BIOINFORMATICS

Vol. 20 no. 7 2004, pages 1053–1059
DOI: 10.1093/bioinformatics/bth037

Good spaced seeds for homology search

Kwok Pui Choi[1,2,*], Fanfan Zeng[3] and Louxin Zhang[1]

[1]Department of Mathematics, [2]Department of Statistics and Applied Probability and [3]School of Computing, National University of Singapore, Singapore 117543

Received on October 7, 2003; revised October 7, 2003; accepted on November 15, 2003
Advance Access publication February 12, 2004

ABSTRACT

Motivation: Filtration is an important technique used to speed up local alignment as exemplified in the BLAST programs. Recently, Ma *et al*. discovered that better filtering can be achieved by spacing out the matching positions according to a certain pattern, instead of contiguous positions to trigger a local alignment in their PatternHunter program. Such a match pattern is called a spaced seed.

Results: Our numerical computation shows that the ranks of spaced seeds (based on sensitivity) change with the sequences similarity. Since homologous sequences may have diverse similarity, we assess the sensitivity of spaced seeds over a range of similarity levels and present a list of good spaced seeds for facilitating homology search in DNA genomic sequences. We validate that the listed spaced seeds are indeed more sensitive using three arbitrarily chosen pairs of DNA genomic sequences.

Contact: matzlx@nus.edu.sg

1 INTRODUCTION

The program of aligning genomic sequences from different species has been extensively used in various applications, such as gene detection (Yeh *et al*., 2001), inferring SNPs, tandem and segmental duplications, and locating intronic and intergenic regions with potential biological functions (Delcher *et al*., 1999; Hardison *et al*., 1997; Li *et al*., 2001). With the fast growing number of genomes being completely sequenced, sequence alignment has become an indispensable tool in comparative genomics. This unprecedented demand for comparing long genomic DNA sequences has stimulated the need to design faster and yet sensitive alignment tools. In recent years, there has been a surge of alignment programs designed to meet this need for different purposes, e.g. Lipman and Pearson, 1985, Altschul *et al*. (1990, 1997), Huang and Miller (1991), Gish and States (1993), Zhang *et al*. (2000), Ning *et al*. (2001), Schwartz *et al*. (2003), Kent (2002), Ma *et al*. (2002), to name but a few.

One popular approach to speed up alignment is the filtration technique as exemplified in the BLAST programs (Altschul *et al*., 1990). This approach consists of two steps: (i) 'search step'—it first picks up short contiguous regions in the target sequence that have a perfect match in the query sequence and (ii) 'alignment step'—it detects whether each short region obtained in (i) can be extended into a significant alignment, and it outputs this alignment, if so. For example, the BLASTN program of the earliest version first finds perfect matches of consecutive 11 nt bases between a query sequence and a target DNA sequence, and then extends these exact matches into local alignments, keeping those with scores that exceed a preassigned threshold. Another program called BLAT developed by Kent (2002) allows single or near multiple hits of predetermined patterns such as short perfect matches and single almost perfect matches to trigger a local alignment.

Two conflicting factors—search speed and sensitivity are at play in the design of sequence alignment programs when the filtration technique is used. If a smaller k had been used, the search step would have picked up more shorter regions due to chance but many of them would have been discarded in the alignment step, hence an increase in computing time. On the other hand, if a larger k had been used, significant alignment regions without any perfect k contiguous matches would have been missed in the search step, hence a decrease in the sensitivity of the homology search.

Recently, a novel approach in the search step to trigger a local alignment was introduced by Ma *et al*. (2002). Their program PatternHunter (PH) utilizes a single optimal match pattern to improve the alignment sensitivity. Such an innovation is important since the general sequence search aims to identify more homologous sequences, in which the mismatch positions are unknown. More specifically, PH looks for runs of 18 consecutive nucleotide bases in each sequence, in which the nucleotide matches are required at the 11 positions according to the 1s in the string $111*1**1*1**11*111$. Such a pattern is called a spaced seed. Even in a personal computer with moderate memory space, PH is able to compare prokaryotic genomes in seconds, *Arabidopsis* chromosomes in minutes and human or mouse chromosomes in hours (Waterston *et al*., 2002; Scherer *et al*., 2003; Ureta-Vidal *et al*., 2003).

The spaced seed idea in PH motivated several research groups to work on the problem of identifying optimal spaced seeds in different sequence alignment models (Keith *et al*., 2002; Buhler *et al*., 2003; Brejovà *et al*., 2003; Choi and Zhang, 2003). Assuming that the similarity of the sequences

*To whom correspondence should be addressed.

to be aligned follows a Markov chain model, Buhler *et al*. (2003) adapted the dynamical programming technique in Keith *et al*. (2002) to calculate the sensitivity of a spaced seed, from which the optimal spaced seeds can be identified.

Brejovà *et al*. (2003) worked on identifying optimal spaced seeds in the context of detecting homologous coding regions in unannotated genomic sequences. They introduced a 3-period model to take into account the dependence structure of the bases within a codon, and two more hidden Markov models to depict local alignments of coding regions. They modified the dynamical programming technique in Keich *et al*. (2002) to calculate the sensitivity of a spaced seed, from which they identified the optimal spaced seeds for aligning coding regions. We also note that the WABA program (Kent and Zahler, 2000) uses the seed consisting of repeats of $11*$ for aligning coding sequences.

This paper is a sequel to our earlier work (Choi and Zhang, 2003). This work focuses on identifying good spaced seeds under the PH model, where the match in each position is modeled as an independent and identically distributed Bernoulli random variable. While the hidden Markov model approach is well adapted for aligning sequences in the coding regions, the PH model remains relevant when it comes to comparing long genomic sequences or searching a large database that contains many diverse sequences. Furthermore, whether (i) calculating the sensitivity of a spaced seed or (ii) identifying if the optimal spaced seeds are polynomial-time solvable, even under the PH model, have not yet been settled. Choi and Zhang (2003) derived a set of recurrence formulas for computing the sensitivity of a spaced seed and presented some theoretical results for comparing spaced seeds. Based on these results, they proposed a fast heuristic algorithm for identifying optimal spaced seeds.

The objective of this work is to provide flexibility for researchers to customize their choice of spaced seeds in the PH program in terms of the sequence similarity, the desired sensitivity and specificity. For a spaced seed to be of practical use for homology search in a large database, the spaced seed should be optimal or near optimal over a wide range of similarity levels. This is because homologous sequences can have diverse similarity levels ranging from below 65% to over 90%. Therefore, we assess spaced seeds over a range of similarity levels, instead of working on one particular similarity level as previous works, and we provide good spaced seeds over a range of similarity levels and validate them using arbitrarily chosen DNA genomic sequences.

The rest of this paper is divided into three more sections. Section 2 lists good spaced seeds of different weights. Section 3 introduces the method that we used for identifying the good spaced seeds. It is a refinement of the method proposed in Choi and Zhang (2003). In Section 4, we also validate these spaced seeds on several genomic sequence datasets. Our experiments show that the identified spaced seeds of different weights work quite well.

2 GOOD SPACED SEEDS

2.1 Sensitivity of spaced seeds in local alignment

As mentioned before, the BLAST programs look for a perfect match of k contiguous bases that appear in both the query and target sequences in the search step in the filtration method. The novelty of the idea introduced in Ma *et al*. (2002) is that better filtering can be achieved by spacing out the k matching positions. Since we still require only k matches, better filtering is achieved without sacrificing the speed in the search step. Such a pattern of the matching positions is called a 'spaced seed' in their paper. We denote a spaced seed by a string on $\{1, *\}$, where 1s indicate exact match positions; and $*$s indicate positions which are not required to match (called the 'don't care' positions). Suppose that the spaced seed $Q = 1**11*1$ is adopted, then for all 7mer from the query sequence, we require a match at the positions 0, 3, 4 and 6 (where we number the positions of 1s in the spaced seed from 0). For example, if the query and target sequences are respectively `gcaattgccg` and `acgattgctg`, then the 7mer `caattgc` and `attgccg` in the query sequence hit the target sequence at positions 8 and 10, respectively; whereas the 7mer `gcaattg` does not hit the target sequence at all. Alternatively, we specify a spaced seed by the relative positions of the 1s in the seed (Burkhardt and Kärkkäinen, 2001). For example, the seed Q given above has the set of relative positions $\{0, 3, 4, 6\}$. The number of 1s in a seed is called its weight and its overall length is called its length.

To measure the sensitivity of a given spaced seed, we adopt the same probability model (PH model) as in Ma *et al*. (2002). Assume that S' and S'' are two DNA sequences of length n such that the events that S' and S'' are identical at position i (or $S'[i] = S''[i]$) are jointly independent and each event is of probability p. In other words, p measures the level of similarity of the two DNA sequences. By translating a match at a position to 1 and a mismatch to 0, we have the following equivalent formulation.

Let S be a sequence of Bernoulli random variables in which, for $1 \leq i \leq n$, $S[i]$ takes two values 1 and 0 with probabilities p and $1 - p$, respectively. Let Q be a spaced seed of length L and weight w given by its relative position set $\{i_1 = 0, i_2, \ldots, i_w = L - 1\}$. The seed Q is said to hit S at position n if

$$S[n - L + i_j + 1] = 1, \quad j = 1, 2, \ldots, w.$$

Note that we use the ending position as the hitting position.

Let Q_n be the probability that Q hits S before or at position n. Q_n is used to measure the sensitivity of Q for a fixed n. Given weight w and similarity level p and n, a seed Q is said to be optimal if it has the largest hitting probability Q_n among all the spaced seeds (of weight w). Ma *et al*. (2002) chose $p = 70\%$ and $n = 64$ for deriving the default spaced seed in PH by considering the fact that most ungapped homologous regions are typically of size 20–200 bases. In terms of hitting

probability, a spaced seed Q' may lag behind another spaced seed Q'' for small n but it may lead Q'' when n is large enough. Thus, the optimal spaced seeds identified based on n vary with n in general.

Based on a general theorem in Nicodème *et al*. (1999), Buhler *et al*. (2003) showed that, for any spaced seed Q, there exist two positive numbers λ_Q and β_Q such that $\lim_{n\to\infty}(1-Q_n)/[\beta_Q\lambda_Q^n]=1$. In fact, since $(1-Q_n)\le(1-Q_m)(1-Q_{n-m})$ for any $m<n$ (Choi and Zhang, 2003), $\ln(1-Q_n)$ is subadditive and hence $\lim_{n\to\infty}(1-Q_n)^{1/n}$ exists and λ_Q is the limit. This asymptotic theorem leads to a natural question: can one simply select the spaced seeds Q with the minimum λ_Q as good spaced seeds? However, two technical issues need to be resolved first. (i) The λ_Q (depending on Q and p) is not yet known to be polynomial-time computable. In addition, two spaced seeds may have the same λ. For example, take $Q'=11*1$ and $Q''=1*1***1$. We have $1-Q''_{2n}=(1-Q'_n)^2$ (see Choi and Zhang, 2003) and hence, $\lambda_{Q'}=\lambda_{Q''}$ by taking limit. (ii) Most importantly, such good spaced seeds may not be practically useful since they may only catch up other spaced seeds when n is large. We observed that some spaced seeds with small λs will only be able to catch up spaced seeds with larger λs when n is larger than 200.

It is an open question whether Q_n is polynomial-time computable in terms of $L-w$ and n, where L and w are the length and the weight of a given seed Q, respectively. Due to the difficulty in computing Q_n and the huge number $\binom{L-2}{w-2}$ of spaced seeds, identifying the optimal spaced seed seems intractable. Choi and Zhang (2003) presented a set of recurrence formulas to compute Q_n given the spaced seed Q, n and p. Since these formulas will be used in the method described in Section 3 for identifying good spaced seeds, we restate them as follows.

To calculate Q_n, we denote the probability that Q hits S the first time at n by f_n. Define A_n to be the event that Q hits S at n, and $\bar{A}_n$ the complement of A_n. Then, for $n\ge L$, we have

$$f_n = P[\bar{A}_L\bar{A}_{L+1}\cdots\bar{A}_{n-1}A_n]$$

and

$$Q_n = P[A_L\cup\cdots\cup A_n]=\sum_{i=L}^{n} f_i. \tag{1}$$

Given a spaced seed Q, let

$$W_Q=\{x_1,x_2,\ldots,x_m\}$$

be the set of all $m:=2^{L-w}$ distinct strings x_j obtained from the seed Q by filling 1 in the 'care' positions i_k ($1\le k\le w$), i.e. $x_j[i_k]=1$, and 0 or 1 in the 'don't care' positions. For example, $W_{1*1*1}=\{10101,11101,10111,11111\}$. It can be easily seen that Q hits S at n if and only if there is a string $x_j\in W_Q$ which occurs at n. For each x_j, we use $A_n^{(j)}$ to denote the event that x_j occurs at n. Then $A_n=\cup_{1\le j\le m}A_n^{(j)}$ and $A_n^{(j)}$'s are disjoint. For $1\le j\le m$, let

$$f_n^{(j)}=P[\bar{A}_L\bar{A}_{L+1}\cdots\bar{A}_{n-1}A_n^{(j)}].$$

If we use $x_j[a,b]$ to denote the substring of x_j from position a to position b, then,

$$f_n=\sum_{j=1}^{m} f_n^{(j)}. \tag{2}$$

and

$$f_n^{(j)}=(1-Q_{n-L})P[x_j] - \sum_{i=1}^{L-1}\sum_{k\in\Gamma_{i,j}} f_{n-i}^{(k)}P[x_j(L-i+1,L)], \tag{3}$$

where $\Gamma_{i,j}=\{k|x_k[i+1,L]=x_j[1,L-i]\}$, and $P[x_j]$ is the probability that x_j occurs at a specific position (Choi and Zhang, 2003).

REMARK.

(1) Formulas (1)–(3) demonstrate that Q_n depends strongly on the relative positions of the 1s in the seed Q. Since there is no known simple formula to compute Q_n, it is extremely hard to study the sensitivity of a spaced seed theoretically.

(2) Let n be fixed. For a fixed weight w, let $\mathcal{S}_l$ denote the set of spaced seeds of length l and weight w. Our numerical calculation shows that the sensitivity function $\max_{Q\in\mathcal{S}_l}Q_n$, as a function of l, is concave, and it reaches its maximum at about $\min(w/p,2w-1)$.

2.2 Good spaced seeds

Here, unlike previous works, we assess spaced seed over a range of similarity levels. We identify good spaced seeds according to their hitting probability Q_{64} following Ma *et al*. (2002). For each $p_s=65,70,75,80,85$ and 90% and a weight w in the range from 9 to 18, we identify the top 20 spaced seeds Q of weight w according to their hitting probabilities Q_{64} using the method in Section 3. Then we list three/four good spaced seeds for each weight in Table 1. They are simply selected based on their rankings in the six top-20 spaced seeds lists, where each list corresponds to a similarity level. Alternatively, one may take the average of these hitting probabilities to assess spaced seeds. However, the outcome is more or less the same. The sensitivity of the best spaced seeds for each weight ($\le$15) and similarity level is given in Table 2.

There are two competing good spaced seeds of weight 11. More specifically, the spaced seed used in PH is good for lower similarity levels from 61 to 73% (where more computation was done), and thus better for detecting remote homologous sequences; the other is better for higher similarity levels from 74 to 96%, which was also found in Buhler *et al*. (2003). Similarly, there are two competing good spaced seeds of weight 13.

Table 1. Good spaced seeds for different weights

W	Good spaced seeds	Rank under a similarity level (%) 65	70	75	80	85	90
9	11∗11∗1∗1∗∗∗111	1	1	1	1	1	1
	11∗1∗11∗∗∗1∗111	2	2	2	2	2	3
	11∗11∗∗1∗1∗∗111	4	4	4	4	4	4
10	11∗11∗∗∗11∗1∗111	1	1	1	1	1	1
	111∗∗1∗1∗∗11∗111	2	2	4	6	8	9
	11∗11∗∗1∗1∗1∗∗111	8	6	2	2	2	5
11	111∗1∗∗1∗1∗∗11∗111	1	1	2	2	2	3
	111∗∗1∗11∗∗1∗1∗111	2	2	1	1	1	1
	11∗1∗1∗11∗∗1∗∗1111	6	3	3	5	5	6
12	111∗1∗11∗1∗∗11∗111	1	1	1	1	1	1
	111∗1∗∗11∗1∗11∗111	2	2	2	5	3	2
	111∗∗1∗1∗1∗∗11∗1111	6	3	3	2	4	4
13	111∗1∗11∗∗11∗∗1∗1111	2	1	1	2	2	2
	111∗1∗∗11∗1∗∗111∗111	7	2	2	1	1	1
	111∗11∗11∗∗1∗1∗1111	1	4	5	7	8	8
14	111∗111∗∗1∗11∗∗1∗1111	2	1	1	1	1	1
	1111∗1∗∗11∗∗11∗1∗1111	5	2	2	3	3	6
	1111∗1∗1∗11∗∗11∗1111	1	3	7	—	—	—
15	1111∗∗1∗1∗1∗11∗∗11∗1111	—	5	1	1	1	1
	111∗111∗∗1∗11∗∗1∗11111	14	1	2	5	5	4
	111∗111∗1∗11∗1∗∗11111	1	2	—	—	—	—
16	1111∗11∗∗11∗1∗1∗11∗1111	7	1	2	6	13	20
	1111∗∗11∗1∗1∗11∗∗11∗1111	—	7	1	1	1	3
	1111∗1∗∗11∗1∗1∗∗111∗1111	—	—	5	2	2	1
	111∗111∗1∗∗111∗11∗1111	1	9	—	—	—	—
17	1111∗1∗1∗111∗∗11∗11∗1111	6	1	2	4	4	5
	1111∗1∗11∗∗11∗∗11∗1∗11111	—	—	1	1	1	1
	1111∗111∗∗11∗11∗1∗11111	1	3	—	—	—	—
18	1111∗11∗∗111∗1∗1∗11∗11111	—	1	1	2	3	2
	111∗1111∗1∗∗111∗1∗∗11∗1111	—	—	4	3	1	1
	1111∗111∗111∗∗1∗11∗11111	1	4	—	—	—	—

The sign '—' means that the corresponding seed is not among the top 20 seeds for the similarity level.

Table 2. The sensitivity of the best seeds for each weight at a similarity level in the PH model

W	Best spaced seeds	Similarity (%)	Sensitivity
9	11∗11∗1∗1∗∗∗111	65	0.52215
		70	0.72916
		75	0.88951
		80	0.97249
		85	0.99687
		90	0.99991
10	11∗11∗∗∗11∗1∗111	65	0.38093
		70	0.59574
		75	0.80112
		80	0.93685
		85	0.99010
		90	0.99957
11	111∗1∗∗1∗1∗∗11∗111	65	0.26721
		70	0.46712
	111∗∗1∗11∗∗1∗1∗111	75	0.69596
		80	0.88240
		85	0.97601
		90	0.99848
12	111∗1∗11∗1∗∗11∗111	65	0.18385
		70	0.35643
		75	0.58709
		80	0.81206
		85	0.95212
		90	0.99583
13	111∗11∗11∗∗1∗1∗1111	65	0.12327
	111∗1∗11∗∗11∗∗1∗1111	70	0.26475
		75	0.48210
	111∗1∗∗11∗1∗∗111∗111	80	0.73071
		85	0.91747
		90	0.99063
14	1111∗1∗1∗11∗∗11∗1111	65	0.08179
	111∗111∗∗1∗11∗∗1∗1111	70	0.19351
		75	0.38805
		80	0.66455
		85	0.87223
		90	0.98168
15	111∗111∗1∗11∗1∗∗11111	65	0.05340
	111∗111∗∗1∗11∗∗1∗11111	70	0.13867
	1111∗∗1∗1∗1∗11∗1111	75	0.30546
		80	0.55623
		85	0.81601
		90	0.96724

For weight greater than 15, some of the rank 1 spaced seeds for one similarity level are not even among the top 20 spaced seeds for another similarity level. It seems that when the weight of a spaced seed is large, its sensitivity could fluctuate greatly with the sequence similarity level. This suggests that, if a large weight spaced seed is used for fast sequence comparison, it should be selected based on the domain knowledge of the genomes involved. Finally, we notice that the third spaced seed $11*11**1*1**111$ of weight 9 was used in YASS (Noé and Kucherov, 2003), a new similarity search program using multiple hits of a spaced seed for improving search sensitivity.

The optimal spaced seed Q of weight 12 in Table 1 is of particular interest due to the following two reasons. First, Q can be a very good choice for database search since it is optimal over a wide range of similarity levels from 59 to 96%; in addition, its hitting probability $Q_{64} = 0.356430$ is larger than the corresponding one (which is 0.300196) of the consecutive seed of weight 11 when $n = 64$ and $p = 70\%$. Such robustness was further demonstrated in a series of experiments described in Section 4. Second, Q is good for aligning coding regions. Genomic coding sequences have a natural 3-periodic

structure of codons. Point mutation in the third position in a codon often does not change the corresponding amino acid. Therefore, the spaced seeds designed for aligning coding regions usually contain repeating '11∗' patterns that ignore every third position (Buhler *et al*., 2003; Brejovà *et al*., 2003). Q contains four repeats of the pattern '11∗' out of its 6-codon span in the fifth reading frame: 11 1∗1∗11 ∗1∗ ∗11 ∗11 1.

3 METHODS

The idea behind our method for identifying good spaced seeds is that they can be predicted pretty early at $2L$. The choice of $2L$ is intuitively due to: (i) Choi and Zhang (2003) who showed that good spaced seeds have already caught up with the consecutive seed well before $2L$ and (ii) calculating Q_{2L} has taken account of all the possible overlapping structures for any given spaced seed (see Lemma 3.1). This method, a refinement of Choi and Zhang (2003), only keeps the top 20 spaced seeds at $n = 2L$ and then computes their sensitivity up to $n = 64$ based on the recurrence formulas (1)–(3). Then we report the spaced seeds with the highest sensitivity. We have verified that these good spaced seeds are indeed optimal for $w \leq 13$ by exhaustive calculation.

For a spaced seed $Q = q_1 q_2 \cdots q_L$, we define its reversal $Q^{(r)} = q_L q_{L-1} \cdots q_1$. Symmetry consideration leads to the observation: If Q is optimal, then so is $Q^{(r)}$. In our method, we only consider Q or its reverse that contains at least $w/2$ 1s in its first half. We also use the following condition to filter the bad spaced seeds.

CONDITION. *The 'don't care' positions of Q cannot be too clustered, i.e. each block of consecutive 'don't care' positions cannot exceed a small b_0. Here, we set $b_0 = \lceil((L-w)/(w-1)) + 2\rceil$, which is slightly over the average number of 'don't care' positions in each block.*

Note that there are at most $\sum_{i=0}^{w-2}(-1)^i \binom{w-1}{i}\binom{L-(b_0+1)i-2}{w-2}$ spaced seeds satisfying this condition (Choi and Zhang, 2003). Our previous study showed that the above condition reduces the number of spaced seeds to be examined by about 60%. To further reduce computing time, we compute their values of Q_{2L} using the following lemma for those spaced seeds passing this screening condition.

LEMMA 3.1. *For a spaced seed Q of length L and weight w. Let b denote the number of blocks of zeros. Then,*

$$Q_{2L} = (L+1)p^w - Lp^{w+b+1} - \sum_{k=1}^{L-1}(L-k)P[A_L\bar{A}_{L+1}\bar{A}_{L+2}\cdots\bar{A}_{L+k}A_{L+k+1}]. \quad (4)$$

PROOF. The formula follows from

$$Q_{2L} = Q_L + Lf_{L+1} - \sum_{k=1}^{L-1}(L-k)(f_{L+k} - f_{L+k+1})$$

since for any $n > L$,

$$f_n = P[\bar{A}_L\bar{A}_{L+1}\cdots\bar{A}_{n-1}A_n] = P[A_L\bar{A}_{L+1}\cdots\bar{A}_{n-1}\bar{A}_n]$$

and

$$f_n - f_{n+1} = P[A_L\bar{A}_{L+1}\bar{A}_{L+2}\cdots\bar{A}_nA_{n+1}].$$

Employing Lemma 3.1 speeds up the calculation of Q_{2L} significantly. By formula (4), we only need to compute $P[A_L\bar{A}_{L+1}\cdots\bar{A}_{L+k}A_{L+k+1}]$. The gain is that, for each $k \leq L/2$, events A_L and A_{L+k+1} fix a lot of 1 in $S[1, L+k+1]$, thus leaving very few positions to check for $\bar{A}_{L+1}\bar{A}_{L+2}\cdots\bar{A}_{L+k}$ to occur. For k in the range from $L/2+1$ to $\leq L-1$, we compute $P[A_L\bar{A}_{L+1}\cdots\bar{A}_{L+k}A_{L+k+1}]$ as

$$P[A_L\bar{A}_{L+1}\cdots\bar{A}_{L+k}A_{L+k+1}] = P[A_L\bar{A}_{L+1}A_{L+k+1}] - \sum_{j=2}^{k}P[A_L\bar{A}_{L+1}\cdots\bar{A}_{L+j-1}A_{L+j}A_{L+k+1}].$$

The computation is about 10–15 times faster than using formulas (2)–(4). For example, in a 500 MHz PentiumIII PC with 900 MB memory, our current program took less than 12 min, while the previous version took 160 min find the top 10 spaced seeds of weight 11 and length 18.

The Mandala program (Buhler *et al*., 2003) is designed to identify optimal single- or multiple-spaced seeds for Markov models. By default, Mandala employs 'hill-climbing' search technique to produce a single good spaced seed with specified weight and, at most, the specified length. Based on experiments with Mandala for the PH model, we observed that the time taken by Mandala to find 10 multiple spaced seeds is roughly equal to that of ours to find the top 10 spaced seeds of given weight and length. Mandala also supports exhaustive search options. Since it only outputs the best spaced seed of the specified weight, it is not suitable for the seed-ranking study over a range of similarity levels that we have done here.

4 TESTING ON BIOLOGICAL DATA

To validate that the spaced seeds presented in Section 2 are indeed good for real biological data, we conducted a series of homology search experiments using PH on the following three selected datasets, where the first named genomic DNA sequence was used as the query sequence:

(i) *Haemorhiza influenza* (NC_000907, 1.83 Mb) and *Escherichia coli* (NC_000913, 4.63 Mb) genomes.

(ii) A 1.7 Mb segment (from 101 to 102.7 Mb) in mouse chromosome X and a 1 Mb segment (from 79 to 80 Mb) in human chromosome X and

(iii) A 1.3 Mb segment (from 61.7 to 63 Mb) in mouse chromosome 10 and a 2 Mb segment (from 0.5 to 2.5 Mb) in human chromosome 19.

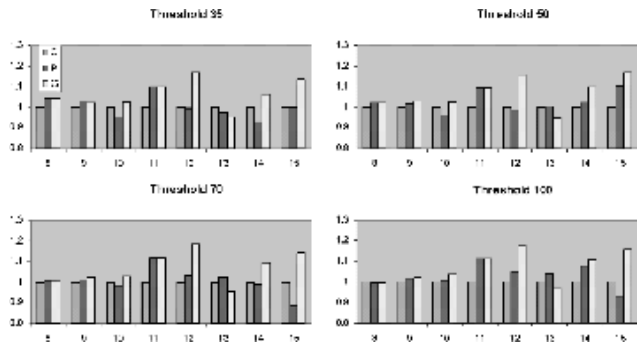

Fig. 1. Relative sensitivity in dataset (i): *H.influenza* versus *E.coli*. Here C, P and G stand for consecutive seed, PH default seed and our good seed, respectively.

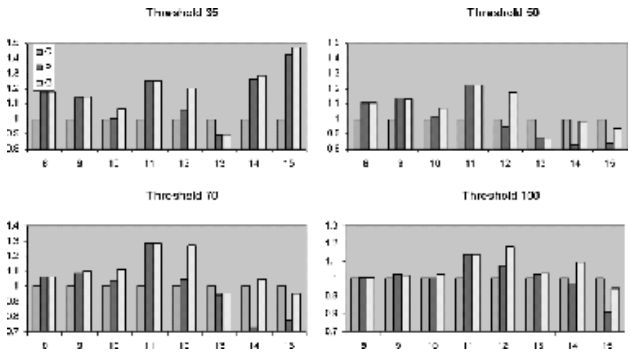

Fig. 2. Relative sensitivity in dataset (ii): a mouse chromosome X segment versus a human chromosome X segment.

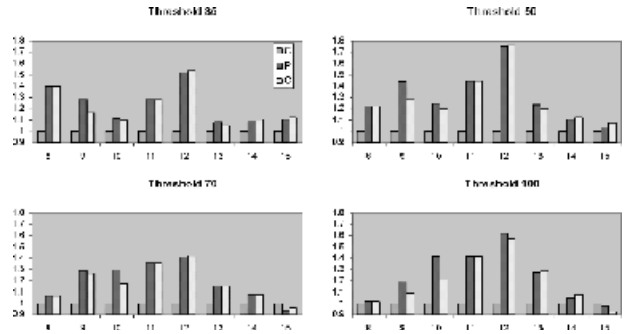

Fig. 3. Relative sensitivity in dataset (iii): a mouse chromosome 10 segment versus a human chromosome 19 segment.

Here, the segments in mouse and human genomes can be obtained using the start and stop positions from www.ncbi.nlm.nih.gov/mapview/map_search.cgi. As we want to test the performance of spaced seeds for similarity search in a large DNA database, we deliberately choose these three datasets without using any genomic information (such as similarity level and coding regions) on these sequences.

The academic version of PH allows users to use either its default spaced seed of weight up to 15 or their own spaced seed of any weight. For each weight w from 8 to 15 and for each test dataset, we run the PH using: (a) the consecutive seed C, (b) its default seed P and (c) the first good spaced seed G of the specified weight w given in Table 1. For each run of the PH, we count the numbers of output as high scoring alignment pairs (HSPs). Using the consecutive seed C as a benchmark, we measure the relative performance r of the seeds P and G of the same weight as

$$r_P(\text{or } r_G) = \frac{\text{number of HSPs using } P \text{ (or } G)}{\text{number of HSPs using } C}.$$

For example, in dataset (i), when the weight is 11 and the threshold is set at 70, the numbers of HSPs found by using the consecutive seed C and our seed G are 780 and 870, respectively. Hence the relative sensitivity of G is $870/780 = 1.115$. Obviously, we set the relative sensitivity of C as 1.

In total, we run the PH 96 times for three datasets, four thresholds (35, 50, 70 and 100) and weights from 8 to 15. Here the threshold specifies the lower bound for the score of matches found by the PH (the seed $11 * *1 * *1 * 1 * 111$ of weight 8 used in our tests was not listed in Table 1). Our test results are summarized in Figures 1–3. The results showed that both the default seeds and our good spaced seeds have a significant advantage over the consecutive seeds of the same weight. Our seeds and the default seeds outperformed the consecutive seeds of the same weight in 82 and 75 cases out of a total of 96, respectively.

For weights 8 and 11, the number of HSPs found by using PH default seeds and our good seeds is exactly the same in all the cases. This suggests that the two seeds are the same for weights 8 and 11. As the results are identical for weights 8 and 11, we only considered cases for the other weights. In the 72 remaining cases, our seeds outperformed PH default seeds in 46 cases.

For datasets (i) and (ii), both our spaced seed and the PH default seed of weight 13 are less sensitive than the consecutive seed. A possible reason could be that the sensitivity of a spaced seed of weight 13 varies with the similarity levels as indicated in Table 1; and the similarity levels of the matching pairs in datasets (i) and (ii) could be quite diverse.

Table 1 shows that the first spaced seed of weight 12 is very robust. This is well reflected in our test results. It performed better than the consecutive seed in all 12 cases, and the PH default seed in only 9 out of 12 cases. Moreover, this seed is more sensitive than the PH default seed in 11 out of 12 cases.

5 CONCLUSION

Given the importance of the database search in biological research, it is crucial to select good spaced seeds for improving the sensitivity of the seeded alignment programs. Using our methods for screening spaced seeds, we assess spaced seeds over a range of similarity levels from 65 to 90% and provide a list of good spaced seeds for different weights ranging from 9 to 18 for facilitating the use of PH for homology search.

We define the optimum span of a spaced seed as the similarity interval in which it is optimal over all the spaced seed of the same weight. Based on our study on spaced seeds, we propose the following three recommendations for choosing a good seed for homology search.

(1) There are two competing spaced seeds of weight 11. The PH default seed $111*1**1*1**11*111$ has optimum span [61%, 73%], while the spaced seed $111**1*11**1*1*111$ found in Buhler *et al*. (2003) has [74%, 96%]. Hence, the former seed is good for remote homology search and the latter for aligning sequences with higher similarity.

(2) The optimal spaced seed $111*1*11*1**11*111$ of weight 12 is good for fast genomic database search. Since it contains one more match position than the current popular default seeds of weight 11, database search using this seed is faster. Most importantly, it is more sensitive than the consecutive seed of weight 11 and has a wide optimum span [59%, 96%]. Such a property is desirable for searching DNA genomic databases in which homologous sequences have diverse similarity. Since it contains four repeats of $11*$ in its 6-codon span (in the reading frame 5), it is also good for aligning coding regions.

(3) When the weight of a spaced seed is large, its sensitivity could fluctuate greatly with the sequence similarity level. In other words, the larger the weight of a spaced seed, the narrower its optimum span. Hence, for database search purpose, a large weight spaced seed should be selected carefully according to the domain knowledge of the genomes involved.

ACKNOWLEDGEMENTS

The authors would like to thank anonymous reviewers for many constructive comments on the work. We also thank M. Li for graciously providing us the PatternHunter program, J. Buhler for the Mandala package, Z. Zhang for helpful discussions and Z. Wang for assistance in PatternHunter. This work is partially supported by BMRC Research Grant BMRC01/1/21/19/140.

REFERENCES

Altschul,S.F., Gish,W., Miller,W., Myers,E. and Lipman,D.J. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Altschul,S.F., Madden,T.L., Schäffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) Gapped Blast and Psi-Blast: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Brejovà,B., Brown,D. and Vinař,T. (2003) Optimal spaced seeds for hidden markov models, with application to homologous coding regions. *Proceedings of the 14th Annual Symposium on Combinatorial Pattern Matching (CPM)*, pp. 42–54.

Buhler,J. (2001) Efficient large-scale sequence comparison by locality-sensitive hashing. *Bioinformatics*, **17**, 419–428.

Buhler,J., Keich,U. and Sun,Y. (2003) Designing seeds for similarity search in genomic DNA. *Proceedings of RECOMB'2003*, pp. 67–75.

Burkhardt,S. and Kärkkäinen,J. (2001) Better filtering with gapped q-grams. *CPM'2001*.

Choi,K.P. and Zhang,L. (2003) Sensitivity analysis and efficient method for identifying optimal spaced seeds. *J. Comp. and Syst. Sci.* (in press).

Delcher,A.L., Kasif,S., Fleischmann,R.D., Peterson,J., White,O. and Salzberg,S.L. (1999) Alignment of whole genomes. *Nucleic Acids Res.*, **27**, 2369–2376.

Gish,W. and States,D.J. (1993) Identification of protein coding regions by database similarity search. *Nat. Genet.*, **3**, 266–272.

Hardison,R.C., Oeltjen,J. and Miller,W. (1997) Long human–mouse sequence alignments reveal novel regulatory elements: a reason to sequence the mouse genome. *Genome Res.*, **7**, 966–969.

Huang,X. and Miller,W. (1991) A time-efficient, linear-space local similarity algorithms. *Adv. Appl. Math.*, **12**, 337–357.

Keich,U., Li,M., Ma,B. and Tromp,J. (2002) On spaced seeds unpublished.

Kent,W.J. (2002) BLAT: the BLAST-like alignment tool. *Genome Res.*, **12**, 656–664.

Kent,W.J. and Zahler,A.M. (2000) Conservation, regulation, synteny, and introns in a large-scale *C.briggsae–C.elegans* genomic alignment. *Genome Res.*, **10**, 1115–1125.

Li,W.-H., Gu,Z., Wang,H. and Nekrutenko,A. (2001) Evolutionary analysis of the human genome. *Nature*, **409**, 847–849.

Lipman,D.J. and Pearson,W.R. (1985) Rapid and sensitive protein similarity searches. *Science*, **227**, 1435–1441.

Ma,B., Tromp,J. and Li,M. (2002) PatternHunter—faster and more sensitive homology search. *Bioinformatics*, **18**, 440–445.

Nicodème,P., Salvy,B. and Flajolet,P. (1999) Motif Sattistics, *Lecture Notes in Computer Sciences*, **1643**, 194–211.

Ning,Z., Cox,A.J. and Mullikin,J.C. (2001) SSAHA: a fast search method for large DNA databases. *Genome Res.*, **11**, 1725–1729.

Noé,L. and Kucherov,G. (2003) YASS: similarity search in DNA sequences. *Technical Report No. 4852*, INRIA, France.

Scherer,S.W., Cheung,J., MacDonald,J.R., Osborne,L.R., Nakabayashi,K., Herwick,J.A., Carson,A.R., Parker-Katiraee,L., Skaug,J., Khaja,R. *et al*. (2003) Human chromosome 7: DNA sequence and biology. *Science*, **300**, 767–772.

Schwartz,S., Kent,W.J., Smith,A., Zhang,Z., Baertsch,R., Hardison,R.C., Haussler,D. and Miller,W. (2003) Human-mouse alignments with BLASTZ. *Genome Res.*, **13**, 103–107.

Ureta-Vidal,A., Ettwiller,L. and Birney,E. (2003) Comparative genomics: genome-wide analysis in metazoan eukaryotes. *Nat. Rev. Genet.*, **13**, 251–262.

Waterston,R.H., Lindblad-Toh,K., Birney,E., Rogers,J., Abril,J.F., Agarwal,P., Agarwala,R., Ainscough,R., Alexandersson,M. and An,P. (2002) Initial sequencing and comparative analysis of the mouse genome. *Nature*, **420**, 520–562.

Yeh,R.F., Lim,L.P. and Burge,C.B. (2001) Computational inference of homologous gene structures in the human genome. *Genome Res.*, **11**, 803–816.

Zhang,Z., Schwartz,S., Wagner,L. and Miller,W. (2000) A greedy algorithm for aligning DNA sequences. *J. Comput. Biol.*, **7**, 203–214.