Multiple Seeds Sensitivity Using A single Seed With Threshold*

Lavinia Egidi[1] and Giovanni Manzini[1]

[1]Computer Science Institute, University of Eastern Piedmont, Italy.

Abstract

Spaced seeds are a fundamental tool for similarity search in biosequences. The best sensitivity/selectivity trade-offs are obtained using many seeds simultaneously: this is known as the *multiple seed* approach. Unfortunately, spaced seeds use a large amount of memory and the available RAM is a practical limit to the number of seeds one can use simultaneously.

Inspired by some recent results on lossless seeds, we revisit the approach of using a single spaced seed and considering two regions homologous if the seed hits in at least t sufficiently close positions. We show that by choosing the locations of the don't care symbols in the seed using quadratic residues modulo a prime number, we derive single seeds that when used with a threshold $t > 1$ have competitive sensitivity/selectivity trade-offs, indeed close to the best multiple seeds known in the literature. In addition, the choice of the threshold t can be adjusted to modify sensitivity and selectivity a posteriori, thus enabling a more accurate search in the specific instance at issue. The seeds we propose also exhibit robustness and allow flexibility in usage.

1 Introduction

Spaced seeds are a fundamental tool for approximate string matching [6, 22]. A good spaced seed must be able to detect all homologous regions (high sensitivity) and have a small ratio of false positives (high selectivity). After almost a decade of research it is now generally accepted that the best results are obtained using multiple seeds [4, 10, 15, 20] that have a much higher sensitivity than single seeds with the same selectivity.

In their seminal paper [20], the authors highlighted two major issues hindering the widespread use of multiple seeds. The first one was the difficulty of finding optimal multiple seeds: there are exponentially many candidate sets of seeds to consider and evaluating the sensitivity of each set also takes exponential time. Researchers have addressed this problem developing heuristics that provide multiple seeds which are sub-optimal but still very effective in practice. The second issue mentioned in [20] is the huge space requirements of multiple seeds. Since a different hash table is used for each seed, using k seeds decreases by a factor k the size of the largest sequence that can be searched in one shot. Unless a completely new indexing scheme is found to replace hash tables, this limitation of multiple seeds is unavoidable.

Another approach for using spaced seeds is to choose a threshold $t > 1$ and consider two regions homologous only if the seed hits in t sufficiently close positions. This idea was mentioned in the very first papers on spaced seeds [6, 22] and analyzed in the wider context of vector seeds in [3]. More recently, [10] used multiple spaced seeds with threshold $t = 2$ (in [1] a threshold $t = 2$ was used for non gapped seeds).

*Postprint version. The final publication is available on `http://dx.doi.org/10.1142/S0219720015500110`

Supported by recent theoretical results on lossless seeds [11] in this paper we propose and analyze spaced seeds whose shape is derived by non-elementary number theoretic properties of quadratic residues modulo a prime number. We call them *Quadratic Residue seeds*, or simply *QR-seeds*.

We show that using this approach a single seed with the appropriate threshold can achieve sensitivity close to the one of multiple seeds with essentially the same selectivity. The value of the threshold can be adjusted *in itinere* in order to achieve best results in a search. Moreover, whereas other seeds available in the literature have a structure that depends on parameters of the specific search (length of the homologous region and probability that it contains matches), QR-seeds are designed independently of such parameters, which is very convenient since estimates of the probability of matches are not always available *a priori*. Finally, since we use a single seed the space requirements noted in [20] are not an issue here.

The paper is structured as follows. In Section 2 we state the basic definitions, outline the setting and specify precisely the problem we address. In Section 2.1 we give a formal definition of selectivity for multiple seeds and formulas for upper and lower bounds to it. In Section 3, we introduce QR-seeds. In Section 3.1 we propose a way to estimate the selectivity of QR-seeds when used with threshold t, and we compute upper and lower bounds according to the methods introduced in Section 2.1. Section 3.2 is devoted to the evaluation of the sensitivity of QR-seeds when used with threshold t. Finally, in Section 4 we compare the sensitivity/selectivity trade-off of the seeds we propose with those available in the literature. The final section recaps our results. All software tools and seeds used in pur analysis are available on the web [13].

2 Spaced Seed design

We consider the standard Bernoulli model [20] in which a homologous region is represented by a binary i.i.d. string in which **1** represents a match and **0** a mismatch. Two important problem parameters are the length of the region, denoted by N, and the probability of a match inside the region, denoted by q and usually called the *similarity* level. Outside homologous regions matches occur by chance with probability σ^{-1}, where σ is the size of the underlying alphabet. Hence, non homologous regions are also binary i.i.d. strings with the difference that **1**'s occur with probability σ^{-1}.

A spaced seed is a binary string where `1` represents a "required match" position and `0` a "don't care" position. We say that a seed s *hits* a region R at position i, with $0 \leq i \leq |R| - |s|$, if $s[j] = \texttt{1}$ implies $R[i + j] = \mathbf{1}$. Thus, we must have a match in R corresponding to each "required match" position in s. In the standard single seed setting a region is marked as "probably homologous" if s hits R in at least one position i. Since a seed may hit also outside homologous regions, every time a region is marked as homologous this hypothesis is verified via Smith-Waterman dynamic programming.

This simple scheme can be generalized in two ways. In the *multiple seed* setting, we are given a set of seeds $M = \{s_1, s_2, \ldots, s_k\}$ and we mark a region R as "probably homologous" if at least one s_j hits R. In the *seed with threshold* setting, we are given a seed s, a threshold $t > 1$ and a maximum distance d, and we mark region R as "probably homologous" if there are at least t positions $i_1 < i_2 < \cdots < i_t$ such that $i_t - i_1 \leq d$ and s hits R at positions $i_1, i_2, \ldots, i_t$. In both generalizations, the dynamic programming verification phase is still necessary.

In all of the above settings, we define the *sensitivity* of a (multiple) seed as the probability that the (multiple) seed detects a length-N homologous region randomly generated according to the Bernoulli model with match probability q (so the sensitivity always depends on N and q). Clearly, higher sensitivity is desirable since it implies that a smaller number of homologous regions go undetected. However, we cannot achieve high sensitivity by simply using seeds with only a few "required match" positions since the overall running time of the algorithm also depends on the number of *false positives*, that is, the number of times in which the algorithm marks a region as probably homologous only to be proved wrong by the (time consuming) verification phase.

The main goal of seed design is therefore to find a (multiple) seed with high sensitivity and producing a small number of false positives. Note that the sensitivity depends on the region length N and similarity

q, while the number of false positives depends on the probability σ^{-1} of a match outside a homologous region. Since sensitivity and false matches are related, we usually fix the three parameters N, q, and σ and compare the sensitivity of algorithms producing the same average number of false positives, or we compare the average number of false positives of algorithms with the same sensitivity.

Setting aside the problem of finding "good" seeds, the main disadvantage of multiple seeds is that a separate data structure, either a hash table or a modified suffix array[8, 14], is used for each seed, resulting in a huge overall space usage. The main disadvantage of seeds with threshold is that finding the occurrence of t close hits is more complex than just finding all the hits; however this is a well studied problem (see [9] and references therein) with efficient solutions especially for small t.

2.1 Selectivity of a seed

Under our assumption that non-homologous regions are random binary i.i.d. strings, the average number of false positives produced by a seed is proportional to the probability that the seed incorrectly marks a single position as "probably homologous". Since in non-homologous regions a single character match occurs with probability σ^{-1}, for a single seed (without threshold) the probability of a false positive is equal to σ^{-w}, where w is the *weight* of the seed, that is, the number of 1's it contains. This observation is the basis of a large body of literature in which the objective is to find (single) seeds with high sensitivity and the largest possible weight [22, 18, 21, 23, 12].

To compare the effectiveness of seeds of different kinds, we define the *selectivity* of a seed S as

$$\mathsf{sel}(S) \quad =_{\text{def}} \quad -\log_\sigma \Pr(\mathsf{false\ positive}). \tag{1}$$

Notice that, the higher is the selectivity the lower is the (average) number of false positives generated by a seed. By the above discussion, in our Bernoulli i.i.d. model, the selectivity of a single seed coincides with its weight. For other kinds of seeds the selectivity is harder to compute, even in this simple model. For a multiple seed $M = \{s_1, s_2, \ldots, s_k\}$, the probability of a false positive is given by

$$\Pr(\mathsf{false\ positive}) = \Pr\left(\bigcup_{j=1}^{k} \{s_j \text{ hits}\}\right).$$

By the inclusion/exclusion principle, if w_j denotes the weight of seed s_j, such probability is upper bounded by

$$\Pr(\mathsf{false\ positive}) \leq \sum_{j=1}^{k} \Pr(s_j \text{ hits}) = \sum_{j=1}^{k} \sigma^{-w_j}. \tag{2}$$

Consequently, if all $s_j \in M$ have the same weight $w_j = w$, it is

$$\mathsf{sel}(M) = \ -\log_\sigma(\Pr(\mathsf{false\ positive})) \ \geq w - \log_\sigma k.$$

Indeed, as observed in [20], for DNA sequences where $\sigma = 4$, quadrupling the number of seeds is roughly equivalent to reducing by one the number of 1's in a single seed. Note that (2) is just a special case of the inclusion/exclusion principle. For $i = 1, \ldots, k$, we define

$$B_i = \sum_{\substack{H \subseteq \{1,\ldots,k\} \\ |H|=i}} \Pr\left(\cap_{j \in H} s_j \text{ hits}\right), \tag{3}$$

that is, B_i is the sum over all subsets of M of size i of the probability that all seeds in the subset hit. Then, for odd positive $\ell \leq k$ we have the upper bound

$$\Pr(\mathsf{false\ positive}) \leq \sum_{i=1}^{\ell} (-1)^{i-1} B_i, \tag{4}$$

for even positive $\ell \leq k$ we have the lower bound

$$\Pr(\mathsf{false\ positive}) \geq \sum_{i=1}^{\ell} (-1)^{i-1} B_i, \tag{5}$$

and for $\ell = k$ the alternating sum equals $\Pr(\mathsf{false\ positive})$ by the inclusion–exclusion principle.

Inequalities (4) and (5) are also known as Bonferroni inequalities (see [2] page 23 or [7] Section 4.1). When k is large, computing the exact probability using the inclusion–exclusion principle becomes prohibitive. Bonferroni inequalities provide convenient, less computationally intensive, upper and lower bounds. On the other hand, it must be noticed that there is no guarantee that the approximations given by Bonferroni inequalities become more accurate as ℓ increases. In particular, the upper bounds can diverge and the lower bounds can be negative (cf. for instance, [24]). Using (4) and (5) we derive in Section 3.1 upper and lower bounds on the selectivity of any multiple seed M.

3 Quadratic Residue seeds

For any odd prime $p \geq 11$ we consider the seed S_p of length p, such that $S_p[j] = \mathtt{0}$ if and only if j is a Quadratic Residue modulo p, that is, there exists i, $0 < i < p$, such that $i^2 \bmod p = j$. For example, for $p = 11$ the squares modulo 11 are $\{1, 3, 4, 5, 9\}$ so it is $S_{11} = \mathtt{10100011101}$ (see [11] for more details). By elementary algebra we know that S_p contains $\frac{p+1}{2}$ $\mathtt{1}$'s and $\frac{p-1}{2}$ $\mathtt{0}$'s. In the following we call S_p a QR-seed.

The fact that makes S_p interesting for seed design is that the distribution of the $\mathtt{0}$'s in S_p has many remarkable properties. For example, setting $n = 1$ in Theorem 4.1 in [11] we get:

Theorem 1 *Let $p \geq 11$ be a prime such that $(p \bmod 4) = 3$, and let R denote a binary string of length $2p - 1$ containing exactly two $\mathbf{0}$'s. Then S_p hits R in at least $(p - 3)/4$ positions.*

For a binary string R containing more than two $\mathbf{0}$'s a similar result still holds, but only for sufficiently large primes (see Theorem 5.2 in [11]). These results combined with an analysis of false positive ratios, show that QR-seeds used with a threshold $t > 1$ are extremely effective *lossless* seeds.

In this paper, we consider the more practical setting of lossy seeds and analyze the trade-off offered by QR-seeds in terms of sensitivity vs selectivity. The results in [11] suggest to consider the seed S_p with a threshold $t \leq (p - 3)/4$, and maximum distance $d = p - 1$ (notice that in Theorem 1 the hits occur at distance at most $p - 1$). Hence, in the following we will only consider the parameters p and t since the maximum distance will be always implicitly assumed to be $p - 1$.

3.1 Selectivity of QR-seeds

In the following we write $\langle S_p, t\rangle$ to denote the QR-seed S_p used with threshold $t \leq (p-3)/4$ and maximum distance $p - 1$. An important observation for our analysis is that to every seed $\langle S_p, t\rangle$ we can associate an equivalent multiple seed. Given $\langle S_p, t\rangle$ we consider all t-uples of distinct shifted copies of S_p, with maximum shift $p - 1$, and for each t-uple we consider the bitwise OR of its elements (see Figure 1 for an example for $p = 11$, $t = 2$). The resulting set contains $\binom{p-1}{t-1}$ binary strings and will be denoted by $M(p, t)$. Such set is equivalent to $\langle S_p, t\rangle$ in the sense that S_p hits at least t times, if and only if one of the strings in $M(p, t)$ hits. In view of this equivalence it is

$$\mathsf{sel}\Big(\langle S_p, t\rangle\Big) \quad = \quad \mathsf{sel}\Big(M(p, t)\Big). \tag{6}$$

We can now estimate $\mathsf{sel}(\langle S_p, t\rangle)$ using (6) and Bonferroni's inequalities. Assuming the values B_i are defined for the multiple seed $M(p, t)$ as in (3), for odd ℓ we have

$$\mathsf{sel}(\langle S_p, t\rangle) \ \geq \ -\log_\sigma(B_1 - B_2 + \cdots + B_\ell),$$

while for even ℓ, whenever $B_1 - B_2 + \cdots - B_\ell > 0$, we have

$$\mathsf{sel}(\langle S_p, t\rangle) \;\leq\; -\log_\sigma(B_1 - B_2 + \cdots - B_\ell). \tag{7}$$

Table 1 reports the bounds on the selectivity of QR-seeds derived by the above inequalities for $\ell \leq 5$ (for $p = 11, \ldots, 17$) and $\ell \leq 3$ (for $p = 19, \ldots, 41$). We only consider thresholds $t \leq 5$, since for larger t the size of the equivalent multiset $M(p, t)$ becomes too large to use Bonferroni inequalities. For the largest values of p and t Table 1 only reports a lower bound on the selectivity. This happens because the arguments of the logarithm in (7) are negative, so we cannot derive a valid upper bound (recall the discussion on Bonferroni inequalities at the end of Section 2.1).

```
 10100011101    10100011101     10100011101                    10100011101
10100011101   10100011101    10100011101        ···     10100011101
111100111111  1010101111101  10110111111101              10100011101010100011101
```

Figure 1: **Construction of the multiple seed $M(11, 2)$ equivalent to the QR-seed $\langle 11, 2\rangle$.** To build $M(11, 2)$ we consider all possible pairs of shifted copies of S_{11}. Here we show only the first three and the last one but there are 10 of them since the maximum distance is $11 - 1 = 10$. For each pair we compute the bitwise OR: the resulting set of strings in the third row (with gray background in the figure) is $M(11, 2)$.

	2	3	4	5
11	8.47– 8.47			
13	9.83– 9.83			
17	12.69–12.69	14.57–15.64		
19	14.41–14.41	16.99–17.20	18.88–	
23	17.31–17.31	20.13–20.33	22.12–	23.52–
29	21.62–21.62	25.17–25.28	27.46–30.27	29.12–
31	23.19–23.19	27.02–27.08	29.62–30.88	31.41–
37	27.59–27.59	32.16–32.22	34.82–36.07	36.59–
41	30.43–30.43	34.89–34.97	37.39–	39.01–

Table 1: Upper and lower bounds to the selectivity of QR-seeds for $p = 11, \ldots, 41$, $\sigma = 4$, and thresholds $t = 2, 3, 4, 5$.

From Table 1 we see, for example, that the QR-seed $\langle 37, 2\rangle$ has a selectivity ≈ 27.59, thus slightly smaller than a single seed with weight 28, and slightly larger than a single seed with weight 27. Note that although the selectivity of a single seed does not depend on the alphabet size, our experiments suggest that the selectivity of multiple seeds and QR-seeds does. We tested the cases $\sigma = 20$ and $\sigma = 200$ and found that the selectivity of QR-seeds increases with the alphabet size (see Table 2 in the appendix), although we can offer so far no mathematical proof or no intuitive explanation of why it is so.

3.2 Sensitivity of QR-seeds

As we mentioned in Section 1, the sensitivity of a seed is the probability that it detects a homologous region of length N in which there is a match with probability q; the parameter q is called the similarity level of the region. Following the recent literature on spaced seeds [16, 17], we consider N between 50 and 200 and q in the range $0.75, \ldots, 0.95$. Also following the literature we report the sensitivity as a percentage computed multiplying by 100 the probability of detecting a homologous region.

The algorithm **DP-k-hits** from [20] can be used to compute the sensitivity of a seed with threshold $t > 1$ (see also the slightly more general Dynamic Programming algorithm in [3]). However, that algorithm assumes that the t hits can be anywhere inside the length-N region. Since for the seeds $\langle S_p, t\rangle$ we require that the t hits are within distance $p - 1$, we can use that algorithm only for relatively small regions

($N < 2p$). For larger regions, we considered the multiple seed $M(p,t)$ equivalent to $\langle S_p, t\rangle$ and we computed its sensitivity using the algorithm **DP-hits** also from [20]. Unfortunately such algorithm is exponential in time and space and we were able to use it only for $p \leq 23$. For larger p, we estimated the sensitivity by generating W random binary strings according to the Bernoulli model with parameter q, and counting how many of them were detected by $\langle S_p, t\rangle$. We used $W = 10^7$ since we found that, for $p \leq 23$, $W = 10^6$ suffices to estimate up to the third significant digit the true sensitivity reported by **DP-hits**, and, even for larger p, increasing W from 10^6 to 10^7 does not significantly affect the estimate of the sensitivity.

Figure 2 shows the (estimated) sensitivity of some QR-seeds as a function of the similarity level for region length $N = 50$ and $N = 100$. Other plots for different values of q and N are given in Fig. 7 in the appendix. For each $\langle S_p, t\rangle$ we report also the lower bound on its selectivity taken from Table 1 (in the legend seeds are sorted by increasing selectivity). We see that, in general, a seed with a lower selectivity has a higher sensitivity for a given similarity (compare for example $\langle 23, 2\rangle$ and $\langle 23, 3\rangle$ on the plot for $N = 50$). This means that the seed with the lowest selectivity generates more false positives but is able to detect a larger number of homologous regions. However, there is an exception: for $N = 50$ seed $\langle 29, 2\rangle$ has a smaller selectivity than $\langle 23, 4\rangle$ and also a smaller sensitivity for any given similarity level. This means that for $N = 50$ $\langle 23, 2\rangle$ always outperforms $\langle 29, 2\rangle$. The reason for this phenomenon is that the length of the region $N = 50$ is relatively short compared to the length of the seed S_{29}. Indeed, Theorem 1 establishes the effectiveness of $\langle S_p, t\rangle$ as a lossless seed for $t > 1$ for a region of length at least $2p - 1$. The intuitive reason is that if the region has size $N < 2p - 1$, we can have multiple hits inside the region at distance at most $N - p < p - 1$ whereas the nice theoretical properties of QR-seeds hold assuming the hits can be at distance up to $p - 1$. This property reflects also QR-seeds used in the lossy settings since, also in other experiments not reported here, we found that, regardless of the threshold, the seed *Spt* is not competitive when used in regions of size $N < 2p$.

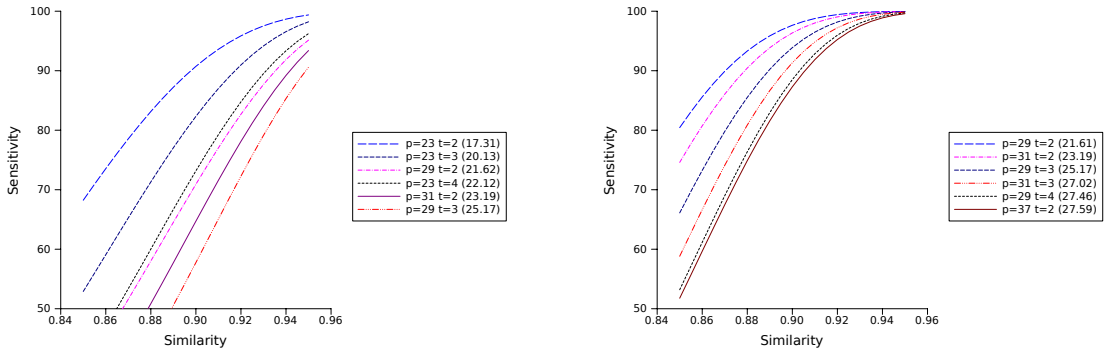

Figure 2: **Sensitivity vs Similarity for some QR-seeds.** Sensitivity as a function of the similarity q for $q = 0.85 \ldots 0.95$ and region length $N = 50$ (left) and $N = 100$ (right).

Figure 2 shows that by varying p and t we get, for any given similarity level, different trade-offs between selectivity and sensitivity. To establish whether QR-seeds are of practical interest in the next sections we compare these trade-offs with those of best (multiple) seeds known in the literature.

3.3 QR-seeds with threshold $t = 1$

Since QR-seeds were originally designed to be used with a threshold $t > 1$, we expect that even in the lossy setting they are competitive especially for $t > 1$. To verify this intuition we compare the sensitivity/selectivity tradeoff of QR-seeds with $t = 1$ and $t > 1$. Recall by Section 3.1 that the selectivity of $\langle S_p, 1\rangle$ is equal to the weight of S_p, that is, $(p + 1)/2$. Figure 3 compares selectivity and sensitivity of QR-seeds for regions of length $N = 50$ and $N = 150$ and similarity $q = 0.90$. We can clearly see a qualitative difference between $t = 1$ and $t > 1$: seeds with the similar selectivity have a much higher

sensitivity for $t > 1$. Note also that while there are significant differences between $t = 1$ and $t = 2$, seeds with $t = 3$ or $t = 4$ do not significantly outperform seeds with $t = 2$.

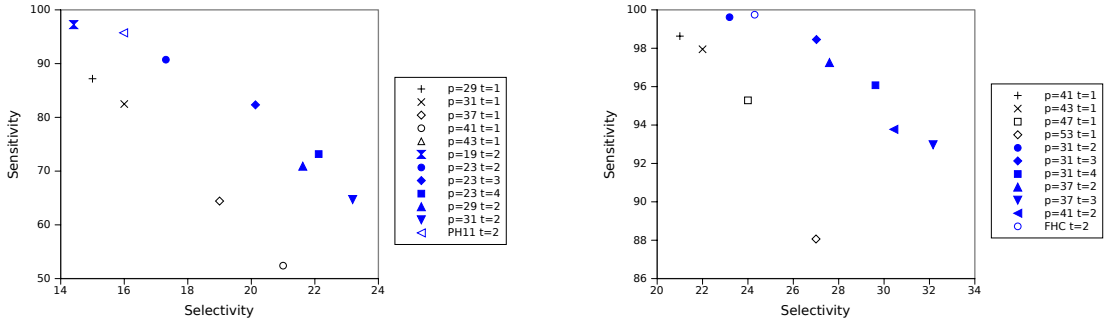

Figure 3: **Comparison of QR-seeds for $t = 1$ vs $t > 1$.** The diagrams show the sensitivity/selectivity tradeoffs of QR-seeds for different values of the threshold t for similarity $q = 0.90$ and region length $N = 50$ (left) and $N = 150$ (right). The diagrams also show the sensitivity/selectivity tradeoffs for two seeds (PH11 and FHC) designed for $t = 1$ when used with $t = 2$ (see text).

To get a complete picture, we considered also two seeds designed to work with $t = 1$, namely the Pattern Hunter [22] seed PH11 = **111010010100110111** and the seed FHC = **1110101000101001010010011001111** we generated with the tool FastHC tool [17]. We analyzed the performance of these seeds with $t = 2$ and maximum hit distance equal to the length of the seed minus one (as for QR-seeds). As we can see from Figure 3, their sensitivity/selectivity tradeoff is similar to that of QR-seeds with $t > 1$. This experiment suggests that, despite their nice theoretical properties in the lossless setting, QR-seeds are not necessarily the best possible choice as lossy seeds when using a threshold $t > 1$. We plan to investigate this issue in a future research.

4 Comparison of QR-seeds with multiple seeds

We compare the sensitivity/selectivity trade-off of our seeds with those of the multiple seeds available in the literature. For computing sensitivity and selectivity of a multiple seed we need its complete definition, that is, the individual seeds it contains. Among recent homology search tools based on multiple seeds, the only one giving the complete definition of its seeds is BFAST [15]. In addition, we tested the multiple seeds generated by the FastHC tool [17]. The experiments in [17] show that FastHC generates multiple seeds with a better sensitivity than those produced by older tools, such as Pattern Hunter 2 [20], Mandala [5], Iedera [19], and SpEED [16]. The improvement in sensitivity offered by FastHC is not marked but it is consistent over all classes of multiple seeds considered. In the following we refer to the seeds generated by FastHC as FHC seeds.

Recall from Section 2 that the size of a multiple seed is the number of individual seeds in it. We have estimated the selectivity of FHC and BFAST seeds for $\sigma = 4$ using Bonferroni inequalities (4) and (5) with $\ell = 5$ and $\ell = 4$ respectively. We found that for each multiple seed S of size k, the selectivity $\mathsf{sel}(S)$ was within 0.1% of the value $w - \log_\sigma k$, where w is the weight of the seeds in S (all FHC/BFAST multiple seeds consist of seeds with the same weight). The sensitivity of BFAST and FHC multiple seeds was taken from [17] or computed using the **DP-hits** algorithm [20]. Since **DP-hits** uses exponential space, on our 64GB machine we could not run this algorithm for region length $N \geq 100$. Thus, for large N we replaced the **DP-hits** algorithm with an estimate of the sensitivity obtained generating random strings as described in Section 3.2.

The diagrams in Figure 4 compare selectivity and sensitivity of QR-seeds with those of FHC multiple seeds of size 1, 2, 3, 4, 5, and 10, and BFAST multiple seeds of size 10 for region length $N = 50$ and similarity $q = 0.90$ (left), $q = 0.95$ (right). The individual FHC and BFAST seeds all have weight 22. FHC seeds were taken from [17] or generated using the FastHC tool. Note that FHC multiple seeds are

designed ad-hoc for each pair N, q. So, for example, the set of ten seeds FHC 10 used in the left diagram is different from the set FHC 10 used in the right diagram, see [17] for details. BFAST multiple seeds are optimized for the region length only, so in both diagrams we use the multiple seed given in Table 1 in [15]. QR-seeds are the simplest: for each prime p there is a unique seed S_p which is used, possibly with different thresholds t, in all experiments.

The diagrams in Figure 5 compare selectivity and sensitivity of QR-seeds with those of FHC multiple seeds of size 1, 2, 4, 8, and 16, for $N = 150$ and 200 and $q = 0.90$. We used the FHC multiple seeds from [17] in which the individual seeds have weight 28. The diagram in Figure 6 compares selectivity and sensitivity of QR-seeds with those of FHC multiple seeds of size 1, 2, 4, 8, and 16, for $N = 100$ and $q = 0.90$. In addition to the multiple seeds from [17], in which the individual seeds have weight 28, we generated with the FastHC tool a second family in which the individual seeds have weight 30.

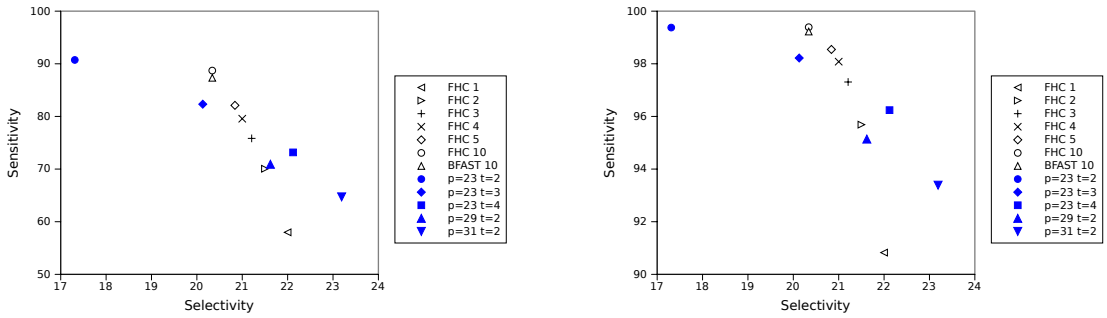

Figure 4: **QR-seeds vs multiple seeds for $N = 50$.** The diagrams show the sensitivity/selectivity tradeoffs of QR-seeds compared with those of FHC multiple seeds of size 1, 2, 3, 4, 5, and 10, and BFAST multiple seeds of size 10 for region length $N = 50$ and similarity $q = 0.90$ (left) and $q = 0.95$ (right).

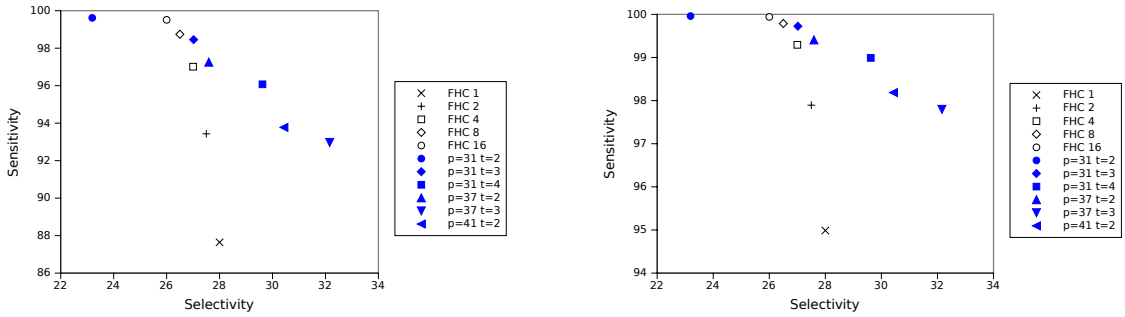

Figure 5: **QR-seeds vs multiple seeds for $N = 150$ and $N = 200$.** The diagrams show the sensitivity/selectivity tradeoffs of QR-seeds compared with those of FHC multiple seeds of size 1, 2, 4, 8, 16 for similarity $q = 0.90$ and region length $N = 150$ (left) and $N = 200$ (right).

To compare the sensitivity/selectivity trade-off of different seeds, consider for example the left diagram in Figure 4. Since high sensitivity and high selectivity are both desirable, we see that, for example, QR-seed $\langle 29, 2\rangle$ outperforms FHC 2 since it has higher sensitivity *and* higher selectivity. Similarly, BFAST 10 and FHC 10 outperform QR-seed $\langle 23, 3\rangle$ since they have higher sensitivity and higher selectivity. In most cases, however, seeds are not immediately comparable since they offer different trade-offs: for example FHC 3 has a slightly larger sensitivity than $\langle 23, 4\rangle$ but it also has a slightly smaller selectivity. Similar considerations apply to all the other diagrams. From Figure 4 we see that for $N = 50$ there are QR-seeds outperforming FHC seeds of size 2 or less, while from Figure 5 we see that for $N = 150$ and 200 QR-seeds outperform FHC seeds of size 4 or less. Finally, from Figure 6 we see that for $N = 100$ QR-seeds have very similar sensitivity/selectivity trade-offs than FHC seeds of size 4.

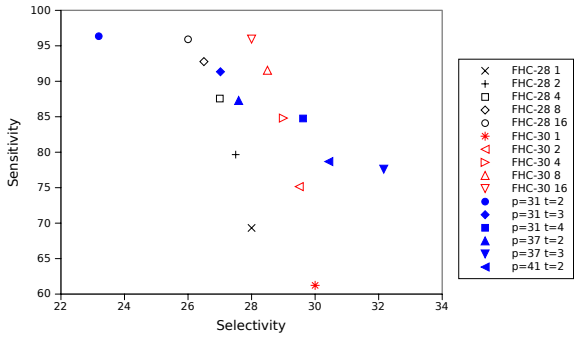

Figure 6: **QR-seeds vs multiple seeds for $N = 100$.** The diagram shows the sensitivity/selectivity tradeoffs of QR-seeds compared with those of FHC multiple seeds of size 1, 2, 4, 8, 16 for similarity $q = 0.90$ and region length $N = 100$. The diagram reports sensitivity and selectivity for two sets of FHC seeds with individual seeds of size 28 and 30 respectively.

We can summarize the above results as follows. By increasing the size of multiple seeds, we obtain sensitivity/selectivity trade-offs which are superior of those of QR-seeds. However, since multiple seeds of size k require k times the memory space of a single seed, whenever memory usage is an issue QR-seeds become a valid alternative since they offer good sensitivity/selectivity trade-offs using the same space of a single seed.

Another advantage of QR-seeds is that their shapes do not depend on the similarity level q and on the size of the homologous region N, two parameters for which we usually only have estimates. In the diagrams above FHC seeds are used in the "ideal" setting in which the values N and q coincide with those used for the construction of the seeds. If this is not the case there could be a degradation in performance; because of their simpler structure QR-seeds are more robust in this respect.

Finally, we note that when using QR-seeds we do not necessarily need to choose the threshold t in advance. We can start using a seed S_p and record the hits; if we find regions with, say, $t = 3$ hits at distance at most $p - 1$ we proceed as usual; if no such region exists we have the option of considering regions with only $t = 2$ hits which represents regions with are less homologous but still potentially interesting.

5 Conclusions and further work

We introduced a class of spaced seeds whose shape is related to quadratic residues modulo a prime number p. Such seeds have been proven effective in the lossless setting when we require that they hit in $t > 1$ sufficiently close positions. We have measured the sensitivity and selectivity of these seeds in the lossy setting and found that they are as effective as multiple seeds, but, being based on a single seed, they require much less memory.

We observe that increasing the size of multiple seeds yields diminishing returns. For example in Figure 4 we see that there is a huge increase in sensitivity when the size of FHC seeds changes from 1 to 2, but a much smaller increase when the size changes from 4 to 5. In Figures 5 and 6 we see that we have diminishing returns even when the size of the multiple seed doubles. This phenomenon suggests that, even setting aside the issue of memory usage, simply increasing the size of multiple seeds should not be the only strategy for designing better homology search tools. New approaches need to be considered and we believe our experiments show that using a threshold $t > 1$ is a promising approach. Indeed, we plan to combine the two ideas and design *multiple seeds* using a threshold $t > 1$.

We have also measured the sensitivity/selectivity tradeoff of some "hand-selected" single seeds used with threshold $t = 2$ and found that they are as good as QR-seeds. This suggests that the properties

of quadratic residues are not essential for designing effective seeds for $t > 1$. This opens the way to investigations on effective heuristics for finding "good" seeds for $t > 1$. A natural starting point for such investigation would be to combine the Hill Climbing strategies from [16, 17] with the dynamic programming algorithm **DP-k-hits** from [20]. The practical challenge here is to cope with the huge space requirements of **DP-k-hits** for large seeds. Note that our experimental results suggest that it could be sufficient to use a threshold $t = 2$ to get significant improvements over the standard setting $t = 1$.

References

[1] S. Altschul, T. Madden, A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. Lipman. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–402, 1997.

[2] C. E. Bonferroni. Teoria statistica delle classi e calcolo delle probabilità. *Pubblicazioni del R. Istituto Superiore di Scienze Economiche e Commerciali di Firenze*, 8:1–62, 1936.

[3] Broňa Brejová, Daniel G Brown, and Tomáš Vinař. Vector seeds: An extension to spaced seeds. *Journal of Computer and System Sciences*, 70(3):364–380, 2005.

[4] Daniel G. Brown. A survey of seeding for sequence alignment. In I. Măndoiu and A. Zelikovsky, editors, *Bioinformatics Algorithms: Techniques and Applications*, pages 126–152. Wiley-Interscience, Hoboken, New Jersey, 2008.

[5] Jeremy Buhler, Uri Keich, and Yanni Sun. Designing seeds for similarity search in genomic DNA. *Journal of Computer and System Sciences*, 70(3):342–363, 2005.

[6] Stefan Burkhardt and Juha Kärkkäinen. Better filtering with gapped q-grams. In *Proc. 12th Symposium on Combinatorial Pattern Matching (CPM '01)*, pages 73–85, Berlin, 2001. Springer-Verlag LNCS n. 2089.

[7] L. Comtet. *Advanced Combinatorics*. Springer-Verlag, New-York, 2010.

[8] M. Crochemore and G. Tischler. The gapped suffix array: A new index structure for fast approximate matching. In *SPIRE*, pages 359–364. Springer Verlag LNCS n. 6393, 2010.

[9] J. Shane Culpepper and Alistair Moffat. Efficient set intersection for inverted indexing. *ACM Transactions on Information Systems*, 29:1–25, 2010.

[10] M. David, M. Dzamba, D. Lister, L. Ilie, and M. Brudno. SHRiMP2: Sensitive yet practical short read mapping. *Bioinformatics*, 27(7):1011–1012, 2011.

[11] Lavinia Egidi and Giovanni Manzini. Better spaced seeds using quadratic residues. *Journal of Computer and System Sciences*, 79(7):1144–1155, 2013.

[12] Lavinia Egidi and Giovanni Manzini. Design and analysis of periodic multiple seeds. *Theoretical Computer Science*, pages 62–76, 2014.

[13] Lavinia Egidi and Giovanni Manzini. http://people.unipmn.it/manzini/qrseeds/qrlossy.zip, 2014.

[14] Travis Gagie, Giovanni Manzini, and Daniel Valenzuela. Compressed spaced suffix arrays. In *Proceedings of the 2nd International Conference on Algorithms for Big Data*, pages 37–45. CEUR-WS, vol. 1146, 2014. http://ceur-ws.org/Vol-1146/paper7.pdf.

[15] Nils Homer, Barry Merriman, and Stanley F Nelson. BFAST: An alignment tool for large scale genome resequencing. *PloS one*, 4(11), January 2009.

[16] Lucian Ilie, Silvana Ilie, and Anahita Mansouri Bigvand. SpEED: Fast Computation of Sensitive Spaced Seeds. *Bioinformatics*, 27(17):2433–4, 2011.

[17] Silvana Ilie. Efficient Computation of Spaced Seeds. *BMC research notes*, 5(1):123, 2012.

[18] Uri Keich, Ming Li, Bin Ma, and John Tromp. On spaced seeds for similarity search. *Discrete Applied Mathematics*, 138(3):253–263, 2004.

[19] Gregory Kucherov, Laurent Noé, and Mikhail Roytberg. A unifying framework for seed sensitivity and its application to subset seeds. *Journal of Bioinformatics and Computational Biology*, 4:553–69, 2006.

[20] Ming Li, Bin Ma, Derek Kisman, and John Tromp. PatternHunter II: Highly sensitive and fast homology search. *Journal of Bioinformatics and Computational Biology*, 2(3):417–440, 2004.

[21] Bin Ma and Ming Li. On the complexity of the spaced seeds. *Journal of Computer and System Sciences*, 73(7):1024–1034, 2007.

[22] Bin Ma, John Tromp, and Ming Li. PatternHunter: Faster and more sensitive homology search. *Bioinformatics*, 18(3):440–445, 2002.

[23] François Nicolas and Eric Rivals. Hardness of optimal spaced seed design. *Journal of Computer and System Sciences*, 74(5):831–849, 2008.

[24] Steven J. Schwager. Bonferroni sometimes loses. *The American Statistician*, 38(3):192–197, Aug. 1984.

A Additional Tables and Figures

	2	3	4	5
11	8.94– 8.94			
13	10.46–10.46			
17	13.40–13.40	15.58–15.59		
19	14.94–14.94	17.74–17.74	19.90–19.91	
23	17.91–17.91	20.92–20.92	23.40–23.41	24.82–24.85
29	22.40–22.40	25.95–25.95	28.69–28.69	30.66–30.67
31	23.74–23.74	27.74–27.74	30.84–30.84	32.84–32.86
37	28.40–28.40	33.32–33.32	36.31–36.32	38.33–38.36
41	31.30–31.30	35.87–35.87	38.66–38.67	40.53–40.56

	2	3	4	5
11	9.00– 9.00			
13	10.70–10.70			
17	13.66–13.66	15.79–15.79		
19	15.00–15.00	17.87–17.87	19.99–19.99	
23	17.99–17.99	21.00–21.00	23.69–23.69	24.99–24.99
29	22.66–22.66	26.00–26.00	28.86–28.86	30.86–30.86
31	23.87–23.87	27.87–27.87	30.99–30.99	32.99–32.99
37	28.66–28.66	33.63–33.63	36.66–36.66	38.69–38.69
41	31.61–31.61	35.99–35.99	38.86–38.86	40.79–40.79

Table 2: **Selectivity of QR-seeds for alphabet size 20 and 200.** Upper and lower bounds to the selectivity of QR-seeds for $p = 11, \ldots, 41$, and alphabet size $\sigma = 20$ (top) and $\sigma = 200$ (bottom), for thresholds $t = 2, 3, 4, 5$. All values were computed using Bonferroni inequalities.

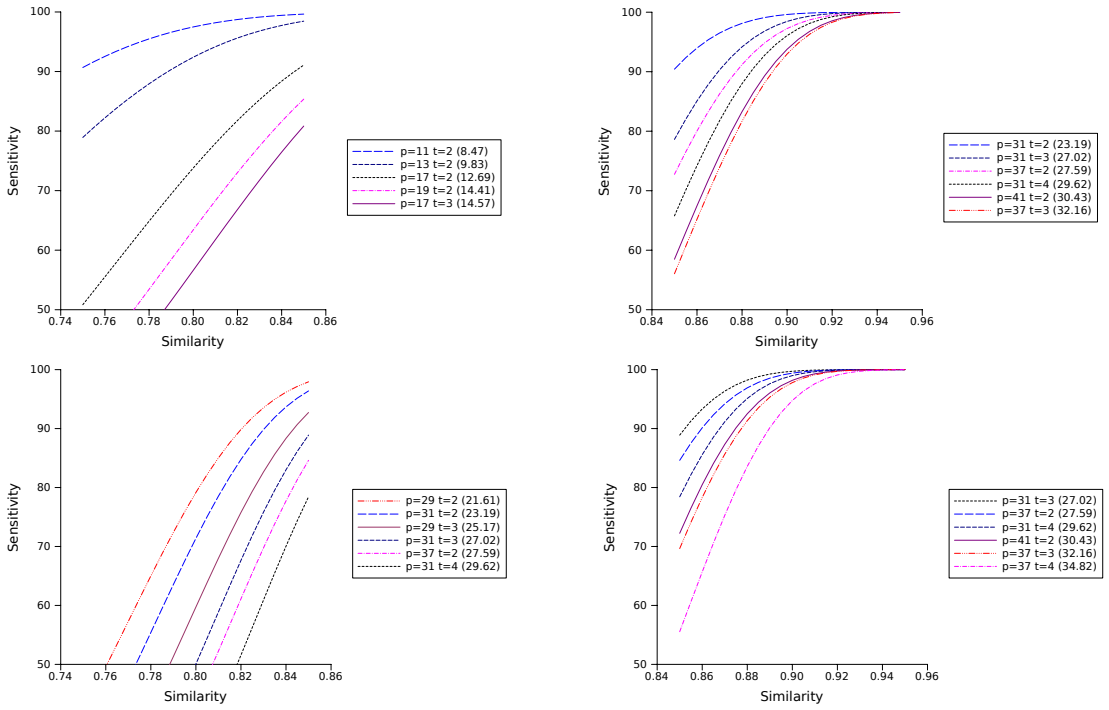

Figure 7: **Sensitivity vs Similarity for QR-seeds for different region lengths.** Sensitivity as a function of the similarity q for $N = 50$ (top-left), $N = 150$ (top-right) and $N = 200$ (bottom left and right).