298-307

WANG Jian-Xin, born in 1969, Ph. D., professor, Ph. D. supervisor. His current research interests include algorithm analysis and optimization, computer network and bioinformatics.

LIU Yun-Long, born in 1983, master. His research interests include computer algorithm and bioinformatics.

CHEN Jian-Er, born in 1954, Ph. D., professor, Ph. D. supervisor. His research interests include algorithms and complexity, computer networks, and computer graphics.

Background

This work is supported by the National Natural Science Foundation of China (60433020) and the Program for New Century Excellent Talents in University (NCET-05-0683).

With the development of computing theories, more and more real-world applications have been proved NP-Hard in the basis of model view. Therefore, to formulate some practical algorithms for these problems, researchers have tried various solutions, for example, a novel technique named color-coding. And this technique is being applied to more and more practical problems in recent years.

Color-coding can be applied to some NP-Hard problems in subset selection. The key idea in this technique is make a transformation from the restriction of distinct elements to the restriction of distinct colors, and then the transformation will lead to the shrink of solution space therefore make the problem easier. The application of color-coding can be classified into two categories according to different coloring schemes: Randomized color-coding and deterministic color-coding.

Although color-coding improves the efficiency on solve subset selection-like problems, it still has many deficiencies as an inchoate solution:

(1) Despite the complexity of randomized color-coding is acceptable in most of the cases, its accuracy cannot meet the demands from time to time, especially in those situations where all solutions except exact ones are invalid;

(2) The complexity of deterministic color-coding has been reduced a lot with the efforts of researchers, albeit most coloring schemes provided by these algorithms are oversized and impractical;

(3) At present, all deterministic color-codings are based on perfect hash functions and fix parameter theory, therefore there is a presumption that the number of elements is far greater than the number of colors, and there are few colors. This presumption constrains the application of color-coding for it is not always stands.

Thus, in order to improve the practicality of color-coding and enable the applications of color-coding in more areas, this paper mainly concerns on the circumstance where the size of element set and color set are close. To make full use of this condition, this paper proposes a non-hash style coloring algorithm named PBCC (Partition-Based Color-Coding). This algorithm intends to make every coloring covers as many subsets as possible therefore minimize the size of coloring scheme through equal partitioning and the consideration of coloring distribution between subsets.

PBCC is proved to be efficient under the circumstance where the size of element set and color set are close by both application and theoretical analysis. It can generate much smaller coloring schemes than other algorithms such as those hash-style ones, therefore it can be applied to computing biology problems such as motif-finding and its similar ones.

Furthermore, since deciding lower bound of coloring schemes using just maximum coverage cannot points out where the bottleneck on reducing colorings, the authors also study on the lower bound in this situation, and show that repetitions between colorings are the obstacles in reducing colorings, and the obstacles cannot be annihilated. Therefore, a stricter lower bound is got by study of repetitions between colorings.