Bioinformatics Advance Access published February 26, 2004

Primer Design Using Genetic Algorithm

Jain-Shing Wu[1], Chungnan Lee[1], Chien-Chang Wu [2] and Yow-Ling Shiue[3]

[1]: Department of Computer Science and Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan
[2]: Department of Food Engineering, Da-Yeh University, Taiwan
[3]: Institute of Biomedical Sciences, National Sun Yat-Sen University, Kaohsiung, Taiwan

E-mail: wujs@mail.cse.nsysu.edu.tw, cnlee@mail.cse.nsysu.edu.tw

ABSTRACT

Motivation: Before performing the PCR experiment, a pair of primers to clip the target DNA subsequence is required. However, this is a tedious task as too many constraints need to be satisfied. Various kinds of approaches for designing a primer have been proposed in the last few decades, but most of them don't have restriction sites on the designed primers and don't satisfy the specificity constraint.

Results: The proposed algorithm imitates nature's process of evolution and genetic operations on chromosomes in order to achieve the optimal solutions, and is a best fit for DNA behavior. Experimental results indicate that the proposed algorithm can find a pair of primers that not only obeys the design properties but also has a specific restriction site and specificity. Gel electrophoresis verifies that the proposed method really can clip out the target sequence.

Availability: A public version of software is available on request from the authors.

Contact: wujs@mail.cse.nsysu.edu.tw

Keywords: Genetic Algorithm, PCR, Primer Design

INTRODUCTION

With the growing number of researches in bioinformatics, many biotechnologies have improved considerably. However, most of them need to amplify the number of DNA sequences, since the experimental results of the biotechnologies are not so clearly recognized by the naked eye using only a small amount of DNA. Consequently DNA cloning is becoming increasingly important. The polymerase chain reaction (PCR) is a way for fast mass duplication of DNA sequences (Mullis and Faloona, 1987). Prior to performing a PCR, a pair of subsequence of DNA called "primers" must be found in order to clip the target in a long DNA sequence.

Due to the properties of oligonucleotides that influence the efficiency of the PCR amplification, the optimal primer design includes criteria (McPherson et al., 1993; Sambrook et al., 2001) such as melting temperatures, length, base composition, 3' termini, repeated and self-complementary sequences, and complementarity between members of a primer pair, etc. To ensure the primers will be efficiently annealed during each cycle of the PCR, the calculated values for the melting temperature of a primer pair should not differ by more than 5 . Primers length should be laid in the interval of 16-28 nucleotides long. The length of the members of a primer pair should not differ more than 3 base pairs (bp). The GC content of the members of a primer pair should be between 40% and 60%. The nature of the 3' end of the primers is crucial. If possible, the 3' end of each primer should be G or C. Actual differences for these criteria are aggregated by weighting sums. The prime design is to construct candidates and to select the best.

Various kinds of approaches in designing primers have been proposed in the last few decades. The manual primer design method can find a primer that fits the primer design constraints. However, it is too time consuming to find a good primer. Besides, accuracy can easily be lost through human errors. Since experiments are expensive, and a minor mistake may cause the experiment to fail, the manual primer design method is considered to be potentially unstable.

To improve the efficiency in primer design, many primer design softwares have been developed for DNA sequences already identified. However, if the designed primer doesn't satisfy a user's requirements, then it's up to the user to modify the primers in order to meet the requirements. The "GeneFisher" system proposed by Meyer et al. (1995) first aligns the unknown DNA sequence with the known DNA sequence, which has the same functions, and then designs the primer that can deal with the unknown DNA sequence. The system Consensus-DEgenerate Hybrid Oligonucleotide Primer

(CODEHOP) proposed by Rose et al. (1998) can find primers for known amino acid sequences. It first calls upon a sequence alignment program, such as FASTA, to find a similar sequence. Then it retransforms the amino acid sequences into DNA sequences, and designs the primer for the similar DNA sequences. Although these systems have provided a convenient way for primer design, most of them don't have restriction sites on the designed primers. Singh et al. (1998) proposed the system "Primer Premier" which provides a good performance on primer designs. It provides a restriction enzyme graph for the DNA sequences by finding the positions of the restriction sites on the aligned DNA sequences. According to the graph, it designs the primer which contains specific restriction sites within the primers. However, it sometimes cannot find the solution, due to the fact that design properties may not be followed. Recently, Kämpke et al. (2001) employed dynamic programming to design primers. Their algorithm can solve the situation of designing multiple primers for multiple target DNA sequences. But, they did not consider some necessary criteria of primer design, such as to avoid "T" or "A" at the 3' end. Besides, in order to reduce the computational complexity, the algorithm removes some potential solutions that may take a long time to compute.

Since the search space of the primer design problem is huge and complex, a better method to solve this problem is needed. Genetic algorithms (GA) are well-known heuristic algorithms based on the imitation of natural systems. Their effectiveness in search and optimization problems has received extensive attention. GA imitates nature's evolutionary process and the genetic operations on chromosomes in order to achieve optimal solutions. In each run of a search, it generates a new and usually a better generation of solutions than the previous run (Goldberg, 1989; Jong, 1989). Due to the nature of the GA process being similar to the evolution of DNA, it is suitable for solving the primer design problem.

In this paper, a new algorithm using the GA for designing primers for PCR is presented. Since most of the design properties are treated as fitness rules, solutions that do not obey the fitness rule are eliminated through competition. The proposed algorithm also searches for restriction sites and specificity. If the restriction sites are closed to a user's requirements, then the rank of this primer is promoted so that it can be listed on the top of the solution ranking set.

The rest of this paper is organized as follows. In next section, we present some essential definitions for the proposed algorithm. In the following section, we describe the process of the proposed algorithm for PCR primer design. Experimental results are given in succeeding section. Discussions are given in fifth section, and finally conclusions are drawn in the last section.

DEFINITION OF THE PROPOSED ALGORITHM

Let G_D be the DNA sequence template, which is denoted as the template of the base-nucleic acid code sequence of DNA. For example, G_D is represented as

$$G_D = \text{AATCGACCAT}\ldots,$$

where A, T, C, and G are the base-nucleic acid codes. $\overline{G_D}$ is denoted as the complement code of the original base-nucleic acid code. The complement of A is T, and vice versa. Similarly, C and G are the complement of each other. For example, G_D, which is described above, its complement $\overline{G_D}$ is

$$\overline{G_D} = \text{TTAGCTGGTA}\ldots$$

The forward primer of G_D is denoted as B_f, and is defined as follows:

$$B_f = \{\, b_i \mid i \text{ is the index of } G_D \text{ between } F_s \text{ and } F_e \,\},$$

where F_s and F_e are respectively denoted as the start index and the end index of B_f in G_D. The reverse primer of G_D is denoted as B_r, and is defined as follows:

$$B_r = \{\, b_i \mid i \text{ is the index of } \overline{G_D} \text{ between } R_s \text{ and } R_e \,\},$$

where R_s and R_e are respectively denoted as the start index and the end index of B_r in $\overline{G_D}$. The individual of the proposed algorithm is denoted as one pair of primers, which is presented as a vector P_t, and is written as

$$P_t = (F_s, F_e, R_s, R_e)$$

Since the four components of the individual are dependent, if one of them is changed by the crossover or mutation process, it may sometimes cause an error, violating the length constraint. To avoid this problem, we transform the dependence of these four components into an independent form. Hence, the individual is transformed from dependent form (F_s, F_e, R_s, R_e) into independent form $(F_s, \alpha, \beta, \gamma)$, and is defined as

$$P_t' = (F_s, \alpha, \beta, \gamma),$$

where α, β, and γ are given as follows:

$$\alpha = (F_e - F_s)$$

$$\beta = (R_s - F_e)$$

$$\gamma = (R_e - R_s)$$

The dependent form represents the actual position of the primer, and the independent form represents the relative position. For example, the dependent form of the individual P_t is (145,164,989,1011). Therefore, the independent form P_t' is (145,19,825,22). Since the dependence of the four components are removed, one of the four components being changed doesn't

cause a violation. Suppose P_1 is a primer of G_D, then the length $|P_1|$ is the sum of the numbers of all base-nucleic acid codes, and is written as

$$|P_1| = \#G + \#C + \#A + \#T$$

The melting temperature of P_1 is $T_m(P_1)$, which is a reference temperature for a primer to perform annealing, and known as the Wallace formula, is written as

$$T_m(P_1) = (\#G + \#C) * 4 + (\#A + \#T) * 2$$

The GC content, $GC(P_1)$, is the ratio of base-nucleic acid codes G and C of sequence P_1. $GC(P_1)$ is written as

$$GC(P_1) = \frac{\#G + \#C}{|D_A|} \times 100\%$$

The specificity of the primer P_1 is denoted as $Uni(P_1)$, which is to examine the annealing position of the primer P_1in the G_D, and it is

$$Uni(P_1) = \begin{cases} 0, & \text{if } P_1 \text{ appear in } G_D \text{ once} \\ 1, & \text{if } P_1 \text{ appear in } G_D \text{ more than once} \end{cases}$$

The termination of primer P_1 is denoted as $Term(P_1)$, which is to examine whether the 3' end of the primer P_1 is G or C, and is defined as

$$\text{Term}(P_1) = \begin{cases} 0, & \text{if 3' end is G, C, CG or GC} \\ 1, & \text{otherwise} \end{cases}$$

THE PROPOSED ALGORITHM FOR PCR THE PRIMER DESIGN

The proposed algorithm for the PCR primer design consists of the initialization process, evaluation process, crossover process, and the mutation process. Figure 1 shows the flowchart of the proposed algorithm. Detailed steps of the proposed GA for the PCR primer design are described as follows:

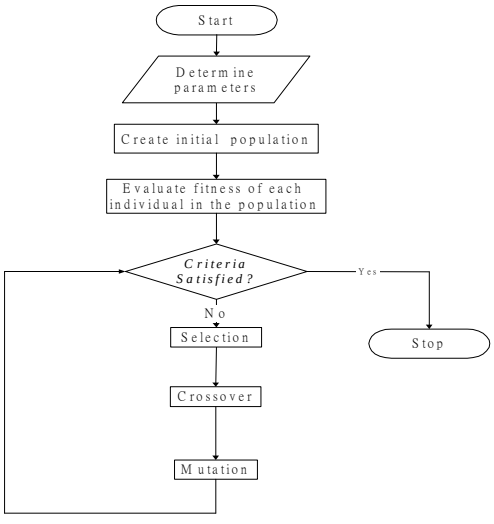

Fig. 1. The flowchart of the proposed algorithm for the PCR primer design

The initialization process randomly generates the initial individuals of population P, which is a set of 500 solutions P_t. In this phase, the F_s of one forward B_f in G_D is generated randomly. Then the α and γ are generated randomly within 18 to 26. The β is generated randomly around the length of the target product. Then the variable F_s, α, β, *and* γ are recorded in the matrix, in order to avoid the duplicate primer pair being generated in the population.

The evaluation process checks whether the solutions satisfy the design constraints or not. When solution P_t is generated, the length of each primer P_t should be within 18 to 26. If the length is less than 18, it is not easily to control the melting temperature. Besides, if one wants to optimize the melting temperature, then the primer must become GC-rich, which may cause a violation of specificity. Suppose that B_f is the forward primer of P_t *and* B_r is the reverse primer of P_t. The length check function is denoted as $leng(P_t)$ and is defined as

$$leng(P_t) = \begin{cases} 0, & \text{if } 18 \le |B_f|, |B_r| \le 26 \\ 1, & \text{otherwise} \end{cases}$$

Besides, the length different function $lengd(P_t)$ between B_f and B_r must be less 3 mer, and it is

$$\text{lengd}(P_t) = \begin{cases} 0, & \text{if ABS}(|B_f| - |B_r|) \le 3 \\ 1, & \text{otherwise} \end{cases}$$

where ABS(.) denotes the absolute value. The evaluation process evaluates the population P based on the design properties, which are considered as constraints. The first constraint, the melting temperature, ensures that the primer pair will smoothly anneal to G_D at the same temperature. The temperature difference Tmd between B_f and B_r is defined as

$$Tmd(P_t) = \begin{cases} 0, \text{if } ABS(Tm(B_f) - Tm(B_r) \le 5 \\ 1, \text{otherwise} \end{cases}$$

The GC content check function GC_p of the Pt is denoted as

$$GC_P(P_t) = \begin{cases} 0, \text{if } 40\% \le GC(B_f), GC(B_r) \le 60\% \\ 1, \text{otherwise} \end{cases}$$

The specificity of the primers should be zero.

$$Uni(B_f) = 0$$

$$Uni(B_r) = 0$$

$$Uni(P_t) = Uni(B_f) + Uni(B_r)$$

The termination of the primers should be zero, too.

$$Term(B_f) = 0$$

$$Term(B_r) = 0$$

$$Term(P_t) = Term(B_f) + Term(B_r)$$

No inverted repeat sequence or self- complementary sequence greater than a 3 base pair (bp) in length should be allowed. The 3' terminal sequences of one

primer should not be able to bind to any site on the other primer. Self-complementary sequences of primers should be avoided.

$$Sc(P_t) = \begin{cases} 0, \text{ if there is no self - complementary} \\ \quad \text{of } B_f \text{ and } B_r \\ 1, \text{ otherwise} \end{cases}$$

In a primer pair, one primer should not be the complement of the other one.

$$PC(P_t) = \begin{cases} 0, \text{ if there is no pair - complementary} \\ \quad \text{of } B_f \text{ and } B_r \\ 1, \text{ otherwise} \end{cases}$$

Since the constraint termination should be obeyed, but because three or more Cs or Gs at the 3'-ends of primers may promote error annealing at G or C-rich sequences, this condition should be avoided. Restriction sites are situated at the primer if possible. The proposed algorithm searches the individual to see if there is a similar restriction site on it. The typical restriction sites used in this paper are listed in the following table.

Restriction enzyme	Sequence
Apa I	GGGCCC
Avr II	CCTAGG
Bam HI	GGATCC
Bgl II	AGATCT
Dra I	TTTAAA

In order to recruit a specific restriction site, the proposed algorithm first checks the individual from 5' end to 3' end for a proceed pattern match, and verifies whether the enzyme is on it or not. If there is a similar restriction site (the matched pattern's length $|P_m|$ is less or equal to the length of the enzyme L_e and more or equal to the length of the enzyme minus 3, $((L_e - 3) \le |P_m| \le L_e)$, than the proposed algorithm adjusts the fitness value to allow the individual most likely to pass the evaluation. The restriction site check function $R_t(P_t)$ is denoted as

$$R_t(P_t) = \begin{cases} 0, \text{ if there exists a restriction site} \\ \quad \text{of } B_f \text{ or } B_r \\ 1, \text{ otherwise} \end{cases}$$

In addition to the examination of the restriction site, the other time consuming activity is specificity. The specificity tries to find an actual position for a primer pair to anneal to. If there are several positions in the target that are not mismatched to other positions of the target sequence for the primer pair to anneal to, then the product will consist of non-specific amplified sequences, so that the result affects the experiments that follow. Due to the time consuming process of examining all individuals in order to reduce the possibility of mis-priming, we use a matrix, which records the position and the specificity fitness value of the position, to speed up the examination process. Based on the simulation, with the use of the matrix the algorithm can run 4-5 times faster than the algorithm without using the matrix.

The fitness value is computed by the design constraints mentioned above, and it is:

$$Fitness(P_t) = leng(P_t) + 3*lengd(P_t) + 3*Tmd(P_t) + 3*GC_P(P_t) + 3*(Term(B_f) + Term(B_r)) + 50*Uni(P_t) + 10*Sc(P_t) + 10*PC(P_t) + R_t(P_t)$$

Selection applies the Roulette Wheel method to allow the individuals with a high weight to have a higher chance to be selected, and sends these two individuals into the mating pool. The weight mentioned here is the inverse of the fitness value, which is calculated in the evaluation process. Figure 2 shows the flowchart of the crossover process and the mutation process.

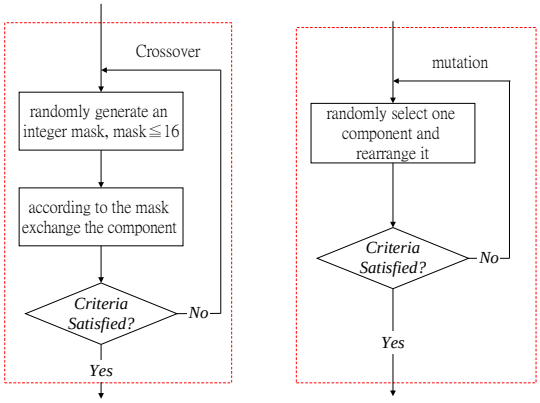

Fig. 2. The flowchart of the crossover and the mutation processes of the proposed algorithm

The crossover process generates a random number R that is smaller than 16. It uses the binary form of R as a mask to decide which components of individuals X and Y should be exchanged. For example, a random number R is $11_{(10)}$ and its binary is $1011_{(2)}$. The first, third, and fourth components of individuals X and Y should be exchanged. Then, the crossover process examines the offspring individuals as to whether the offspring violate the constraints or not. Figure 3 shows an example of the crossover process.

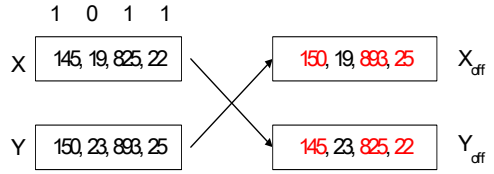

Fig. 3. An example of the crossover process

The four components of X are represented as (145,19,825,22); and the four components of Y are represented as (150,23,893,25). After crossover, offspring X_{off} is (150, 19, 893, 25) and offspring Y_{off} is (145, 23, 825, 22). The mutation process is to randomly change the four components of the selected individual. First, it generates a random number to decide which one component shall mutate. And then the mutation process examines the individual as to whether that individual fits the constraints or not. Figure 4 shows an example of the mutation process.

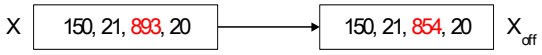

Fig. 4. An example of the mutation process

The four components of individual X are (150, 21, 893, 20). Then, it randomly selects one component to mutate. In this example, the third component is selected and is changed to 854. Hence, the offspring X_{off} is (150, 21, 854, 20).

For the GA, it contains three main parameters: *max_gen*, *M*, and *pop_size*. The parameter *max_gen* is the maximum number of generations allowed for the GA process. The parameter *pop_size* is the size of the population, and it is used for controlling the individual amounts. The mating pool is used for stocking the new individuals produced by the crossover and the mutation process. After evaluation, the individuals are sorted, and only the pop_size amounts of individuals are reserved. The variable M is used to control the size of the mating pool. If the mating pool size is greater than the original population size, then the advantage of the original generation might disappear. There are some minor parameters, like P_e and P_m. P_e is the probability of performing the crossover process, if a random number within 0 and 1 is smaller than P_e, then the process crossover proceeds. Similar to P_e, P_m is the probability of performing the mutation process, if a random number within 0 and 1 is smaller than P_m, then the process crossover proceeds. The higher P_e is, the faster the process is, but the more inexact the solution is. The higher P_m is, the more precise the solution is, but the slower the process becomes. There is an obvious trade-off between these two parameters.

EXPERIMENTS

Materials

cDNA templates

Two cDNA templates were used as our targets for PCR amplification, and subjected to primer designs in our experiments: Pseudomonas mendocina PHA synthase 1 (phaC1), PHA depolymerase (phaZ), and PHA synthase 2 (phaC2) genes, complete cds and *Homo sapiens* CDK2-associated protein 1 (CKD2AP1) coding DNAs (CDs) (GenBank Acc#NM_004642; 523..870). One of the major differences between these two DNA sequences is their length.

Preparation of cDNA templates

Total RNA was first extracted from the *HeLa* cell line. Reverse transcription into cDNAs were followed. The cDNAs were then subjected to being quality checked, quantified and adjusted to an appropriate concentration for further PCR reaction.

Dry dock experiments

The environment

The proposed algorithm was run on a Pentium 4, 1.5G Hz, 128 MB, Windows 2000, and jdk1.4.0. platform.

Results of the dry dock experiments

The proposed algorithm was compared with three famous free primer design softwares (Primer3, GeneFisher, and SGD). In Table 1, the primer pair, which the proposed algorithm finds, contains one similar restriction site on the forward primer. This similar restriction site can be corrected as “GGATCC”, so that the enzyme *Bam* HI can be used and “CCTAGG” (*Avr* II) on the reverse primer. Primer 3 finds a primer pair that satisfies the complementary sequence, but has the highest melting temperature. GeneFisher finds a primer pair, of which the product size is the longest, but it doesn’t have specificity. SGD finds a primer pair, which contains the enzyme “AAATTT” (*Dra* I) on the forward primer and “GGGCCC” (*Apa* I) on the reverse primer, but it doesn’t have specificity, and the complementary sequence is not satisfied.

The results of the second experiment are listed in Table 2. Primer 3 finds a primer pair that has a minimum temperature difference between the pair of primers. Besides, the primer pair also satisfies the complementary sequence. The melting temperatures of the two primers are around the optimal experimental temperature of 40 ^{o}C ~ 60 ^{o}C. The PCR product amplified from primers designed by Primer 3 is 267 bps in length. GeneFisher finds a solution whose length is 252 bps. The melting temperatures are within 40 ^{o}C ~ 60 ^{o}C. The primer pair satisfies the pair complementarity constraint, but it doesn’t satisfy the self-complementary constraint, since the forward primer forms a U-turn on the 3’ end. Besides, the length of each primer is less than 18 bps. SGD finds a solution that has the longest product size.

Table 1. Comparisons of primer design among this study, primer3, GeneFisher, and SGD for the first experiment

	The proposed algorithm	Primer 3	GeneFisher	SGD
Forward Primer (5'->3')	TTTCATCCTGGTAACTCTG	TGCCACTGCTGATCTTCAAC	CTCGAACTGAAGAACGTCA	AACCATTCGTATTCCGCA
Reverse Primer (5'->3')	ATCCGTCTAGAGACTTTCAT	CTGGATTCTTCAGGCTCTGG	AGCCGATTTGTAGCAGGA	TTTGGGCATTCATGAAGG
Position (F)	1814-1832	1968-1987	231-249	1890-1907
Position (R)	2905-2924	3064-3083	1413-1430	2987-3005
Product Size (bp)	1112	1116	1200	1116
Primer Length (F/R) (mer)	19/20	20/20	19/18	18/18
Melting temperature (F/R) (^{o}C)	54/56	60/62	56/54	51/51
Temperature difference (^{o}C)	2	2	2	0
GC-content (F) (%)	42	50	47	44
GC-content (R) (%)	40	55	50	44
3' terminus (F)	G	C	A	A
3' terminus (R)	T	G	A	G
Self- complementarity	No	No	Yes	No
Pair- complementarity	No	No	No	No
Specificity	Yes	No	No	No
Restriction site	'GGTACC' (*Bam* HI)/ 'CCTAGG' (*Avr* II)	N/A	N/A	'AAATTT' (*Dra* I)/ 'GGGCCC' (*Apa* I)

Table 2. Comparisons of primer design among this study, primer3, GeneFisher, and SGD for the second experiment

	The proposed algorithm	Primer 3	GeneFisher	SGD
Product Size (bp)	303	267	252	348
Primer Length (mer)	18/18	18/18	16/17	21/18
GC Content (Forward) (%)	39	61	63	47
GC Content (Reverse) (%)	50	56	53	50
Melting Temperature (Forward) (^{o}C)	50	58	54	53
Melting Temperature (Reverse) (^{o}C)	54	56	52	56
Temperature difference in pair (^{o}C)	4	2	2	3
Specificity	Yes	No	No	No
3' terminus (Forward)	C	A	A	C
3' terminus (Reverse)	T	T	A	T
Self- complementarity	No	No	Yes	Yes
Pair- complementarity	No	No	No	Yes

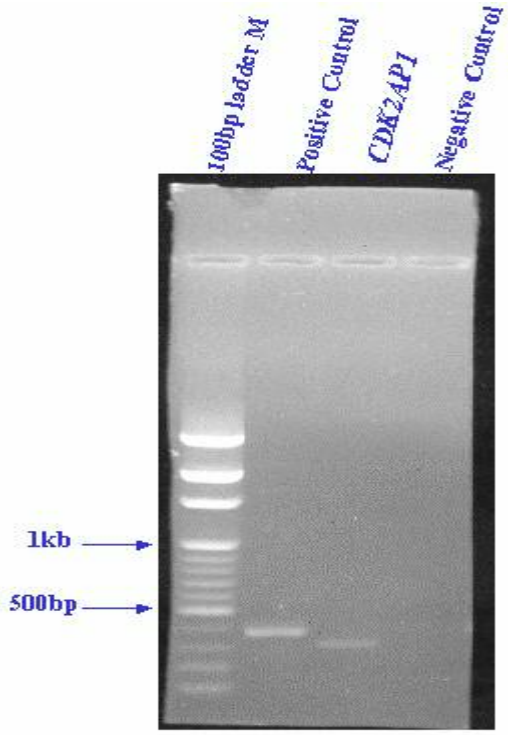

Fig. 5. The result of 1% agarose gel analysis in the second experiment (human CDK2AP1 complete CDs).

However the solution doesn't satisfy the pair complementary sequence and self-complementary constraint. None of the three design softwares can satisfy the specificity.

The product size of the solution obtained from the proposed algorithm is 303 bps. The melting temperatures are within 40 ^{o}C ~ 60 ^{o}C. In addition, the primer pair found by the proposed algorithm also satisfies the complementary sequence. The temperature difference between the pair of primers is the highest among all the design softwares, but still within an acceptable range. It is time consuming to find the primer pair that satisfies specificity and at the same time search for the primer pair that has a restriction site that can be recruited in the primers. However, the difference of the execution time among the proposed algorithm and the other design softwares is insignificant.

Wet experiment

PCR amplifications

One primer pair, designed using our proposed algorithm, and based on CDK2AP1CDs (forward: 5' ATGTCTTACAAACCGAAC 3'; reverse: 5' CAGTCCTCTAGCGTGAAT 3') was used to perform the PCR reaction. The forward and reverse primers spanned from bases 1-18 and 285-303, respectively. A total of 15 µl reagent consisting of 50ng *HeLa* cell cDNA, 10x reaction buffer 1.5µl, dNTP (1.25mM) 1.8µl, forward and reverse primers (10mM) 0.5µl each, *Taq* Polymerase (5 u /µl) 0.05µl and deioned, distilled water, were mixed for PCR amplification. A hot start PCR program was set up as follows: 95 ^{o}C for 5 min; 95 ^{o}C for 45 sec, 55 ^{o}C for 45 sec, 72 ^{o}C for 45sec for 35 cycles; 72 ^{o}C for 10 min or final extension and then ramped to 4 ^{o}C for 10 min.

Agarose gel electrophoresis and amplified *CDK2AP1* PCR product

The amplified PCR products were analyzed in 1% agarose gel electrophoreses containing 0.1 % ethidium bromide. Electrophoreses were conducted in 0.5X TBE buffer. After electrophoreses, the agarose gels were removed from the gel boxes and visualized upon the illumination of UV light. Analysis of amplified *CDK2AP1* PCR product (size= 303 bps) was shown in the middle lane of Figure 5.

The wet experimental results

Figure 5 shows the analysis of the second experimental PCR product, human *CDK2AP1* complete cDNAs. In Figure 5 M shows a 100-bp ladder marker. Positive control (G3PDH) and negative control (water) are also included. At the middle of the gel is the amplified PCR product. Its length, which is approximated to 303 bp, can be clearly observed. The positive control result is at the left. Since there is only one band in positive control,

the experimental result is acceptable. The negative control shows no bands in this experiment, hence it is certain that the experiment is not contaminated.

DISCUSSIONS

Primer design for PCRs is an extremely important issue in any molecular biology laboratory. That is to say, the quality of the primers decides whether or not the experiment will be successful. Many softwares for primer design based on different algorithms have been proposed in the last decade. However, none of these provided a tool to recruit specific restriction sites in a primer pair for further application in molecular cloning. In this study we propose a method using a genetic algorithm that considers all these constraints. When compared to other primer design softwares, our proposed algorithm is able to find a feasible solution that follows all required properties in primer design. Besides, our two experiments show that no matter if the length of the DNA sequence is long or short, the proposed algorithm is able to find a good solution.

One of the most critical properties in primer designs is the specificity. Other properties for primer design, such as the length of the primers can be achieved by using the fixing method-insertion, deletion, and replacement. One method is able to modify the primer pair by extending or decreasing the length of the primer. This method is referred to as the insertion and deletion method. Another process is called the replacement method, and it relies on repairing the result. One can mutate the specific nucleic acid code of the primer to another nucleic acid code. For example, the nucleic acid code 'A' mutates to 'T', hence, it can not be reduced just in order to speed up the evaluation process. In order to satisfy the specificity, the proposed algorithm loosens some constraints, like the 3' termination and the GC content.

If the proposed algorithm ignores the specificity and the restriction site in the second experiment, then the execution time can be reduced to one second. However, in our experiment, although the proposed algorithm spends time on handling the specificity and the restriction site, the running time is about the same as the execution time of the competing softwares without handling the specificity and the restriction site.

For the optimal problem, GA usually converges all solutions into one optimal solution in a short time by using mutation and crossover. Therefore the redundant part can be reduced by examining the matrix. The proposed algorithm uses the matrix to record the position of the primer P_1 and the value of $Uni(P_1)$. The restriction site checking process is similar to the specificity process. It also records the position of the primer P_1 and the value of the matched pattern's length $|P_m|$ to the enzyme sequence.

CONCLUSIONS

We presented a new algorithm using a genetic algorithm for primer design. The proposed algorithm met the design constraints such as melting temperatures, length, base composition, 3' termini, repeated and self-complementary sequences, and complementary sequences between members of a primer pair, especially for recruiting specific restriction sites and specificity. The sequencing result showed that there was a product whose length was approximately the solution that we expected, thus verifying the proposed method really can clip out the target sequence.

REFERENCES

Mullis, K. and Faloona, F. (1987) *Methods in enzymology*, Academic Press, New York and London. **155**, 335

McPherson, M. J., Quirke, P. and Taylor, G. R. (1993 *PCR: A Practical Approach.* Oxford University Press, New York.

Sambrook, J. and Russell, D. W. （2001）*Molecular Cloning 3rd*, Cold Spring Harbor Laboratory Press, New York. **2**, 8.1-8.126

Meyer, F., Schleiermacher, C. and Giegerich, R. (1995) GeneFisher software support for the detection of postulated genes. http://bibiserv.techfak.uni-bielefeld.de/docs/gf_paper.html

Rose, T. M., Schultz, E. R., Henikoff, J. G., Pietrokovski, S., McCallum, C. M. and Henikoff, S. (1998) Consensus-degenerate hybrid oligonucleotide primers for amplification of distantly-related sequences. *Nucleic Acids Res.* **26**, 1628-1635.

Singh, V. K., Mangalam, A. K., Dwived, S. and Naik, S. (1998), Primer premier : Program for design of degenerate primers form a protein sequence. *BioTechniques*. **24**, 318-319.

Kämpke, T., Kieninger, M. and Mecklenburg, M. (2001) Efficient primer design algorithms. *Bioinformatics.* **17**, 214-225.

Goldberg, D. E. (1989) *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, New York.

Jong, K. D. (1988) Learning with genetic algorithms: an overview. *Machine Learning 3.* Kluwer Academic, Hingham, MA. 121-138.