

有向图 k 顶点导出子图的 DNA 粘贴算法

朱维军 张春艳 周清雷 陈永华

(郑州大学信息工程学院 郑州 450001)

摘 要 在经典的电子计算中,有向图 k 顶点导出子图是一个高度复杂的问题。DNA 计算是近年来发展的以 DNA 为载体求解计算问题的非经典计算技术。文中研究了使用 DNA 计算解决有向图 k 顶点导出子图的问题,从而提出了一种在粘贴机上运行的子图生成算法。首先,以粘贴机的标准生化元操作作为算法调用的基本算子;其次,使用顺序与循环等程序结构,把上述基本算子按照一定的逻辑方式组织起来;最后,读取生化反应结果,即可获得给定有向图的所有 k 顶点导出子图。仿真实验结果表明,与经典算法相比,新算法在理想条件下大幅缩短了子图生成时间。

关键词 粘贴机,脱氧核糖核酸,有向图,顶点导出子图

中图分类号 TP384 文献标识码 A DOI 10.11896/j.issn.1002-137X.2019.01.048

DNA Sticker Algorithm for k -vertex Induced Sub-graphs of Directed Graphs

ZHU Wei-jun ZHANG Chun-yan ZHOU Qing-lei CHEN Yong-hua

(School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China)

Abstract How to obtain all the k -vertex induced sub-graphs in a directed graph is a complex computational problem. The DNA computing is a nonclassical computing technique developed in recent years, which can be employed to solve some complex computational problems via DNAs and their biochemical reactions. Aiming at obtaining all the k -vertex induced sub-graphs in a directed graph using the DNA computing, a DNA sticker algorithm for constructing sub-graphs was proposed. First, the existing sticker system provides the basic operators for the new algorithm. Each of these basic operators has its basic computational function, and each of these basic operators can be implemented by a specific standard biochemical reaction. Second, these basic operations can be organized in a certain logical way. As a result, some program structures, such as sequence and loop, are formed. At last, all the k -vertex induced sub-graphs can be obtained for a given directed graph, by reading the results of the biochemical reactions. These simulated results demonstrate that the new algorithm significantly reduces the time which is required by generating sub-graphs under ideal conditions, compared with the classical algorithms.

Keywords Sticker system, DNA, Directed graph, Vertex induced sub-graph

1 引言

著名的理论物理学家 Kaku 曾在 2012 年预言,10 年内电子计算机的工艺制造技术将达到极限^[1]。而 DNA 计算被认为有望成为解决上述问题的一条有效途径。DNA 计算最早由图灵奖得主 Adleman 教授开创,源自 1994 年 Adleman 教授在 *Science* 上发表的一篇关于使用脱氧核糖核酸(Deoxy-ribonucleic Acid, DNA)分子解决一个小规模哈密顿路径问题^[2]的文章。由于 DNA 计算方法相对于经典计算拥有巨大的并行优势,因此使用 DNA 分子来求解计算机科学问题,特别是以 NP 为背景的复杂计算问题,迅速成为一个国际热点研究领域。除了哈密顿路径之外,图着色^[3-5]、图的顶点覆

盖^[6-10]、图的团^[11-15]、图的匹配^[16-17]、旅行商^[18-19]等 NP 问题也相继在 DNA 计算的框架内得到了解决。此外,通用 DNA 计算机的研究也取得了关键突破^[20]。

图作为一种重要的数学模型和数据结构,广泛应用于众多领域。图论问题集为图的应用提供了理论基础。迄今为止,作为一类重要的复杂计算问题,若干常用的图论方法已经可以采用 DNA 分子来实现。然而,这只是针对无向图而言,对于一系列有向图的问题来说,目前尚缺乏有效的 DNA 计算手段解决。特别是有向图 k 顶点导出子图的问题,目前尚缺乏可求解的 DNA 计算方法。这是本工作拟研究的课题。

本文第 2 节介绍粘贴机的相关知识;第 3 节介绍有向图 k 顶点导出子图;第 4 节给出解决该问题的 DNA 算法;第 5

收稿日期:2017-12-28 返修日期:2018-03-24 本文受国家自然科学基金项目(U1204608,61572444)资助。

朱维军(1976-),男,博士,副教授,主要研究方向为 DNA 计算、形式化方法、信息安全, E-mail: zhuweijun76@163.com;张春艳(1990-),女,硕士生,主要研究方向为 DNA 计算;周清雷(1962-),男,教授,主要研究方向为信息安全、自动机理论、DNA 计算等;陈永华(1962-),男,博士,副教授,主要研究方向为 DNA 计算, E-mail: ieyhchen@zzu.edu.cn(通信作者)。

节通过仿真实验评估算法效率;最后总结全文。

2 粘贴机模型

粘贴机是一种非自治的 DNA 计算模型,由 Roweis 等^[21]提出。大致来说,粘贴机模型提供了一些元操作,DNA 计算方法的研究人员可调用这些标准化的元操作,将这些元操作按照顺序、条件、循环等逻辑关系组织起来,就可获得求解某项具体 DNA 计算问题的 DNA 算法。即粘贴机模型提供了一个通用平台,把底层的基于 DNA 分子的生物学实现与高层的计算问题及其求解方案隔离开来,使得 DNA 算法的研究人员无需考虑底层硬件(生物学)实现。

粘贴机的详细结构可以参考文献^[21]。粘贴存储器由存储链和粘贴链两部分构成,其中存储链含有 n 个互不相同且长链都为 m 的子链;每个粘贴链长为 m ,且必须恰与存储链中的一个子链互补。据此,我们可以巧妙地将存储复合物用二进制表示。存储链的每个子链可看成 1 个位(bit),若该子链与其互补粘贴子链发生退火反应,则该子链表示 1,否则表示 0。例如:



则该存储复合物即可表示为 0101。

如上所述,粘贴机上运行的所有 DNA 算法可由 5 种元操作构成,根据不同的需求,设计不同的逻辑结构,从而编码出不同的 DNA 算法,解决相应的问题。这 5 种基本元操作分别是:合并、分离、设置、清除和清空^[22]。

合并(merge):取给定试管的并集,并将最终产物倒入同一试管。即 $\text{merge}(T_1, \dots, T_n)$ 操作最终得到试管 T_n ,其中包含试管 T_1 到 T_n 中的所有存储链。

分离(separate):对于试管 T 和整数 i ,该操作产生两个新的试管 T^+ 和 T^- ,其中 T^+ 包含 T 中第 i 个子链为 1 的所有存储复合物, T^- 包含 T 中第 i 个子链为 0 的所有存储复合物^[23],其中 $1 \leq i \leq n$ 。

设置(set):将试管 T 中所有存储链某一位的值全部置为 1。即 $\text{set}(T, i)$ 操作后,试管 T 中每个存储复合物的第 i 个子链值都为 1,其中 $1 \leq i \leq n$ 。

清除(clear):将试管 T 中所有存储链某一位的值全部置为 0。即 $\text{clear}(T, i)$ 操作后,试管 T 中每个存储复合物的第 i 个子链值都为 0,其中 $1 \leq i \leq n$ 。而此操作一般采用加热的方法将特定位的值由 1 变为 0。

清空(discard):该操作将清空试管 T 中所有的存储复合物。

对于 DNA 粘贴计算模型,其输入是一个包含可行解编码链的初始试管,即一个含有许多 DNA 链的链库,输出是生化反应产生生物,这些分子产生生物包含的编码信息就是所需要的计算结果。此处,链库的定义如下。

定义 1^[24] 设 $n \geq k, m \geq 0$ 。设 $\left[n+m, \binom{n}{k} \right]$ 链库是一个

包含 $n+m$ 位数的 DNA 分子多重集试管,其中 $n+m$ 位数的前 n 个字符是 n 的所有 k 组合,后 m 个字符全为 0。

3 有向图 k 顶点导出子图问题

有向图和 k 顶点导出子图^[22-23]的定义如下。

定义 2 有向图 $G=(V, E)$ 是一个二元组,其中顶点集 $V=(v_1, \dots, v_n)$,边集 $E=(e_1, \dots, e_m)$ 。对于每一个 $e_i=(v_{i_1}, v_{i_2}) \in E$,有 $v_{i_1} \in V \wedge v_{i_2} \in V$,称 v_{i_1} 为边 e_i 的起点, v_{i_2} 为边 e_i 的终点,其中 n 为顶点数, m 为边数。

定义 3 在有向图 $G=(V, E)$ 中,图 $G^*=(V^*, E^*)$,若 $V^* \subseteq V, E^* \subseteq E \cap (V^*/2)$,其中 $(V^*/2)$ 是 V^* 的二元子集组成的边集合,当 $|V^*|=k$ 时,称 G^* 为 G 的 k 顶点导出子图。

图 1 给出了一个有向图的例子。由图 1 可见,每条边都具有方向性,并有特定的起点和终点。设 k 的值为 3,根据定义 3 可知图 1 中所有的 3-顶点导出子图,如图 2 所示。

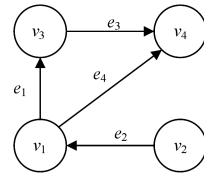


图 1 有向图 G_1

Fig. 1 Directed graph G_1

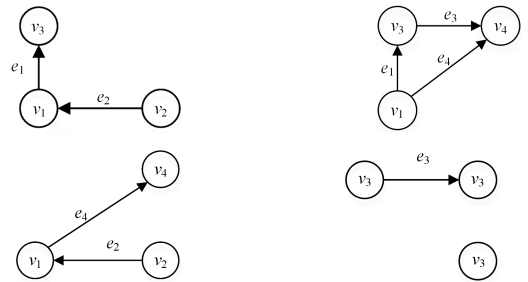


图 2 有向图 G_1 的 3-顶点导出子图

Fig. 2 3-vertex induced sub-graphs of directed graph G_1

4 算法

4.1 算法描述

根据粘贴机模型,本文设计了可求解有向图 k 顶点导出子图的 DNA 粘贴算法,如算法 1 所示。

算法 1 有向图 k 顶点导出子图的 DNA 粘贴算法

Input: $T=(n+nm+m, C_n^k)$

Output: 所有 k 顶点导出子图的二进制表达形式

Begin

1. for $i \leftarrow 1$ to m do

2. separate(T, T^+, T^-, i_1); /* i_1 是第 i 条边的起点 */

3. separate(T^+, T^{++}, T^{+-}, i_2); /* i_2 是第 i 条边的终点 */

4. set($T^{++}, n+(i-1) \times m+i_1$); /* 第 i 条边对应的第 1 个顶点位置设为 1 */

5. set($T^{++}, n+(i-1) \times m+i_2$); /* 第 i 条边对应的第 2 个顶点位置设为 1 */

6. merge(T^-, T^{+-}, T^{++}, T)

7. end for

8. return T

End

下面通过例 1 来解释算法的原理与运行过程。

例 1 求图 1 的所有 4 个 3-顶点导出子图。

设计粘贴机数据结构 $T=(n+nm+m, C_n^k)=(4+4\times 4+4, C_4^3)$ (见表 1)。从 4 个顶点中选出 3 个,有 $C_4^3=4$ 种选法。表 1 中的每一行分别表示一种选法(也就是一个子图),以及其对应的 DNA 分子编码。每一行的前 4 位(分子编码的前 4 位)表示对应选出的顶点子集,分别是 $v_1, v_2, v_3; v_1, v_2, v_4; v_1, v_3, v_4; v_2, v_3, v_4$ 。所有分子编码的后 4 位表示对应 4 条边的方向。如果一条边 e_i 的方向是从小数字序号顶点指向大数字序号顶点,那么对应的 $p(e_i)$ 被设置为 1;如果一条边 e_i 的方向是从大数字序号顶点指向小数字序号顶点,那么对应的 $p(e_i)$ 被设置为 0。换句话说,每行前 4 位与后 4 位的初始化取值在整个算法运行过程中保持不变。每行中间的 4×4 位编码在算法运行前被初始化为 0,在算法运行过程中,这部分编码的取值动态改变,算法结束时它们的值就表示从子图选出的边,以及这些边的两个顶点。

表 1 图 2 中各子图中的顶点集/边集与二进制的对应关系
(算法运行之前的试管状态)

Table 1 Relationship between vertex set/edge set and binary strings in each sub-graph of Fig. 2(state of test tube before algorithm runs)

v_1, v_2, v_3, v_4	$e_1(v_1, v_2, v_3, v_4)$	$e_2(v_1, v_2, v_3, v_4)$	$e_3(v_1, v_2, v_3, v_4)$	$e_4(v_1, v_2, v_3, v_4)$	$p(e_1), p(e_2), p(e_3), p(e_4)$
1110	0000	0000	0000	0000	1011
1101	0000	0000	0000	0000	1011
1011	0000	0000	0000	0000	1011
0111	0000	0000	0000	0000	1011

算法的每次循环对一条边进行处理。以第 1 条边为例,处理过程如下:首先,算法执行两个 separate 语句,把第 1 条边对应的两个顶点(即 v_1, v_3)均出现的两个子图,也就是第 1 个和第 3 个子图(对应于表 1 中的第 1 行和第 3 行)的 DNA 分子置入试管 T^{++} ;然后,算法执行两个 set 语句,把 T^{++} 中的所有分子(上述两个子图)的 $e_1(v_1, v_2, v_3, v_4)$ 部分做如下修改,第 1 条边所对应的两个顶点(v_1, v_3)在第 1 行和第 3 行的相应位置被置为 1,即 $e_1(v_1, v_2, v_3, v_4)$ 的第 1 行被设置为 1010,第 3 行被设置为 1010。因为其他两行不在 T^{++} 中,所以 set 语句不对它们进行修改。因此 $e_1(v_1, v_2, v_3, v_4)$ 的第 2 行和第 4 行的取值不发生改变,仍为 0000。这样,算法的第 1 次循环结束,对第 1 条边处理完毕, $e_1(v_1, v_2, v_3, v_4)$ 的所有 4 行取值被最终确定,并不再被算法的后续执行过程修改。

以此类推,算法的第 i 次循环处理第 i 条边,并最终确定 $e_i(v_1, v_2, v_3, v_4)$ 的所有 4 行取值。这样,当算法结束时,所有的 $e_i(v_1, v_2, v_3, v_4)$ 的 4 行取值均被最终确定,如表 2 所列,这就是整个 DNA 计算得到的结果。

表 2 图 2 各子图中的顶点集/边集与二进制的对应关系
(算法运行之后的试管状态)

Table 2 Relationship between vertex set/edge set and binary strings in each sub-graph of Fig. 2(state of test tube after algorithm runs)

v_1, v_2, v_3, v_4	$e_1(v_1, v_2, v_3, v_4)$	$e_2(v_1, v_2, v_3, v_4)$	$e_3(v_1, v_2, v_3, v_4)$	$e_4(v_1, v_2, v_3, v_4)$	$p(e_1), p(e_2), p(e_3), p(e_4)$
1110	1010	1100	0000	0000	1011
1101	0000	1100	0000	1001	1011
1011	1010	0000	0011	1001	1011
0111	0000	0000	0011	0000	1011

根据表 2 结果即可画出 4 个子图,以第 1 个子图为例。参见表 2 第 1 行,前 4 位是 1110,因此 v_1, v_2, v_3 这 3 个顶点被选中; $e_1(v_1, v_2, v_3, v_4)$ 是 1010,因此在 v_1, v_3 之间画一条边,标记为 e_1 ; $e_2(v_1, v_2, v_3, v_4)$ 是 1100,因此在 v_1, v_2 之间画一条边,标记为 e_2 ; $e_3(v_1, v_2, v_3, v_4)$ 是 0000,因此不在该子图中增加 e_3 ;同理,不增加 e_4 。可见,该子图有两条边,分别是 e_1 和 e_2 。接下来为这两条边确定方向。参见表 2 第 1 行的最后 4 位,由于 $p(e_1)=1$,因此 e_1 的方向是由小序号顶点到大序号顶点,即从 v_1 到 v_3 ;因为 $p(e_2)=0$,所以 e_2 的方向是由大序号顶点到小序号顶点,即从 v_2 到 v_1 。这样,就画出了第 1 个子图。

重复上述过程即可画出所有子图,如图 2 的 4 个子图所示。表 2 的每一行对应一个子图的所有信息,由于这些信息不仅指出了子图选中的边,而且给出了这些边分别对应的顶点,甚至边的方向。因此,表 2 信息完备,不需要表格之外的任何其他信息,特别是不需要对照原图,即可画出所有子图。

例 1 结束。

4.2 复杂度分析

对新算法进行复杂度分析,算法中的所有生化元操作和电子元操作的执行时间均被统一定义为标准 CPU 时间,即 $O(1)$,可证明如下命题成立。

命题 1 新算法的最坏时间复杂度为 $O(m)$ 。

证明:算法的每次循环执行了 2 次 separate 生化元操作、2 次 set 生化元操作、1 次 merge 生化元操作,以及 4×2 次电子元操作(加减乘除),因此 1 次循环所需时间为 $O(13)$ 。由于循环被执行了 m 次,因此需要的时间复杂度为 $O(13\times m)=O(m)$ 。

命题得证。

5 仿真实验

5.1 实验目的、实验平台与实验方法

我们从生物可实现和时间效率两个角度考量新算法。

粘贴机提供的元操作集是鲁棒的,使用该集合写出的算法程序可确保其完整实现计算功能^[22],免于不确定的生物因素影响。因此,新算法的生物有效性可由粘贴机模型的生物有效性^[22,25]来保证。

为了对基于 DNA 粘贴机的有向图 k 顶点导出子图算法与基于电子计算的有向图 k 顶点导出子图算法的时间效率进行比较,使用如下平台进行仿真实验:CPU 为 Intel(R) Core (TM) i7,内存 8 GB, windows 7, MATLAB R2014a。为了使两种方法具有可比性,两个算法中所有的生化元操作与电子元操作均被定义为标准 CPU 时间。使用 MATLAB 仿真算法的主要功能即可评估算法运行所需时间。

5.2 实验结果与分析

仿真实验结果如图 3 所示。图 3 中斜面表示新算法的运行时间,曲面则表示经典算法的运行时间。随着图 3 中节点数 n 的增加,经典算法所需时间急剧增加,而新算法所需时间仅仅随着图中边数 m 的增加而线性增加。造成这一现象的原因在于:DNA 算法使用了大量的分子参与并行计算,而经典算法的电子硬件结构限制了算法的并行性。由此可见,

DNA 计算用于求解有向图 k 顶点导出子图问题具有显著的优势。

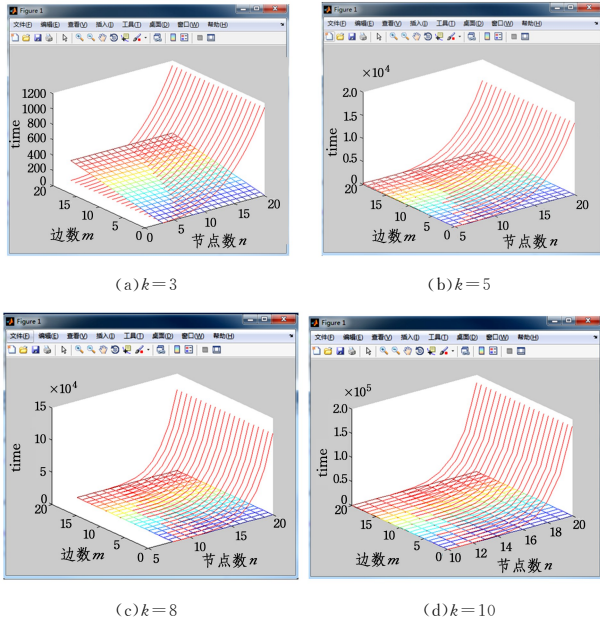


图3 新算法与经典算法运行时间的比较

Fig. 3 Comparison of running time between new algorithm and classical algorithm

6 相关工作比较

作为古老的计算问题,有向图解决方案目前仍在继续发展,一些研究工作对有向图论展开了研究。例如,在给定时空限制下计算强连通成份^[26]、寻找稠子图^[27]等,而本文工作使用了DNA计算技术来求解有向图问题。

除了研究将粘贴机模型用于求解计算问题之外,近年来也有一些工作致力于研究模型本身,主要表现在扩展模型和自身理论的发展。关于前者,双面粘贴系统^[28]、带复杂结构的变体粘贴模型^[29]、概率粘贴系统^[30]、权重粘贴系统^[31]被相继提出,以满足解决不同计算问题的需要。关于后者,展开了粘贴机的表达能力^[32]以及与WC自动机关系^[33]的研究,以便探索它的可用范围。与这些工作不同,本文工作借助了粘贴机模型来求解具体的图论计算问题。

文献^[24]分别给出了求解 k 边导出子图问题和 k 顶点导出子图问题的DNA算法,这两种算法均可处理无向图,但不能处理有向图。本文提出的新算法可同时对无向图与有向图的 k 顶点导出子图问题进行求解。文献^[24]中两种算法运行获得的生化反应结果不包含顶点与边的位置关系,因此,读取生化反应结果之后,还需要人工对比原图才能画出相应的无向子图。而本文针对处理 k 顶点导出子图,给出的新算法运行获得的生化反应结果,不仅包含顶点与边的所有位置关系信息,而且包含边的方向信息,因而无需人工参考原图就能直接由生化反应结果画出有向子图。

文献^[23]给出求解 k 边导出子图问题的一种DNA算法。该算法与本文提出的新算法的不同点在于:前者处理 k 边导出子图,而后者处理 k 顶点导出子图。二者的相同点在于均可处理有向图。相比而言,新算法具有如下优势:由文献

^[23]中所提算法运行后得到的生化反应结果可直接得出所需子图中的顶点与边的位置关系;而新算法运行后的生化反应结果不仅可以直接得出上述位置关系,而且直接蕴含了边的方向,即使用新算法获得子图的过程中人工参与度更低。

结束语 本文提出了一种使用DNA分子求解有向图 k 顶点导出子图的方法。从已有文献来看,未见可求解该问题的DNA算法的报道。新方法在DNA计算的框架内初步解决了有向图 k 顶点导出子图问题,这是本文工作的主要成果与贡献。

作为一种DNA计算方法,新算法在实际使用中需要非常谨慎地处理DNA分子。粘贴机可以被认为是一个机器人工作站,它由一些机器人(手臂、泵装置、加热器、冷却器)、控制机器人的微处理器和若干试管组成,其中,微处理器控制机器人和试管,使得粘贴算法的操作按顺序执行^[24]。在粘贴机上实现合并元操作时,如果DNA处理不慎,可能会被剪切力切成碎片,而且,DNA分子可能会残留在试管壁上,最终导致它们丢失^[24]。然而,这个瑕疵并非新算法特有,而是粘贴机算法的共性不足。此外,文献^[24]指出,粘贴机算法的最大共性问题在于清除元操作——粘贴链一旦失活,解链将无法进行。然而即便如此,该生化元操作仍可在不影响计算完备性的前提下实现清除,何况新算法并未使用清除这一粘贴机中问题最大^[24]的生化元操作,由此可见,新算法固然与其他DNA算法一样对生化实验操作水平提出要求,但这并不影响其在生物学上的可实现性。由粘贴机模型带来的粘贴算法的不足,是我们下一步需要研究的问题。

参考文献

- [1] XU J. The age of the biological computer is coming [J]. Bulletin of Chinese Academy of Sciences, 2014, 29(1): 42-54. (in Chinese)
许进. 生物计算机时代即将来临 [J]. 中国科学院院刊, 2014, 29(1): 42-54.
- [2] ADLEMAN L M. Molecular computation of solutions to combinatorial problems [J]. Nature, 1994, 369: 40.
- [3] LIU W, ZHANG F, XU J. A DNA algorithm for the graph coloring problem [J]. Journal of Chemical Information and Computer Sciences, 2002, 42(5): 1176-1178.
- [4] WANG S D, LIU W B, XU J. DNA sticker algorithm for vertex-coloring problems of graph [J]. Journal of Systems Engineering and Electronics, 2005, 27(3): 568-572. (in Chinese)
王淑栋, 刘文斌, 许进. 图顶点着色问题的DNA粘贴算法 [J]. 系统工程与电子技术, 2005, 27(3): 568-572.
- [5] BACH E, GLASER E, CONDON A, et al. DNA models and algorithms for NP-complete problems [C] // Proceedings of Eleventh Annual IEEE Conference on Computational Complexity. IEEE, 1996: 290-300.
- [6] XU J, LI S P, DONG F Y, et al. The Sticker models of DNA computer (II): Application [J]. Chinese Science Bulletin, 2004, 49(4): 299-307. (in Chinese)
许进, 李三平, 董亚非, 等. 粘贴DNA计算机模型(II): 应用 [J]. 科学通报, 2004, 49(4): 299-307.
- [7] CHANG W L, GUO M. Solving the clique problem and the ver-

- tex cover problem in Adleman-Lipton's model[C]// IASTED International Conference on Networks, Parallel and Distributed Processing, and Applications, Japan, 2002; 431-436.
- [8] PAN L Q, XU J, LIU Y C. A surface-based DNA algorithm for the minimal vertex cover problem[J]. *Progress in Natural Science: Materials International*, 2003, 13(1): 78-80.
- [9] ZHOU K, XU J. Closed Circle DNA Algorithm of the Minimal Covering Problem[J]. *Computer Engineering and Applications*, 2006, 42(20): 7-9. (in Chinese)
周康, 许进. 最小顶点覆盖问题的闭环 DNA 算法[J]. *计算机工程与应用*, 2006, 42(20): 7-9.
- [10] OUALI M E, FOHLIN H, SRIVASTAV A. An approximation algorithm for the partial vertex cover problem in hypergraphs [J]. *Journal of Combinatorial Optimization*, 2012, 7659 (2): 1-19.
- [11] NING J, LI Y, YU W. Molecular Sticker Model Stimulation on Silicon for a Maximum Clique Problem[J]. *International Journal of Molecular Sciences*, 2015, 16(6): 13474-13489.
- [12] AL JUNID S A M, TAHIR N M, MAJID Z A, et al. Potential of graph theory algorithm approach for DNA sequence alignment and comparison[C]// 2012 Third International Conference on Intelligent Systems, Modelling and Simulation (ISMS). IEEE, 2012; 187-190.
- [13] WANG Z, TAN J, ZHU L, et al. A Biological Algorithm to Solve the Maximum Complete Subgraph Problem Based on Adleman-Lipton Model[J]. *Journal of Computational & Theoretical Nanoscience*, 2014, 11(11): 2310-2312.
- [14] XU J, TAN G J, FAN Y K, et al. DNA Computer Principle, Advances and Difficulties(IV): On the Models of DNA Computer [J]. *Chinese Journal of Computers*, 2007, 30(6): 881-893. (in Chinese)
许进, 谭钢军, 范月科, 等. DNA 计算机原理、进展及难点(IV): 论 DNA 计算机模型[J]. *计算机学报*, 2007, 30(6): 881-893.
- [15] SHEN L J, SONG Z C, WU L Q, et al. A Molecular Computing Model for Maximum Clique Problem Based on DNA Nanoparticles[J]. *Journal of Computational & Theoretical Nanoscience*, 2014, 11(10): 2120-2124(5).
- [16] WU X, SONG C Y, ZHANG N, et al. DNA Algorithm for Maximum Matching Problem Based on Sticker Computation Model [J]. *Computer Science*, 2013, 40(12): 127-132. (in Chinese)
吴雪, 宋晨阳, 张楠, 等. 最大匹配问题的粘贴 DNA 算法[J]. *计算机科学*, 2013, 40(12): 127-132.
- [17] ZHOU X, LI K L, YUE G X, et al. A Volume Molecular Solution for the Maximum Matching Problem on DNA-Based Computing[J]. *Journal of Computer Research and Development*, 2011, 48(11): 2147-2154. (in Chinese)
周旭, 李肯立, 乐光学, 等. 一种最大匹配问题 DNA 计算算法 [J]. *计算机研究与发展*, 2011, 48(11): 2147-2154.
- [18] HAN A, ZHU D. A new DNA encoding method for traveling salesman problem [C]// International Conference on Intelligent Computing. Springer, Berlin, Heidelberg, 2006; 328-335.
- [19] LEE J Y, SHIN S Y, PARK T H, et al. Solving traveling salesman problems with DNA molecules encoding numerical values [J]. *BioSystems*, 2004, 78(1): 39-47.
- [20] XU J. Probe machine [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2016, 27(7): 1405-1416.
- [21] ROWEIS S, WINFREE E, BURGOYNE R, et al. A sticker-based model for DNA computation[J]. *Journal of Computational Biology*, 1998, 5(4): 615-629.
- [22] MARTÍNEZ-PÉREZ I M, ZIMMERMANN K H. Parallel bioinspired algorithms for NP complete graph problems[J]. *Journal of Parallel and Distributed Computing*, 2009, 69(3): 221-229.
- [23] ZHU W J, XU Z H, ZHANG H B, et al. DNA algorithm for k -edge induces sub-graphs of directed graphs[J]. *Journal of Xi-dian University*, 2013, 40(5): 175-180. (in Chinese)
朱维军, 徐朝辉, 张海宾, 等. 有向图 k 边导出子图的 DNA 粘贴算法[J]. *西安电子科技大学学报*, 2013, 40(5): 175-180.
- [24] IGNATOVA Z, MARTÍNEZ-PÉREZ I, ZIMMERMANN K H. DNA computing models [M]. Berlin: Springer Science & Business Media, 2008.
- [25] KARI L, PAUN G, ROZENBERG G, et al. DNA computing, sticker systems and universality [J]. *Acta Informatica*, 1998, 35: 401-420.
- [26] GEORGIADIS L, ITALIANO G F, PAROTSIDIS N. Strong connectivity in directed graphs under failures, with applications [C]// Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms. 2017; 1880-1899.
- [27] KHULLER S, SAHA B. On finding dense subgraphs [C]// Lecture Notes in Computer Science; 36th International Colloquium on Automata, Languages and Programming. Berlin: Springer-Verlag, 2009; 597-608.
- [28] ALHAZOV A, CAVALIERE M. Computing by observing bio-systems; The case of sticker systems[C]// International Workshop on DNA-Based Computers. Springer, Berlin, Heidelberg, 2004; 1-13.
- [29] SAKAKIBARA Y, KOBAYASHI S. Sticker systems with complex structures[J]. *Soft Computing*, 2001, 5(2): 114-120.
- [30] SELVARAJOO M, HENG F W, SARMIN N H, et al. Probabilistic sticker systems[J]. *Malaysian Journal of Fundamental and Applied Sciences*, 2013, 9(3): 402-420.
- [31] GAN Y S, FONG W H, SARMIN N H, et al. Some Properties and Variants of Weighted Sticker Systems [J]. *International Journal of Applied Mathematics and Statistics™*, 2013, 45(15): 367-375.
- [32] SIANG G Y, HENG F W, SARMIN N H, et al. The generative power of weighted one-sided and regular sticker systems[C]// AIP Conference Proceedings. AIP, 2014, 1602: 855-862.
- [33] KUSKE D, WEIGEL P. The role of the complementarity relation in Watson-Crick automata and sticker systems[C]// Developments in Language Theory. Springer, Berlin, Heidelberg, 2004; 272-283.