

# 基于 DNA 计算的线性时序逻辑模型检测方法

朱维军<sup>1)</sup> 周清雷<sup>1)</sup> 张钦宪<sup>2)</sup>

<sup>1)</sup>(郑州大学信息工程学院 郑州 450001)

<sup>2)</sup>(郑州大学基础医学院 郑州 450001)

**摘 要** 该文研究基于脱氧核糖核酸(Deoxyribonucleic Acid, DNA)计算的线性时序逻辑(Linear Temporal Logic, LTL)模型检测问题. 为此, 该文给出了使用粘贴自动机实现 LTL 模型检测的方法. 首先, 使用 3'-5'型单链 DNA 分子对 LTL 公式的有穷状态自动机(Finite State Automata, FSA)模型进行编码, 从而获得实现公式的粘贴自动机; 其次, 使用 5'-3'型单链 DNA 分子对系统模型进行编码, 从而获得粘贴自动机的输入字符串; 最后, 对表征粘贴自动机的 DNA 单链分子和表征输入字符串的 DNA 单链分子实施一系列生化反应, 即可判定系统是否满足公式. 分子生物学仿真实验结果表明: 给出的 DNA 编码序列能达到 99.9% 的碱基配对正确率, 且新方法成功地对所有 4 种 LTL 基本公式与 5 种 LTL 常见公式实施了检测; 与之对照, 已有的方法只能有效检测 1 种 LTL 基本公式与 0 种 LTL 常见公式. 在此基础上, 对本实验给出的 DNA 编码方案直接作位数扩展即可拥有对任意给定 LTL 一般公式的(理论)检测能力.

**关键词** 模型检测; 脱氧核糖核酸; 线性时序逻辑; 粘贴自动机; 有穷状态自动机; DNA 计算

中图法分类号 TP301; TP384

DOI号 10.11897/SP.J.1016.2016.02578

## A LTL Model Checking Approach Based on DNA Computing

ZHU Wei-Jun<sup>1)</sup> ZHOU Qing-Lei<sup>1)</sup> ZHANG Qin-Xian<sup>2)</sup>

<sup>1)</sup>(School of Information Engineering, Zhengzhou University, Zhengzhou 450001)

<sup>2)</sup>(School of Basic Medical Sciences, Zhengzhou University, Zhengzhou 450001)

**Abstract** This paper investigates the Linear Temporal Logic (LTL) model checking problem under the circumstance of Deoxyribonucleic Acid (DNA) computing. To address this problem, we presented a method for checking LTL formulas via some sticker automata. Firstly, 3'-5' single-stranded DNA molecules are applied to encoding a Finite State Automaton (FSA) model for the given LTL formula, so that a sticker automata which is a DNA model of the formula can be obtained. Secondly, the given system model is encoded by 5'-3' single-stranded DNA molecules, by which the input strings of sticker automata can be generated. Finally, a series of biochemical reactions are implemented on the single-stranded DNA molecules characterizing sticker automata or input strings, in order to determine whether the system meets the formula or not. The experimental results show that the generated DNA coding sequences achieve in 99.9% by accuracy for base pairing. Furthermore, the proposed method successfully checks four kinds of basic LTL formulas and five kinds of popular LTL formulas, whereas the existing methods can only check one kind of basic LTL formula and none popular LTL formula, as shown in the experiments. By extending more bits on presented DNA encoding scheme, the proposed method can check any given LTL formulas theoretically.

**Keywords** model checking; deoxyribonucleic acid; linear temporal logic; sticker automata; finite state automata; DNA computing

收稿日期:2015-04-16;在线出版日期:2015-11-10. 本课题得到国家自然科学基金(61250007, U1204608, U1304606, 61373043, 61572444)、中国博士后科学基金(2012M511588, 2015M572120)和河南省高等学校青年骨干教师资助计划(2014GGJS-001)资助. 朱维军, 男, 1976年生, 博士, 副教授, 中国计算机学会(CCF)高级会员, 主要研究方向为 DNA 计算、形式化方法. E-mail: zhuweijun76@163.com. 周清雷, 男, 1962年生, 博士, 教授, 博士生导师, 中国计算机学会(CCF)理事, 主要研究领域为 DNA 计算、形式化方法、信息安全. 张钦宪, 男, 1953年生, 教授, 博士生导师, 主要研究领域为 DNA 计算、分子生物学.

## 1 引言

模型检测由图灵奖得主 Clarke 教授等人<sup>[1]</sup>提出,是一种能够自动验证系统是否满足给定性质的共性技术. 目前已被广泛地应用于处理器设计验证<sup>[2]</sup>、网络协议验证、安全协议验证<sup>[3]</sup>、软件规范验证<sup>[4]</sup>等领域,并被美国国家航空航天局<sup>①</sup>以及 Intel、IBM、摩托罗拉等信息技术领军企业使用<sup>②</sup>.

模型检测的一般原理是:在模型检测系统中,有穷状态自动机建立系统模型,通过时序逻辑公式描述系统需满足的性质. 若有穷状态机满足相应的逻辑公式,则其满足性质;否则,不满足. 图灵奖得主 Pnueli 教授开创性地引入计算机科学的线性时序逻辑<sup>[5-6]</sup>与图灵奖得主 Clarke 教授提出的计算树逻辑<sup>[7]</sup>(Computation Tree Logic, CTL),是目前在计算机工业界使用的两大主流时序逻辑,分别被用来描述系统的线性时序性质和分支时序性质.

生物计算机以生物分子作为信息存储的载体,以生物酶和生物操作作为计算工具,是一种与传统的电子计算机不同的计算机模型. 1994 年,图灵奖得主 Adleman 教授<sup>[8]</sup>在《Science》上发表的一篇文章用 DNA 实验求解了一个小规模汉密尔顿路径问题,被公认是 DNA 计算领域的开创性工作. 由于 DNA 分子天然的巨大并行计算优势,一系列 DNA 计算模型和方法被提出用于解决复杂计算问题,例如:Lipton<sup>[9]</sup>在《Science》上发表文章给出命题公式可满足性问题的 DNA 解法;Ouyang 等人<sup>[10]</sup>在《Science》上发表文章给出最大团问题的 DNA 解法;Shapiro 等人<sup>[11]</sup>在《Nature》上发表文章给出两状态两字符自动机问题的 DNA 解法等.

此外,过滤模型<sup>[12]</sup>、粘贴系统<sup>[13]</sup>、长度一编码型自动机模型<sup>[14]</sup>、粘贴自动机模型<sup>[15]</sup>、DNA 图灵机模型<sup>[16]</sup>、非枚举型 DNA 计算模型<sup>[17]</sup>、巨磁电阻型 DNA 计算模型<sup>[18]</sup>、DNA 分子逻辑计算模型<sup>[19]</sup>、纳米分子逻辑计算模型<sup>[20]</sup>等多种 DNA 计算模型相继被构建,各种非自治算法或自组装方法也被提出用于解决诸如最大团<sup>[21]</sup>、图顶点着色<sup>[17,22]</sup>、可满足性<sup>[18]</sup>、N 皇后<sup>[23]</sup>、最大匹配<sup>[24]</sup>、最小顶点覆盖<sup>[25]</sup>、最小集合覆盖与最小精确覆盖<sup>[26]</sup>、子集和<sup>[27]</sup>、Ramsey 数<sup>[28]</sup>等若干非确定多项式(Non-deterministic Polynomial, NP)问题.

一方面,试管生化反应、纳米器件与分子自组装可被用于求解计算机科学问题<sup>[8,29-31]</sup>. 另一方面,由于拥有良好的信息处理机制与巨大的并行性,因此活细胞也可被用于执行计算. 许进教授等人<sup>[32]</sup>以合成生物学中广泛使用的生物砖为材料,以大肠杆菌 *Escherichia coli* 为宿主细胞,以 DNA 重组酶 Hin 为计算工具,构建了一个解决可满足性问题的细胞计算模型. 该模型中,每个细胞可独立生成并判定可满足性问题的一个解,数以亿计的细胞检查数以亿计的可能解,显示了细胞计算的巨大潜力. 文献<sup>[33]</sup>则使用大肠杆菌细胞构建一个条件学习系统,在“学习”信号的帮助下可识别“坏人”信号,这是构建人工分子生物智能系统的一个有益尝试.

计算机与其他计算工具的一个关键区别在于通用性. 许进教授<sup>[34]</sup>为通用 DNA 计算机找到了一种称之为“探针机”的数学模型,并将存储系统、运算系统、检测系统与控制系统集合为一个整体,逐步形成一个真正意义上的通用 DNA 计算机——“中州型 DNA 计算机”<sup>[34]</sup>.

模型检测问题是与可满足性问题相关但又非常不同的又一个重要计算问题,在 DNA 计算的框架内寻求模型检测问题的解法始终是研究人员追求的目标. 直到 2006 年,图灵奖得主 Emerson 教授等人<sup>[35]</sup>提出了一种使用 DNA 分子对 CTL 基本公式 EFp 进行模型检测的方法. 对于 LTL 逻辑而言,其模型检测在经典计算中是一个 PSPACE 问题. 我们的前期研究找到了可对 LTL 逻辑一个基本时序算子“until”实施检测的 DNA 方法<sup>[36]</sup>. 对于更多的 LTL 基本时序公式,形式化验证工业实践中常见的 LTL 公式,乃至任意给定的 LTL 一般公式,如何在 DNA 计算框架内实施模型检测?这些问题仍悬而未决. 本文将对此进行研究.

## 2 基础知识

### 2.1 线性时序逻辑

LTL 逻辑的一阶部分不可判定,因此我们只需研究其命题部分(Propositional LTL, PLTL).

① Mordechai B. Principles of the spin model checker. [www.ki.pwr.edu.pl/cichon/2014\\_15\\_b/AL/PrinciplesSpin-Checker.pdf](http://www.ki.pwr.edu.pl/cichon/2014_15_b/AL/PrinciplesSpin-Checker.pdf), 2008

② Clarke E. Grand challenge problem: Model check concurrent software. <http://www.csl.sri.com/users/shankar/VGC05/Clarke.ppt>

**定义 1.** 设  $AP$  是原子命题集合, 如果  $\varphi$  和  $\psi$  是 PLTL 公式, 那么,  $p \in AP, \neg\varphi, \varphi \vee \psi, \bigcirc\varphi, \varphi \triangleright \psi$  都是 PLTL 公式.

**定义 2.** 一个 PLTL 模型是一个三元组  $M = (S, R, L)$ , 其中:

- (1)  $S$  是非空有穷状态集,  $s \in S$  是状态;
- (2)  $R: S \rightarrow S, R(s)$  是状态  $s$  的后继状态;
- (3)  $L: S \rightarrow 2^{AP}, L(s)$  是在状态  $s$  中成立的原子命题的集合.

**定义 3.** 满足关系  $\models$  定义如下:

- (1)  $s \models p$  当且仅当  $p \in L(s)$ ;
- (2)  $s \models \neg\varphi$  当且仅当  $\neg(s \models \varphi)$ ;
- (3)  $s \models \varphi \vee \psi$  当且仅当  $(s \models \varphi) \vee (s \models \psi)$ ;
- (4)  $s \models \varphi \triangleright \psi$  当且仅当  $\exists j \geq 0, R_j(s) \models \psi \wedge (\forall 0 \leq k < j, R_k(s) \models \varphi)$ , 其中,  $R_0(s) = s, R_{n+1}(s) = R(R_n(s))$ ;
- (5)  $s \models \bigcirc\varphi$  当且仅当  $R(s) \models \varphi$ .

**定义 4.** 导出公式定义如下:

- (1)  $\diamond\varphi \triangleq \text{true} \triangleright \varphi$ ;
- (2)  $\square\varphi \triangleq \neg \diamond \neg \varphi$ ;
- (3)  $\varphi \wedge \psi \triangleq \neg(\neg\varphi \vee \neg\psi)$ ;
- (4)  $\varphi \rightarrow \psi \triangleq \neg\varphi \vee \psi$ .

并发程序验证的基本问题是活性(会不会死锁)、公平性(会不会死等)、安全性(会不会出错), 以及可达性, 均为形式化验证实践中常见的需求性质, 且与 4 个基本时序算子共同构成该逻辑中最具研究价值与实用价值的公式子集, 如表 1 所示.

表 1 线性时序逻辑的基本时序公式与常用公式

公式	文中编号	公式的自动机模型	公式类型
$\varphi_1 = p \triangleright q$	公式 1	图 1(a)	四个基本时序公式之一
$\varphi_2 = \square p$	公式 2	图 1(b)	四个基本时序公式之一
$\varphi_3 = \diamond p$	公式 3	图 1(c)	四个基本时序公式之一
$\varphi_4 = \bigcirc p$	公式 4	图 1(d)	四个基本时序公式之一
$\varphi_5 = \square \neg p$	公式 5	图 1(e)	常用公式: 描述安全性
$\varphi_6 = \diamond p$	公式 6	图 1(f)	常用公式: 描述可达性
$\varphi_7 = \square(p \rightarrow \diamond q)$	公式 7	图 1(g)	常用公式: 描述活性
$\varphi_8 = \square \diamond p \rightarrow \square \diamond q$	公式 8	图 1(h)	常用公式: 描述强公平性
$\varphi_9 = \diamond \square p \rightarrow \square \diamond q$	公式 9	图 1(i)	常用公式: 描述弱公平性

上述 9 个公式的直观含义涉及命题或性质成立的如下时序关系——从当前时刻开始, 给定原子性质未来是否成立? 9 个公式的具体含义分别如下.

公式 1. 原子性质  $p$  始终成立, 直到原子性质  $q$  成立.

公式 2. 原子性质  $p$  永远成立.

公式 3. 原子性质  $p$  终将成立.

公式 4. 下一时刻原子性质  $p$  成立.

公式 5. 原子性质  $p$  永远不成立. 如果某程序满足公式 5, 则该程序运行永远不会进入某种不安全状态.

公式 6. 含义同公式 3. 如果某程序满足公式 6, 则该程序运行终将到达某种状态.

公式 7. 在未来任意时刻, 如果原子性质  $p$  成立, 那么此后原子性质  $q$  终将成立. 如果某程序满足公式 7, 则该程序运行不会出现死锁.

公式 8. 如果从将来任意时刻开始, 原子性质  $p$  均最终成立, 那么从将来任意时刻开始, 原子性质  $q$  均最终成立. 如果某程序满足公式 8, 则该程序运行使用各种资源是“强公平”的.

公式 9. 如果从将来某一时刻开始, 原子性质  $p$  永远成立, 那么从将来任意时刻开始, 原子性质  $q$  均最终成立. 如果某程序满足公式 9, 则该程序运行使用各种资源是“弱公平”的.

图 1 则用(FSA)图的方式表达了 9 个公式的含义, 其中 FSA 的定义参见 2.2 节.

## 2.2 有穷状态自动机与模型检测

**定义 5.** 一个 FSA  $A$  是一个五元组  $(\Sigma, Q, T, q_0, F)$ , 其中:

- (1)  $\Sigma$  是一个有穷非空字母表(原子命题集);
- (2)  $Q$  是一个有穷非空状态集;
- (3) 转移函数(转移规则)  $T: Q \times \Sigma \rightarrow R(Q)$ ;
- (4)  $q_0 \in Q$  是起始状态;
- (5)  $F \subseteq Q$  是接受状态集.

**定义 6.** 设  $w = y_1, \dots, y_m$  是定义在  $\Sigma$  上的字符串, 如果存在  $Q$  中状态序列  $r_0, r_1, \dots, r_m$  满足:

- (1)  $r_0 = q_0$ ;
- (2)  $r_{i+1} \in T(r_i, y_{i+1}), i = 0, 1, \dots, m-1$ ;
- (3)  $r_m \in F$ ,

则称  $w$  是被  $A$  接受的文字.

**定义 7.**  $A$  识别的语言被定义为  $A$  接受的文字的集合.

在本文中, 系统建模使用的自动机均为带标签之 FSA, 其原子命题在状态而非边上被满足, 例如

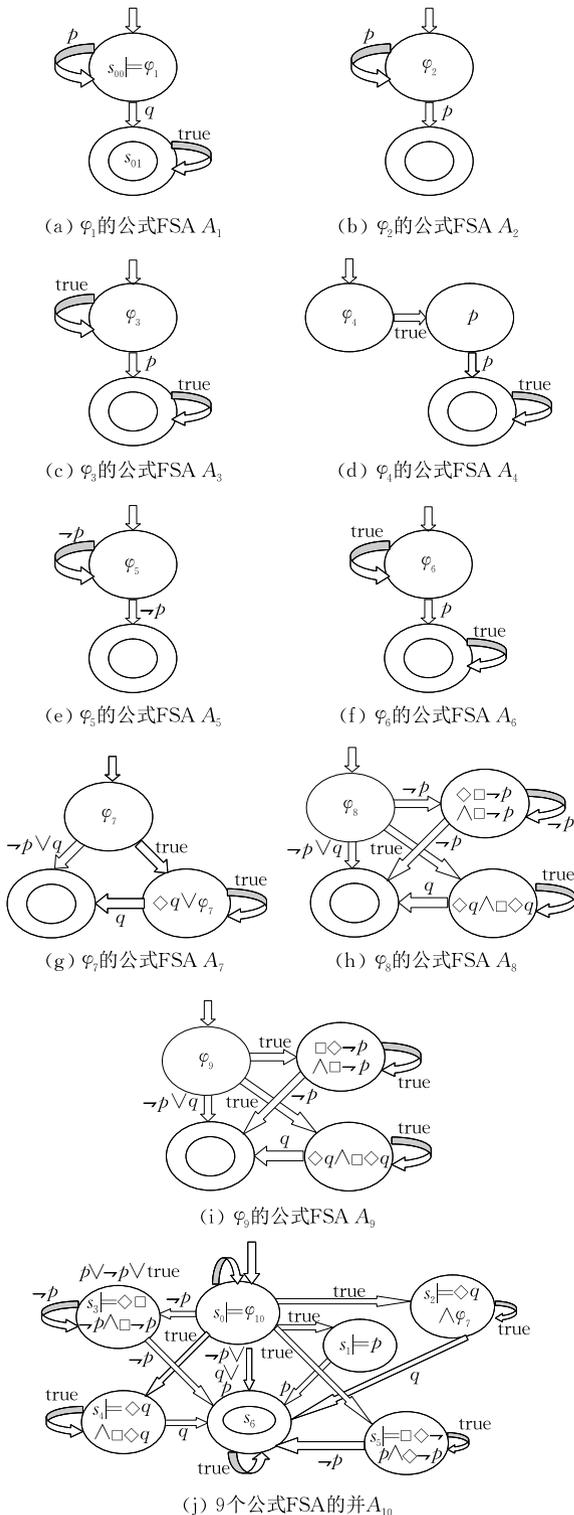


图 1 公式对应的 FSA

图 2;与之相反,公式的自动机模型均为不带标签之 FSA,其原子命题在边上被满足,例如如图 1.

**例 1.** 图 1(a) 示例了一个不带标签之 FSA. 在此例中,FSA 存在 2 个状态(2 个节点),分别是  $s_{00}$  和  $s_{01}$ ,其中, $s_{00}$  是初始状态, $s_{01}$  是终止状态. 此公式 FSA 有如下 3 个转移规则(3 个边):转移规则 1(从

状态  $s_{00}$  出发输入字母  $p$  即进入状态  $s_{00}$ )、转移规则 2(从状态  $s_{00}$  出发输入字母  $q$  即进入状态  $s_{01}$ )以及转移规则 3(从状态  $s_{01}$  出发输入字母  $p$  或  $q$  即进入状态  $s_{01}$ ),而  $ppq$ 、 $pq$ 、 $pqp$  则为此 FSA 的接受字集中的 3 个接受字.

**例 2.** 图 2 示例了一个带标签之 FSA. 在此例中,FSA 存在 3 个状态,分别是 0、1 和 2,其中,0 是初始状态,2 是终止状态;状态 0 满足原子命题  $p$  和  $q$ ,状态 1 满足原子命题  $p$ ,状态 2 满足原子命题  $q$ ;FSA 可从状态 0 转移至状态 1,也可从状态 1 转移至状态 0,还可从状态 1 转移至状态 2. 从图中显而易见:0-1-2 是 FSA 的一个接受运行经过的路径(状态序列), $ppq$  是该路径先后满足的一个原子命题序列;0-1-0-1-2 是 FSA 的另一个接受运行经过的路径(状态序列), $ppppq$  是此路径先后满足的一个原子命题序列. 依次类推,由于长度的不同,存在无穷多个类似路径,因而存在无穷多个原子命题序列.

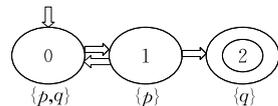


图 2 本文实验使用的系统模型 FSA

2.1 节给出了 LTL 公式的定义,2.2 节给出了系统模型 FSA 的定义,在此基础上,模型检测问题被定义为算法判定系统模型 FSA 是否满足 LTL 公式.

LTL 模型检测方法的原理如图 3 所示. 如果对于系统 FSA 的每一个接受运行(路径)经过的状态序列,存在一个它满足的原子命题序列,使得该原子

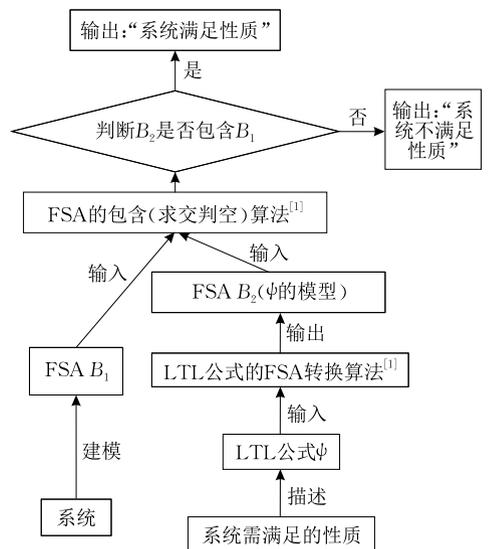


图 3 基于经典计算的 LTL 模型检测算法原理

命题序列组成的字符串是公式 FSA 的一个接受文字,那么,系统 FSA 满足公式.反之也成立.简而言之,模型检测结果为“yes”,当且仅当系统 FSA 的每一个路径都是公式 FSA 路径集合的一个元素.

**例 3.** 图 2 所示的系统 FSA 满足公式 1,计算原理如下.把图 2 中路径 0-1-2 对应的一个字符串  $ppq$  输入图 1(a) 所示的公式 1 的 FSA,计算得知  $ppq$  是公式 1 的 FSA 接受的文字,即图 2 中路径 0-1-2( $ppq$ ) 是图 1(a) 中路径集合的一个元素( $ppq$  也是公式 1 的 FSA 的一个路径).把图 2 中其他路径对应的字符串作为输入,也获得了同样的结果.因此,图 2 的每一条(从 0 开始到 2 结束的)路径都是公式 1 的 FSA 路径集合的一个元素.根据图 3 所示的算法原理,模型检测的结果为“yes”,即系统满足公式.

### 2.3 粘贴自动机

粘贴自动机<sup>[37]</sup>是一种实现 FSA 的 DNA 计算模型.给定一个表征输入字符串(文字)的 DNA 链,给定一个 FSA,粘贴自动机模型能够判断该字符串是否被 FSA 接受.

#### 2.3.1 FSA 和输入字符串的编码方式

文献[37]给出了如下编码方式.

假设  $M=(\Sigma, S, T, s_0, F)$  是一个 FSA,字母表  $\Sigma$  中的任意一个字符  $a$  都可以用单链 DNA 编码为  $C(a)$ .

(1)  $\Sigma$  上的输入字符串  $a_1, \dots, a_n$  由 DNA 单链分子编码:  $5' I_1 X_0 \dots X_m C(a_1) \dots X_0 \dots X_m C(a_n) X_0 \dots X_m I_2 3'$ , 其中,  $I_1$  是启动区序列,  $X_0 \dots X_m$  是间隔序列,以间隔字符  $a_i$  的编码  $C(a_i)$ ,  $I_2$  是终止区序列.

(2) 状态转移函数  $T(s_i, a) = s_j$  被编码为  $3' \overline{X_{i+1}} \dots \overline{X_m C(a)} X_0 \dots X_j 5'$ , 其中,  $\overline{X}$  表示核苷酸  $X$  的 Watson-Crick 补链,  $\overline{C(a)}$  表示  $a$  的 DNA 序列的 Watson-Crick 补链.

(3) 每个初态  $s_i$  的编码:  $3' \overline{I_1 X_0 \dots X_i} 5'$ .

(4) 每个终态  $s_j$  的编码:  $3' \overline{X_{j+1} \dots X_m I_2} 5'$ .

#### 2.3.2 计算过程

粘贴自动机模型的计算过程分为以下 3 步<sup>[37]</sup>.

第 1 步. 数据预处理.

(1) 合成输入字符串和自动机的编码链.

(2) 将所有的 DNA 链放到试管  $T$  中,退火使互补链充分杂交,碱基互补配对形成双链;在试管  $T$  中加入连接酶,得到(完全或部分)双链 DNA 分子.

第 2 步. 计算.

经过第 1 步的数据处理,如果输入字符串被自动机接受,那么试管  $T$  中是以启动区开始、以终止区结束的完全双链 DNA 分子;否则,试管  $T$  中的 DNA 分子不是完全双链.原因在于,表达输入字符串的单链分子之部分片段被表达转移规则的单链分子配对成功,而其余部分片段则不存在表达转移规则的单链分子可与之配对.故此,我们向试管  $T$  中加入 Mung Bean 核酶,降解其中的单链 DNA 片段,保留完全双链 DNA 分子,以完成本步.

第 3 步. 输出结果.

利用电泳技术分离不同长度的 DNA 分子.若存在多种长度 DNA 分子,则表明在加入核酶前试管  $T$  中存在部分双链 DNA 分子,这提示输入字符串不被自动机接受;否则,在加入核酶前,  $T$  中均为完全双链 DNA 分子,这提示输入字符串被自动机接受.

### 2.4 Mung Bean 核酶及其应用于粘贴自动机

$S_1$  核酶来源于米曲霉,可降解单链 DNA 或核糖核酸(Ribonucleic Acid, RNA),是一种单链核酸酶,产生带 5' 磷酸的单核苷酸或寡核苷酸双链,对双链 DNA、双链 RNA 和 DNA-RNA 杂交体不敏感.酶浓度大时可完全消化双链,中等浓度时可在切口或小缺口处切割双链.

Mung Bean 核酶是来源于绿豆芽的核酸酶,将单链 DNA 降解为 5' 端带磷酸基团的单核苷酸或寡核苷酸.双链 DNA、双链 RNA 及 DNA-RNA 杂交体对此酶相对不敏感.然而此酶大量使用时也能将双链核酸完全降解. Mung Bean 核酶的物理性质及催化性质与  $S_1$  核酶相似,但 Mung Bean 核酶的作用比  $S_1$  核酶更温和. Mung Bean 核酶用于使 DNA 突出端变为平端.

总之,上述两种核酶都可从单链 DNA 分子上逐个删除核苷酸,  $S_1$  核酶有时也将单链 DNA 分子强行插入双链 DNA,因此 Mung Bean 核酶更适合于粘贴自动机计算<sup>[37]</sup>.两种酶都可降解无法接受的输入字符串的单链片段,而被接受的输入字符串完全双链 DNA 分子在酶降解后仍保持完整无缺<sup>[37]</sup>.

文献[38]使用 Mung Bean 核酶降解粘贴自动机单链片段,该文以真实生化实验的方式正确计算出 0、1 两个二进制数之间的“逻辑与”、“逻辑或”的运算结果.从而再次证实,使用 Mung Bean 核酶降解粘贴自动机单链片段,是生物学可实现的.此外,该实验也得到了一些有关使用 Mung Bean 核酶降

解粘贴自动机单链片段的如下技术要点<sup>[38]</sup>:

(1) 掌握好酶量与反应时间. 酶量与反应时间过少则单链降解不完全, 过多则可导致双链降解. 该实验给出的酶量是输入字符串分子量的 2/7, 反应时间是 3 小时<sup>[38]</sup>.

(2) 需注意 DNA 分子的呼吸作用, 即双链分子以单双链转变的形式动态存在. 该作用可导致酶降解单链状态下互补分子, 从而改变 DNA 分子动态平衡条件, 最终导致双链分子被降解. 该实验的对策是, 不采用热失活方式终止酶切反应, 加入蛋白变性剂失活 Mung Bean 核酶, 同时在退火后先用 T4 连接酶将 DNA 分子的缺口补上, 以稳定双链分子<sup>[38]</sup>.

### 3 基于粘贴自动机的模型检测方法

#### 3.1 将 9 个 LTL 公式转换为 FSA

对于表 1 给出的 LTL 的 9 个基本公式或常用公式, 分别构造它们的 FSA 模型, 如图 1 中的前 9 个子图所示. 这些子图的并, 如图 1(j) 所示.

按照粘贴自动机模型中 FSA 的编码方式编码图 1(j) 中的自动机. 将每一个实施编码后的 DNA 单链按如下规则选择放置: 如果该单链是对自动机  $A_i$  中的某个节点或边实施的编码, 则把该单链放置于试管  $R_i$  中, 其中  $i$  是从 1 到 9 之间的自然数. 若 LTL 公式不属于上述 9 个公式, 则对公式的 FSA 模型使用同样的方法实施编码后放入试管  $R_0$ .

#### 3.2 按照粘贴自动机的输入字符串格式生成系统的运行

##### 3.2.1 状态编码

状态用图 4 所示的单链 DNA 编码. 其中  $X_0 \cdots X_m$  是粘贴自动机的间隔序列, 为满足粘贴自动机输入字符串的格式而增加;  $L(s)$  是状态标签函数;  $Y$  用于调节编码, 使得每个状态编码的长度均为彼此相等的偶数, 并且确保状态和状态编码之间一一对应.



图 4 状态的 DNA 编码

##### 3.2.2 生成系统 FSA 的所有以开始状态为始点以终止状态为终点的路径

**算法 1.** 系统模型自动机的运行路径生成算法<sup>[36]</sup>.

输入: 系统自动机的模型  $M$

输出: 表达  $M$  所有路径的所有 DNA 分子

BEGIN

第 1 步. 对于每个状态  $s$ , 向试管  $T$  中加入  $s$  的编

码; 对于每条边  $e$ , 向试管  $T$  中加入  $e$  的编码. 经过退火和连接反应, 生成系统的 FSA 模型的随机路径.

第 2 步. 对试管  $T$  执行提取前缀操作, 使试管  $T$  中只含有以开始状态为起点的路径.

第 3 步. 对试管  $T$  执行提取后缀操作, 使试管  $T$  中只含有以终止状态为终点的路径.

第 4 步. 在试管  $T$  中加入 DNA 分子  $\frac{I_1}{I_1 H_0}$  和  $\frac{X_0 \cdots X_m I_2}{B X_0 \cdots X_m I_2}$ , 其中  $I_1$  是启动区,  $H_0$  是初始状态编码的前半段,  $B$  是终止状态编码的后半段,  $X_0 \cdots X_m$  是间隔序列,  $I_2$  是终止区. 通过退火和连接反应, 在单链 DNA 分子 5' 端添加启动区, 并且在单链 DNA 分子 3' 端添加间隔区和终止区.

第 5 步. 使用亲和纯化技术, 过滤出试管  $T$  中所有包含  $I_1$  和  $I_2$  序列的单链 DNA 分子.

END

其中, 第 5 步除去第 4 步中多余的 DNA 分子, 得到符合粘贴自动机输入字符串格式要求的单链 DNA 分子, 每条单链 DNA 对应系统的一个运行.

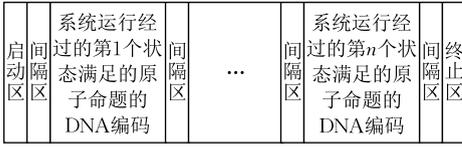
#### 3.3 用粘贴自动机验证系统是否满足公式

2.3 节给出的粘贴自动机计算原理可概括如下: 一些单链 DNA 分子(本文称之为 I 类分子)用于编码字符串, 另一些单链 DNA 分子(本文称之为 II 类分子)用于编码 FSA, 上述两种单链分子充分杂交, 如果形成完全双链分子, 则字符串是 FSA 接受的文字; 否则, 即为不被接受的文字.

进一步地, 我们可以把 LTL 模型检测问题规约为粘贴自动机的计算问题, 规约原理如图 5 所示.

参见图 5(e). 执行算法 1 得到的所有 I 类分子即表征着系统 FSA 运行的所有路径, 即为将参与粘贴自动机计算的已编码的字符串, 编码格式如图 5(a) 所示. 使用 2.3.1 节的编码方式(2)、(3)和(4)可以对给定的 LTL 公式的 FSA 模型设计出 II 类分子, 编码格式如图 5(b) 所示. 把 I 类分子和 II 类分子作为粘贴自动机计算的两个输入, 根据粘贴自动机计算原理(特别是 DNA 编码设计的两类分子之碱基配对关系), 两类分子杂交形成(或未形成)完全双链当且仅当系统运行(字符串)是(或不是)公式 FSA 的接受字, 参见图 5(c)和图 5(d). 根据图 3 所示的模型检测原理(特别是两个 FSA 的路径包含关系), 如果系统 FSA 的所有路径(字符串)均为此 LTL 公式 FSA 模型接受的文字, 那么系统模型 FSA 满足 LTL 公式, 否则, 不满足.

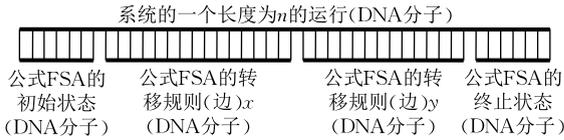
以上即为 LTL 模型检测的 DNA 算法的核心思想与基本原理. 它解释了该算法能够验证系统是否满足公式的原因. 算法描述如下.



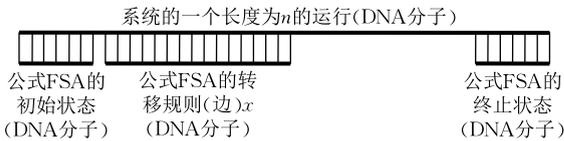
(a) 表征系统FSA运行的DNA分子(I类分子)



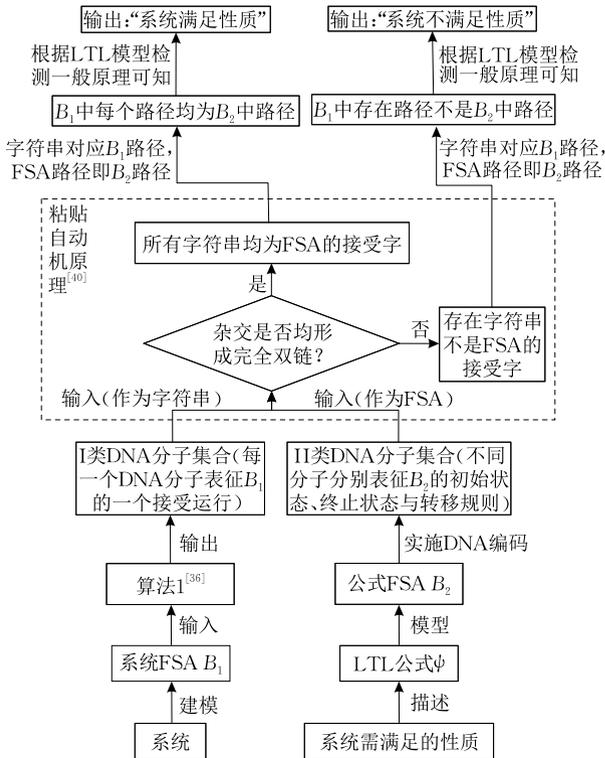
(b) 表征公式FSA的3种DNA分子(II类分子)



(c) 两类分子杂交之后形成完全双链  
(当系统某运行是公式FSA的接受字)



(d) 两类分子杂交之后形成不完全双链  
(当系统某运行不是公式FSA的接受字)



(e) 算法流程

图 5 基于 DNA 的 LTL 模型检测方法原理

**算法 2.** LTL 逻辑公式的 DNA 模型检测算法.

输入: 3.1 节的试管  $R = \{R_0, R_1, \dots, R_9\}$ , 即 LTL 公式之粘贴自动机编码; 算法 1 得到的试管  $T$ , 即系统的所有运行

输出: 系统模型是否满足公式

BEGIN

$$l_1, l_3, l_6 := |V| \times 2^{|V|-1} + |E|$$

$$l_2, l_5 := |V|$$

$$l_4 := 2$$

$$l_7, l_8, l_9 := |V| \times 2^{|V|-1} + |V| + |E|$$

$$l_0 := \text{len}(f) \times (|V| \times 2^{|V|-1} + |E|)$$

IF 待检测的公式为  $\varphi_i$ , 其中  $i$  是从 1 到 9 之间的某个自然数, THEN

第 1 步. 用长度分离操作提出试管  $T$  中长度小于  $l_i$  的单链 DNA.

第 2 步. 合并试管  $R_i$  到  $T$ , 在  $T$  中加入 DNA 连接酶, 使互补链充分杂交.

ELSE /\* 待检测的为一般公式

第 1 步. 用长度分离操作提出试管  $T$  中长度小于  $l_0$  的单链 DNA.

第 2 步. 合并试管  $R_0$  到  $T$ , 在  $T$  中加入 DNA 连接酶, 使互补链充分杂交.

ENDIF

第 3 步. 在试管  $T$  中加入 Mung Bean 核酶, 降解所有单链 DNA 分子.

IF 待检测的公式为  $\varphi_i$ , 其中  $i$  是从 1 到 9 之间的某个自然数, THEN

第 4 步. 对所有的从 1 到  $l_i$  之间的自然数  $j$ , 使用凝胶电泳技术分离出长度为  $|I_1 X_0 \dots X_m C(a_1) \dots X_0 \dots X_m C(a_j) X_0 \dots X_m I_2|bp$  的 DNA 分子, 通过凝胶成像系统显示结果. 考察杂交反应前系统路径 DNA 分子浓度是否约等于反应后完全双链 DNA 分子浓度. 如果是, 模型检测结果返回“*Yes*”. 否则, 模型检测结果返回“*No*”, 并使用 sanger 测序法生成一系列长度相差 1 个碱基的 PCR 产物混合物, 根据不同长度片段的电泳速度不同, 电泳技术直接读取 DNA 上的核苷酸顺序, 比较不完全双链分子与转移规则编码, 即可获得不满足公式的自动机路径.

ELSE /\* 待检测的为一般公式

第 4 步. 对所有的从 1 到  $l_0$  之间的自然数  $j$ , 使用凝胶电泳技术分离出长度为  $|I_1 X_0 \dots X_m C(a_1) \dots X_0 \dots X_m C(a_j) X_0 \dots X_m I_2|bp$  的 DNA 分子, 通过凝胶成像系统显示结果. 考察杂交反应前系统路径 DNA 分子浓度是否约等于反应后完全双链 DNA 分子浓度. 如果是, 模型检测结果返回“*Yes*”. 否则, 模型检测结果返回“*No*”, 并使用 sanger 测序法生成一系列长度相差 1 个碱基的 PCR 产物混合物, 根据不同长度片段的电泳速度不同, 电泳技术直接读取 DNA 上的核苷酸顺序, 比较不完全双链分子与转移规则编码, 即可获得不满足公式的自动机路径.

ENDIF

END

公式 FSA 接受运行可产生完全双链 DNA 分子, 它既含启动区又含终止区, 无法被 Mung Bean

核酶切开. 而不被公式 FSA 接受的运行则产生部分双链 DNA 分子, 将被 Mung Bean 核酶切开, 切后的产物或者只含启动区, 或者只含终止区. 算法 2 就是根据此现象获得检测结果.

读出检测结果需要使用凝胶电泳技术. 当接受运行经过  $j$  个原子命题时, 分子编码为  $I_1 X_0 \cdots X_m C(a_1) \cdots X_0 \cdots X_m C(a_j) X_0 \cdots X_m I_2$ , 其长度  $|I_1 X_0 \cdots X_m C(a_1) \cdots X_0 \cdots X_m C(a_j) X_0 \cdots X_m I_2| = |I_1| + |I_2| + (j+1) \times |X_0 \cdots X_m| + j \times |C(a_1)|$ , 而  $|I_1|$ ,  $|I_2|$ ,  $|X_0 \cdots X_m|$ ,  $|C(a_1)|$  均为在编码设计阶段即被确定的常数, 因此在使用凝胶电泳技术时, DNA 分子的长度可利用上式计算. 由于经过原子命题的数目不同, 因此不同的接受运行都有其不同的长度. 单链分子、不同时包含启动区与终止区的双链分子并非接受运行, 其长度均不符合上式计算结果. 因此可知, 长度符合上式计算结果的双链分子与接受运行存在对应关系. 当所有运行(路径)均满足公式时, 系统满足公式.

在真实的检测中, 情况更加复杂. DNA 计算本质上是一种概率计算, 真阳性的概率无法达到百分之百. 在本算法中如果特异性生化反应概率接近 1, 即可判定模型检测结果有效. 杂交后长度为  $|I_1 X_0 \cdots X_m C(a_1) \cdots X_0 \cdots X_m C(a_j) X_0 \cdots X_m I_2|$  的分子浓度若约等于杂交前 I 类分子的浓度, 则真阳性率接近于 1. 其中, DNA 分子浓度的测定可使用分光光度法, 通过分光光度计读取 OD260 值, 根据 DNA 浓度公式计算分子浓度.

当真阳性率远小于 1 时, 系统不满足公式, 模型检测算法需给出反例. 此时, 并非所有路径分子均形成长度为  $|I_1 X_0 \cdots X_m C(a_1) \cdots X_0 \cdots X_m C(a_j) X_0 \cdots X_m I_2|$  的完全双链分子, 对部分双链分子, 经 Mung Bean 核酶降解, 获得字符串分子与部分转移规则分子的配对, 如此配对在逻辑层面并非完全, 但在分子层面则为双链分子片段. 使用 sanger 测序法可读出双链片段如下信息: 何种字符串与何种转移规则实现了匹配, 代表何种系统运行路径的字符串未能实现与转移规则的完全匹配. 使用电泳技术读出此字符串, 即为算法给出的反例.

**例 4.** 图 2 所示的系统 FSA 满足公式 1, 其 DNA 计算原理如图 6 所示. 图 6(a) 给出了 1 种 DNA 分子, 被用于表征系统 FSA 的一个接受运行 0-1-2, 此运行经过的 3 个状态满足的原子命题序列可为  $ppq$  也可为  $qpq$ . 图 6(c) 给出了 5 种 DNA 分子, 分别表征公式 FSA 的初始状态  $s_{00}$ 、终止状态  $s_{01}$ 、转移规则 1、转移规则 2 以及转移规则 3 (3 个转移规则的含义参见例 1). 如图 6(d) 所示, 粘帖自动

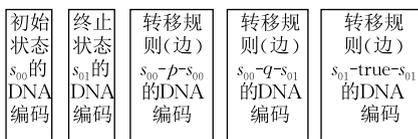
机编码框架内恰当的 DNA 编码设计可导致如下现象发生——表征状态  $s_{00}$  的 1 个分子、表征状态  $s_{01}$  的 1 个分子、表征转移规则 1 的 2 个分子、表征转移规则 2 的 1 个分子与表征系统运行 0-1-2 的 1 个分子通过碱基配对形成 1 个完全双链. 参见图 6(d), 从左往右观察此完全双链的上下两条单链, 可见它们携带的信息均符合如下特征: 从初始状态出发, 到终止状态结束, 中间经过的原子命题序列为  $ppq$ . 这意味着, 系统运行 0-1-2 经过的字符串  $ppq$  是公式 1 的 FSA 的一个接受字. 同理可见, 所有 I 类分子与所有 II 类分子均形成完全双链, 因此可知, 系统的所有接受运行都是公式 1 的 FSA 的接受字. 因此, 可以获得如下模型检测结果: 图 2 所示的系统 FSA 满足公式 1.



(a) 表征系统FSA一个运行0-1-2的DNA分子



(b) 表征系统FSA一个运行0-1-0-1-2的DNA分子



(c) 表征公式FSA的5种DNA分子

系统的一个运行0-1-2(满足 $ppq$ 或 $qpq$ )(DNA分子)



(d) 两类分子杂交之后形成完全双链

图 6 一个模型检测例子: 图 2 所示 FSA 满足公式 1

### 3.4 复杂度分析

文献[36]给出了算法 1, 并指出其复杂度为  $O(x+y)$ , 其中  $x$  和  $y$  分别为自动机的节点数和边数. 算法 2 需要调用算法 1 作为前者的输入.

设算法 2 本身调用的基本操作包括基本生化操作、基本逻辑操作(基本电子操作). 其中, 基本生化操作包括向试管中加入编码、退火、连接、提取前缀操作(使用探针杂交或 PCR 技术)、提取后缀操作(使用探针杂交或 PCR 技术)、亲和纯化、凝胶电泳、合并试管、添加酶、测序; 基本逻辑操作包括赋值、按

给定式子计算数值.

算法 2 始于 10 个赋值语句, 执行了 10 个基本逻辑操作. 第 1 步使用 1 次凝胶电泳技术以分离不同长度的分子. 第 2 步调用合并试管与添加酶各 1 次, 计 2 个基本生化操作. 第 3 步添加酶 1 次. 第 4 步使用 2 次凝胶电泳技术和 1 次 sanger 测序. 综上, 算法 2 需执行  $O(1)$  个基本操作, 调用算法 1 作为输入需时  $O(x+y)$ , 因此算法 2 共计执行  $O(x+y)$  个基本操作.

表 2 给出了新算法与相关方法的复杂度比较. 可见, 与基于电子计算的经典方法相比, 基于 DNA 计算的新方法把基本操作次数的数量级降为线性. 然而, 生物操作所需时间既不统一也不固定, 一些杂交反应甚至需要数小时才能达到平衡状态<sup>[37]</sup>, 这也是当前的 DNA 计算面临的共性问题之一. 两种机制的计算方法存在互补性.

表 2 时序逻辑模型检测的 DNA 方法的复杂度比较

检测 LTL 的 DNA 算法 (新方法)	检测 CTL 公式的 DNA 方法 <sup>[35]</sup>	检测 LTL 的 经典方法 <sup>[1]</sup>
$O(x+y)$ 个基本操作	$O(x+y)$ 个基本操作	PSPACE 个基本操作

## 4 仿真实验

### 4.1 实验所需的 DNA 编码设计

本文算法在粘贴自动机通用模型上构建, 因此必须满足粘贴自动机的编码设计规则, 仅在规则框架内细化设计. 需要具体设计的编码包括启动区  $I_1$ 、终止区  $I_2$ 、间隔区  $X = X_1 \cdots X_n$  以及原子命题的编码. 如图 1(j) 所示, 本实验拟检测的 9 个公式合并 FSA 包含 7 个节点, 根据粘贴自动机模型的原理, 间隔区需要由 7 个以上的  $X_i$  组成 ( $i$  为自然数). 当每一个  $X_i$  的碱基位点数均为  $j$  时可编码  $2j$  个  $X = X_1 \cdots X_n$ , 因此  $j$  的最小值为 3. 可为原子命题设计 3 个碱基位, 为启动区和终止区分别设计 4 个碱基位, 以避免位数过高带来的生化反应效率低下以及位数过低带来的编码重复. 在此基础上可推导获得图 1(j) 所示 FSA 的粘贴自动机编码.

DNA 编码序列设计是特异性杂交的物质保证, 编码合理与否事关生化实验成败. 为保证特异性杂交, DNA 编码设计须满足一些分子生物学约束条件, 包括物理约束和热力学约束<sup>[39]</sup>.

如上文所述, 本实验编码仅在给定框架内设计, 因而物理约束不可变. 我们研究其热力学约束, 考察解链温度与自由能.

NUPACK<sup>①</sup> 是一款免费软件, 可用于 DNA 序

列编码设计与分析, 由加州理工学院开发. 在此通用平台上, 我们设计出表征系统运行的 DNA 编码序列, 参见图 7 (以运行  $pq$  为例, 加号前是设计的编码序列, 加号后为配对序列, 需要注意的是两序列在图中的显示都是按照 5'-3' 方向的顺序, 这是 NUPACK 工具的一个输出特点, 下同), 设计序列在 10 摄氏度下的热力学分析如图 8、图 9 和图 10 所示.

```
CGCGAACGCCCTAATTGCGCAATCAATGTGACAGCGCCCTAATTGCGCAATCAATGTCTGAA
CGCCCTAATTGCGCAATCAATGTCCG+CGGAACATTGATTGCGAATTAGGGCGTTCCAGAC
ATTGATTGCGAATTAGGGCGTTCTGACATTGATTGCGAATTAGGGCGTTCCGC
```

图 7 设计出的 DNA 编码序列

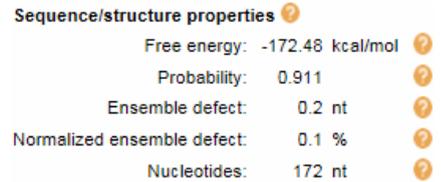


图 8 编码序列的结构性质

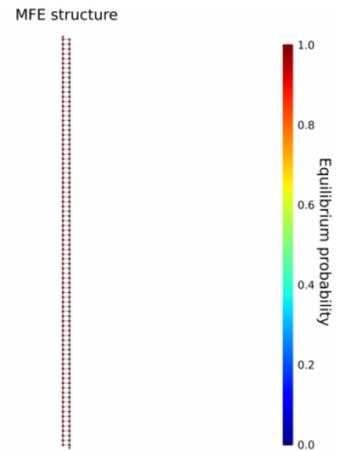


图 9 热力学分析: 最小自由能结构

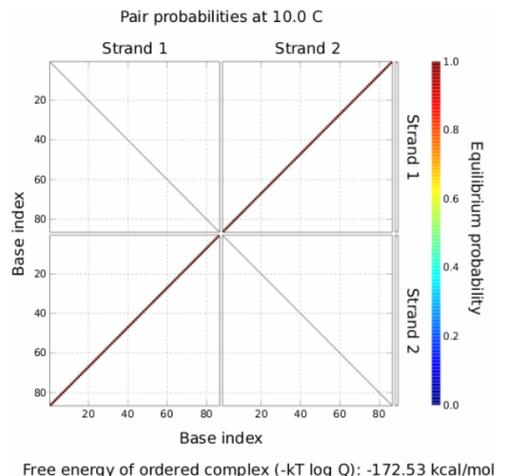


图 10 平衡状态下的配对概率

① NUPACK: Nucleic Acid Package. <http://www.nupack.org>, 2015

如图 8 所示, Normalized Ensemble Defect (NED) 的含义是编码在生化反应达到平衡时核苷酸的不正确配对比例, 0% 提示设计最优, 100% 则提示设计最差, 设计出的该编码序列的 NED 值为 0.1%。

最小自由能原理指出, 生化反应达到平衡时自由能最小, 如图 9 所示: 两种分子的配对颜色呈深红色, 对照右侧指示平衡概率的竖条颜色变化可知, 双链分子完全配对达到平衡的概率几乎为 1. 可见, 其自由能约等于最小自由能。

如图 10 所示, 红线位置提示两单链所有碱基位点完全按顺序互补配对, 红线颜色提示所有配对的概率均约等于 1.

综上, 本文设计的 DNA 序列编码满足最小自由能约束, 参加反应的 DNA 具有基本一致的解链温度. 以此编码序列实施仿真实验, 获得的结果在生物学意义上是可信、有效的。

图 7 给出了输入字符串的编码规则, 如表 3 所示. 根据表 3, 可推导出 9 个公式的粘贴自动机编码, 如表 4 与表 5 所示.

表 3 输入字符串的编码规则

(表征系统运行, 按粘贴自动机输入字符串的方式编码)

编码对象	编码
启动区	$I_1 = 5' \text{ CGCG } 3'$
间隔区	$X_0 = 5' \text{ AAC } 3', X_1 = 5' \text{ GCC } 3', X_2 = 5' \text{ CTA } 3',$ $X_3 = 5' \text{ ATT } 3', X_4 = 5' \text{ CGC } 3', X_5 = 5' \text{ AAT } 3',$ $X_6 = 5' \text{ CAA } 3', X_7 = 5' \text{ TGT } 3'$
终止区	$I_2 = 5' \text{ TCCG } 3'$
原子命题	$p = 5' \text{ CAG } 3', q = 5' \text{ CTG } 3', r = \neg p = 5' \text{ TTT } 3'$

表 4 9 个公式对应的自动机的 DNA 编码

(分别置于试管 R 的 9 个分管, 按粘贴自动机的方式编码)

编码对象	DNA 编码(从左到右是 3'-5' 方向)
开始状态 $s_0$	$\text{GCGC}\overline{X_0} 5' = 3' \text{ GCGCTTG}$
结束状态 $s_6$	$\overline{X_7} \text{AGGC } 5' = 3' \text{ ACAAGGC}$
转移规则 $t(s_0, p) = s_0$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{GTC}\overline{X_0}$
转移规则 $t(s_0, q) = s_6$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{GAC}\overline{X_0 X_1 X_2 X_3 X_4 X_5 X_6}$
转移规则 $t(s_0, \text{true}) = s_6$	$\overline{X_7} \text{GTC} \text{GAC}\overline{X_0 X_1 X_2 X_3 X_4 X_5 X_6}$
转移规则 $t(s_0, p) = s_6$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{GTC}\overline{X_0 X_1 X_2 X_3 X_4 X_5 X_6}$
转移规则 $t(s_0, \text{true}) = s_0$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{GTC} \text{GAC}\overline{X_0}$
转移规则 $t(s_0, \text{true}) = s_1$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{GTC} \text{GAC}\overline{X_0 X_1}$
转移规则 $t(s_1, p) = s_6$	$\overline{X_2 X_3 X_4 X_5 X_6 X_7} \text{GTC}\overline{X_0 X_1 X_2 X_3 X_4 X_5 X_6}$
转移规则 $t(s_0, r) = s_0$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{AAA}\overline{X_0}$
转移规则 $t(s_0, r) = s_6$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{AAA}\overline{X_0 X_1 X_2 X_3 X_4 X_5 X_6}$
转移规则 $t(s_0, \text{true}) = s_2$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{GTC} \text{GAC} \text{AAA}\overline{X_0 X_1 X_2}$

(续 表)

编码对象	DNA 编码(从左到右是 3'-5' 方向)
转移规则 $t(s_2, \text{true}) = s_2$	$\overline{X_3 X_4 X_5 X_6 X_7} \text{GTC} \text{GAC} \text{AAA}\overline{X_0 X_1 X_2}$
转移规则 $t(s_0, q \vee r) = s_6$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{GAC} \text{AAA}\overline{X_0 X_1 X_2 X_3 X_4 X_5 X_6}$
转移规则 $t(s_2, q) = s_6$	$\overline{X_3 X_4 X_5 X_6 X_7} \text{GAC}\overline{X_0 X_1 X_2 X_3 X_4 X_5 X_6}$
转移规则 $t(s_0, r) = s_3$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{AAA}\overline{X_0 X_1 X_2 X_3}$
转移规则 $t(s_0, \text{true}) = s_4$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{GTC} \text{GAC} \text{AAA}\overline{X_0 X_1 X_2 X_3 X_4}$
转移规则 $t(s_3, r) = s_3$	$\overline{X_1 X_5 X_6 X_7} \text{AAA}\overline{X_0 X_1 X_2 X_3}$
转移规则 $t(s_3, r) = s_6$	$\overline{X_4 X_5 X_6 X_7} \text{AAA}\overline{X_0 X_1 X_2 X_3 X_4 X_5 X_6}$
转移规则 $t(s_4, \text{true}) = s_4$	$\overline{X_5 X_6 X_7} \text{GTC} \text{GAC} \text{AAA}\overline{X_0 X_1 X_2 X_3 X_4}$
转移规则 $t(s_4, q) = s_6$	$\overline{X_5 X_6 X_7} \text{GAC}\overline{X_0 X_1 X_2 X_3 X_4 X_5 X_6}$
转移规则 $t(s_0, \text{true}) = s_5$	$\overline{X_1 X_2 X_3 X_4 X_5 X_6 X_7} \text{GTC} \text{GAC} \text{AAA}\overline{X_0 X_1 X_2 X_3 X_4 X_5}$
转移规则 $t(s_5, \text{true}) = s_5$	$\overline{X_6 X_7} \text{GTC} \text{GAC} \text{AAA}\overline{X_0 X_1 X_2 X_3 X_4 X_5}$
转移规则 $t(s_5, r) = s_6$	$\overline{X_6 X_7} \text{AAA}\overline{X_0 X_1 X_2 X_3 X_4 X_5 X_6}$

表 5 表 4 编码对象所属公式及所在试管

编码对象	所属公式	编码所在试管
开始状态 $s_0$	所有公式	所有 $R_i$
结束状态 $s_6$	所有公式	所有 $R_i$
转移规则 $t(s_0, p) = s_0$	$\varphi_1, \varphi_2$	$R_1, R_2$
转移规则 $t(s_0, q) = s_6$	$\varphi_1$	$R_1$
转移规则 $t(s_6, \text{true}) = s_6$	$\varphi_1, \varphi_3, \varphi_4, \varphi_6$	$R_1, R_3, R_4, R_6$
转移规则 $t(s_0, p) = s_6$	$\varphi_2, \varphi_3, \varphi_6$	$R_2, R_3, R_5$
转移规则 $t(s_0, \text{true}) = s_0$	$\varphi_3, \varphi_6$	$R_3, R_6$
转移规则 $t(s_0, \text{true}) = s_1$	$\varphi_4$	$R_4$
转移规则 $t(s_1, p) = s_6$	$\varphi_4$	$R_4$
转移规则 $t(s_0, r) = s_0$	$\varphi_5$	$R_5$
转移规则 $t(s_0, r) = s_6$	$\varphi_5$	$R_5$
转移规则 $t(s_0, \text{true}) = s_2$	$\varphi_7$	$R_7$
转移规则 $t(s_2, \text{true}) = s_2$	$\varphi_7$	$R_7$
转移规则 $t(s_0, q \vee r) = s_6$	$\varphi_7, \varphi_8, \varphi_9$	$R_7, R_8, R_9$
转移规则 $t(s_2, q) = s_6$	$\varphi_7$	$R_7$
转移规则 $t(s_0, r) = s_3$	$\varphi_8$	$R_8$
转移规则 $t(s_0, \text{true}) = s_4$	$\varphi_8, \varphi_9$	$R_8, R_9$
转移规则 $t(s_3, r) = s_3$	$\varphi_8$	$R_8$
转移规则 $t(s_3, r) = s_6$	$\varphi_8$	$R_8$
转移规则 $t(s_4, \text{true}) = s_4$	$\varphi_8, \varphi_9$	$R_8, R_9$
转移规则 $t(s_4, q) = s_6$	$\varphi_8, \varphi_9$	$R_8, R_9$
转移规则 $t(s_0, \text{true}) = s_5$	$\varphi_9$	$R_9$
转移规则 $t(s_5, \text{true}) = s_5$	$\varphi_9$	$R_9$
转移规则 $t(s_5, r) = s_6$	$\varphi_9$	$R_9$

## 4.2 检测 9 个公式的仿真实验

在 4.1 节给出的 DNA 编码的基础上, 可实施仿真实验. 通用工具 NUPACK 充分考虑了计算的生物学实现, 我们以该工具为实验平台, 对系统运行实施仿真检测. 需要注意的是, 在 4.2 节, 所有 DNA 分子的编码从左至右均为 5'-3' 方向显示, 这是为了与 NUPACK 工具保持一致.

在本实验中, 我们将检测图 2 所示的系统模型是否满足 9 个公式. 按照 4.1 节给出的编码, 可得系

统运行的所有路径,如表 6 所示,其中  $k$  为任意自然数.

表 6 系统运行(置于试管 T 中)

路径	路径的 DNA 编码或者路径经过的顶点顺序
1 号 路径的 编码	CGCG AACGCCCTAATTCGCAATCAATGT CAG CTG AACGCCCTAATTCGCAATCAATGT CAG AACGCCCTAATTCGCAATCAATGT CTG AACGCCCTAATTCGCAATCAATGT TCCG
1 号路径 顶点顺序	0,1,2
$k$ 号 路径的 编码	CGCG AACGCCCTAATTCGCAATCAATGT (CAG CTG AACGCCCTAATTCGCAATCAATGT CAG AACGCCCTAATTCGCAATCAATGT) <sup>k</sup> CTG AACGCCCTAATTCGCAATCAATGT TCCG
$k$ 号路径 顶点顺序	(0,1) <sup>k</sup> ,2

首先给出对 1 号路径的检测(不失一般性,设在 1 号路径上的运行经过的原子命题序列为  $ppq$ ).

任意检测实例均以初始状态 CGCGAAC 开头,且以 TGTTCG 结束,因此只需检测  $d = \text{GCCCT-AATTCGCAATCAATGT CAG AACGCCCTAATTCGCAATCAATGT CTG AACGCCCTAATTCGCAAT-}$

CAA 即可. 简而言之,考察对状态转移规则实施编码的 DNA 分子是否与分子  $d$  杂交即可.

#### 4.2.1 对运行 $ppq$ 检测公式 1

公式 1 的 3 种状态转移规则分别对应试管中的 3 种 DNA 分子,编码分别是

$t_{0p0} = \text{GTT CTG ACA TTG ATT GCG AAT TAG GGC};$

$t_{0q6} = \text{TTG ATT GCG AAT TAG GGC GTT CAG ACA TTG ATT GCG AAT TAG GGC};$

$t_{6true6} = \text{TTG ATT GCG AAT TAG GGC GTT CTG | CAG ACA}$ ,其中,“CTG | CAG”表示“CTG”与“CAG”任选其一.

把  $d$ 、 $t_{0p0}$ 、 $t_{0q6}$  和  $t_{6true6}$  这 4 种 DNA 分子置入容积为  $10^{-3}$  L 的试管 1,其浓度分别为  $100 \mu\text{M}$ 、 $200 \mu\text{M}$ 、 $100 \mu\text{M}$ 、 $100 \mu\text{M}$ ,因此分子数量分别为  $10^{16}$ 、 $2 \times 10^{16}$ 、 $10^{16}$ 、 $10^{16}$ ,当温度缓慢降到 10 摄氏度时,仿真杂交结果如图 11 所示. 其中,strand1 表示分子  $d$ ,strand2 表示分子  $t_{0p0}$ ,strand3 表示分子  $t_{0q6}$ ,strand4 表示分子  $t_{6true6}$ .

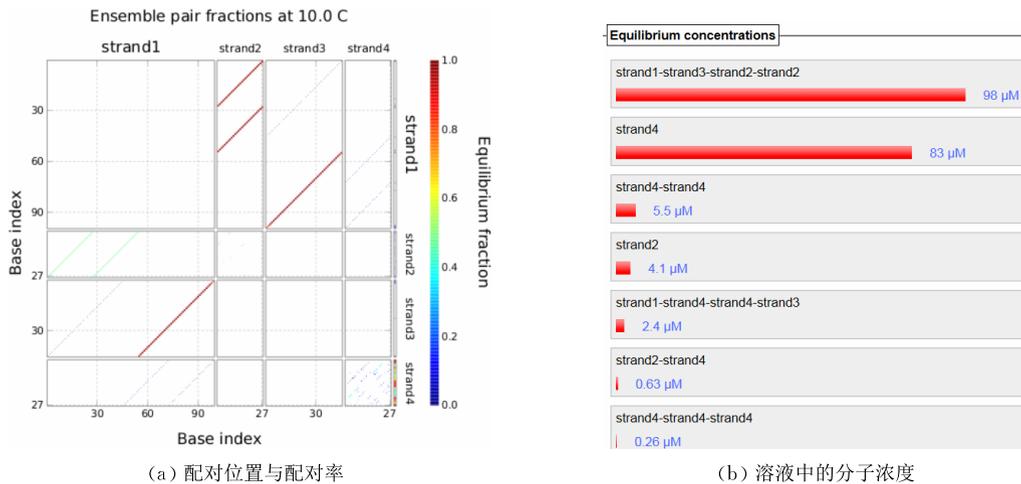


图 11 检测公式 1 的分子杂交反应结果

参见图 11(a). 图中从上至下第 1 条红线的坐标位置表明,分子  $d$  的  $5'$ - $3'$  方向前 27 个位点上的碱基序列与分子  $t_{0p0}$  的  $3'$ - $5'$  方向所有 27 个位点上的碱基序列形成完全互补配对;图中从上至下第 2 条红线的坐标位置表明,分子  $d$  的  $5'$ - $3'$  方向从第 28 至第 54 位点上的碱基序列与分子  $t_{0p0}$  的  $3'$ - $5'$  方向所有 27 个位点上的碱基序列形成完全互补配对;图中从上至下第 3 条红线的坐标位置表明,分子  $d$  的  $5'$ - $3'$  方向从第 55 至第 99 位点上的碱基序列与分子  $t_{0q6}$  的  $3'$ - $5'$  方向所有 45 个位点上的碱基序列形成完全互补配对. 此现象提示杂交具有特异

性. 根据算法 2,完全双链的形成表明运行  $ppq$  满足公式 1.

对比图 11(a)中上述 3 条红线的颜色与该图中右侧的颜色变化竖条,可见 3 条红线的颜色与竖条顶端颜色极为接近,这表明上述碱基序列配对的发生概率接近 1. 此现象提示特异性程度较高.

参见图 11(b).  $98 \mu\text{M}/100 \mu\text{M} = 98\%$  的分子  $d$  参加了与分子  $t_{0p0}$  和分子  $t_{0q6}$  的特异性杂交(图中的 strand1-strand3-strand2-strand2,而其余 6 个均为非特异);分子  $t_{0p0}$  的假阴性率为  $4.1 \mu\text{M}/200 \mu\text{M} = 2.05\%$ ,假阳性率为  $0.63 \mu\text{M}/200 \mu\text{M} = 0.315\%$ ,特

异性杂交真阳性率约等于 98%; 同理可见分子 t0q6 的假阴性率为 0, 假阳性率为 2.4%, 真阳性率约等于 98%, 再次提示杂交的强特异性。

#### 4.2.2 对运行 $ppq$ 检测公式 2

公式 2 的 3 种状态转移规则分别对应试管中的 3 种 DNA 分子, 编码分别是

t0p0 = GTT CTG ACA TTG ATT GCG AAT TAG GGC;

t0p6 = TTG ATT GCG AAT TAG GGC GTT CTG ACA TTG ATT GCG AAT TAG GGC.

把 d、t0p0、t0p6 这 3 种 DNA 分子置入容积为  $10^{-3}$  L 的试管 2, 其浓度分别为  $100 \mu\text{M}$ 、 $200 \mu\text{M}$ 、 $100 \mu\text{M}$ , 因此分子数量分别为  $10^{16}$ 、 $10^{16}$ 、 $10^{16}$ , 当温度缓慢降到 10 摄氏度时, 仿真杂交结果如图 12 所示。其中, strand1 表示分子 d, strand2 表示分子 t0p0, strand3 表示分子 t0p6。

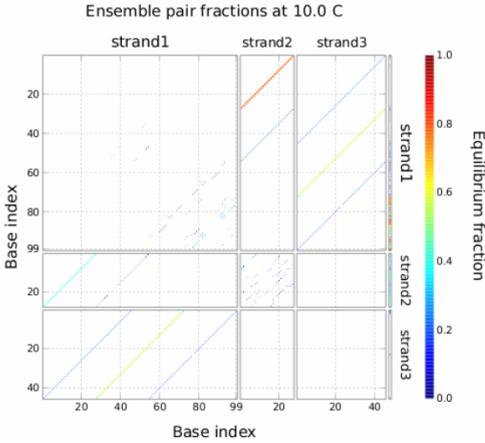


图 12 检测公式 2 的分子杂交结果——配对位置与配对率

参见图 12。图中红线的坐标位置表明, 分子 d 的 5'-3'方向前 27 个位点上的碱基序列与分子 t0p0 的 3'-5'方向所有 27 个位点上的碱基序列形成完全互补配对, 根据颜色可知, 配对概率约为 70%~80%; 图中 strand1 中段位点和 strand2 的交叉矩形区域存在一条蓝线, 图中 strand1 中段位点和 strand3 的交叉矩形区域存在一条蓝线和一条黄线, 这些位置表明一些分子 d 与分子 t0p0 配对而另一些分子 d 与分子 t0p6 配对, 其颜色则表明上述两种配对概率较低。在 strand1 末段位点, 则只存在与 strand3 的蓝线, 配对概率极低, 蓝线中存在较多断点, 表明存在一部分碱基完全无法配对。上述现象提示: 分子 d 的 5 端位点序列形成了部分双链, 而 3 端位点序列仍为单链。根据算法 2, 运行  $ppq$  不满足公式 2。

#### 4.2.3 对运行 $ppq$ 检测公式 3

公式 3 的 3 种状态转移规则分别对应试管中的 3 种 DNA 分子, 编码分别是

t0true0 = GTT CTG | CAG ACA TTG ATT GCG AAT TAG GGC;

t0p6 = TTG ATT GCG AAT TAG GGC GTT CTG ACA TTG ATT GCG AAT TAG GGC;

t6true6 = TTG ATT GCG AAT TAG GGC GTT CTG | CAG ACA.

把 d、t0true0、t0p6、t6true6 这 4 种 DNA 分子置入容积为  $10^{-3}$  L 的试管 3, 其浓度分别为  $100 \mu\text{M}$ 、 $100 \mu\text{M}$ 、 $100 \mu\text{M}$ 、 $100 \mu\text{M}$ , 因此分子数量分别为  $10^{16}$ 、 $10^{16}$ 、 $10^{16}$ 、 $10^{16}$ 。当温度缓慢降到 10 摄氏度时, 仿真杂交结果如图 13 所示。其中, strand1 表示分子 d, strand2 表示分子 t0true0, strand3 表示分子 t0p6, strand4 表示分子 t6true6。

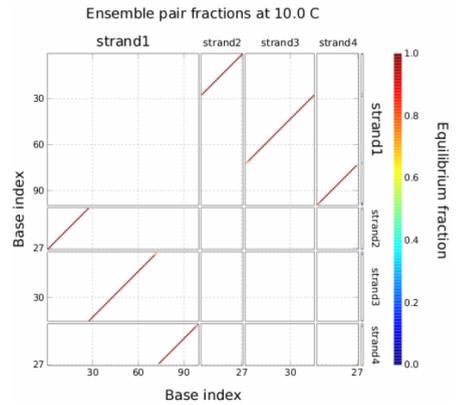


图 13 检测公式 3 的分子杂交结果——配对位置与配对率

参见图 13。从图中可知, 分子 d 的 5 端位点与分子 t0true0 完全配对, 分子 d 的中端位点与分子 t0p6 完全配对, 分子 d 的 3 端位点与分子 t6true6 完全配对, 且 3 条线颜色均为深红, 因此杂交形成完全双链。根据算法 2, 运行  $ppq$  满足公式 3。

#### 4.2.4 对运行 $ppq$ 检测公式 4

公式 4 的 3 种状态转移规则分别对应试管中的 3 种 DNA 分子, 编码分别是

t0true1 = GGC GTT CTG | CAG ACA TTG ATT GCG AAT TAG GGC;

t1p6 = TTG ATT GCG AAT TAG GGC GTT CTG ACA TTG ATT GCG AAT TAG;

t6true6 = TTG ATT GCG AAT TAG GGC GTT CTG | CAG ACA.

把 d、t0true1、t1p6、t6true6 这 4 种 DNA 分子置入容积为  $10^{-3}$  L 的试管 4, 其浓度分别为  $100 \mu\text{M}$ 、

100  $\mu\text{M}$ 、100  $\mu\text{M}$ 、100  $\mu\text{M}$ ，因此分子数量分别为  $10^{16}$ 、 $10^{16}$ 、 $10^{16}$ 、 $10^{16}$ 。当温度缓慢降到 10 摄氏度时，仿真杂交结果如图 14 所示。其中，strand1 表示分子 d，strand2 表示分子 t0true1，strand3 表示分子 t1p6，strand4 表示分子 t6true6。

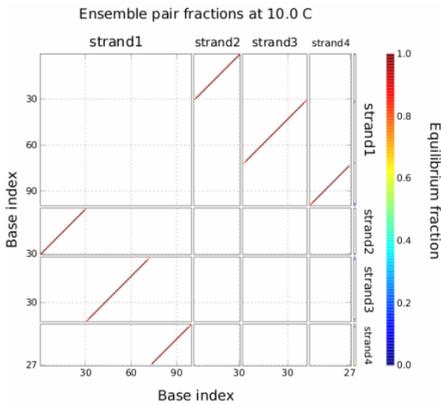


图 14 检测公式 4 的分子杂交结果——配对位置与配对率

参见图 14。从图中可知，分子 d 的 5 端位点与分子 t0true1 完全配对，分子 d 的中端位点与分子 t1p6 完全配对，分子 d 的 3 端位点与分子 t6true6 完全配对，且 3 条线颜色均为深红，因此杂交形成完全双链。根据算法 2，运行  $ppq$  满足公式 4。

#### 4.2.5 对运行 $ppq$ 检测公式 5

公式 5 的 3 种状态转移规则分别对应试管中的 3 种 DNA 分子，编码分别是

t0r0=GTT AAA ACA TTG ATT GCG AAT TAG GGC;

t0r6=TTG ATT GCG AAT TAG GGC GTT AAA ACA TTG ATT GCG AAT TAG GGC.

把 d、t0r0、t0r6 这 3 种 DNA 分子置入容积为  $10^{-3}$  L 的试管 5，其浓度分别为 100  $\mu\text{M}$ 、100  $\mu\text{M}$ 、100  $\mu\text{M}$ ，因此分子数量分别为  $10^{16}$ 、 $10^{16}$ 、 $10^{16}$ 。当温度缓慢降到 10 摄氏度时，仿真杂交结果如图 15 所

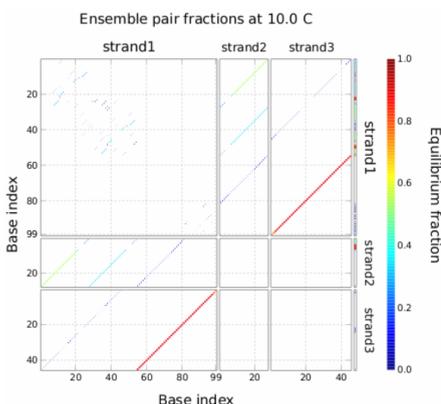


图 15 检测公式 5 的分子杂交结果——配对位置与配对率

示。其中，strand1 表示分子 d，strand2 表示分子 t0r0，strand3 表示分子 t0r6。

参见图 15。分子 d 的 5 端位点几乎不与其他分子配对，分子 d 的 3 端位点与分子 t0r6 完全配对，因此杂交未能形成完全双链。根据算法 2，运行  $ppq$  不满足公式 5。

#### 4.2.6 对运行 $ppq$ 检测公式 6

由于公式 6 等于公式 3，因此检测结果与公式 3 完全相同。

#### 4.2.7 对运行 $ppq$ 检测公式 7

公式 7 的 4 种状态转移规则分别对应试管中的 4 种 DNA 分子，编码分别是

t0true2 = TAG GGC GTT CTG | CAG ACA TTG ATT GCG AAT TAG GGC;

t0q|r6 = TTG ATT GCG AAT TAG GGC GTT CAG | AAA ACA TTG ATT GCG AAT TAG GGC;

t2true2 = TAG GGC GTT CTG | CAG ACA TTG ATT GCG AAT;

t2q6 = TTG ATT GCG AAT TAG GGC GTT CAG ACA TTG ATT GCG AAT.

只需把 d、t0true2、t0q|r6、t2true2、t2q6 这 5 种 DNA 分子置入容积为  $10^{-3}$  L 的试管 7，然后按照上文所述的实验流程操作，即可获得结果，即运行  $ppq$  满足公式 7。

为了更清楚地观察结果，我们微调实验流程，实施 4 组实验，每组实验的试管 7 中均置入 100  $\mu\text{M}$  的分子 d，分子 t0true2 不在第 1 组实验的试管 7 中出现、分子 t0q|r6 不在第 2 组实验的试管 7 中出现，依次类推。运行经过 3 个状态，因此使用的转移规则不超过 3 种。从 4 种转移规则中挑出 3 种是一个组合问题，共有 4 种挑法，分别对应于如上所述的 4 组实验。因此这 4 组实验是完备的，得出的结果是充分的。

第 1 组实验结果参见图 16(a)。最上方红线存在断点，因此 strand2 和 strand1 并未完全配对。可见若无分子 t0true2 参加则无法生成完全双链。

第 2 组实验结果参见图 16(b)。分子 d (strand1) 的 5 端位点与分子 t0true2 (strand2) 完全配对，分子 d 的中端位点与分子 t2true2 (strand3) 完全配对，分子 d 的 3 端位点与分子 t2q6 (strand4) 完全配对，且 3 条线颜色均为深红，因此杂交形成完全双链。

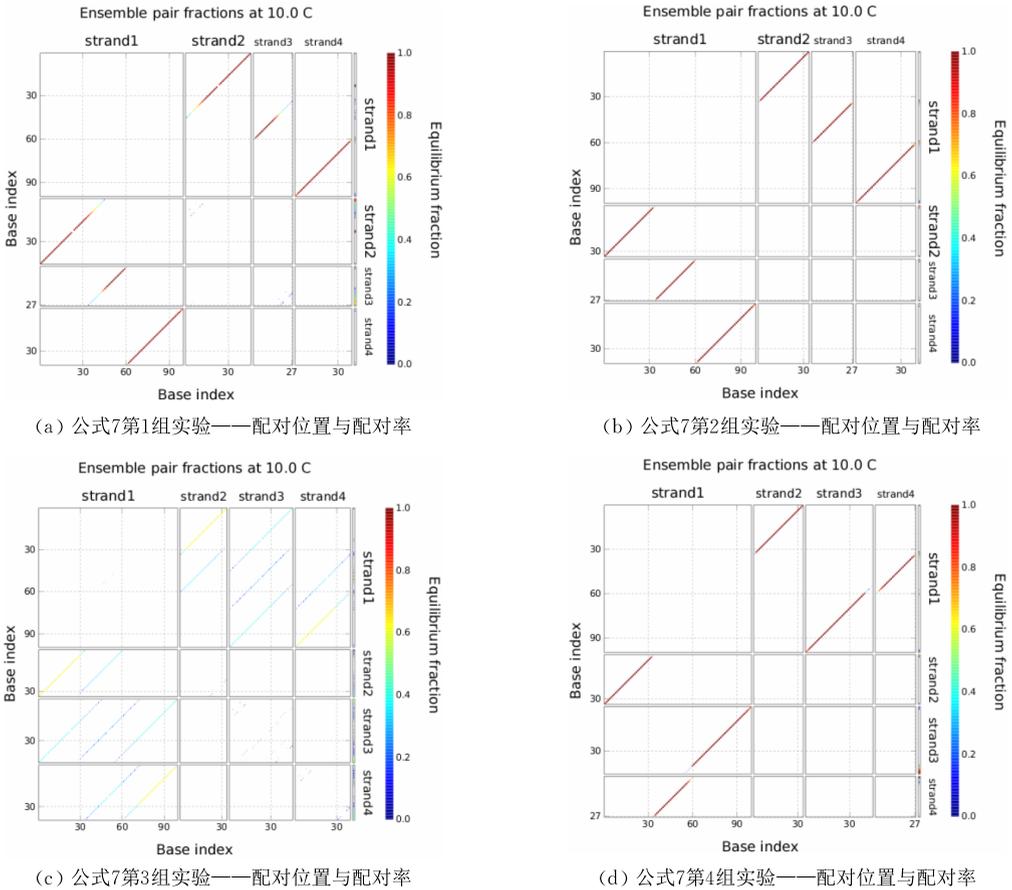


图 16 公式 7 的多组实验

第 3 组实验结果参见图 16(c). 从图中可见, 杂交极少发生, 因此未形成完全双链.

第 4 组实验结果参见图 16(d). 把右侧两个红线向 strand1 轴上做投影, 可发现投影存在断点, strand1 的位点 60 与 strand4 的位点 1 仅有 1% 的配对概率, 从而导致两种分子未能完全杂交, 因此未形成完全双链.

根据以上 4 组实验, 分子 d、分子 t0true2、分子 t2true2、分子 t2q6 可形成完全双链. 因此可知, 运行 ppq 满足公式 7.

4. 2. 8 对运行 ppq 检测公式 8

公式 8 的 7 种状态转移规则分别对应试管中的 7 种 DNA 分子, 编码分别是

t0r3 = AAT TAG GGC GTT AAA ACA TTG ATT GCG AAT TAG GGC;

t0true4 = GCG AAT TAG GGC GTT CTG | CAG ACA TTG ATT GCG AAT TAG GGC;

t0q | r6 = TTG ATT GCG AAT TAG GGC GTT CAG | AAA ACA TTG ATT GCG AAT TAG GGC;

t3r3 = AAT TAG GGC GTT AAA ACA TTG ATT GCG;

t3r6 = TTG ATT GCG AAT TAG GGC GTT AAA ACA TTG ATT GCG;

t4true4 = GCG AAT TAG GGC GTT CTG | CAG ACA TTG ATT;

t4q6 = TTG ATT GCG AAT TAG GGC GTT CAG ACA TTG ATT.

把分子 d 置入容积为  $10^{-3}$  L 的试管 8, 浓度为  $100 \mu\text{M}$ . 从 t0r3、t0true4、t0q | r6、t3r3、t3r6、t4true4、t4q6 这 7 种分子中挑出 3 种, 有 35 种挑法, 分别对应 35 组实验. 其中, 只有如下一组可获得完全双链: 把 t0true4、t4true4、t4q6 置入试管 8, 浓度分别为  $100 \mu\text{M}$ 、 $100 \mu\text{M}$ 、 $100 \mu\text{M}$ .

实验结果参见图 17. 分子 d(strand1) 的 5 端位点与分子 t0true4(strand2) 完全配对, 分子 d 的中端位点与分子 t4true4(strand3) 完全配对, 分子 d 的 3 端位点与分子 t4q6(strand4) 完全配对, 且 3 条线的颜色均为深红, 因此杂交形成完全双链. 根据算法 2, 运行 ppq 满足公式 8.

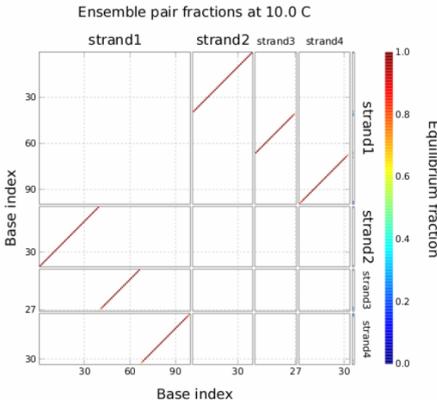


图 17 公式 8 成功杂交组实验——配对位置与配对率

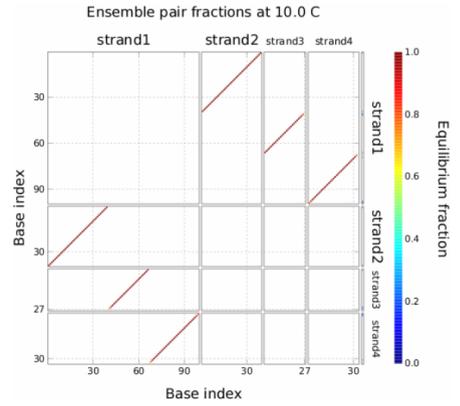


图 18 公式 9 成功杂交组实验——配对位置与配对率

4.2.9 对运行 *ppq* 检测公式 9

公式 9 的 7 种状态转移规则分别对应试管中的 7 种 DNA 分子, 编码分别是

t0true4 = GCG AAT TAG GGC GTT CTG | CAG ACA TTG ATT GCG AAT TAG GGC;

t0true5 = ATT GCG AAT TAG GGC GTT CTG | CAG ACA TTG ATT GCG AAT TAG GGC;

t0q | r6 = TTG ATT GCG AAT TAG GGC GTT CAG | AAA ACA TTG ATT GCG AAT TAG GGC;

t4true4 = GCG AAT TAG GGC GTT CTG | CAG ACA TTG ATT;

t4q6 = TTG ATT GCG AAT TAG GGC GTT CAG ACA TTG ATT;

t5true5 = ATT GCG AAT TAG GGC GTT CTG | CAG ACA TTG;

t5r6 = TTG ATT GCG AAT TAG GGC GTT AAA ACA TTG.

把分子 d 置入容积为  $10^{-3}$  L 的试管 9, 浓度为  $100 \mu\text{M}$ . 从 t0true4、t0true5、t0q | r6、t4true4、t4q6、t5true5、t5r6 这 7 种分子中挑出 3 种, 有 35 种挑法, 分别对应 35 组实验. 其中, 只有如下一组可获得完全双链: 把 t0true4、t4true4、t4q6 置入试管 9, 浓度分别为  $100 \mu\text{M}$ 、 $100 \mu\text{M}$ 、 $100 \mu\text{M}$ .

实验结果参见图 18. 分子 d (strand1) 的 5 端位点与分子 t0true4 (strand2) 完全配对, 分子 d 的中端位点与分子 t4true4 (strand3) 完全配对, 分子 d 的 3 端位点与分子 t4q6 (strand4) 完全配对, 且 3 条线的颜色均为深红, 因此杂交形成完全双链. 根据算法 2, 运行 *ppq* 满足公式 9.

以上为检测 1 号路径是否满足 9 个公式. 同理, 可检测 *k* 号路径是否满足 9 个公式. 按照附录给出的一系列定理, 计算不同公式所需检测的路径长度与路径 (运行) 数, 结果如表 7 所示. 针对每一个公式, 按照前文所述的实验过程, 分别检测它所需检测的所有路径, 获得的结果如表 8 的第 2~5 列所示. 如果某公式所需检测的所有路径均满足该公式, 则系统满足该公式, 结果见表 8 的最后一列.

表 7 不同公式需检测的路径长度与路径数

公式	需检测的路径长度	需检测的路径数
公式 1	$l_1 =  V  \times 2^{ V -1} +  E  = 3 \times 2^{3-1} + 3 = 15$	7
公式 2	$l_2 =  V  = 3$	1
公式 3	$l_3 =  V  \times 2^{ V -1} +  E  = 3 \times 2^{3-1} + 3 = 15$	7
公式 4	$l_4 = 2$	1
公式 5	$l_5 =  V  = 3$	1
公式 6	$l_6 =  V  \times 2^{ V -1} +  E  = 3 \times 2^{3-1} + 3 = 15$	7
公式 7	$l_7 =  V  \times 2^{ V -1} +  V  +  E  = 3 \times 2^{3-1} + 3 + 3 = 18$	8
公式 8	$l_8 =  V  \times 2^{ V -1} +  V  +  E  = 3 \times 2^{3-1} + 3 + 3 = 18$	8
公式 9	$l_9 =  V  \times 2^{ V -1} +  V  +  E  = 3 \times 2^{3-1} + 3 + 3 = 18$	8

表 8 对不同路径检测 9 个公式的结果 (是否被满足)

公式	1 号路径	$k > 1$ 号路径	7 号路径	8 号路径	系统是否满足公式
公式 1	是	是	是	无需检测	是
公式 2	否	无需检测	无需检测	无需检测	否
公式 3	是	是	是	无需检测	是
公式 4	是	无需检测	无需检测	无需检测	是
公式 5	否	无需检测	无需检测	无需检测	否
公式 6	是	是	是	无需检测	是
公式 7	是	是	是	是	是
公式 8	是	是	是	是	是
公式 9	是	是	是	是	是

### 4.3 相关工作比较

表 9 给出了新方法及相关方法的能力比较, 对于表中列出的公式类型,  $EFp$  是一个 CTL 公式, 本文涉及的其他公式均为 LTL 公式.

表 9 模型检测的不同 DNA 方法之实际检测能力比较

公式类型	本文方法	文献[35]中方法	文献[36]中方法
$EFp$	不能检测	能检测	不能检测
$p \triangleright q$	可直接检测	不能检测	能检测
$\square p$	可直接检测	不能检测	方法可检测, 编码限制导致实际不可检测
$\diamond p$	可直接检测	不能检测	方法可检测, 编码限制导致实际不可检测
$\bigcirc p$	可直接检测	不能检测	不能检测
安全性公式	可直接检测	不能检测	不能检测
可达性公式	可直接检测	不能检测	不能检测
活性公式	可直接检测	不能检测	不能检测
公平性公式	可直接检测	不能检测	不能检测
其他情况	理论可检测	不能检测	不能检测

文献[35]给出了检测  $EFp$  的 DNA 方法, 并未给出检测其他 CTL 公式的 DNA 方法, 更不涉及 LTL 公式模型检测的 DNA 方法. 作为对照, 本文研究的正是使用 DNA 分子对 LTL 公式实施模型检测.

我们的前期工作给出了可直接检测 LTL 基本公式  $p \triangleright q$  的 DNA 方法<sup>[36]</sup>. 由于  $\square p$ 、 $\diamond p$  等 LTL 基本公式可由  $p \triangleright q$  表达, 因此文献[36]中的方法在理论上也可用于检测前两个公式. 然而, 由于 DNA 编码位数的限制, 文献[36]中的编码序列没能对上述两个公式实施编码, 导致两公式实际无法检测. 作为对照, 新方法则可以实施此项检测.

文献[36]中的方法专用于检测公式  $p \triangleright q$ , 而  $\bigcirc p$ 、安全性、可达性、活性、公平性以及其它 LTL 公式均不能被  $p \triangleright q$  表示, 因此它们均不能被文献[36]中的方法检测. 作为对照, 新方法则可以直接检测  $\bigcirc p$ 、安全性、可达性、活性、公平性等公式.

以上的检测能力比较是建立在生物学实现被证实的前提之下. 文献[40]指出, 粘贴自动机在理论上可对有  $m$  个节点  $n$  条边的有穷状态自动机的语言接受问题实施判定, 其中  $n$  和  $m$  均为自然数(所谓理论上的能力, 是指在逻辑层面已得到肯定的证明, 而在生物学层面目前尚未得到实验证实因而能否生物实现尚未可知). 新方法的核心思想在于: 把基于 DNA 计算的 LTL 模型检测问题规约到基于粘贴自动机的有穷状态自动机接受语言问题. 据此我们可知, 新方法在理论上可对 LTL 一般公式实施模型检测, 只需扩展本文 DNA 编码位数即可.

对本文编码方案与文献[36]中编码方案进行理论上的比较. 计算结果参见图 19, 本文编码方案能

对图模型规模在 7 个节点 42 条边之内的 LTL 公式实施检测, 而文献[36]中的编码方案最多只能对图模型规模在 3 个节点 6 条边之内的 LTL 公式实施检测. 如果对新方案进一步实施位数扩展, 可获得更强的理论检测能力, 直至实现对任意 LTL 公式的检测. 这是因为编码方案决定了  $m$  位编码可检测图模型具有  $m$  个节点的公式, 当  $m$  足够大时, 即可覆盖任意公式.

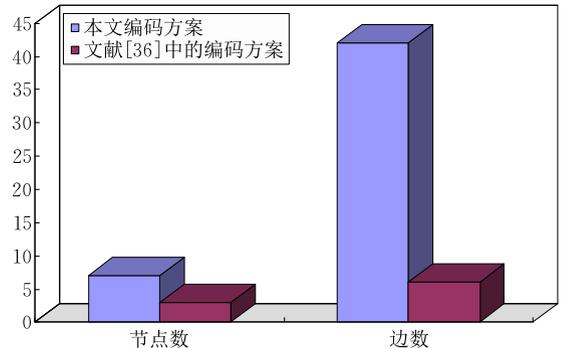


图 19 不同 DNA 编码方案带来的理论检测能力比较 (理论上可对如此规模的公式实施检测: 公式的图模型中节点数与边数分别不超过  $m$  和  $n$ )

## 5 结 论

本文给出了针对 LTL 模型检测的 DNA 方法. 我们的仿真实验以 DNA 分子为载体对 LTL 逻辑的全部基本时序公式与实际应用中常见的 LTL 公式实施了有效检测. 特别地, 新方法实现了使用 DNA 分子对安全性公式进行检测——“在实际的系统中, 超过 90% (很可能超过 95%) 的错误是因为违背了安全性”, 图灵奖得主 Lamport 教授指出<sup>①</sup>.

此外, 执行算法 2 (至少在理论上) 可实现对 LTL 任意给定公式  $f$  进行检测: 首先, 对本文实验中的 DNA 编码方案直接进行位数扩展, 引入更多的间隔区  $X_i$ , 当位数  $\max(i)$  扩展到  $n$  位时, 即可对其图模型节点数在  $n$  个以下的 LTL 公式进行编码; 其次, 按照位数扩展后的编码方案对  $f$  的自动机模型进行 DNA 编码, 并作为本文算法的一个输入; 再次, 系统模型的 DNA 编码作为本文算法的另一个输入; 最后, 运行算法即可获得检测结果.

基于 DNA 计算的方法可利用其巨大的天然的并行计算优势缓解 LTL 模型检测当前面临的状态空间爆炸这一瓶颈, 虽然这是一种以海量编码分子

① Liveness Manifestos. [www.cs.nyu.edu/acsys/beyond-safety/liveness.htm](http://www.cs.nyu.edu/acsys/beyond-safety/liveness.htm)

并行计算换取海量组合搜索串行计算的方案. 这是新方法在计算机科学中的潜在应用价值.

在细胞内进行的 LTL 模型检测计算, 将使得对肿瘤分子标志发展过程的动态识别与动态地自动调整药物用法与用量成为可能. 特别是文献[41]给出的方法只能对肿瘤分子标志是否满足“条件 1 并且条件 2”、“条件 3 或者条件 4”等静态条件进行判断, 进而由细胞内计算控制给药; 在细胞内进行的 LTL 模型检测计算则可对诸如“某时刻满足条件 1, 后来又满足条件 2”、“曾经满足条件 3”、“始终满足条件 4”等表征肿瘤发生、发展动态变化的条件实施检测, 以自动控制给药或者调整用药, 从而有望为人类重大基因突变疾病的分子诊疗、基因修复提供更好的自治智能方法. 这是新方法在分子生物学与医学中的潜在应用价值.

### 参 考 文 献

- [1] Clarke E, et al. Model Checking. Massachusetts: MIT Press, 1999
- [2] Barnat J, Bauch P, Brim L, et al. Designing fast LTL model checking algorithms for many-core GPUs. Journal of Parallel and Distributed Computing, 2012, 72(9): 1083-1097
- [3] Carbone R. LTL model-checking for security protocols. AI Communications, 2011, 24(4): 281-283
- [4] Classen A, Cordy M, Schobbens P. Featured transition systems: Foundations for verifying variability-intensive systems and their application to LTL model checking. IEEE Transactions on Software Engineering, 2013, 39(8): 1069-1089
- [5] Pnueli A. The temporal logic of programs//Proceedings of the 18th Annual Symposium on Foundations of Computer Science. Washington, USA, 1977: 46-57
- [6] Benari M, Pnueli A, Manna Z. The temporal logic of branching time. Acta Informatica, 1983, 20(3): 207-226
- [7] Emerson E, Clarke E. Using branching time temporal logic to synthesize synchronization skeletons. Science of Computer Programming, 1982, 2(3): 241-266
- [8] Adleman L. Molecular computation of solutions to combinatorial problems. Science, 1994, 266(5187): 1021-1023
- [9] Lipton R. DNA solution of hard computational problems. Science, 1995, 268(5210): 542-545
- [10] Ouyang Q, Kaplan P D, Liu S, et al. DNA solution of the maximal clique problem. Science, 1997, 278(17): 446-449
- [11] Shapiro E, Benenson Y, Adar R, et al. Programmable and autonomous computing machine made of biomolecules. Nature, 2001, 414(6862): 430-434
- [12] Adleman L. On constructing a molecular computer//Proceedings of the Conference on DNA Based Computers. Princeton University, Princeton, New Jersey, USA, 1995
- [13] Roweis S, Winfree E, Burgoyne R, et al. A sticker-based model for DNA computation. Journal of Computational Biology, 1998, 5(4): 615-629
- [14] Kuramochi J, Sakakibara Y. Intensive in vitro experiments of implementing and executing finite automata in test tube//Proceedings of the 11th International Workshop on DNA Computing. Lecture Note in Computer Science. Berlin Heidelberg, Germany: Springer Press, 2006: 193-202
- [15] Zimmermann K, Zhang G, Ignatova Z, et al. Computational genes: A tool for molecular diagnosis and therapy of aberrant mutational phenotype. BMC Bioinformatics, 2007, 8(1): 365
- [16] Peng Y, Andrew J, Sudheer S, et al. Design of an autonomous DNA nanomechanical device capable of universal computation and universal translational motion//Proceedings of the 10th International Meeting on DNA Computing. Lecture Note in Computer Science 3384. Milan, Italy, 2005: 426-444
- [17] Xu J, Qiang X, Yang Y, et al. An unenumerative DNA computing model for vertex coloring problem. IEEE Transactions on Nanobioscience, 2011, 10(2): 94-98
- [18] Xiao Jian-Hua, Xu Jin. The DNA computation model based on giant magnetoresistance for SAT problem. Chinese Journal of Computers, 2013, 36(4): 829-835(in Chinese)  
(肖建华, 许进. 可满足性问题的巨磁电阻型 DNA 计算模型. 计算机学报, 2013, 36(4): 829-835)
- [19] Zhang Cheng, Ma Li-Na, Dong Ya-Fei, et al. Molecular logic computing model based on DNA self-assembly strand branch migration. Chinese Science Bulletin, 2012, 57(31): 2909-2915(in Chinese)  
(张成, 马丽娜, 董亚非等. 自组装 DNA 链置换分子逻辑计算模型. 科学通报, 2012, 57(31): 2909-2915)
- [20] Zhang Cheng, Yang Jing, Xu Jin, Molecular logic computing model based on self-assembly of DNA nanoparticles. Chinese Science Bulletin, 2011, 56(33): 3566-3571(in Chinese)  
(张成, 杨静, 许进. 自组装 DNA/纳米颗粒分子逻辑计算模型. 科学通报, 2011, 56(27): 2276-2282)
- [21] Li Ken-Li, Luo Xing, Wu Fan, et al. An algorithm in tile assembly model for maximum clique problem. Journal of Computer Research and Development, 2013, 50(3): 666-675 (in Chinese)  
(李肯立, 罗兴, 吴帆等. 基于自组装模型的最大团问题 DNA 计算算法. 计算机研究与发展, 2013, 50(3): 666-675)
- [22] Xu J, Qiang X, Zhang K, et al. A parallel type of DNA computing model for graph vertex coloring problem//Proceedings of the IEEE 5th International Conference on Bio-Inspired Computing: Theories and Applications. Changsha, China, 2010: 231-235
- [23] Wu Fan, Li Ken-Li. An algorithm in tile assembly model for N queen problem. Acta Electronica Sinica, 2013, 41(11): 2174-2180(in Chinese)  
(吴帆, 李肯立. 基于自组装的 N 皇后问题 DNA 计算算法. 电子学报, 2013, 41(11): 2174-2180)

- [24] Zhou Xu, Li Ken-Li, Le Guang-Xue, et al. A volume molecular solution for the maximum matching problem on DNA-based computing. *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. *计算机研究与发展*, 2011, 48(11): 2147-2154)
- [25] Wu F, Li K, Sallam A, Zhou X. A molecular solution for minimum vertex cover problem in tile assembly model. *The Journal of Supercomputing*, 2013, 66(1): 148-169
- [26] Zhou X, Zhou Y, Li K, et al. Molecular solutions for minimum and exact cover problems in the tile assembly model. *The Journal of Supercomputing*, 2014, 69(2): 976-1005
- [27] Li Ken-Li, Yao Feng-Juan, Xu Jin, et al. An  $O(1.414n)$  volume molecular solutions for the subset-sum problem on DNA-based supercomputing. *Chinese Journal of Computers*, 2007, 30(11): 1947-1953(in Chinese)  
(李肯立, 姚凤娟, 许进等. 子集和问题的  $O(1.414n)$  链数 DNA 计算机算法. *计算机学报*, 2007, 30(11): 1947-1953)
- [28] Zhou X, Li K, Goodman M, et al. A novel approach for the classical ramsey number problem on DNA-based supercomputing. *Communications in Mathematical and in Computer Chemistry*, 2011, 66(1): 347-370
- [29] Yang J, Dong C, Dong Y, et al. Logic nanoparticle beacon triggered by the binding induced effect of multiple inputs. *ACS Applied Materials & Interfaces*, 2014, 6(16): 14 486-14 492
- [30] Zhang C, Wu L, Yang J, et al. A molecular logical switch beacon controlled by thiolated DNA signals. *Chemical Communications*, 2013, 49(96): 11 308-11 310
- [31] Zhang C, Ma J, Yang J. Nanoparticle aggregation logic computing controlled by DNA branch migration. *Applied Physics Letter*, 2013, 103(9): 93-106
- [32] Chen Mei, Chen Xiang-Qun, Zhang Lu, Xu Jin. A biobrick inversion cellular computing model for satisfiability problem. *Chinese Journal of Computers*, 2013, 36(12): 2537-2544(in Chinese)  
(陈梅, 陈向群, 张路, 许进. 可满足性问题生物砖翻转细胞计算模型. *计算机学报*, 2013, 36(12): 2537-2544)
- [33] Chen M, Xu J. Construction of a genetic conditional learning system in *Escherichia coli*. *Science China: Information Sciences*, 2015, 58(11): 112107:1-112107:6
- [34] Xu Jin. Forthcoming era of biological computer. *Bulletin of the Chinese Academy of Sciences*, 2014, 29(1): 42-54(in Chinese)  
(许进. 生物计算机时代即将来临. *中国科学院院刊*, 2014, 29(1): 42-54)
- [35] Emerson E, Hager K, Konieczka J. Molecular model checking. *International Journal of Foundations of Computer Science*, 2006, 17(4): 733-741
- [36] Zhu Wei-Jun, Zhou Qing-Lei, Li Yong-Liang. LTL model checking based on DNA computing. *Acta Electronica Sinica*, 2016, 44(6): 1265-1271(in Chinese)  
(朱维军, 周清雷, 李永亮. 以 DNA 为载体的线性时序逻辑模型检测. *电子学报*, 2016, 44(6): 1265-1271)
- [37] Zimmermann K, Ignatova Z, Perez M. *DNA Computing Models*. USA: Springer Press, 2008
- [38] Qiao Zhu. *Molecular Computing Using DNA Length Coding* [M. S. dissertation]. University of Electronic Science and Technology of China, Chengdu, 2010(in Chinese)  
(乔柱. 利用 DNA 长度编码实现分子计算[硕士学位论文]. 电子科技大学, 成都, 2010)
- [39] Wang Yan-Feng, Cui Guang-Zhao. *The Design and Optimization of DNA Coding Sequence*. Beijing: Publishing House of Electronics Industry, 2013(in Chinese)  
(王延峰, 崔光照. DNA 编码序列的设计与优化. 北京: 电子工业出版社, 2013)
- [40] Perez M, Zimmermann K H, Ignatova Z. An autonomous DNA model for finite state automata. *International Journal of Bioinformatics Research and Applications*, 2009, 5(1): 81-96
- [41] Zimmermann K, Ignatova Z, Perez M. Patent US20090018809. *Computer gene*, USA, 2009

## 附录. 若干性质的证明.

公式模型含有穷路径和无穷路径, DNA 计算只考虑有穷的情况. 本附录给出的一系列命题确保了本文给出的 DNA 方法只需在系统模型的一定长度之内的路径上对公式实施检测, 需要检测的路径长度因具体公式而有所不同.

**定义 8.** 顶点集不等的初级回路. 假设  $r_1, r_2$  是图  $G = (V, E)$  中的两个不同的初级回路, 顶点集合  $R_i = \{v \mid v \text{ 是 } r_i \text{ 中的顶点}\}, i = 1, 2$ . 如果  $R_1 \neq R_2$ , 那么称  $r_1, r_2$  为两个顶点集不等的初级回路; 否则, 称  $r_1$  和  $r_2$  是两个顶点集相等的初级回路.

**命题 1.** 假设顶点序列  $r_1 = uvwx y$  是系统模型  $M$  的一个运行, 子序列  $v = v_1 \cdots v_m \cdots v_i$  是  $M$  中的一个初级回路(圈), 子序列  $x = x_j \cdots x_k \cdots x_j$  也是  $M$  中的一个初级回路(圈),

记顶点序列  $r_2 = uvwx y$ . 如果  $v$  和  $x$  是两个顶点集相等的初级回路, 那么以下结论成立: 若  $r_2 \models p \triangleright q$ , 则  $r_1 \models p \triangleright q$ .

证明.

(1)  $(r_2 = uvwx y) \models p \triangleright q$ , 在  $r_2$  中, 如果原子命题  $q$  在顶点  $x_j$  之前第一次被满足, 那么, 在  $x_j$  之后插入任意的顶点序列不影响  $r_2$  对公式  $p \triangleright q$  的满足性, 因此, 此时, 若  $r_2 \models p \triangleright q$ , 则  $r_1 \models p \triangleright q$ .

(2)  $(r_2 = uvwx y) \models p \triangleright q$ , 在  $r_2$  中, 如果原子命题  $q$  在顶点  $x_j$  之后第一次被满足, 那么对于所有的顶点  $a \in V$ , 有  $a \models p$ ; 因为  $V = X$ , 所以对于所有的顶点  $b \in X$ , 有  $b \models p$ , 因此将  $r_2$  中的顶点  $x_j$  替换为子序列  $x$  不影响  $r_2$  对  $p \triangleright q$  的满足性. 证毕.

**命题 2.** 假设顶点序列  $r$  是系统模型  $M$  的一个运行,  $r$  中存在  $n(n \geq 2)$  个初级回路  $r_i (2 \leq i \leq n)$ , 如果对于任意的  $2 \leq i, j \leq n (i \neq j)$ ,  $r_i$  和  $r_j$  都是两个顶点集不等的初级回路, 那么对于任意的符合上述条件的  $r$ , 有  $|r| < |V| \times 2^{|\mathcal{V}| - 1} + |E|$ .

证明.

(1) 在  $M$  是一个有向完全图的情况下,  $r$  最多经过  $C(|V|, 1) + C(|V|, 2) + \dots + C(|V|, |V|) = 2^{|\mathcal{V}| - 1}$  个顶点集不等的初级回路, 它们的长度分别是  $1, 2, \dots, |V|$ , 此时  $r$  的长度  $|r| = C(|V|, 1) + 2C(|V|, 2) + \dots + |V|C(|V|, |V|) < |V| \times 2^{|\mathcal{V}| - 1}$ , 如果  $r$  再经过  $|E|$  条边, 那么  $r$  一定会再经过一条初级回路, 并且经过的这条回路与以前经过的  $|V|$  个初级回路中的某个是顶点集相等的, 与假设条件矛盾. 因此  $|r| < |V| \times 2^{|\mathcal{V}| - 1} + |E|$ .

(2) 在  $M$  是一个有向完全图的情况下, 如果  $r$  只经过所有初级回路中的一部分, 此种情况  $r$  的长度  $|r| < |V| \times 2^{|\mathcal{V}| - 1} + |E|$ .

(3) 如果  $M$  是一个有向图, 但不是完全的, 那么任意一个运行  $r$  经过的满足命题假设的初级回路数  $n < 2^{|\mathcal{V}| - 1}$ , 此时  $r$  的长度  $|r| < |V| \times 2^{|\mathcal{V}| - 1}$ , 如果  $r$  再经过  $|E|$  条边, 那么  $r$  一定会再经过一条初级回路, 并且经过的这条回路与以前经过的  $n$  个初级回路中的某个是顶点集相等的, 与假设条件矛盾. 因此  $|r| < |V| \times 2^{|\mathcal{V}| - 1} + |E|$ . 证毕.

**命题 3<sup>[4]</sup>.** 一个系统满足一个指定的 PLTL 公式, 当且仅当系统的所有运行都被这个 PLTL 公式对应的有限状态自动机接受.

**命题 4.** 假设系统的有限状态自动机模型  $M$  中有  $|E|$  条迁移,  $r$  是由初始状态到终止状态的运行, 记运行的集合

$$X = \{r \mid |r| < |V| \times 2^{|\mathcal{V}| - 1} + |E|\},$$

$$Y = \{r \mid |r| \geq |V| \times 2^{|\mathcal{V}| - 1} + |E|\}.$$

如果  $X$  中的所有运行都满足  $p \triangleright q$ , 那么  $Y$  中的所有运行也满足  $p \triangleright q$ .

证明. 假设  $r$  是  $Y$  中的任意一个运行, 那么  $r$  一定经过初级回路, 由命题 2 可知,  $r$  一定经过两个或两个以上的顶点集相等的初级回路, 去掉重复出现的顶点集相等的初级回路后得到运行  $s$ , 由命题 2 可知,  $s \in X$ , 如果  $X$  中的所有运行都满足  $p \triangleright q$ , 那么  $s \models p \triangleright q$ , 由命题 1 可知,  $r \models p \triangleright q$ . 证毕.

**命题 5.** 一个系统满足 PLTL 公式  $\varphi_1 = p \triangleright q$ , 当且仅当长度小于  $|V| \times 2^{|\mathcal{V}| - 1} + |E|$  的所有运行满足  $\varphi_1$ .

证明.

⇒ 如果一个系统满足 PLTL 公式  $p \triangleright q$ , 那么长度小于  $|V| \times 2^{|\mathcal{V}| - 1} + |E|$  的运行满足  $p \triangleright q$ .

⇐ 如果长度小于  $|V| \times 2^{|\mathcal{V}| - 1} + |E|$  的运行满足  $p \triangleright q$ , 由命题 3 可知, 长度大于  $|V| \times 2^{|\mathcal{V}| - 1} + |E|$  的运行也满足  $p \triangleright q$ , 也就是说, 系统的所有运行都满足  $p \triangleright q$ . 证毕.

由命题 5 可知, 要验证一个系统是否满足 PLTL 公式  $p \triangleright q$ , 只要验证系统的长度小于  $|V| \times 2^{|\mathcal{V}| - 1} + |E|$  的所有运行是否满足  $p \triangleright q$  就行了.

**命题 6.** 假设顶点序列  $r_1 = uvwx_y$  是系统模型  $M$  的

一个运行, 子序列  $v = v_i \dots v_m \dots v_i$  是  $M$  中的一个初级回路(圈), 子序列  $x = x_j \dots x_k \dots x_j$  也是  $M$  中的一个初级回路(圈), 记顶点序列  $r_2 = uvwx_y$ . 如果  $v$  和  $x$  是两个顶点集相等的初级回路, 那么以下结论成立: 若  $r_2 \models \diamond p$ , 则  $r_1 \models p$ .

证明.

(1) 若  $p$  在  $x_j$  之前或  $x_j$  第一次被满足, 则有子序列  $uvwx_j \models \diamond p$ , 因此  $r_1 = uvwx_j \dots x_k \dots x_j y \models \diamond p$ .

(2) 若  $p$  在  $x_j$  之后第一次被满足, 则有子序列  $v$  上任意顶点均满足 true; 由于  $v$  与  $x$  顶点集相同, 因此  $x$  上任意顶点均满足 true, 即子序列  $x \models \text{true}$ ; 根据前提假设有子序列  $uvw \models \text{true}$ , 因此  $uvwx \models \text{true}$ ; 根据前提假设又有  $y \models \diamond p$ , 因此  $r_1 = uvwx_y \models \diamond p$ . 证毕.

**推论 1.** 一个带标签的有穷状态自动机满足 PLTL 公式  $\varphi_3 = \varphi_6 = \diamond p$ , 当且仅当长度小于  $|V| \times 2^{|\mathcal{V}| - 1} + |E|$  的所有运行满足  $\varphi_3 = \varphi_6$ .

**命题 7.** 一个带标签的有穷状态自动机从  $s_0$  出发按路径长度  $l$  作广度优先搜索, 当  $l = |V| - 1$  时, 必可覆盖所有顶点.

命题 8 阐述了满足特定约束的有向图的一个基本性质, 在图论中早已得到证明, 这里不再重复给出.

**命题 8.** 一个带标签的有穷状态自动机  $A$  满足 PLTL 公式  $\varphi_2 = \square p$  (或  $\varphi_5 = \square r$ ), 当且仅当长度小于  $|V|$  的所有运行满足  $\varphi_2$  (或  $\varphi_5$ ).

证明.

⇒ 若  $A \models \square p$ , 即所有(接受)运行满足  $\square p$ , 那么长度小于  $|V|$  的所有运行满足  $\square p$ .

⇐ 若长度小于  $|V|$  的所有运行满足  $\square p$ , 根据命题 8, 这些运行覆盖了所有顶点, 因此  $A$  中所有顶点满足  $p$ , 故此  $A$  的所有(接受)运行满足  $\square p$ , 即  $A \models \square p$ . 证毕.

**推论 2.** 一个带标签的有穷状态自动机  $A$  满足 PLTL 公式  $\varphi_4 = \square p$ , 当且仅当长度等于 2 的所有运行满足  $\varphi_4$ .

**命题 9.** 一个带标签的有穷状态自动机  $A$  满足 PLTL 公式  $\varphi_7 = \square(p \triangleright \diamond q) = \square(\neg p \vee \diamond q) = \square(r \vee \diamond q)$ , (其中  $r = \neg p$ ), 当且仅当长度小于  $|V| \times 2^{|\mathcal{V}| - 1} + |V| + |E|$  的所有运行满足  $\varphi_7$ .

**命题 10.** 一个带标签的有穷状态自动机  $A$  满足 PLTL 公式  $\varphi_8 = \square \diamond p \rightarrow \square \diamond q = \square \diamond \neg p \vee \square \diamond q = \square \diamond r \vee \square \diamond q$  (其中  $r = \neg p$ ), 当且仅当长度小于  $|V| \times 2^{|\mathcal{V}| - 1} + |V| + |E|$  的所有运行满足  $\varphi_8$ .

**命题 11.** 一个带标签的有穷状态自动机  $A$  满足 PLTL 公式  $\varphi_9 = \diamond \square p \rightarrow \square \diamond q = \square \diamond \neg p \vee \square \diamond q = \square \diamond r \vee \square \diamond q$  (其中  $r = \neg p$ ), 当且仅当长度小于  $|V| \times 2^{|\mathcal{V}| - 1} + |V| + |E|$  的所有运行满足  $\varphi_9$ .

篇幅所限, 这里只给出命题 11、命题 12 和命题 13 的证明思路:

(1)  $\square(r \vee \diamond q)$  的含义在于: 在从  $A$  中任意顶点出发的子图上, 检测  $r \vee \diamond q$  是否被出发点满足. 在出发点即可判断  $r$  是否被满足, 而判断  $\diamond q$  则是把出发点“暂时看作”  $s_0$  的

一个检测过程. ①根据推论 7, 这个检测过程需要考察所有  $|V| \times 2^{|V|-1} + |E|$  长度之内的路径; ②根据命题 9, 从  $s_0$  到  $A$  中任意顶点, 考察长度在  $|V|$  之内的所有路径即可覆盖这些顶点. 综合①和②可知, 从  $s_0$  出发, 考察长度在  $|V| \times 2^{|V|-1} + |V| + |E|$  之内的所有路径即可判定  $\square(r \vee \diamond q)$  是否被满足.

(2) 同理, 从  $s_0$  出发, 考察长度在  $|V| \times 2^{|V|-1} + |V| + |E|$  之内的所有路径即可判定  $\square \diamond q, \square \diamond r$  是否被满足.

(3)  $\diamond \square r$  的含义在于: 从  $s_0$  出发的所有路径最终到达如下顶点: 从该顶点出发的子图中所有顶点均满足  $r$ . ①根据命题 9, 判断从某顶点出发的子图中所有顶点均满足  $r$ , 需考察长度在  $|V|$  之内的所有路径; ②根据推论 7, 从  $s_0$  出发的所有路径最终到达满足给定性质的顶点, 需考察长度在

$|V| \times 2^{|V|-1} + |E|$  之内的所有路径. 综合①和②可知, 从  $s_0$  出发, 考察长度在  $|V| \times 2^{|V|-1} + |V| + |E|$  之内的所有路径即可判定  $\diamond \square r$  是否被满足.

综合上述(1)、(2)和(3)可知, 命题 11、命题 12 和命题 13 成立.

由上述这些命题的证明过程可见, 对于所有 4 个基本时序算子, 每一个“ $\triangleright$ ”或“ $\diamond$ ”需要增加的考察路径长度为  $|V| \times 2^{|V|-1} + |E|$ , 每一个“ $\square$ ”需要增加的考察路径长度为  $|V|$ , 每一个“ $\circ$ ”需要增加的考察路径长度为 1. 故此, 对于 LTL 一般公式  $f$  而言, 在最坏情况下, 需要考察的路径长度不超过  $\text{len}(f) \times (|V| \times 2^{|V|-1} + |E|)$ , 其中  $\text{len}(f)$  为 LTL 公式  $f$  的长度.



**ZHU Wei-Jun**, born in 1976, Ph.D., associate professor. His research interests include DNA computing and formal methods.

**ZHOU Qing-Lei**, born in 1962, Ph.D., professor, Ph.D. supervisor. His research interests include DNA computing, formal methods and information security.

**ZHANG Qin-Xian**, born in 1953, professor, Ph.D. supervisor. His research interests include DNA computing and molecular biology.

## Background

The DNA computing is different from the classical computing. The former technique employs DNA molecules to conduct computations. Turing Award winner Prof. Adleman published the article in Science in 1994, using DNA molecules to solve the Hamiltonian path problem with a small scale, which is recognized as the pioneering work of DNA computing. Due to parallel advantages, the DNA computing has been successfully employed to solve many complex computational problems, for example, the satisfiability problem, the maximum clique problem, etc. In order to solve the Computational Tree Logic (CTL) model-checking problem via DNA molecules, Turing Award winner Prof. Emerson gave a DNA computing method for checking a basic temporal construct called “EFp” in CTL formulas. In order to solve the Linear Temporal Logic (LTL) model-checking problem via DNA molecules, a DNA-based method for checking a basic LTL formula called “pUq” was given in our previous study. However, the existing approaches can only check the temporal formulas mentioned above via DNA molecules under the

circumstances of molecular biology. To the best of our knowledge, other temporal logic formulas cannot be checked effectively via DNA molecules yet, for lack of available approaches. Up to now, we failed to see significant advances in DNA model checking. To solve this problem, we carry out research on LTL model checking methods based on DNA computing in this paper, and we present a new method. The simulate experimental result confirmed that the proposed approach successfully checks all basic LTL formulas and the most of popular LTL formulas under the circumstances of molecular biology, as well as other LTL formulas in theory. This work is supported by the National Natural Science Foundation of China under Grant No. 61250007, No. U1204608, No. U1304606, No. 61373043, No. 61572444, and the China Postdoctoral Science Foundation under Grant No. 2012M511588, No. 2015M572120, as well as Foundation for University Key Teacher by the Henan Province under Grant No. 2014GGJS-001.