第30卷 第6期
2007年6月

计 算 机 学 报
CHINESE JOURNAL OF COMPUTERS

Vol. 30 No. 6
June 2007

DNA计算机中队列数据结构的设计及实现

李汪根[1),3)] 丁永生[1),2)]

[1)](东华大学信息科学与技术学院 上海 201620)

[2)](东华大学数字化纺织服装技术教育部工程研究中心 上海 201620)

[3)](安徽师范大学数学计算机学院 安徽 芜湖 241000)

摘 要 提出了DNA计算机中队列数据结构的设计方法,该方法利用两种不同的限制性内切酶完成队列的入队和出队操作,并给出了队列的DNA编码和仿真实例.首先给出了DNA计算机中队列存储结构的形式描述;然后详细给出了DNA计算机中队列初始化、入队和出队等操作的生物实现方法;最后给出了一个具体算法的实例,仿真了DNA计算机上该算法的运行机制.仿真结果表明文中提出的队列的设计方法在DNA计算机上切实可行.这种方法可推广到DNA计算机上其他类型的数据结构,帮助DNA计算机合理、有效地组织需要处理的信息,从而使DNA计算机走向实际应用.

关键词 DNA计算机;数据结构;队列;DNA编码;存储结构

中图法分类号 TP384

Design and Implementation of Queue Data Structure in DNA Computer

LI Wang-Gen[1),3)] DING Yong-Sheng[1),2)]

[1)](*College of Information Sciences and Technology, Donghua University, Shanghai* 201620)

[2)](*Engineering Research Center of Digitized Textile & Fashion Technology of Ministry of Education, Donghua University, Shanghai* 201620)

[3)](*College of Mathematics and Computer Sciences, Anhui Normal University, Anhui, Wuhu* 241000)

Abstract This paper proposes the method of designing a queue in DNA computer, which utilizes two different kinds of restriction enzymes to complete entering and leaving of the queue; And the nucleotide encodings of the queue and an instance of simulation are given out. Firstly, the storage structure of the queue in DNA computer is formally described. Then, the bio-operations of initialization, entering and leaving on the queue are described in detail. Furthermore, to prove the feasibility of this method, an actual queue with detailed nucleotide encodings is given out. The process of an algorithm implemented on this queue is demonstrated. Based on this method, more other data structures in DNA computer can be developed, which can help to organize the information processed by DNA computer correctly and efficiently, and make DNA computer for practical applications.

Keywords DNA computer; data structure; queue; DNA encoding; storage structure

1 引 言

DNA计算是分子计算的一种,DNA计算中被处理的数据存储在DNA分子中,数据的处理通过诸如合成、杂交、剪切等基因工程的方法来实现.DNA计算的最大优点是具有极大的存储密度和计算的高度并行性.科学家在DNA计算领域已经取得了

收稿日期:2006-03-01;修改稿收到日期:2006-12-28.本课题得到国家自然科学基金(60474037)、教育部新世纪优秀人才支持计划(NCET-04-415)、教育部高等学校博士学科点专项科研基金(20060255006)和安徽省教育厅青年基金(2005jql043)资助.**李汪根**,男,1973年生,博士研究生,从事DNA计算和生物信息学等研究.**丁永生**,男,1967年生,博士,教授,博士生导师,从事智能系统、网络智能、DNA计算、人工免疫系统、生物网络结构、生物信息学、数字化纺织服装、智能决策与分析等研究.E-mail: ysding@dhu.edu.cn.

许多研究成果[1-2]:1994 年,Adleman 首次成功地在实验室应用 DNA 分子解决了 7 节点的 Hamilton 路径问题,证明了运用 DNA 分子进行计算的可行性[3];1995 年,Lipton 提出了运用 DNA 实验解决可满足性问题的方法[4];1997 年,Ouyang 等人通过对全体可能的连通子图构建相应的 DNA 分子计算池,从而解决了 6 节点的最大连通子图问题[5]. 1996 年,Guarnieri 等首次给出了 DNA 计算机运算系统中正有理数的进位加法运算[6];1999 年,Yurke 等提出了一种对输入串与运算串的 DNA 分子编码是分离的 DNA 计算机系统中的加法运算模型[7]. 2001 年,Shapiro 等提出了一种以 DNA 分子为载体的有限状态自动机模型[8],该模型的硬件由生物操纵酶实现,软件和输入数据是预先编码的 DNA 分子,通过在试管中自动完成一系列剪切、连接循环最后可以得到编码了最终状态的输出分子;2004 年,Shapiro 等又提出了一种用于控制基因表达的分子自动计算机并应用于疾病诊断治疗[9].

DNA 计算机要走向实际应用,必须像电子计算机一样,需要解决 DNA 计算机中信息的组织问题,这就需要合理的数据结构来有效地组织 DNA 计算机需要处理的信息. 数据结构在计算机中占有重要地位,它有多种类型,如:队列、堆栈、链表、树、图等[10]. 本文以计算机中典型的数据结构——队列为例,讨论了 DNA 计算机上队列数据结构的设计及其编码实现,详细描述了 DNA 计算机上队列基本操作的实现,并给出了其应用的算法实例.

2 DNA 计算机中队列的设计及其编码实现

2.1 队列的基本概念

队列结构是一种特殊的线性表,这种结构只允许在表的一端进行插入操作(称之为队尾),而在另一端进行删除操作(称之为队头). 由于最先出队列的元素也是最先进入队列的元素,因此这种结构也称为先进先出表. 队头和队尾分别由队头指针和队尾指针指示. 当队列中不含任何数据元素时,称为空队列.

队列在内存中有两种存储结构:顺序存储结构和链式存储结构. 顺序存储结构的队列,其元素按照逻辑顺序存储在连续有序的存储空间. 链式存储结构的队列,其元素按照逻辑顺序存储,但存储空间不要求连续有序,一般用链表来存储所有元素.

2.2 DNA 计算机中队列的存储结构

用 DNA 计算机来求解问题,首要步骤是将问题映射到 DNA 域,即 DNA 编码. 本文采用 DNA 域中 A,G,C,T 的不同组合来表示队列不同组件的编码方式.

在实现队列的基本操作中,需要用到合成、杂交、连接、剪切等生化反应,这些生化反应在分子生物学上已经有很成熟的应用[11],本文不作详细讨论.

这里,我们用双链 DNA 分子来表示 DNA 计算机中的队列,队列元素的编码按顺序存放在 DNA 双链中. 因此,DNA 计算机中队列的存储结构更像顺序表. 由于 DNA 分子的海量存储容量,这里我们考虑队列的长度,也不考虑队列操作中会发生溢出.

表示队列的 DNA 双链由 7 个部分组成. 作为一个示例,图 1 给出了 DNA 计算机中的队列示意图.

(1) 队列元素:图 1 中,含有三个队列元素 A, B 和 C,其中,C 是队尾元素,A 是队头元素.

(2) *Enzyme_I*:某种限制性内切酶的识别位点. 在该酶的活性条件下,编码队列的 DNA 链将在队尾的位置被剪切. 在连接酶的作用下,待入队的队列元素将与被这种内切酶剪下的片段进行杂交和连接反应,从而完成该元素的入队操作.

(3) *Enzyme_II*:也是某种限制性内切酶的识别位点. 在该酶的活性条件下,队列将在队头的位置被剪切. 为了正确完成下次队头元素的出队操作,一个预先设计好的称作 detector 的新片段会在连接酶的作用下与被这种内切酶剪下的片段进行杂交和连接反应. 因此这部分用来完成出队操作. 同时,从参加连接反应的 detector 分子可以知道当前出队的队列元素的值.

由于入队和出队操作在队列的不同位置完成,因此分别实现这两种操作的限制性内切酶不能为同一种酶,否则在这种酶的活性条件下,完成这两种操作中的任一种都会影响到另一种操作.

(4) *Segment_L* 和 *Segment_R*:这两个部分包含合适长度的碱基对,这些碱基对确保队列入队和出队操作时,编码队列的 DNA 双链在合适的位置被内切酶剪切成两个片段.

(5) *Primer_L* 和 *Primer_R*:这两个部分是设计好的 PCR 反应的左右引物. 反应物的浓度直接影响生化反应的效率. 为了保证操作的可靠性,分子生物学中采用 PCR 来扩增目标 DNA 链的数量.

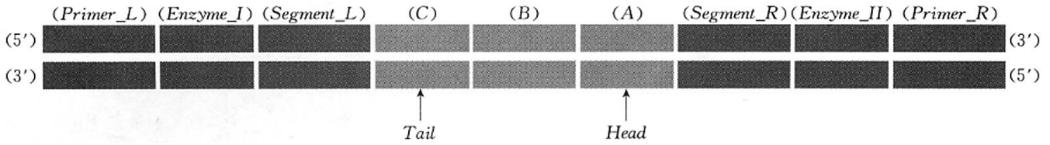

图 1　DNA 计算机中含有 A，B 和 C 三个元素的队列

2.3　DNA 计算机中队列的基本操作

这一节讨论如何用上述生物操作来实现 DNA 计算机中队列的基本操作，这些操作包括初始化一个队列、元素入队和出队等. 这里，我们用符号 Q 表示 DNA 计算机中的一个队列.

(1) *Bio-Initiate*(Q)

一个不含任何元素的空队列如图 2 所示. *Bio-Initiate*(Q)表示初始化一个空队列，其在 DNA 计算机中的实现过程可用图 3 来表示. 首先按照队列的 DNA 编码合成两条单链：顺义链 $Q1$ 和它的反义链 $A1$. 然后在合适的温度条件下，对这两条互补链进行退火反应从而生成代表空队列的 DNA 双链.

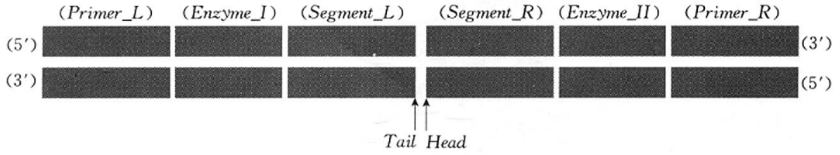

图 2　不含任何元素的空队列

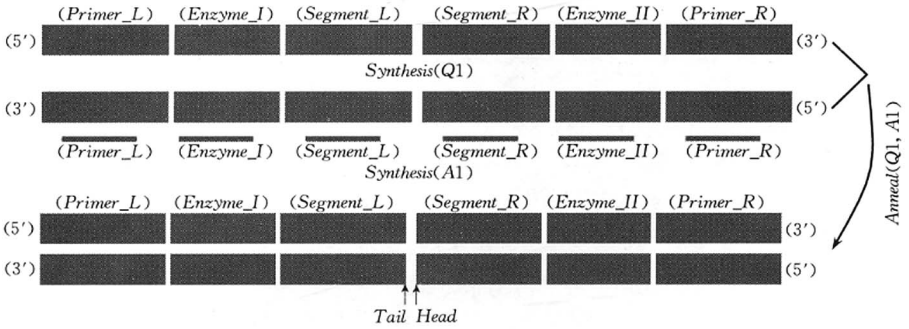

图 3　*Bio-Initiate*(Q)过程示意图

(2) *Bio-Enter*(Q,X)

Bio-Enter(Q, X)表示将元素 X 插入队列队尾，这个过程可用图 4 表示. 为实现元素 X 的入队操作，需要在 *Enzyme_I* 酶的活性条件下，在编码队列的双链 DNA 分子其队尾元素所在位置处剪切该双链. 这将产生两种片段：左半部分和右半部分，每部分含有一个粘性末端. 由于元素 X 的编码含有与从 Q 中剪切下来的右半部分互补的粘性末端，所以在连接酶的作用下它们之间会通过杂交和连接反应，形成完整的 DNA 双链，从而将元素 X 放到队列

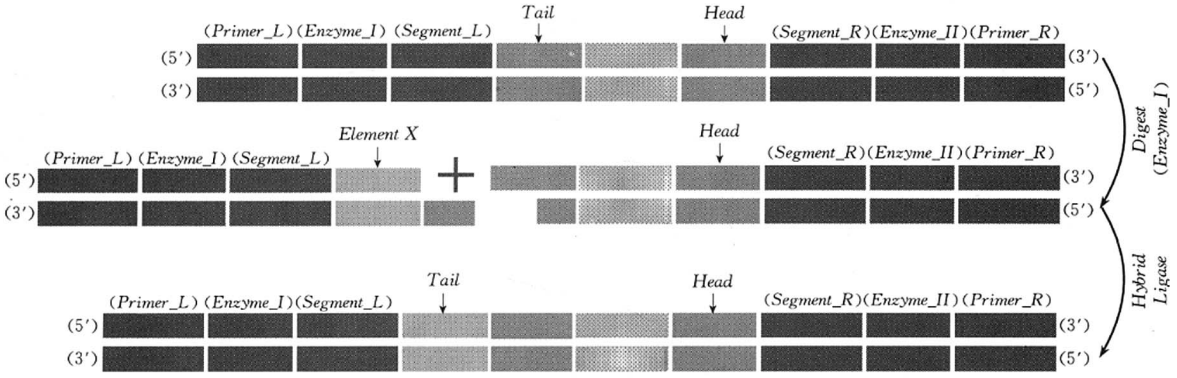

图 4　*Bio-Enter*(Q,X)过程示意图

的尾部. 进一步,由于元素 X 的编码中含有内切酶 *Enzyme_I* 的识别位点,因此下一次入队操作能正确完成.

(3) *Bio-Getout*(Q,X)

Bio-Getout(Q,X)表示将队头元素移出队列,并获取队头元素的值,如图 5 所示. 为完成队头元素的出队,需在内切酶 *Enzyme_II* 的活性条件下,在编码队列 Q 的 DNA 分子其队头位置剪切该双链,从而产生两个具有互补粘性末端的片段:左半部分和右半部分. 预先设计好的某片段 detector 含有与其左半部分互补的粘性末端,因此在连接酶的作用下会发生杂交和连接反应,从而得到代表新队列的 DNA 分子. 另一方面,由于为每一个队列元素都设计了一个唯一的 detector 分子,因此可以从当前被连接上的 detector 分子得知当前出队的队列元素的值. 换言之,每一个出队的元素都对应唯一一个 detector,其粘性末端刚好与被剪切下来的左半部分互补配对. 另外,还有一个特定的 *detector_Empty* 分子,如果当前被连接的是该分子,则当前队列是一个空队列.

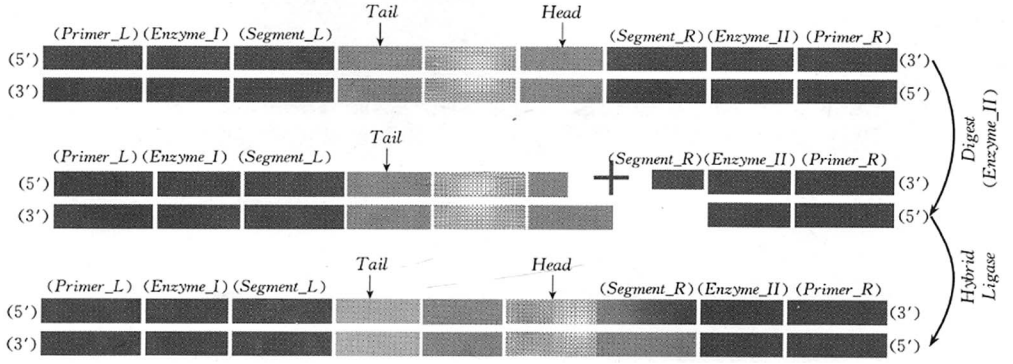

图 5 *Bio-Getout*(Q,X)的过程示意图

3 DNA 计算机中队列的算法实例

为了验证以上 DNA 计算机中队列设计的可行性,我们详细给出含有 3 个元素 A,B,C 的队列 Q 各部分的 DNA 编码,并给出基于该队列的一个算法实例.

3.1 核苷酸编码

这里,限制性内切酶 *Enzyme_I* 采用 *FokI*,其识别位点是 5′-GGATG-3′,剪切位点是(9,13). 内切酶 *Enzyme_II* 采用 *BsmFI*,其识别位点是 5′-GGGAC-3′,剪切位点是(10,14). 在整个队列的 DNA 编码序列中有且仅有这两个内切酶的一个识别位点.

在下面的核苷酸编码中,方框中的数字代表任意可能的碱基,但是要求这些碱基构成的 DNA 序列中不包含更多的上述两种内切酶的识别位点.

空队列 Q 的一种 DNA 编码如图 6(a)所示. Q 中 3 个元素的 DNA 编码如图 6(b)所示. 每个队列元素的编码中包含有内切酶 *FokI* 的识别位点,并且具有相同的粘性末端,以便下一次入队操作能正确完成. 编码中紧靠粘性末端的左边 4 个不同的碱基用来区分队列的不同元素. 图 6(c)给出了分别与这 3 种队列元素对应的 detector 分子以及识别队列为空队列的 *detector_Empty* 分子的编码.

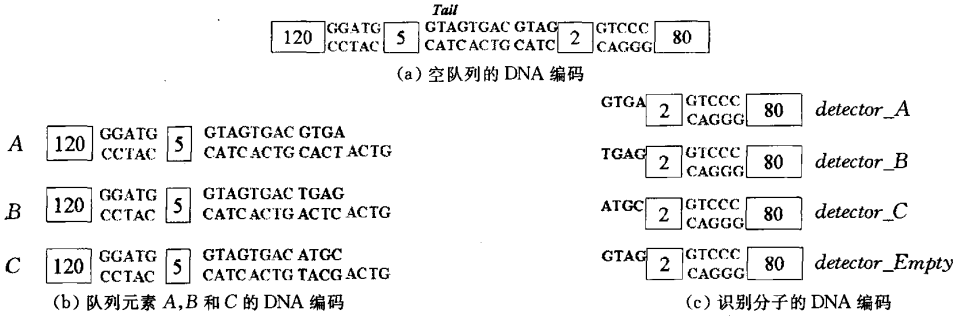

图 6 队列 Q 的 DNA 编码

3.2 算法实例

DNA 计算机上基于上述队列的一个简单算法如图 7 所示.

图 8 演示了该算法的运行过程. 在算法运行的第 4 步，由于被连接上的识别分子是 *detector_A* 分子，所以当前出队的队首元素是 *A*.

```
Algorithm // ** Queue Q is empty when starting ** //
{
  Bio-Initiate(Q);
  Bio-Enter(Q, A);
  Bio-Enter(Q, B);
  Bio-Getout(Q, X);
  Bio-Enter(Q, C);
}
```

图 7　基于队列 *Q* 的算法实例

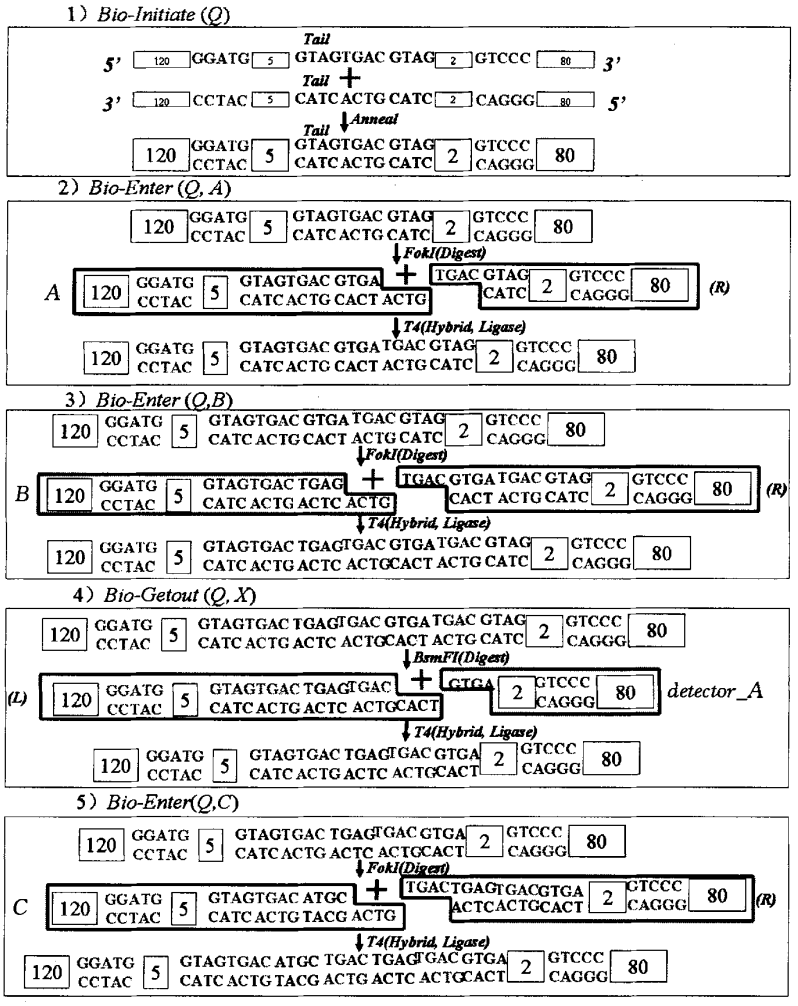

图 8　DNA 计算机中算法的运行过程

4 结　论

本文研究了在 DNA 计算机中实现队列数据结构的可行性. 讨论了 DNA 计算机中队列的存储结构、队列元素以及队列基本操作的 DNA 编码方式，并给出了一个具体队列的实例及其详细的 DNA 编码实现. 算法的演示结果表明本文提出的 DNA 计算机中队列数据结构的实现及编码方法具有可行性. 本文工作对进一步研究 DNA 计算机中其它数据结构具有很好的借鉴意义.

参 考 文 献

[1] Wang Lei, Lin Ya-Ping, Li Zhi-Yong. DNA computation for a category of special integer planning problem. Journal of Computer Research and Development, 2005, 42(8): 1431-1437(in Chinese)
(王雷，林亚平，李智勇. 一类特殊整数规划问题的 DNA 计算. 计算机研究与发展，2005，42(8)：1431-1437)

[2] Xu Jin, Zhang Lei. DNA computer principle, advances and

difficulties (I): Biological computing system and its applications to graph theory. Chinese Journal of Computers, 2003, 26(1): 1-11(in Chinese)
(许进，张雷. DNA 计算机原理、进展及难点(I):生物计算系统及其在图论中的应用. 计算机学报，2003，26(1)：1-11)

[3] Adleman L M. Molecular computation of solutions to combinatorial problems. Science, 1994, 266(11): 1021-1024

[4] Lipton R J. DNA solution of hard computational problem. Science, 1995, 268(4): 542-545

[5] Ouyang Q, Kaplan P D, Liu S, Libchaber A. DNA solution of the maximal clique problem. Science, 1997, 278(17): 446-449

[6] Guarnieri F, Fliss M, Bancroft C. Making DNA add. Science, 1996, 273(7): 220-223

[7] Benenson Y et al. DNA implementation of addition in which the input strands are separate from the operator strands. Biosystems, 1999, 52(1-3): 165-174

[8] Benenson Y, Paz-Elizur T, Adar R, Keinan E, Livneh Z, Shapiro E. Programmable and autonomous computing machine made of biomolecules. Nature, 2001, 414(22): 430-434

[9] Benenson Y, Gil B, Ben-Dor U, Adar R, Shapiro E. An autonomous molecular computer for logical control of gene expression. Nature, 2004, 429(27): 1-6

[10] Yan Wei-Min, Wu Wei-Min. Data Structure (Second Edition). Beijing: Tsinghua University Press, 1992(in Chinese)
(严蔚敏，吴伟民. 数据结构(第 2 版). 北京：清华大学出版社，1992)

[11] Ding Yong-Sheng, Shao Shi-Huang, Ren Li-Hong. DNA Computing and Soft Computing. Beijing: Science Press, 2002 (in Chinese)
(丁永生，邵世煌，任立红. DNA 计算与软件算. 北京：科学出版社，2002)

LI Wang-Gen, born in 1973, Ph. D. candidate. His research interests include bio-computer, bioinformatics and intelligent computing.

DING Yong-Sheng, born in 1967, professor, Ph. D. supervisor. His scientific interests include computational intelligence, network intelligence, DNA computing, artificial immune systems, bio-network architecture, bio-informatics, digitized textile & fashion technology, and intelligent decision-making.

Background

DNA computing is a novel method developed for computing purpose recently, where data is stored and processed on molecule level. Many researches have reported on DNA computer. Most of which have been directed towards solving difficult NP complete problems, supporting for basic computational operations such as logic operations and arithmetic operations, and mimicking the processing of Turing machine. Being similar to electronic computer, data structures in DNA computer can help to organize the information processed by DNA computer correctly and efficiently. However, data structures in DNA computer is rarely touched till now. This paper proposes the design, encodings and implementation of queue in DNA computer. Based on this method, more other data structures in DNA computer can be developed. This work was supported in part by the National Natural Science Foundation of China (No. 60474037), Program for New Century Excellent Talents in University from Ministry of Education of China (No. NCET-04-415), Specialized Research Fund for the Doctoral Program of Higher Education from Ministry of Education of China (No. 20060255006), and the Youth Foundation from Educational Committee of Anhui Province (No. 2005jql043).