经典 Ramsey 数 DNA 计算模型(Ⅰ):位序列计算模型

许 进[1),2)]　　范月科[2)]

1)(北京大学信息科学技术学院高可信软件技术教育部重点实验室 北京 100871)

2)(华中科技大学分子生物计算机研究所 武汉 430074)

摘 要 Ramsey 数问题是组合数学乃至整个数学中最具魅力的研究领域,也是最困难的数学问题之一.对于经典 Ramsey 数,至今只有 9 个 Ramsey 数得到解决.按照传统的算法,其搜索空间太大,当前的电子计算机无法胜任.研究表明,DNA 计算在求解困难的 NP-完全问题上优于电子计算机.目前已经建立了众多求解 NP-完全问题的 DNA 计算模型,但未见到用于求解 Ramsey 数的 DNA 计算模型.作者建立了一种新颖的 DNA 计算模型,用于一般经典 Ramsey 数的求解.全文共分两篇,该文属首篇,建立了一种可适用于 DNA 计算模式的所谓的求解 Ramsey 数的位序列计算模型,其中的位序列是以图的相邻矩阵下三角阵中行从左到右、列从上到下的排列次序.文中重点对该模型的机理与使用方法进行了分析研究.

关键词 经典 Ramsey 数;DNA 计算;位序列计算模型

中图法分类号 TP301

Classical Ramsey Number DNA Computing Model (Ⅰ): Add-Bit-Sequence Model

XU Jin[1),2)]　FAN Yue-Ke[2)]

1)(*Institute of Software, School of Electronics Engineering and Computer Science, Key Laboratory of High Confidence Software Technologies of Ministry of Education, Peking University, Beijing* 100871)

2)(*Institute of Biomolecular Computer, Huazhong University of Science and Technology, Wuhan* 430074)

Abstract Classical Ramsey number problem is a NP complete problem. It takes exponential time to solve classical Ramsey number problem with traditional electronics computer. It is necessary to study new computation methods because traditional electronics computer faces with greatly difficulty in solving NP complete problem. DNA computing possesses high parallelism in data and higher storage capacity than normal systems. Hence, in theory, it is feasible to solve NP complete problems with DNA computing. A novel method is proposed and used to solve the classical Ramsey number problem in this paper. Firstly, a method for DNA sequence code based on graphic sequence is presented. In order to reflect all possible information of p-order graphs, it is focused on the encoding of p-order complete graph. The number of the edges in the graph is $p*(p-1)/2!$. The delta-encoding method is selected. The sequences of encodings are the edges $1,2,\cdots, p*(p-1)/2!$. Here, the edges $1,2,\cdots,p*(p-1)/2!$ represent adjacent lower triangular matrix array in order from left to right, top to bottomin the graph G. The character of this encoding is to sort them into $p-1$ classes, and the edges in the i class are composed of $\{v_1, v_{u+1}\}$, $\{v_2, v_{i+1}, \cdots, \{v_i, v_{i+1}\}$, $i=1,2,\cdots,p-1$. According to the encoding, the length $p*(p-1)/2!$ of 0-1 sequence can be mapped to all labeled sub-graphs with p vertexes. It means that arbitrary 0-1 sequence with length $p*(p-1)/2!$ corresponds to the graph with the order p; On the other hand, any order p for the labeled graph corresponds to 0-1 sequence with length $p*(p-1)/2!$.

收稿日期:2008-10-30. **许 进**,男,1959 年生,教授,博士生导师,研究兴趣为 DNA 计算和 DNA 计算机、神经网络、遗传算法、图论等. **范月科**,男,1959 年生,博士,研究兴趣为 DNA 计算和 DNA 计算机、图论等.

Secondly, some problems are discussed, such as the set problem of labeling sub-graph and un-labeling sub-graph, especial the method of sequence denoted. Finally, in order to delete the incorrect solutions in the first, the idea, method and process of deleting incorrect solutions are presented, and an instance is presented to illuminate the proposed method. The basic idea is to construct a graph by bit by step. In order to delete incorrect solutions as soon as possible the graph constructed dose not include the K_m and N_n. The method, which deletes the set of sub-graphs including K_m and N_n from the set of all K_m-order sub-graph labeled, deletes the corresponding bit, which is corresponding to the edges set of all m-order Complete sub-graph, from the corresponding sequence L_q; including the set of sequences corresponding bit. The key step for the bit sequence based-on DNA computation can delete the incorrect solution whenever it is brought up. Consequently, it can weaken the speed of which the number of sub-graph increases, and also provides a feasible method for finding the Ramsey number with much higher order. In addition, it is demonstrated the possibility of using the advantage inherent in DNA computation, including vast information storage capacities and much highly computing ability. The more detail researches about how to solve the Ramsey number using DNA computer with be given in the second paper.

Keywords Ramsey number; DNA computing; add-bit-sequence computing model

1 引 言

给定任意两个正整数 k,l，总存在一个最小的正整数 $r(k,l)$，使得任意 $r(k,l)$ 个顶点的图，或者含有 k 个顶点的团，或者含有 l 个顶点的独立集. 我们把这个最小的正整数 $r(k,l)$ 称为关于 (k,l) 的 Ramsey 数. 容易算出 $r(1,l)=r(k,1)=1$，$r(2,l)=l$，$r(k,2)=k$，$r(k,l)=r(l,k)$. 我们把 $r(k,l)$ 称为经典 Ramsey 数.

本文所言之图皆指有限、无向、简单图. 设正整数 $m,n,p\geq 1$，Ramsey (m,n)-图是指既不含 m 个顶点的团也不含 n 个顶点独立集的图；Ramsey (m,n,p)-图是指阶数为 p 的 Ramsey (m,n)-图. 分别用 $G(m,n)$，$G(m,n,p)$ 表示所有 Ramsey (m,n)-图和 Ramsey (m,n,p)-图的集合. Ramsey 数 $r(m,n)$ 定义为不存在 Ramsey (m,n,p)-图的最小数 p. 目前已确定的经典 Ramsey 数有：$r(3,3)=6$，只有 1 个 Ramsey 图；$r(3,4)=9$，共有 3 个 Ramsey 图；$r(3,5)=14$，只有 1 个 Ramsey 图；$r(3,6)=18$，共有 191 个 Ramsey 图；$r(3,7)=23$，共有 191 个 Ramsey 图；$r(3,8)=28$，已知道它有 430215 个 Ramsey 图；$r(3,9)=36$，只有 1 个 Ramsey 图；$r(4,4)=18$，只有 1 个 Ramsey 图；$r(4,5)=25$，已发现 350904 个 Ramsey 图(可能更多)[1-3].

目前已找到 105 个 34 阶 Ramsey (4,6)-图，人们推测 35～40 阶 Ramsey (4,6)-图可能存在. 目前已找到 656 个 42 阶 Ramsey (5,5)-图，人们推测 43～49阶 Ramsey (5,5)-图可能存在(这些结果参见文献[1]). 注意，每个 Ramsey (5,5)-图的补图也是 Ramsey (5,5)-图.

近 20 年来，电子计算机在求解 Ramsey 数问题的研究上起到了巨大的促进作用，如 Ramsey 数 $r(3,8)$ 和 $r(4,5)$ 均是借助于电子计算机来完成的[2-3]. 但随着问题规模的增大，如 $r(3,10)$ 至少需要从 40 个顶点的图集中搜索是否存在 Ramsey 图，而这个搜索次数理论上需要约 2^{760} 个图，显然电子计算机对此是无法实现的. 这就迫使科学家在 Ramsey 数问题的研究上另辟蹊径.

最近 10 多年来，DNA 计算机的研究得到了长足的发展，已建立了不少求解组合优化中 NP-完全问题的 DNA 计算模型，诸如 Hamilton 路与圈问题[4-6]、图的最大团与最大独立集问题[7-8]、图顶点着色问题[9-11]、中国邮递员问题[12]、0-1 规划问题[13-14]等，但至今尚未见到一篇关于求解 Ramsey 数这个组合数学领域困难问题的 DNA 计算模型. 2007 年，我们在 DNA 计算机研究方面有一个较大的突破：给出了消除解空间指数爆炸问题的一种新方法，在此基础上建立了一个求解 61 个顶点图的 3-着色问题 DNA 计算模型并成功地进行了实验，这标志着 DNA 计算机不仅已经具备了进行某些大规模信息处理的能力，而且也可以用于解决电子计算机无法解决的问题了. 正是受到这个工作的启迪，我们在本文中建立了用于求解经典 Ramsey 数的

DNA 计算机模型.

借助于电子计算机在求解经典 Ramsey 数模型的研究上，Mckay 等人作出了杰出的贡献[2-3]. 他们所建立模型的基本思想是利用图的自同构群来删除同构子图，其具体做法是：首先建立 10 个顶点所有非标定子图的集合，然后以这 10 个顶点非标定子图的集合为基础，进一步构造所需要的图集合.

Mckay 等人在进一步构造所需的图集合时，一般通过每次增加一个顶点来完成. 但是，即使如此，一下子增加的非解也是很多，从而导致电子计算机超过一定的范围就无能为力. 这也就是目前利用电子计算机求解 Ramsey 数停滞不前的根本原因所在.

本文提出的模型，在非解的删除上与 Mckay 等人的模型相比，只增加一个位，也就是说，只增加一条边所占位置的一半，这样可以过早地删除非解，使得非解空间大大降低. 例如，对于一个 p 阶图而言，按照 Mckay 的方法，需要增加的位数是 $2(p-1)$，而按照本文的方法，仅仅 1 位. 也就是说，Mckay 一次增加的解空间为 $2^{2(p-1)}$ 倍，而本模型仅为 2 倍.

我们把这种模型称为求解经典 Ramsey 数的边序列模型，而 Mckay 的模型称为点序列模型. 边序列模型适应于电子计算机的计算，更适应于 DNA 计算机模型.

文中约定，在整篇论文中，从 x 个元素中取出 y 个元素的组合数的两种表示方式，C_x^y 和 $\binom{x}{y}$ 是一样的，可根据不同的排版需要任意选取. 有关图与 Ramsey 数方面的术语与记号可参见文献[15-16].

2　序列编码

由于我们所采用信息处理的“数据”是线性 DNA 序列，因而必须将一个给定图的全部信息在一个序列中完整地反应出来. 众所周知，用图的顶点序列是不可能反应出图的完整的信息，而图的边序列可完整地反应出图的信息.

为了反应出所有可能的 p-阶图的信息，我们针对 p-阶完全图进行编码. p-阶完全图共有边的数目是：

$$\binom{p}{2}=\frac{1}{2}p(p-1).$$

因此，给定的编码序列为 $1,2,\cdots,C_p^2$. 那么，边如何排序呢？即，序列 $1,2,\cdots,C_p^2$ 中每个元素如何与这 C_p^2 条边一一对应起来. 我们在本文中选择了所谓的 Δ-编码方法. 具体方法如下：对于具有 p 个顶点的图 G，令图的顶点集 $V=V(G)=\{v_1,v_2,\cdots,v_p\}$，边集 $E=\{\{v_i,v_j\};i,j\in V,i\neq j\}$. 编码的序列，即边的排列序列为 $1,2,\cdots,C_p^2$. 其中 $1,2,\cdots,C_p^2$ 所表示的边为图 G 相邻矩阵中下三角阵依次从左到右、从上到下的次序，如图 1 所示.

$$\begin{array}{c} 1\\ 2\\ 3\\ 4\\ 5\\ \vdots\\ C_p^2 \end{array}\left(\begin{array}{cccccc} & & & & & \\ 1 & & & & & \\ 2 & 3 & & & & \\ 4 & 5 & 6 & & & \\ 7 & 8 & 9 & 10 & & \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ C_{p-1}^2+1 & C_{p-1}^2+2 & C_{p-1}^2+3 & \cdots & \cdots & C_p^2 \end{array}\right)$$

图 1　码的定义

现在以 $p=6$ 为例给予说明：$V=\{v_1,v_2,v_3,v_4,v_5,v_6\}$，序列的长度为 $C_6^2=15$，即序列为 $1,2,\cdots,15$，其中，

$1=\{v_1,v_2\}$，$2=\{v_1,v_3\}$，$3=\{v_2,v_3\}$，$4=\{v_1,v_4\}$，$5=\{v_2,v_4\}$，$6=\{v_3,v_4\}$，$7=\{v_1,v_5\}$，$8=\{v_2,v_5\}$，$9=\{v_3,v_5\}$，$10=\{v_4,v_5\}$，$11=\{v_1,v_6\}$，$12=\{v_2,v_6\}$，$13=\{v_3,v_6\}$，$14=\{v_4,v_6\}$，$15=\{v_5,v_6\}$.

这种编码的特点是将码分成了 $p-1$ 类，第 i 类中的边的构成是 $\{v_1,v_{i+1}\}$，$\{v_2,v_{i+1}\}$，$\{v_3,v_{i+1}\}$，$\cdots$，$\{v_i,v_{i+1}\}$，其中，$i=1,2,\cdots,p-1$.

按照这种编码，长度为 C_p^2 的 0-1 序列集合与具有 p 个顶点的所有标定子图构成的集合之间，通过映射：

$$f(e=\{v_i,v_j\})=\begin{cases}1, & e\in E(G)\\ 0, & e\notin E(G)\end{cases}\qquad(1)$$

可建立起一一对应的关系. 也就是说，任意一个长度为 C_p^2 的 0-1 序列，通过式(1)唯一对应一个阶数为 p 的标定图；反过来，任意一个阶数为 p 的标定图唯一对应于一个 C_p^2 的 0-1 序列. 下面举例给予说明.

例 1.　对于 5-圈图 C_5，如图 2 所示，共有 5 条边，分别是 $1=\{v_1,v_2\}$，$3=\{v_2,v_3\}$，$6=\{v_3,v_4\}$，$7=\{v_1,v_5\}$，$10=\{v_4,v_5\}$，这 5 条边对应位置为 1，其余为 0，于是，这个图对应的序列为 1010011001. 按这种编码，我们自然会提出下列问题.

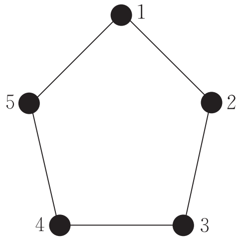

图 2　5-圈图

问题：什么样的序列对应的标定图是同构的？如何消除？如 1010011001，1011000011，1000111010，1100100011，1001010110，1100010101，0110101001，0111000101，0100111100，0101100110，0011101010，0011011100. 按照式(1)所对应的图依次由图 3 给出. 显然，这些图均是同构的，但对应的 0-1 序列却不同. 这个事实使得我们需要构造的，或者寻找 Ramsey 图显得非常困难.

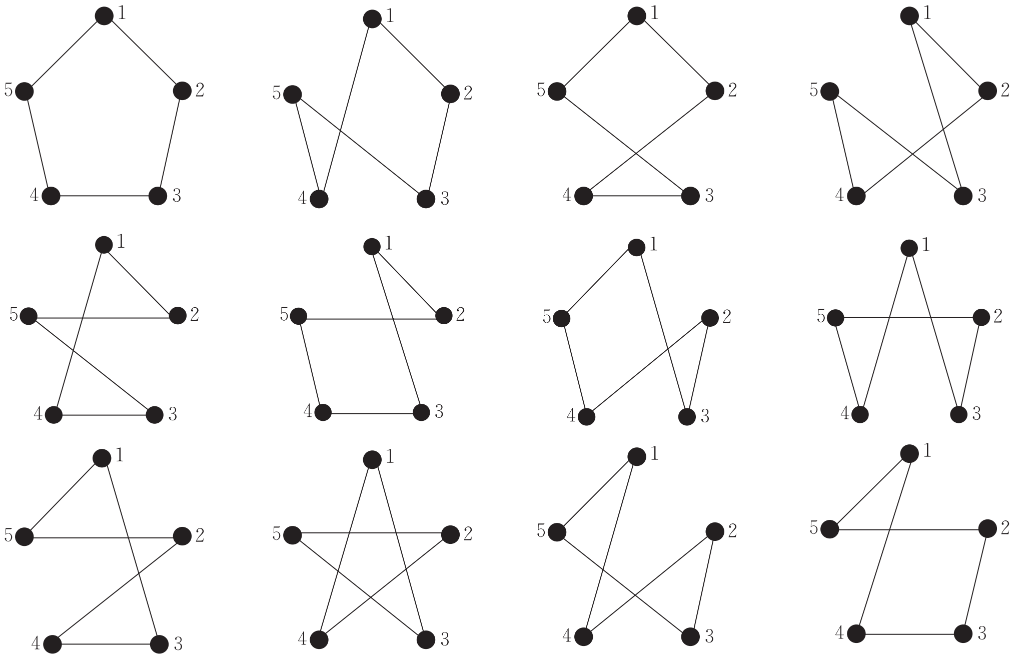

图 3　12 个同构的图

3　非解的删除问题

本模型的基本思想是：按位逐步构造出所需要的，既不含 K_m 也不含 N_n 的图. 这样的目的是尽可能早地删除非解.

从标定的所有 p-阶子图集合 G_p 中删除既含有 K_m 且 N_n 的子图集合方法是从对应的序列 L_q 中删除对应含有所有可能的构成 m-完全子图对应的边集合对应位，以及含有所有可能的构成 n-完全空图的所有边的对应位的序列集合. 也就是说，需要删除的次数是

$$\binom{p}{m}+\binom{p}{n}.$$

为后面方便，我们约定：G_p 的顶点集合用 $\{1,2,\cdots,p\}$ 来代替 $\{v_1,v_2,\cdots,v_p\}$，并在此强调：构成表示图集合 G_p 的所有序列构成的集合 L_p 的第 i 位用 l_i 表示，这里，$i=1,2,\cdots,q=C_p^2$.

总之，对于 $r(m,n)$ 而言，需要实实在在地运行 $C_p^m+C_p^n$ 次，特别当 $m=n$ 时，可设计使运行次数为 $2C_p^m$ 次. 且删除算法的步骤简要如下：

1. 构造出含有可构成 m-阶完全子图 K_m 的 C_p^m 个集合；
2. 应用对照表，得到所删除对应序列中的位置集合；
3. 构造出含有可构成 n-阶完全空图 N_n 的 C_p^n 个集合；
4. 应用对照表，得到所删除对应序列中的位置集合.

4　求解 $r(3,4)$ 的计算实例

业已得知 $r(3,4)=9$，即需要从 8 个顶点的图集中找到 Ramsey 图，而 8 个顶点共有长度为 $C_8^2=\frac{8\times 7}{2}=28$ 位的 0-1 序列. 具体步骤如下.

步骤 1. 建立对照表.

表 1　对照表

序列位	对应边	序列位	对应边	序列位	对应边
l_1	{1,2}	l_{11}	{1,6}	l_{21}	{6,7}
l_2	{1,3}	l_{12}	{2,6}	l_{22}	{1,8}
l_3	{2,3}	l_{13}	{3,6}	l_{23}	{2,8}
l_4	{1,4}	l_{14}	{4,6}	l_{24}	{3,8}
l_5	{2,4}	l_{15}	{5,6}	l_{25}	{4,8}
l_6	{3,4}	l_{16}	{1,7}	l_{26}	{5,8}
l_7	{1,5}	l_{17}	{2,7}	l_{27}	{6,8}
l_8	{2,5}	l_{18}	{3,7}	l_{28}	{7,8}
l_9	{3,5}	l_{19}	{4,7}		
l_{10}	{4,5}	l_{20}	{5,7}		

步骤 2. 确定需要删除 K_3 的子集合及对应 L_8 中的位集合.

对于 8 个顶点的图，需要删除 K_3 的子集合的数目是 $C_8^3=8\times7\times6/6=56$ 个. 这 56 个子集中每个对应完全子图的边集以及该边集对应序列集 L_8 中的位置集合分别是

$\{1,2,3\}\rightarrow\{12,13,23\}=\{l_1,l_2,l_3\}$,

$\{1,2,4\}\rightarrow\{12,14,24\}=\{l_1,l_4,l_5\}$,

$\{1,2,5\}\rightarrow\{12,15,25\}=\{l_1,l_7,l_8\}$,

$\{1,2,6\}\rightarrow\{12,16,26\}=\{l_1,l_{11},l_{12}\}$,

$\{1,2,7\}\rightarrow\{12,17,27\}=\{l_1,l_{16},l_{17}\}$,

$\{1,2,8\}\rightarrow\{12,18,28\}=\{l_1,l_{22},l_{23}\}$,

$\{1,3,4\}\rightarrow\{13,14,34\}=\{l_2,l_4,l_6\}$,

$\{1,3,5\}\rightarrow\{13,15,35\}=\{l_2,l_7,l_9\}$,

$\{1,3,6\}\rightarrow\{13,16,36\}=\{l_2,l_{11},l_{13}\}$,

$\{1,3,7\}\rightarrow\{13,17,37\}=\{l_2,l_{16},l_{18}\}$,

$\{1,3,8\}\rightarrow\{13,18,38\}=\{l_2,l_{22},l_{24}\}$,

$\{1,4,5\}\rightarrow\{14,15,45\}=\{l_4,l_7,l_{10}\}$,

$\{1,4,6\}\rightarrow\{14,16,46\}=\{l_4,l_{11},l_{14}\}$,

$\{1,4,7\}\rightarrow\{14,17,47\}=\{l_4,l_{16},l_{19}\}$,

$\{1,4,8\}\rightarrow\{14,18,48\}=\{l_4,l_{22},l_{25}\}$,

$\{1,5,6\}\rightarrow\{15,16,56\}=\{l_7,l_{11},l_{15}\}$,

$\{1,5,7\}\rightarrow\{15,17,57\}=\{l_7,l_{16},l_{20}\}$,

$\{1,5,8\}\rightarrow\{15,18,58\}=\{l_7,l_{22},l_{26}\}$,

$\{1,6,7\}\rightarrow\{16,17,67\}=\{l_{11},l_{16},l_{21}\}$,

$\{1,6,8\}\rightarrow\{16,18,68\}=\{l_{11},l_{22},l_{27}\}$,

$\{1,7,8\}\rightarrow\{17,18,78\}=\{l_{16},l_{22},l_{28}\}$,

$\{2,3,4\}\rightarrow\{23,24,34\}=\{l_3,l_5,l_6\}$,

$\{2,3,5\}\rightarrow\{23,25,35\}=\{l_3,l_8,l_9\}$,

$\{2,3,6\}\rightarrow\{23,26,36\}=\{l_3,l_{12},l_{13}\}$,

$\{2,3,7\}\rightarrow\{23,27,37\}=\{l_3,l_{17},l_{18}\}$,

$\{2,3,8\}\rightarrow\{23,28,38\}=\{l_3,l_{23},l_{24}\}$,

$\{2,4,5\}\rightarrow\{24,25,45\}=\{l_5,l_8,l_{10}\}$,

$\{2,4,6\}\rightarrow\{24,26,46\}=\{l_5,l_{12},l_{14}\}$,

$\{2,4,7\}\rightarrow\{24,27,47\}=\{l_5,l_{17},l_{19}\}$,

$\{2,4,8\}\rightarrow\{24,28,48\}=\{l_5,l_{23},l_{25}\}$,

$\{2,5,6\}\rightarrow\{25,26,56\}=\{l_8,l_{12},l_{15}\}$,

$\{2,5,7\}\rightarrow\{25,27,57\}=\{l_8,l_{17},l_{20}\}$,

$\{2,5,8\}\rightarrow\{25,28,58\}=\{l_8,l_{23},l_{26}\}$,

$\{2,6,7\}\rightarrow\{26,27,67\}=\{l_{12},l_{17},l_{21}\}$,

$\{2,6,8\}\rightarrow\{26,28,68\}=\{l_{12},l_{23},l_{27}\}$,

$\{2,7,8\}\rightarrow\{27,28,78\}=\{l_{17},l_{23},l_{28}\}$,

$\{3,4,5\}\rightarrow\{34,35,45\}=\{l_6,l_9,l_{10}\}$,

$\{3,4,6\}\rightarrow\{34,36,46\}=\{l_6,l_{13},l_{14}\}$,

$\{3,4,7\}\rightarrow\{34,37,47\}=\{l_6,l_{18},l_{19}\}$,

$\{3,4,8\}\rightarrow\{34,38,48\}=\{l_6,l_{24},l_{25}\}$,

$\{3,5,6\}\rightarrow\{35,36,56\}=\{l_9,l_{13},l_{15}\}$,

$\{3,5,7\}\rightarrow\{35,37,57\}=\{l_9,l_{18},l_{20}\}$,

$\{3,5,8\}\rightarrow\{35,38,58\}=\{l_9,l_{24},l_{26}\}$,

$\{3,6,7\}\rightarrow\{36,37,67\}=\{l_{13},l_{18},l_{21}\}$,

$\{3,6,8\}\rightarrow\{36,38,68\}=\{l_{13},l_{24},l_{27}\}$,

$\{3,7,8\}\rightarrow\{37,38,78\}=\{l_{18},l_{24},l_{28}\}$,

$\{4,5,6\}\rightarrow\{45,46,56\}=\{l_{10},l_{14},l_{15}\}$,

$\{4,5,7\}\rightarrow\{45,47,57\}=\{l_{10},l_{19},l_{20}\}$,

$\{4,5,8\}\rightarrow\{45,48,58\}=\{l_{10},l_{25},l_{26}\}$,

$\{4,6,7\}\rightarrow\{46,47,67\}=\{l_{14},l_{19},l_{21}\}$,

$\{4,6,8\}\rightarrow\{46,48,68\}=\{l_{14},l_{25},l_{27}\}$,

$\{4,7,8\}\rightarrow\{47,48,78\}=\{l_{19},l_{25},l_{28}\}$,

$\{5,6,7\}\rightarrow\{56,57,67\}=\{l_{15},l_{20},l_{21}\}$,

$\{5,6,8\}\rightarrow\{56,58,68\}=\{l_{15},l_{26},l_{27}\}$,

$\{5,7,8\}\rightarrow\{57,58,78\}=\{l_{20},l_{26},l_{28}\}$,

$\{6,7,8\}\rightarrow\{67,68,78\}=\{l_{21},l_{27},l_{28}\}$.

步骤 3. 确定需要删除 N_4 的子集合及对应 L_8 中的位集合.

对于 8 个顶点的图，需要删除 N_4 的子集合的数目是：$C_8^4=8\times7\times6\times5/(4\times3\times2\times1)=70$ 个. 这 70 个子集中每个对应完全空图的构成的所有空边集以及该空边集对应序列集 L_8 中的位置集合分别是

$\{1,2,3,4\}\rightarrow\{12,13,23,14,24,34\}$
$=\{l_1,l_2,l_3,l_4,l_5,l_6\}$,

$\{1,2,3,5\}\rightarrow\{12,13,23,15,25,35\}$
$=\{l_1,l_2,l_3,l_7,l_8,l_9\}$,

$\{1,2,3,6\}\rightarrow\{12,13,23,16,26,36\}$
$=\{l_1,l_2,l_3,l_{11},l_{12},l_{13}\}$,

$\{1,2,3,7\}\rightarrow\{12,13,23,17,27,37\}$
$=\{l_1,l_2,l_3,l_{16},l_{17},l_{18}\}$,

$\{1,2,3,8\}\rightarrow\{12,13,23,18,28,38\}$
$=\{l_1,l_2,l_3,l_{22},l_{23},l_{24}\}$,

$\{1,2,4,5\}\rightarrow\{12,14,24,15,25,45\}$
$=\{l_1,l_4,l_5,l_7,l_8,l_{10}\}$,

$\{1,2,4,6\}\rightarrow\{12,14,24,16,26,46\}$
$=\{l_1,l_4,l_5,l_{11},l_{12},l_{14}\}$,

$\{1,2,4,7\}\rightarrow\{12,14,24,17,27,47\}$
$=\{l_1,l_4,l_5,l_{16},l_{17},l_{19}\}$,

$\{1,2,4,8\}\rightarrow\{12,14,24,18,28,48\}$
$=\{l_1,l_4,l_5,l_{22},l_{23},l_{25}\}$,

$\{1,2,5,6\}\rightarrow\{12,15,25,16,26,56\}$
$=\{l_1,l_7,l_8,l_{11},l_{12},l_{15}\}$,

$\{1,2,5,7\}\rightarrow\{12,15,25,17,27,57\}$
$=\{l_1,l_7,l_8,l_{16},l_{17},l_{20}\}$,

$\{1,2,5,8\} \to \{12,15,25,18,28,58\}$
$= \{l_1, l_7, l_8, l_{22}, l_{23}, l_{26}\}$,
$\{1,2,6,7\} \to \{12,16,26,17,27,67\}$
$= \{l_1, l_{11}, l_{12}, l_{16}, l_{17}, l_{21}\}$,
$\{1,2,6,8\} \to \{12,16,26,18,28,68\}$
$= \{l_1, l_{11}, l_{12}, l_{22}, l_{23}, l_{27}\}$,
$\{1,2,7,8\} \to \{12,17,27,18,28,78\}$
$= \{l_1, l_{16}, l_{17}, l_{22}, l_{23}, l_{28}\}$,
$\{1,3,4,5\} \to \{13,14,34,15,35,45\}$
$= \{l_2, l_4, l_6, l_7, l_9, l_{10}\}$,
$\{1,3,4,6\} \to \{13,14,34,16,36,46\}$
$= \{l_2, l_4, l_6, l_{11}, l_{13}, l_{14}\}$,
$\{1,3,4,7\} \to \{13,14,34,17,37,47\}$
$= \{l_2, l_4, l_6, l_{16}, l_{18}, l_{19}\}$,
$\{1,3,4,8\} \to \{13,14,34,18,38,48\}$
$= \{l_2, l_4, l_6, l_{22}, l_{24}, l_{25}\}$,
$\{1,3,5,6\} \to \{13,15,35,16,36,56\}$
$= \{l_2, l_7, l_9, l_{11}, l_{13}, l_{15}\}$,
$\{1,3,5,7\} \to \{13,15,35,17,37,57\}$
$= \{l_2, l_7, l_9, l_{16}, l_{18}, l_{20}\}$,
$\{1,3,5,8\} \to \{13,15,35,18,38,58\}$
$= \{l_2, l_7, l_9, l_{22}, l_{24}, l_{26}\}$,
$\{1,3,6,7\} \to \{13,16,36,17,37,67\}$
$= \{l_2, l_{11}, l_{13}, l_{16}, l_{18}, l_{21}\}$,
$\{1,3,6,8\} \to \{13,16,36,18,38,68\}$
$= \{l_2, l_{11}, l_{13}, l_{22}, l_{24}, l_{27}\}$,
$\{1,3,7,8\} \to \{13,17,37,18,38,78\}$
$= \{l_2, l_{16}, l_{18}, l_{22}, l_{24}, l_{28}\}$,
$\{1,4,5,6\} \to \{14,15,45,16,46,56\}$
$= \{l_4, l_7, l_{10}, l_{11}, l_{14}, l_{15}\}$,
$\{1,4,5,7\} \to \{14,15,45,17,47,57\}$
$= \{l_4, l_7, l_{10}, l_{16}, l_{19}, l_{20}\}$,
$\{1,4,5,8\} \to \{14,15,45,18,48,58\}$
$= \{l_4, l_7, l_{10}, l_{22}, l_{25}, l_{26}\}$,
$\{1,4,6,7\} \to \{14,16,46,17,47,67\}$
$= \{l_4, l_{11}, l_{14}, l_{16}, l_{19}, l_{21}\}$,
$\{1,4,6,8\} \to \{14,16,46,18,48,68\}$
$= \{l_4, l_{11}, l_{14}, l_{22}, l_{25}, l_{27}\}$,
$\{1,4,7,8\} \to \{14,17,47,18,48,78\}$
$= \{l_4, l_{16}, l_{19}, l_{22}, l_{25}, l_{28}\}$,
$\{1,5,6,7\} \to \{15,16,56,17,57,67\}$
$= \{l_7, l_{11}, l_{15}, l_{16}, l_{20}, l_{21}\}$,
$\{1,5,6,8\} \to \{15,16,56,18,58,68\}$
$= \{l_7, l_{11}, l_{15}, l_{22}, l_{26}, l_{27}\}$,
$\{1,5,7,8\} \to \{15,17,57,18,58,78\}$
$= \{l_7, l_{16}, l_{20}, l_{22}, l_{26}, l_{28}\}$,
$\{1,6,7,8\} \to \{16,17,67,18,68,78\}$
$= \{l_{11}, l_{16}, l_{21}, l_{22}, l_{27}, l_{28}\}$,
$\{2,3,4,5\} \to \{23,24,34,25,35,45\}$
$= \{l_3, l_5, l_6, l_8, l_9, l_{10}\}$,
$\{2,3,4,6\} \to \{23,24,34,26,36,46\}$
$= \{l_3, l_5, l_6, l_{12}, l_{13}, l_{14}\}$,
$\{2,3,4,7\} \to \{23,24,34,27,37,47\}$
$= \{l_3, l_5, l_6, l_{17}, l_{18}, l_{19}\}$,
$\{2,3,4,8\} \to \{23,24,34,28,38,48\}$
$= \{l_3, l_5, l_6, l_{23}, l_{24}, l_{25}\}$,
$\{2,3,5,6\} \to \{23,25,35,26,36,56\}$
$= \{l_3, l_8, l_9, l_{12}, l_{13}, l_{15}\}$,
$\{2,3,5,7\} \to \{23,25,35,27,37,57\}$
$= \{l_3, l_8, l_9, l_{17}, l_{18}, l_{20}\}$,
$\{2,3,5,8\} \to \{23,25,35,28,38,58\}$
$= \{l_3, l_8, l_9, l_{23}, l_{24}, l_{26}\}$,
$\{2,3,6,7\} \to \{23,26,36,27,37,67\}$
$= \{l_3, l_{12}, l_{13}, l_{17}, l_{18}, l_{21}\}$,
$\{2,3,6,8\} \to \{23,26,36,28,38,68\}$
$= \{l_3, l_{12}, l_{13}, l_{23}, l_{24}, l_{27}\}$,
$\{2,3,7,8\} \to \{23,27,37,28,38,78\}$
$= \{l_3, l_{17}, l_{18}, l_{23}, l_{24}, l_{28}\}$,
$\{2,4,5,6\} \to \{24,25,45,26,46,56\}$
$= \{l_5, l_8, l_{10}, l_{12}, l_{14}, l_{15}\}$,
$\{2,4,5,7\} \to \{24,25,45,27,47,57\}$
$= \{l_5, l_8, l_{10}, l_{17}, l_{19}, l_{20}\}$,
$\{2,4,5,8\} \to \{24,25,45,28,48,58\}$
$= \{l_5, l_8, l_{10}, l_{23}, l_{25}, l_{26}\}$,
$\{2,4,6,7\} \to \{24,26,46,27,47,67\}$
$= \{l_5, l_{12}, l_{14}, l_{17}, l_{19}, l_{21}\}$,
$\{2,4,6,8\} \to \{24,26,46,28,48,68\}$
$= \{l_5, l_{12}, l_{14}, l_{23}, l_{25}, l_{27}\}$,
$\{2,4,7,8\} \to \{24,27,47,28,48,78\}$
$= \{l_5, l_{17}, l_{19}, l_{23}, l_{25}, l_{28}\}$,
$\{2,5,6,7\} \to \{25,26,56,27,57,67\}$
$= \{l_8, l_{12}, l_{15}, l_{17}, l_{20}, l_{21}\}$,
$\{2,5,6,8\} \to \{25,26,56,28,58,68\}$
$= \{l_8, l_{12}, l_{15}, l_{23}, l_{26}, l_{27}\}$,
$\{2,5,7,8\} \to \{25,27,57,28,58,78\}$
$= \{l_8, l_{12}, l_{20}, l_{23}, l_{26}, l_{28}\}$,
$\{2,6,7,8\} \to \{26,27,67,28,68,78\}$
$= \{l_{12}, l_{17}, l_{21}, l_{23}, l_{27}, l_{28}\}$,

$\{3,4,5,6\}\to\{34,35,45,36,46,56\}$
$=\{l_6,l_9,l_{10},l_{13},l_{14},l_{15}\}$,
$\{3,4,5,7\}\to\{34,35,45,37,47,57\}$
$=\{l_6,l_9,l_{10},l_{18},l_{19},l_{20}\}$,
$\{3,4,5,8\}\to\{34,35,45,38,48,58\}$
$=\{l_6,l_9,l_{10},l_{24},l_{25},l_{26}\}$,
$\{3,4,6,7\}\to\{34,36,46,37,47,67\}$
$=\{l_6,l_{13},l_{14},l_{18},l_{19},l_{21}\}$,
$\{3,4,6,8\}\to\{34,36,46,38,48,68\}$
$=\{l_6,l_{13},l_{14},l_{24},l_{25},l_{27}\}$,
$\{3,4,7,8\}\to\{34,37,47,38,48,78\}$
$=\{l_6,l_{18},l_{19},l_{24},l_{25},l_{28}\}$,
$\{3,5,6,7\}\to\{35,36,56,37,57,67\}$
$=\{l_9,l_{13},l_{15},l_{18},l_{20},l_{21}\}$,
$\{3,5,6,8\}\to\{35,36,56,38,58,68\}$
$=\{l_9,l_{13},l_{15},l_{24},l_{26},l_{27}\}$,
$\{3,5,7,8\}\to\{35,37,57,38,58,78\}$
$=\{l_9,l_{18},l_{20},l_{24},l_{26},l_{28}\}$,
$\{3,6,7,8\}\to\{36,37,67,38,68,78\}$
$=\{l_{13},l_{18},l_{21},l_{24},l_{27},l_{28}\}$,
$\{4,5,6,7\}\to\{45,46,56,47,57,67\}$
$=\{l_{10},l_{14},l_{15},l_{19},l_{20},l_{21}\}$,
$\{4,5,6,8\}\to\{45,46,56,48,58,68\}$
$=\{l_{10},l_{14},l_{15},l_{25},l_{26},l_{27}\}$,
$\{4,5,7,8\}\to\{45,47,57,48,58,78\}$
$=\{l_{10},l_{19},l_{20},l_{25},l_{26},l_{28}\}$,
$\{4,6,7,8\}\to\{46,47,67,48,68,78\}$
$=\{l_{14},l_{19},l_{21},l_{25},l_{27},l_{28}\}$,
$\{5,6,7,8\}\to\{56,57,67,58,68,78\}$
$=\{l_{15},l_{20},l_{21},l_{26},l_{27},l_{28}\}$.

通过以上 3 个步骤的操作最后得出的就是我们所要找的图.

5　结　论

以上所述的位序列计算模型其核心就是在加位过程中随时删除非解，这样就大大地减缓了子图数目增长的速度，从而为较大阶数的 Ramsey 数求解找到了一个可行方法. 另外，我们利用存储量更大、运算速度更快的 DNA 计算机求解使得解决部分 Ramsey 数变得有了希望，关于运用 DNA 计算的相关研究，我们将在本文的(Ⅱ)中给出.

致　谢　陈梅同学对本文给予了详细地校对，并发现其中的一个错误，给予更正. 在此表示感谢！

参　考　文　献

[1] Radziszowski S P. Small ramsey numbers. Electronic Journal of Combinatorics Dynamical Survey, 2006, 11: 3-36

[2] Mckay B D, Radziszowski S P. $R(4,5)=25$. Journal of Graph Theory, 1995, 19: 309-322

[3] Mckay B D, Zhang K M. The Value of the Ramsey Number $R(3,8)$. Journal of Graph Theory, 1992, 16: 99-105

[4] Adleman L M. Molecular computation of solution to combinatorial problems. Science, 1994, 266: 1021-1024

[5] Liu W B, Xu J. The hamiltonian cycle problem based on DNA computing//Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning. 2002, 1: 313-317

[6] Morimoto N, Arita M, Suyama A. Solid-phase DNA solution to the hamiltonian path problem. DIMACS, Series in Discrete Mathematics and Theoretical Computer Science, 1999, 48: 193-206

[7] Ouyang Q et al. DNA solution of the maximal clique problem. Science, 1997, 278: 446-449

[8] Pan L Q, Liu Y C et al. A surface-based DNA algorithm for the maximal clique problem. Chinese Journal of Electronics, 2002, 11(4): 469-471

[9] Xu Jin, Qiang Xiao-Li et al. An unenumerative DNA computing model for vertex coloring problem. IEEE Transactions on Computer, 2009, to appear

[10] Liu W B, Xu J. A DNA algorithm for the graph coloring problem. Journal of Chemical Information and Computers, 2002, 42 (5): 1176-1178

[11] Liu Y C, Pan L Q, Xu J et al. DNA solution of graph coloring problem. Journal of Chemical Information and Computer Sciences, 2002, 42(3): 529-534

[12] Yin Z X, Zhang F Y, Xu J. A Chinese postman problem based on DNA computing. Journal of Chemical Information and Computer Sciences, 2002, 42(2): 224-224

[13] Yin Z X, Zhang F Y, Xu J. The general form of 0-1 programming problem based on DNA computing. Biosystems, 2003, 70(1): 73-79

[14] Yin Zhi-Xiang, Zhang Feng-Yue, Xu Jin. 0-1 planning problem based on DNA computing. Journal of Electronics & Information Technology, 2003, 25(1): 62-66(in Chinese)
(殷志祥，张凤月，许进. 0-1 规划问题的 DNA 计算. 电子与信息学报，2003，25(1)：62-66)

[15] Harari F. Graph Theory. Reading, Mass: Addison-Wesley, 1969
(哈拉里著，李慰萱译. 图论. 上海：上海科技出版社，1980)

[16] Bondy J A, Murty U S R. Graph Theory with Applications. London: The Macmillan Press Ltd, 1976

XU Jin, born in 1959, professor, Ph. D. supervisor. His research interests include DNA computing and DNA computer, neural networks, genetic algorithms, graph theory etc.

FAN Yue-Ke, born in 1959, Ph. D.. His research interests include DNA computing and DNA computer, graph theory etc.