文章编号: 1001-0920(2006)06-0701-04

插入/切割DNA计算系统模型研究

范忠骏[1,2], 王 波[1], 叶 林[3]
(1. 上海理工大学 管理学院, 上海 200093; 2. 上海电力学院 经济管理系 上海 200090; 3. 青岛理工大学 计算机工程学院, 山东 青岛 266033)

摘 要: 为建立以载体分子为基础的DNA 计算系统, 根据目的基因和载体分子连接操作的特性构造一类DNA 计算模型. 该模型基于上下文的插入/删除模型, 将线性分子拓展到环形质粒分上进行讨论. 以剪接系统理论为基础, 对在单一限制性内切酶作用下的插入/切割操作进行模拟. 最后证明了这类操作的图灵机表达能力.

关键词: DNA 计算; 剪接系统; 插入/切割计算; 图灵机
中图分类号: TP301.1 **文献标识码**: A

Research of Insertion/Cutting DNA Computing System Model

FAN Zhong-jun[1,2], *WANG Bo*[1], *YE Lin*[3]
(1. School of Management, Shanghai University of Science and Technology, Shanghai 200093, China; 2. Department of Economics and Management, Shanghai University of Electric Power, Shanghai 200090, China; 3. School of Computer Engineering, Qingdao University of Technological, Qingdao 266033, China. Correspondent: FAN Zhong-jun, E-mail: fanzj2005@163.com)

Abstract: A DNA computing model is established which simulates the mechanism of the ligation between target gene and vector molecular. The main idea is to study the DNA computing system which is based on vector molecular. This model that based on the theory of splicing systems extends contextual insertions/deletions model from linear molecular to circular plasmids, and simulates the insertion/cutting operation with single restriction enzymes. Finally, the operation's expression ability of Turing machine properties is proved.

Key words: DNA computing; Splicing system; Insertion/cutting computing; Turing machine

1 引 言

理论DNA 计算系统的基本研究方法是通过分析DNA 分子信息处理和传递的机制, 建立相应的数学模型, 其基本理论依据是形式语言和自动机理论. 在分子生物学的背景下, 利用限制性核酸内切酶的切割和连接酶的连接, 能够实现外源DNA 片段(称为目的基因)与载体分子的重组连接, 剪接模型是对这一过程进行抽象[1,2]. 在所使用的载体分子中, 常用的一类称为质粒DNA, 它是一种结构简单且具有自主复制能力的DNA 分子, 一般为双链闭环结构. 对具有这种环形结构的质粒DNA 分子作形式化研究时, 要将它转化为抽象字符集上的环形字符串来讨论. 文献[3, 4]对基于线性字符串和环形字符串剪接操作的一些问题进行了讨论. 文献[5]提出了插入/删除模型并研究了在环形字符串上的操作特性, 但它是将环形字符串切割成线性串, 再与另外的线性字符串连接, 本质上是基于线性字符串来研究. 本文将一个线性字符串插入环形字符串的特定位置并形成新的环形字符串, 由此构建基于环形字符串的操作模型. 文献[6]讨论了基于单一限制性内切酶切割操作的模型, 本文模型不仅考虑了切割操作, 而且考虑了单酶切位点的插入操作, 称为具有剪接规则的插入/切割系统.

通过限制性内切酶切割载体分子可以产生与外

收稿日期: 2005-01-26; **修回日期**: 2005-08-28.
基金项目: 上海市重点学科建设项目(T0502); 上海市教委基金项目(04EB16).
作者简介: 范忠骏(1971—), 男, 江西新建人, 讲师, 硕士, 从事系统控制与优化、DNA 计算的研究; 王波(1960—), 男, 江西东乡人, 副教授, 博士, 从事决策科学、生产运作管理等研究.

源DNA 片段相匹配的末端, 然后在连接酶的作用下实现二者的连接. 如果外源DNA 片段与载体分子由相同的单一限制性核酸内切酶进行消化酶切, 那么二者的两端将产生相同的可匹配粘性末端, 称为单酶切方式[7]. 本文模型就是建立在单一限制性内切酶操作下的插入-剪接系统, 原因是单酶切操作是基本的操作, 模型构造也较为简便

2 带剪接规则的插入/切割DNA 计算系统

基本剪接系统描述了线性DNA 分子片段上的剪接操作. 考虑字符集 Σ 上任意两个字符串

$$x = x_1u_1u_2x_2,\ y = y_1u_3u_4y_2,$$
$$x, y \in \Sigma^*. \tag{1}$$

规定剪接规则为 $r = u_1\# u_2 \$ u_3\# u_4$, 使

$$(x_1u_1u_2x_2, y_1u_3u_4y_2) \to_r (x_1u_1u_4y_2, y_1u_3u_2x_2)$$

成立, 则称为一次剪接操作, 如图 1 所示. 令 $R \subseteq \Sigma^* | \Sigma^* \$ \Sigma^* | \Sigma^*$, 其中 $|, \$ \notin \Sigma$. 则 $r \in R$, R 称为一个剪接规则集. 令 $z = x_1u_1u_4y_2$, $w = y_1u_3u_2x_2$, $z, w \in \Sigma^*$, 则上述剪接操作可表示为 $(x, y) \to_r (z, w)$.

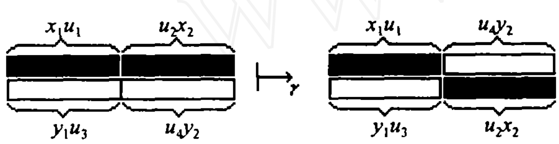

图 1 任意两个字符串的剪接操作

一个剪接系统表示为 $\sigma = (\Sigma, R)$, 记为 H. 对于 $\sigma = (\Sigma, R)$, 若存在 $L \subseteq \Sigma^*$, 使

$$\sigma(L) = \{z \in \Sigma^* \mid (x, y) \to_r (z, w),$$
$$x, y \in L, r \in R\}, \tag{2}$$

则 $\sigma(L)$ 称为剪接系统 H 产生的语言[8].

考虑字符集 Σ 上的任意字符串 $x = a_1a_2\cdots a_m$, 由 x 构成的环形字符串记为 $\sim x = \sim a_1a_2\cdots a_m$, 如图 2 所示. Σ 上的所有环形串构成集合 $\sim\Sigma$, 于是, 一个载体分子可以定义为字符集 $\sim\Sigma$ 上的一个环形字符串.

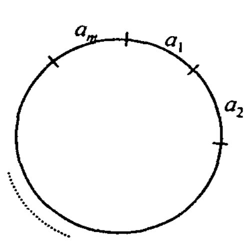

图 2 环形字符串 $\sim x$

令

$$\sim x = \sim u_1u_1'x_1v_1v_1'\cdots u_nu_n'x_nv_nv_n', \tag{3}$$

其中: $u_i'x_iv_i$ 由单一限制性内切酶识别, x_i 表示一段被编码的DNA 片段. $\sim X$ 上的剪接过程描述如下: 在第 i 次操作中, 限制性内切酶识别 $u_i'x_iv_i$ 段, 并进行切割, $\sim X$ 的剩余部分形成线性字符串, 利用连接酶可将此线性字符串重新环化; 若此时有新的带匹配末端的外源DNA 片段插入, 则利用连接酶将其插入相应的位置并形成新的环形字符串[6,9].

考虑 $\sim\Sigma$ 上的环形字符串 $\sim x = \sim x_1u_1u_2x_2$, 其中 u_1u_2 表示一个酶切位点. 若存在 $y = v_1wv_2 \in \Sigma$, 在规则 $r_1 = u_1 | \$ | v_1$ 和 $r_2 = v_2 | \$ | u_2$ 作用下, 使 $\sim x_1u_1v_1wv_2u_2x_2 \in \sim\Sigma$ 成立, 则称为插入操作, 并记为

$$y \to_R \sim x =$$
$$\{\sim x_1u_1v_1wv_2u_2x_2 \mid \sim x = \sim x_1u_1u_2x_2,$$
$$y = v_1wv_2, r_1, r_2 \in R, x, y \in \Sigma^*\}. \tag{4}$$

引入函数 $f: \sim\Sigma \times \Sigma \to \sim\Sigma$, 令

$$\sim z = \sim x_1u_1v_1wv_2u_2x_2 \in \sim\Sigma,$$

则 $\sim z = f(\sim x, y)$. 令 $I = \{\Sigma, R\}$, 对于 $L_1, L_2 \subseteq \Sigma^*$, I 上的插入操作记为

$$L_1 \to_I \sim L_2 = \bigcup_{Y \in L_1, x \in L_2} y \to_I \sim x. \tag{5}$$

类似地, 对于 $\sim\Sigma$ 上的环形字符串 $\sim x = \sim x_1u_1v_1wv_2u_2x_2$ 和 Σ 上的字符串 $y = v_1wv_2$, 若在规则 $r_3 = u_1 | v_1 \$ v_2 | u_2$ 作用下, 使 $\sim x_1u_1u_2x_2$ 成立, 则称为切割操作, 并记为

$$\sim x \to_R y =$$
$$\{\sim x_1u_1u_2x_2 \mid \sim x = \sim x_1u_1v_1wv_2u_2x_2,$$
$$y = v_1wv_2, r_3 \in R, x, y \in \Sigma^*\}. \tag{6}$$

同理, 引入函数 $g: \sim\Sigma \times \Sigma \to \sim\Sigma$, 则 $\sim z = g(\sim x, y)$, 其中 $\sim z = \sim x_1u_1u_2x_2$. 令 $C = \{\Sigma, R\}$, 对于 $L_1, L_2 \subseteq \Sigma^*$, C 上的切割操作记为

$$\sim L_2 \to_C L_1 = \bigcup_{y \in L_1, x \in L_2} \sim x \to_C y. \tag{7}$$

于是, 带剪接规则的插入/切割 $(\sim \mathrm{IC})$ 系统可表示为 $\sim \mathrm{IC} = (\Sigma, \sim T, I, C, \sim A)$, 其中 $T \subseteq \Sigma$ 表示最终字符集, $A \subseteq \Sigma^+$ 表示初始字符串集

定义 1 对于 $x, y, z \in \Sigma^*$, 若 $\sim z = f_R(\sim x, y)$ 或 $\sim z = g_R(\sim x, y)$ 成立, 记为 $\sim x \Rightarrow \sim z$; 一个推导序列为 $\sim x_1 \Rightarrow \sim x_2 \Rightarrow \cdots \Rightarrow \sim x_k\ (k \geq 0)$ 表示为 $\sim x_1 \Rightarrow^* \sim x_k$. 则 $\sim \mathrm{IC}$ 接受的语言为

$$L(\sim \mathrm{IC}) =$$
$$\{\sim z \in \sim T^* \mid z \Rightarrow^* Y_0, Y_0 \in A\ (\subseteq \Sigma^+)\}. \tag{8}$$

3 $(\sim \mathrm{IC})$ 系统的图灵机表达能力

根据文献[10, 11], 首先基于半 Thue 过程给出图灵机的一个描述. 对字符集 Σ^* 上的任意字符串 u 和 v, 表达式 $u \to v$ 称为一个半 Thue 产生式, 半 Thue 产生式的有限集合称为半 Thue 过程, 记为 P. 设 $S, X \subseteq \Sigma$, $S \cap X = \varnothing$, S 称为状态集, X 称为带

字符集. 构造字符集 $\Sigma' = \{S, X, \#, L, R, Y_0, Y_1, Y_2\}$, 则 Σ' 上的任一字符串 $\# u s_i v \#$ 称为一个 Post 字. TM = $(S, X \cup \{\#, L, R, Y_0, Y_1, Y_2\}, P)$ 称为一个图灵机, 满足下列条件:

1) $s_1 \in S$ 称为初始状态, $B \in X$ 称为空字符, $S_f \subseteq S$ 称为最终状态集, $T \subseteq X \setminus \{B\}$ 称为最终字符集;

2) 存在一个半 Thue 过程 P.

引理 1　对于 $w \in T^*$, 图灵机 TM 接受 w 的充要条件是

$$\# s_0 w \# \Rightarrow^* \# w_1 s_f w_2 \# \Rightarrow^* \# Y_0 \#, \qquad (9)$$

由此生成的被图灵机 TM 接受的语言为

$$L(\mathrm{TM}) = \{w \in T^* \mid \# s_0 w \# \Rightarrow^* \# w_1 s_f w_2 \#, \; s_f \in S_f, w_1, w_2 \in X^*\}. \qquad (10)$$

定理 1　若存在 $\sim \Sigma^*$ 上的一个语言 L 被图灵机 TM 接受, 记为 $\sim L(\mathrm{TM})$, 则必然存在一个由 ~ IC 系统确定的语言 $L(\sim \mathrm{IC})$, 使

$$\sim L(\mathrm{TM}) = L(\sim \mathrm{IC}).$$

证明　首先构建基于 ~ IC 系统的半 Thue 过程 P 规则集如下:

1) 对规则 $s_i a \to s_j b$, 剪接规则为 $r_1 = u s_i a | \$ | s_j$, $r_2 = b | \$ | v$, $r_3 = u | s_i \$ a | s_j$, 若 $\sim x = \sim \# l u s_i a v r \#$, $y = s_j b$, 则有

$$\sim \# l u s_i a v r \# \Rightarrow$$
$$\sim \# l u s_i a s_j b v r \# \Rightarrow \sim \# l u s_j b v r \#;$$

2) 对规则 $s_i a c \to a s_j c$, 剪接规则为 $r_1 = u s_i a | \$ | s_j$, $r_2 = R | \$ | c v$, $r_3 = u | s_i \$ s_i | a$, $r_4 = s_j | R \$ R | c v$, 若 $\sim x = \sim \# l u s_i a c v r \#$, $y = s_j R$, 则有

$$\sim \# l u s_i a c v r \# \Rightarrow \sim \# l u s_i a s_j R c v r \# \Rightarrow$$
$$\sim \# l u a s_j R c v r \# \Rightarrow \sim \# l u a s_j c v r \#;$$

3) 对规则 $s_i a \# \to a s_j B \#$, 剪接规则为 $r_1 = u s_i a | \$ | s_j$, $r_2 = B | \$ | \#$, $r_3 = u | s_i \$ s_i | a$, $r_4 = s_j | R \$ R | B$, 若 $\sim x = \sim \# l u s_i a \#$, $y = s_j R B$, 则有

$$\sim \# l u s_i a \# \Rightarrow \sim \# l u s_i a s_j R B \# \Rightarrow$$
$$\sim \# l u a s_j R B \# \Rightarrow \sim \# l u a s_j B \#;$$

4) 对规则 $c s_i a \to s_j c a$, 剪接规则为 $r_1 = u | \$ | s_j$, $r_2 = L | \$ | c s_i a v$, $r_3 = c | s_i \$ s_i | a$, $r_4 = s_j | L \$ L | c a v$, 若 $\sim x = \sim \# l u c s_i a v r \#$, $y = s_j L$, 则有

$$\sim \# l u c s_i a v r \# \Rightarrow \sim \# l u s_j L c s_i a v r \# \Rightarrow$$
$$\sim \# l u s_j L c a v r \# \Rightarrow \sim \# l u s_j c a v r \#;$$

5) 对规则 $\# s_i a \to \# s_j B a$, 剪接规则为 $r_1 = \# | \$ | s_j$, $r_2 = B | \$ | s_i a v$, $r_3 = B | s_i \$ s_i | a v$, $r_4 = s_j | L \$ L | B a v$, 若 $\sim x = \sim \# s_i a v r \#$, $y = s_j L B$, 则有

$$\sim \# s_i a v r \# \Rightarrow \sim \# s_j L B s_i a v r \# \Rightarrow$$
$$\sim \# s_j L B a v r \# \Rightarrow \sim \# s_j B a v r \#;$$

6) 对规则 $s_i a \to Y_1 a$, 剪接规则为 $r_1 = u s_i | \$ | Y_1$, $r_2 = Y_1 | \$ | a v$, $r_3 = u | s_i \$ s_i | Y_1 a v$, 若 $\sim x = \sim \# l u s_i a v r \#$, $y = Y_1$, 则有

$$\sim \# l u s_i a v r \# \Rightarrow$$
$$\sim \# l u s_i Y_1 a v r \# \Rightarrow \sim \# l u Y_1 a v r \#;$$

7) 对规则 $Y_1 a \to Y_1$, 剪接规则为 $r_1 = Y_1 | a \$ a | \#$, 若 $\sim x = \sim \# l u Y_1 a \#$, 则有

$$\sim \# l u Y_1 a \# \Rightarrow \sim \# l u Y_1 \#;$$

8) 对规则 $b Y_1 \to b Y_2$, 剪接规则为 $r_1 = b | \$ | Y_2$, $r_2 = Y_2 | \$ | Y_1$, $r_3 = Y_2 | Y_1 \$ Y_1 | \#$, 若 $\sim x = \sim \# l u b Y_1 \#$, 则有

$$\sim \# l u b Y_1 \# \Rightarrow \sim \# l u b Y_2 Y_1 \# \Rightarrow \sim \# l u Y_2 \#;$$

9) 对规则 $b Y_2 \to Y_2$, 剪接规则为 $r_1 = u | b \$ b | Y_2$, 若 $\sim x = \sim \# l u b Y_2 \#$, 则有

$$\sim \# l u b Y_2 \# \Rightarrow \sim \# l u Y_2 \#;$$

10) 对规则 $Y_2 \# \to Y_0 \#$, 剪接规则为 $r_1 = \# | \$ | Y_0$, $r_2 = Y_0 | \$ | Y_2$, $r_3 = Y_0 | Y_2 \$ Y_2 | \#$, 若 $\sim x = \sim \# Y_2 \#$, 则有

$$\sim \# Y_2 \# \Rightarrow \sim \# Y_0 Y_2 \# \Rightarrow \sim \# Y_0 \#;$$

另外设定规则如下:

11) $\lambda | \$ | \#$, $s_0 | \$ | b$, $b | \$ | \#$, 产生 $\sim w \Rightarrow \sim \# s_0 w \#$;

12) $\lambda | \$ | \#$, $s_0 | \$ | \#$, 产生 $\lambda \Rightarrow \sim \# s_0 \#$;

13) $s_f | \$ | Y_2$, $Y_2 | \$ | \#$, $\# | s_f \$ s_f | Y_2$, 产生 $\sim \# s_f \# \Rightarrow \sim \# s_f Y_2 \# \Rightarrow \sim \# Y_2 \#$.

由规则 1) ~ 13), 对于 $\sim w \in \sim \Sigma^*$, 可证明若 $\sim w \in \sim L(\mathrm{TM})$, 且 $\sim w \in L(\sim \mathrm{IC})$, 则有 $\sim L(\mathrm{TM}) = L(\sim \mathrm{IC})$.

一方面, 若 $\sim w \in \sim L(\mathrm{TM})$, 则 $\sim w \in \sim T^*$, 即有 $\sim \# s_1 w \# \Rightarrow^* \sim \# w_1 s_f w_2 \#$. 另一方面, 若 $\sim w \in L(\sim \mathrm{IC})$, 则表明在 ~ IC 系统规则下由字符串 $\sim w$ 能够推证出 $\sim w \Rightarrow^* \sim Y_0$. 下面分两种情况证明:

1) 若 $\sim w \neq \lambda$, 使用基于 ~ IC 系统的规则集, 则有

$$\sim w \ (\text{使用规则 11)}) \Rightarrow$$
$$\sim \# s_0 w \# \ (\text{使用规则 1) ~ 5)}) \Rightarrow^*$$
$$\sim \# w_1 s_f w_2 \# \ (\text{规则 6)}) \Rightarrow$$
$$\sim \# w_1 Y_1 w_2 \# \ (\text{规则 7)}) \Rightarrow$$
$$\sim \# w_1 Y_1 \# \ (\text{规则 8)}) \Rightarrow$$
$$\sim \# w_1 Y_2 \# \ (\text{规则 9)}) \Rightarrow$$
$$\sim \# Y_2 \# \ (\text{规则 10)}) \Rightarrow \sim \# Y_0 \#.$$

2) 若 $\sim w = \lambda$, 则有

$$\lambda (\text{使用规则 12)}) \Rightarrow$$
$$\sim \# s_0 \# \ (\sim w = \lambda, s_0 = s_f) \Rightarrow$$

$\sim \# s_f \#$ (规则 13)) $\Rightarrow$

$\sim \# Y_2 \# \Rightarrow \sim \# Y_0 \#$.

于是, 由 1) 和 2) 可知定理 1 成立. □

4 结 语

本文提出的(~ IC)系统扩展了基于线性字符串的剪接插入和删除操作, 并证明了它的图灵机表达能力, 目的是建立基于载体分子的DNA 计算系统. 这是因为载体分子系统是目前基因工程的基本材料, 其优良特性使它成为良好的计算平台. 因此开展基于载体分子的计算模型研究具有重要的实际意义.

基于本文提出的思想, 进一步的研究方向和需要解决的问题是: 1) 建立使用双酶切模式的模型; 2) 进一步研究模型的迭代功能及其语言的表达能力.

参考文献(References)

[1] Freund R, Kari L, Păun G. DNA Computing Based on Splicing: The Existence of Universal Computer[J]. *Theory of Computing Systems*, 1999, 32(1): 69-112.

[2] Pixton D. Splicing in Abstract Families of Languages [J]. *Theoretical Computer Science*, 2000, 234 (1-2): 135-166.

[3] Bonizzoni P, Felice D C, Mauri G, et al. Decision Problems for Linear and Circular Splicing Systems[A]. *Lecture Notes in Computer Science* [C]. London: Springer-Verlag, 2002: 78-92.

[4] Bonizzoni P, Felice D C, Mauri G, et al. Circular Splicing and Regularity [J]. *Theoretical Informatics and Applications*, 2004, 38(3): 189-228.

[5] Kari L, Thierrin G. Contextual Insertions Deletions and Computability [J]. *Information and Computation*, 1996, 131(1): 47-61.

[6] Head T. Circular Suggestions for DNA Computing [A]. *Pattern Formation in Biology, Vision and Dynamics* [C]. Singapore: World Scientific, 2000: 325-335.

[7] 杨吉成. *医用分子生物学*[M]. 苏州: 苏州大学, 2003: 199-244.
(Yang J C. *Medical Molecular Biology* [M]. Suzhou: Suzhou University, 2003: 199-244.)

[8] Harju T, Margenstern M. *Remarks on the Universality of Splicing Systems* [R]. Turku: Turku University, 2004.

[9] Daley M, Kari L, Gloor G, et al. Circular Contextual Insertions Deletions with Applications to Biomolecular Computation [A]. *Proc of SPIRE/CRIWG* [C]. Cancun, 1999: 47-54.

[10] 张立昂. *可计算性与计算复杂性导引*[M]. 北京: 北京大学出版社, 1997: 117-121.
(Zhang L A. *Introduction of Computability and Computational Complexity* [M]. Beijing: Beijing University Press, 1997: 117-121.)

[11] 蒋宗礼, 姜守旭. *形式语言与自动机理论*[M]. 北京: 清华大学出版社, 2003: 29-34.
(Jiang Z L, Jiang S X. *Theory of Formal Language and Automata* [M]. Beijing: Tsinghua University Press, 2003: 29-34.)

(上接第 700 页)

参考文献(References)

[1] Xu S Y, Chen T W. Robust H_∞ Filtering for Uncertain Impulsive Stochastic Systems under Sampled Measurements[J]. *Automatica*, 2003, 39(3): 509-516.

[2] Shaked U, de Souza. Robust Minimum Variance Filtering[J]. *IEEE Trans on Signal Processing*, 1995, 43(11): 2474-2483.

[3] Hung Y S, Yang F W. Robust H_∞ Filtering with Error Variance Constraints for Discrete Time-varying Systems with Uncertainty [J]. *Automatica*, 2003, 39 (7): 1185-1194.

[4] Xie L H, Lu L L, Zhang D, et al. Improved Robust H_2 and H_∞ Filtering for Uncertain Discrete-time Systems [J]. *Automatica*, 2004, 40(5): 873-880.

[5] Shi P. Filtering on Sampled-data Systems with Parametric Uncertainty [J]. *IEEE Trans on Automatic Control*, 1998, 43(7): 1022-1227.

[6] Sing Kiong Nguang, Shi P. On Designing Filters for Uncertain Sampled-data Nonlinear Systems[J]. *System & Control Letters*, 2000, 41(5): 305-316.

[7] Sing Kiong Nguang, Shi P. Nonlinear H_∞ Filtering Sampled-data Systems [J]. *Automatica*, 2000, 36 (2): 303-310.