Binning metagenomic reads with probabilistic sequence signatures based on spaced seeds

Samuele Girotto
Department of Information Engineering, University of Padova, Italy

Matteo Comin*
Department of Information Engineering, University of Padova, Italy
Email: comin@dei.unipd.it

Cinzia Pizzi*
Department of Information Engineering, University of Padova, Italy
Email: cinzia.pizzi@dei.unipd.it

***Abstract*—The growing number of sequencing projects in medicine and environmental sciences calls for the development of efficient approaches for the analysis of very large sets of metagenomic reads. Among the challenging tasks in metagenomics, the ability to agglomerate, or "bin" together, reads of the same species, without reference genomes, plays a crucial role in building a comprehensive description of relative abundances and diversity of the species in the sample. Recently, we have proposed an algorithm, called MetaProb, for metagenomic reads binning that reaches a precision that is currently unmatched. The competitive advantage of MetaProb depends on the use of probabilistic sequence signatures based on contiguous k-mers. In this work we explore the use of spaced seeds, rather than contiguous kmers, to build such signatures. The experimental results show that allowing mismatches in carefully chosen predefined positions leads to further benefits both in terms of improved accuracy and of reduction of the memory requirements. Availability: https://bitbucket.org/samu661/metaprob**

I. Introduction

Metagenomics is the study of complex microbial communities without cultivation steps [9]. The samples can be taken from a variety of environments, from soil and water to gut and skin. One of the primary goals of metagenomic studies is to determine the taxonomical identity of the microorganisms that are present in a sample [15]. To this purpose, one can apply direct shotgun sequencing to the sample DNA, followed by the analysis of the set of short reads obtained from sequencing. This analysis can reveal the presence of unexpected bacteria and viruses in a microbial sample, and it also allows the identification and characterization of new bacterial and viral genomes. For example, in the case of the human body, imbalances in the microbiome are known to be related with several diseases, e.g. inflammatory bowel disease (IBD) [20] and colorectal cancer [27].

The taxonomic analysis of microbial communities can be carried out by a process referred to as binning, in which reads from the same species are grouped together without the use of reference genomes. By binning reads, researchers can identify the number and the abundance of species in the environment. Tools such as BiMeta [24], MetaCluster [25], AbundanceBin [26], MetaProb [7], [6], among others, can be used to determine the microbial diversity. We have recently proposed MetaProb in [7] and demonstrated that its one of the best performing methods for metagenomic binning. MetaProb is an assembly-assisted method that groups reads by considering the overlapping information combined with the probabilistic sequence signatures, based on k-mers counts.

The work presented in this manuscript exploits a new approach to improve MetaProb's clustering performances. Here we show that the accuracy can be increased by allowing mismatches, in a limited number of positions, while counting the number of shared k-mers. This concept is also known as spaced seeds [14], where mismatches are allowed at carefully predetermined positions, while optimizing some requirement that spaced seeds counts are discriminative. Recently spaced seeds have been successfully applied to the field of metagenomic [8], [18].

In the following we briefly summarize the idea of spaced seeds and the main associated developments. Then, we describe how spaced seeds can be used instead of contiguous k-mers in the definition of the probabilistic sequence signatures, and how they are implemented in the MetaProbS (S for spaced) pipeline. We report an extensive comparison of MetaProbS, against the original method MetaProb, evaluating nine different spaced seeds. We show that for several metrics MetaProbS outperforms MetaProb.

A. Background on spaced seeds

Contiguous k-mers counts are at the bases of many sequence comparison methods, among which the most widely used and notable example is BLAST [1]. BLAST uses the so-called "hit and extend" method, where a hit consists of a match of a 11-mers between two sequences. Then these matches are potential candidate to be extended and to form a local alignment. It can be easily noticed that not all local alignments include an identical stretch of length 11. As observed in [4] requiring that the matches are not consecutive increases the chances of finding alignments. This idea of optimizing the position of the required matches has been investigated in many studies, and

978-1-4673-8988-4/17/$31.00

it was used in PatternHunter [14], another popular similarity search software.

A spaced seed of length k and weight $w < k$ is a string over the alphabet $\{1, *\}$ that contains w '1' and $(k-w)$ '*' symbols. A spaced seed is a mask where the symbols '1' and '*' denote respectively match and don't care positions. Much work has been dedicated to spaced seeds over the years, we refer the reader to [3] for a survey on the earlier work. The advantage of using spaced seeds is due to the fact that spaced seeds occurrences at neighboring sequence positions are statistically less dependent than occurrences of contiguous k-mers [13].

Spaced seeds are now routinely used in many problems involving sequence comparison like: multiple sequence alignment [5], protein classification [17], read mapping [22] and for alignment-free phylogeny reconstruction [11]. More recently also metagenome reads clustering and classification can benefit form the use of spaced seeds [18], [2].

Despite the fact that spaced seeds are widely used, to find the best spaced seed or set of spaced seeds remains a challenging computational problem. Ideally one would like to maximize the sensitivity of a spaced seeds, however it has been shown that calculating the sensitivity of a spaced seed is NP-hard [13]. The sensitivity can be approximated using dynamic programming, but the running time remains exponential in the length of the seed [14]. In 2011 Ilie et al. introduced SpEED [10] a tool for computing candidate spaced seeds. SpEED is based on the notion of overlap complexity that is correlated with sensitivity but it can be computed in polynomial-time. Recently Morgenstern at al. proposed a new method to compute sets of spaced seeds called rasbhari [8]. Rasbhari uses a hill-climbing algorithm to optimize the overlap complexity of the produced pattern sets.

II. Methods

In metagenomic, one of the basic assumption is that the k-mer frequency distributions of long fragments (or whole genome sequences) are unique to each genome. However the current NGS technologies cannot produce reads long enough to directly apply k-mers/compositional distances. In order to solve this issue MetaProb addresses the problem of metagenomic binning into two phases, by simulating the availability of long fragments. To keep the paper self contained here we briefly describe MetaProb and the required modifications to incorporate spaced seeds.

In Fig.1 is reported an outline of the two phases of MetaProb. In the first phase reads are grouped together based on the extent of their overlap. Ideally the overlap would be computed using a reads overlap graph [16], but this approach is highly demanding in terms of RAM. Instead we use an alignment-free technique frequently used in de-novo genome assembly.

The overlap between reads is estimated by considering the number of shared k-mers between reads. This technique relies on the assumption that, by choosing a sufficiently large value for k, the probability that two k-mers are shared by different genomes is low. For example, a study presented in [24] shows that the average ratio of common k-mers between pairs of bacterial genomes is less than 1.02% when $k = 30$. Therefore, the presence of a shared k-mer between two reads should indicate that the two reads belong to the same species. Moreover, if several such k-mers are shared, this strengthens the probability the two reads overlap. For these reasons, in the first phase, we construct groups where two reads overlap if they share at least m common k-mers. As a result the reads in a group, because of their overlap, are likely to belong to the same species. However, reads from a same species might be distributed in different groups. Thus, further processing is needed to cluster the groups obtained in Phase 1 based on their similarity.

The similarity between groups can be defined in terms of k-mers frequency distribution within each group. However, by construction, the reads in a group must have a significant overlap. Because such overlaps might artificially inflate the count of some k-mers, we developed a strategy, based on independent sets of a graph, to select a subset of reads from a group in order to reduce the redundancy provided by large overlaps. In the second phase, each group is then represented by a set of independent reads on which the k-mers frequency distribution is computed. Because the Euclidean distance of k-mers distributions of different groups can be biased by the stochastic noise in each sequence [12], [23], and by the possibly unbalanced size of the groups, we introduce a similarity measure based on self-standardized probabilistic sequence signatures. Thus in the second phase MetaProb clusters groups based on probabilistic sequence signatures (see for details [7]), so that reads from the same species are grouped together in one cluster.

In this work we explore a new idea of overlap between reads, not based on shared contiguous k-mers, but instead on shared spaced k-mers. A spaced seed is a mask over the alphabet $\{1, *\}$, where the two symbols '1' and '*' denote respectively match and don't care positions, and it is used to incorporate mismatches in several string comparison problems. We say that a spaced seed is of length k and it has weight $w < k$ if it contains w '1' and $(k-w)$ '*' symbols. If $k = w$, we have that the seed does not contain mismatches and it is equivalent to consider contiguous k-mers.

Let us consider for example the spaced seed $111 * 1$, with length 5 and weight 4. It has three consecutive matches followed by a don't care position (match or mismatch are allowed) and another match. By using this spaced seed, it allows to consider as equals the two sequences $TGAAG$ and $TGACG$, thus allowing for one mutation.

We need to choose carefully the parameters k and w of the spaced seed. For contiguous k-mers, precision increases as we increase k. However, the highest sensitivity occurs with somewhat shorter k-mers. For example Clark [19] is more precise for long contiguous k-mers (e.g., k = 31), but the highest sensitivity occurs for k-mers of length between 19 and 22. For these reasons a good compromise is to have spaced seeds where the number of required matches is $w = 22$, and the length is $k = 31$. These values have been reported also in

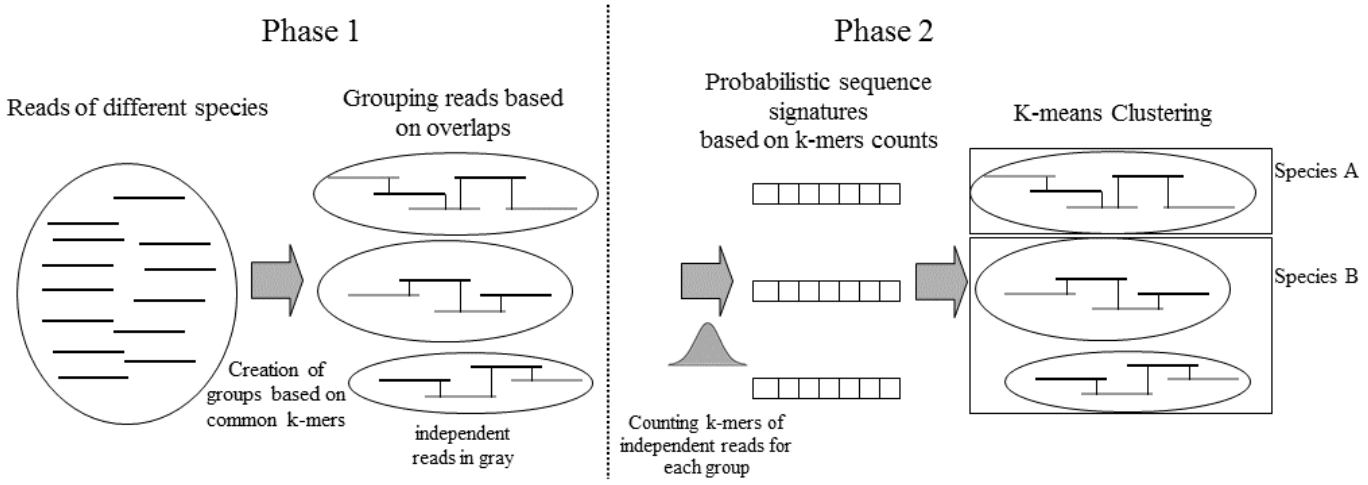


Fig. 1. An overview of the binning process of MetaProb. Phase 1 groups overlapping reads. Phase 2 merges the groups into clusters based on probabilistic sequence signatures of independent reads.

[18], [8] to improve the classification performance.

Also, the structure of spaced seeds is critical to achieve the highest possible precision and sensitivity. As discussed above, given k and w to find the optimal spaced seed that maximizes the hit probability, or other measures, it can be computationally difficult [3]. However, the recent advancements in this field allow us to test spaced seeds optimized for different metrics. For example in [18] the authors report three spaced seeds, with length $k = 31$ and weight $w = 22$, that are designed to maximize the hit probability. Recently, in [8] six new spaced seeds are proposed, three of which are computed minimizing the overlap complexity, while the another three maximize the sensitivity. We selected the nine spaced seeds shown in Table I for testing their impact on MetaProb.

In the context of MetaProb, spaced seeds allow to better detect the overlap between reads. In fact, in the original version of MetaProb, two reads are overlapping only if they share a contiguous stretch of length 30. With spaced seeds we require the two reads to share 22 matching positions that are not consecutive. For this reason spaced seeds can detect more overlaps between reads, by relaxing the requirements to have consecutive matches and by allowing for mutations.

TABLE I
THE NINE SPACED SEEDS USED TO COMPUTE THE PROBABILISTIC SEQUENCE SIGNATURES.

Spaced seeds maximizing the hit probability[18]	
Q1	1111*111*111**1*111**1*11*11111
Q2	11111*1*111**1*11*111**11*11111
Q3	11111*1**111*1*11*11**111*11111
Spaced seeds minimizing the overlap complexity[8]	
Q4	1111*1*111*1**11**111*11111*111
Q5	111*111*111*1111*1**1*11**11111
Q6	11111*1**1*111**11111*1*11*1111
Spaced seeds maximizing the sensitivity[8]	
Q7	1111*1111**11*1*11111*1*1*11*11
Q8	111*1*1*111*11**11*1**111111111
Q9	111111*1*11*1*111**111*11**1111

If more overlaps are discovered, then MetaProb can produce better estimate of the probabilistic sequence signature, and consequently improve the clustering performance.

A. Metrics

In order to estimate the performance of MetaProb using spaced seeds we used metrics to estimate both the quality of the binning, through precision, recall and F-measure, and the amount of computational resources needed, by measuring time and memory required for the analysis.

Furthermore, we evaluate the impact of spaced seeds in building the groups by measuring the average precision and group size.

We repeated these measures in two experimental frameworks: first we assume that the number of species is known and then we assume we do not have this information.

We now recall the definitions of precision, recall and F-measure as defined in [7]. Let m be the number of species in a metagenomic dataset, and k be the number of clusters returned by the binning algorithm. Let A_{ij} be the number of reads from species j assigned to cluster i. The ratio of reads assigned to a cluster that belong to the same species are called *precision* and is defined as follow:

$$precision = \frac{\sum_{i=1}^{k} \max_j A_{ij}}{\sum_{i=1}^{k} \sum_{j=1}^{m} A_{ij}} \quad (1)$$

If we consider groups as clusters, we can define the *groups precision* similarly to the just defined *precision*.

The ratio of reads from the same species that are assigned to the same cluster is called *recall*, and it is formally defined as:

$$recall = \frac{\sum_{j=1}^{m} \max_i A_{ij}}{\sum_{i=1}^{k} \sum_{j=1}^{m} A_{ij} + \#unassigned\_reads} \quad (2)$$

Finally, *F-measure* is a metric that emphasizes comprehensively both precision and recall:

$$F - measure = \frac{2 * precision * recall}{precision + recall} \quad (3)$$

For all metrics we have considered the average on all datasets, so that we can compare the overall behavior of different seeds.

III. Results and Discussion

We compared the performances of MetaProb and MetaProbS on 10 different datasets that were also analyzed in previous works [7], [24], using different spaced seeds and metrics, and assuming that the number of clusters is known. Furthermore, we also compared the two approaches on the subset of the datasets with a large number of species assuming no a-priori knowledge on the number of clusters, and thus also estimating its value. In both cases we compared the performances of MetaProbS with seeds Q1-Q9 against those of MetaProb with $k = 22$, which corresponds to the actual weight of our spaced seeds. Therefore, our comparison will show how, when considering the information coming from the same amount of symbols, using spaced seeds improves the overall performances of the binning process. A comprehensive evaluation of MetaProb can be found in [7], where MetaProb is reported to be the best method under several conditions. Here, we report preliminary results for the comparison of MetaProbS with MetaProb, a more extensive evaluation can be found in [6].

All the experiments where run on a laptop equipped with an Intel i74510U cpu at 2GHz, and 16 GB RAM. We start with a description of the datasets and the metrics we used for comparison, and then discuss the results.

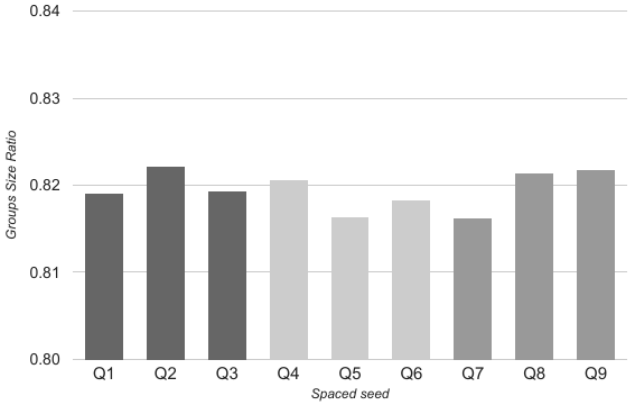
A. Datasets

For our experiments we considered 10 simulated bacterial genomes obtained with MetaSim[21]. These are the datasets $S_1 \dots S_{10}$ and include paired-end short reads (length of approximately 80bp) following an Illumina model with error rate of 1%. These datasets vary in terms of number of species (from 2 to 30), abundance ratio (balanced/unbalanced), and phylogenetic relationships between the species in the sample. More detailed information on the composition of the datasets is shown in Table IV in the Appendix.

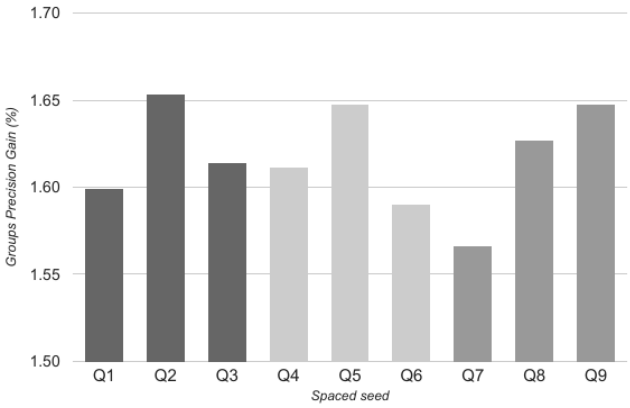
B. Benchmark when the number of species is known

This set of experiments shows how the spaced seeds influence the construction of groups in the first phase of MetaProbS, and consequently how this in turn affect the binning algorithm. In the figures each bar corresponds to a seed, and seeds of the same type are filled with the same color: maximizing the hit probability (Q1,Q2,Q3, dark grey); minimizing overlap complexity (Q4,Q5,Q6, light grey); maximizing the sensitivity (Q7,Q8,Q9, medium grey). These experiments serve as a benchmark as we assume that the number of species in the sample is known a-priori.

Fig. 2. Details of groups characteristics with Spaced-seeds



(a) Groups size



(b) Groups precision

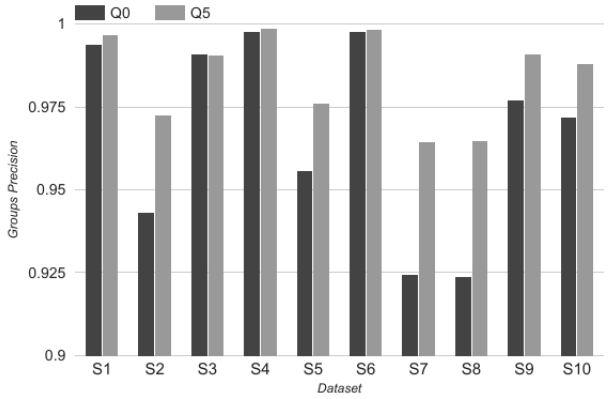
1) Size and precision of groups: First of all we analyzed the features of the groups that are created with seeds. Fig.2 shows the average group size and group precision as defined in Section II-A. We notice that, on average, and for any seed, we obtain groups that are smaller (about 80%) but more precise (about 1.6%) than with MetaProb 22mers. More in details, for what concerns the group precision, we observed that in some cases the gain we obtain with seeds is small because for some datasets the baseline is actually already close to 100%.

As an example, we show in Fig.3 the comparison of the group precision between the Metaprob baseline (Q0) and what we obtain with the seed Q5 that, as we will see in the remaining of the discussion, is among those with the best performances overall.

By using Q5 we have an improvement for all the datasets, with the only exception of S3 where the baseline is 99.09% and the loss is 0.03%, leading to a 99.06% precision, which is still very high. The most interesting aspects is that, although the precision is quite high also by using 22mers, for the two datasets S7 and S8, where the baseline is around 92%, by using Q5 the gain is around 4%, bringing the overall precision above 96% for any dataset.

We recall that, by definition of groups precision, the gain refers to the whole dataset. So if we have 10M reads, an

Fig. 3. Detailed groups precision between 22mers Q0 and the spaced seed Q5



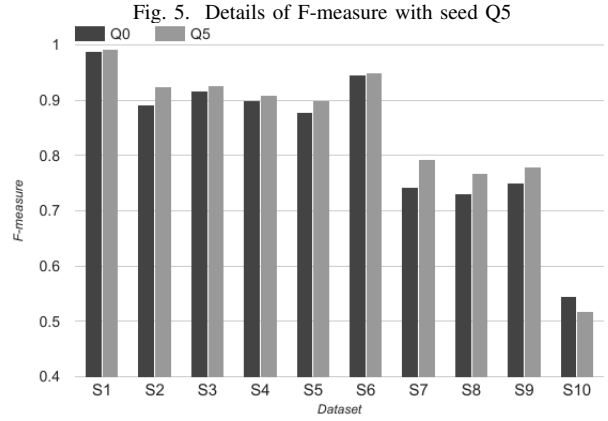
increment of 1% in group precision implies that 100000 more reads are correctly grouped. We underline that this is not a peculiarity of seed Q5, but similar plots can be obtained also with the other seeds, with some slight variations in terms of absolute values.

2) Quality of the clustering: Having smaller and more precise groups has a direct impact on the probabilistic sequence signatures that are extracted from each group. Since these are the features according to which the groups are clustered together, using seeds has an impact on the overall quality of the final clusters that we obtain.

Fig.4 shows the gain in clustering quality obtained with space seeds with respect to the baseline MetaProb 22-mers. We observe that precision, recall and F-measure all improve with respect to the baseline in a range that span from 1% to 1.6%. In general both the dataset composition and the seed structure have an impact on the final quality of clustering.

Fig.5 shows in details what happens when choosing Q5.

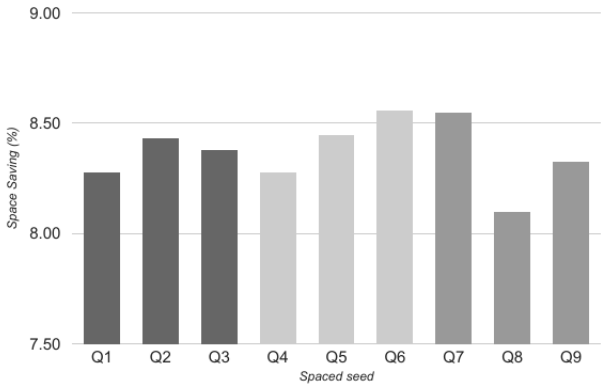
Fig. 5. Details of F-measure with seed Q5



The F-measure improves for all datasets, with the exception of S10. This dataset is among the most difficult to analyze as it has many species, a wide range of species abundance and phylogenetic distances among them. Anyway, even for the dataset S10 the F-measure obtained with Q5 is higher than the one obtained by other state of the art binning algorithms on the same dataset: Abundance Bin scores 0.14, MetaCluster 0.05, and BiMeta 0.43 [7]. As for the other datasets, the gain is relatively small for S1 to S6, where the baseline is close or above 90%. For S7, S8, and S9, where the baseline is around 75% the gain can reach up to 5%.

3) Computational Resources: First, in Fig.6 we observe that, for any seed, we save between 8% and 8.5% of memory.

Fig. 6. Impact on the memory usage with spaced seeds



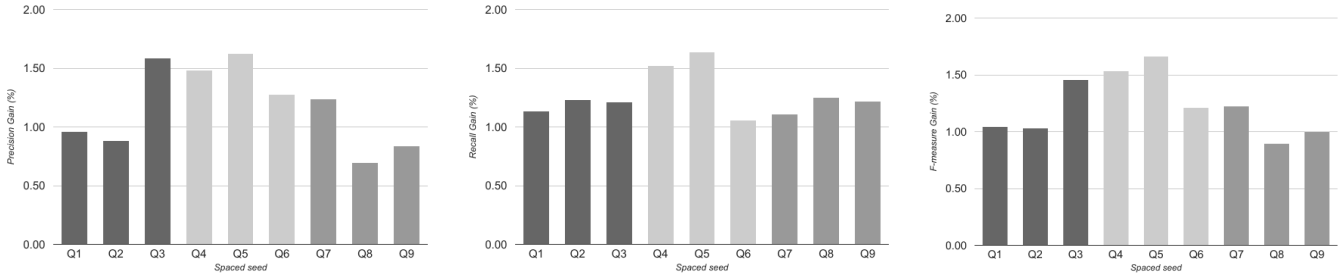
This saving is possible because, although both with MetaProbS and MetaProb we index words of length 22, with seeds we actually need to consider less positions, as the sliding window we use to hash the seeds has length 31 rather than 22.

Using spaced seeds also affects the time required for the computation at several steps. For example, the initial hashing of 22-mers with MetaProb is faster than hashing with spaced seeds because we can reuse the information about the previous k-mer to compute the next. The binning step is also affected by the kmer distribution, which is necessarily different from the 22mers as we still consider 22 positions but they are not necessarily contiguous. This in turn will affect the way the groups are built and the time needed for this step too. In general these differences do not necessarily imply a time overhead by using seeds. In fact, for some datasets, MetaProbS is actually faster than MetaProb with any seed, for some other is slower with any seed and in some cases the behavior is very seed dependent. This said, on average, there is some time overhead with respect to MetaProb, in the range between 1.2% for Q2 and 4.68% for Q6, as shown in Fig.7. The MetaProb baseline with 22mers spans in a range from 14 seconds for processing S1 to 10 minutes for processing S10.

It is also worth noting as the spaced seeds for which the time overhead is higher is the class of seeds that minimize the overlap complexity, which are the ones with the best improvement in terms of quality of the clustering.

Finally, Table II summarizes the results we have shown so far. Using spaced seeds leads to an improvement in terms of both overall quality of the final clustering and memory

Fig. 4. Average difference in clustering efficacy using Spaced-seeds (Qi-Q0), with respect to the baseline of MetaProb (Q0).



(a) Precision (b) Recall (c) F-measure

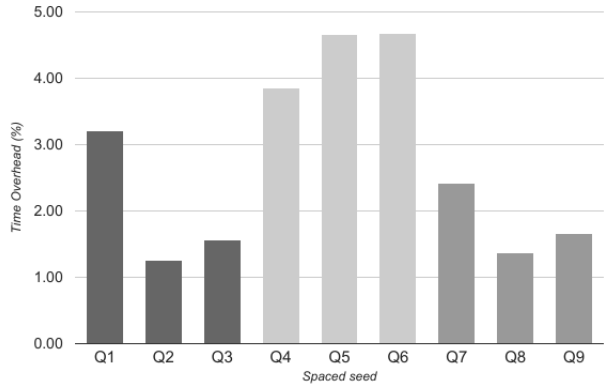
TABLE II
SUMMARY TABLE OF SPACED SEED INFLUENCE ON METAPROBS. (VALUES IN %)

	MinHitProb			MinOverlap			MaxSensitivity		
Seed	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**
Precision gain	0.96	0.88	1.59	1.48	1.63	1.28	1.24	0.70	0.84
Recall gain	1.14	1.23	1.21	1.52	1.64	1.06	1.11	1.25	1.22
F-measure gain	1.04	1.03	1.46	1.53	1.67	1.21	1.22	0.90	1.00
RAM saving	8.28	8.43	8.38	8.28	8.45	8.56	8.55	8.10	8.33
Time overhead	3.21	1.26	1.57	3.86	4.67	4.68	2.43	1.36	1.66
Grp_prec	1.60	1.65	1.61	1.61	1.65	1.59	1.57	1.63	1.65
Grp_size	0.82	0.82	0.82	0.82	0.82	0.82	0.82	0.82	0.82

TABLE III
SUMMARY TABLE OF SPACED SEED INFLUENCE ON METAPROBS WHEN THE NUMBER OF SPECIES IS ESTIMATED. (VALUES IN %)

	MinHitProb			MinOverlap			MaxSensitivity		
Seed	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**
F-measure	2.38	3.02	1.93	2.58	2.27	2.33	0.82	3.64	2.07
RAM	12.08	12.07	12.13	11.86	12.06	12.10	12.28	11.82	12.06
Time	1.03	-2.29	-1.80	-0.25	-0.36	-2.24	5.15	-4.51	4.13
Grp_prec	2.68	2.76	2.67	2.75	2.77	2.63	2.61	2.68	2.70
Grp_size	0.82	0.82	0.82	0.82	0.82	0.82	0.81	0.82	0.82

Fig. 7. Impact on computational time with spaced-seeds



usage. Time overhead is expected when using seeds rather than kmers, as the model is intrinsically more complex. Nevertheless it is bounded within few units percentage.

In terms of class of seeds we observe that those minimizing the overlap complexity are the ones with the best performances in terms of F-measure. However, they are also the ones with the highest time overhead. Among the other classes, we have that Q3, which belongs to the minimizing hit probability class, shows also good performances, slightly less than Q5 and Q4 for what concerns the F-measure, but with a very small time overhead with respect to the baseline.

C. Performance analysis when estimating the number of clusters

This experimental setting is more realistic, as we assumed no prior knowledge on the composition of the sample, and limited the analysis to the datasets with more species: S7, S8, S9 and S10 with, respectively 5, 5, 15, and 30 species.

Table III shows the results of the impact of seeds in this framework. We first observe that the improvements in terms of F-measure, RAM savings, and group precision are higher than in the framework where the number of species is known. As for the F-measure, the actual combinations dataset/seed can vary substantially depending on the choice of the two.

As for the usage of computational resources, there is an homogeneous trend of space saving, even more accentuated that in the previous experiments. Moreover, there is a general speed up with respect to the baseline. The only relevant average slow downs are given by Q7 and Q9 that, although faster than the baseline in the analysis of S7-S9, are substantially slower in the processing of S10. This behavior is indeed shared also by

the other seeds, but for the others the overhead for S10 is less accentuated.

As for the groups composition the observations are similar to the case when the number of species was known. In summary using seeds we obtain smaller but more precise clusters.

IV. Conclusions

In this paper we exploit the spaced seed model to increase the quality of metagenomic reads binning. Several different type of spaced seeds are considered and used within the MetaProb approach instead of contiguous k-mers. Experiments on several datasets showed that using spaced seeds improves the quality of the binning, allowing also for a better use of computational resources.

Funding

This work has been supported by the PRIN project n. 20122F87B2.

References

[1] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of Molecular Biology*, vol. 215, no. 3, pp. 403 – 410, 1990. [Online]. Available: //www.sciencedirect.com/science/article/pii/S0022283605803602

[2] K. Binda, M. Sykulski, and G. Kucherov, "Spaced seeds improve k-mer-based metagenomic classification," *Bioinformatics*, vol. 31, no. 22, p. 3584, 2015. [Online]. Available: +http://dx.doi.org/10.1093/bioinformatics/btv419

[3] D. G. Brown, M. Li, and B. Ma, "A tutorial of recent developments in the seeding of local alignment," *Journal of Bioinformatics and Computational Biology*, vol. 02, no. 04, pp. 819–842, 2004. [Online]. Available: http://www.worldscientific.com/doi/abs/10.1142/S0219720004000983

[4] J. Buhler, "Efficient large-scale sequence comparison by locality-sensitive hashing," *Bioinformatics*, vol. 17, no. 5, p. 419, 2001. [Online]. Available: +http://dx.doi.org/10.1093/bioinformatics/17.5.419

[5] A. E. Darling, T. J. Treangen, L. Zhang, C. Kuiken, X. Messeguer, and N. T. Perna, *Procrastination Leads to Efficient Filtration for Local Multiple Alignment*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 126–137. [Online]. Available: http://dx.doi.org/10.1007/11851561_12

[6] S. Girotto, M. Comin, and C. Pizzi, "Metagenomic reads binning with spaced seeds," *Theoretical Computer Science*, pp. –, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0304397517304632

[7] S. Girotto, C. Pizzi, and M. Comin, "MetaProb: accurate metagenomic reads binning based on probabilistic sequence signatures," *Bioinformatics*, vol. 32, no. 17, pp. i567–i575, Sep. 2016. [Online]. Available: http://bioinformatics.oxfordjournals.org/content/32/17/i567

[8] L. Hahn, C.-A. Leimeister, R. Ounit, S. Lonardi, and B. Morgenstern, "Rasbhari: Optimizing spaced seeds for database searching, read mapping and alignment-free sequence comparison," *PLOS Computational Biology*, vol. 12, no. 10, pp. 1–18, 10 2016. [Online]. Available: http://dx.doi.org/10.1371%2Fjournal.pcbi.1005107

[9] P. Hugenholtz and G. Tyson, "Microbiology: Metagenomics," *Nature*, vol. 455, pp. 481–483, 2008.

[10] L. Ilie, S. Ilie, and A. Mansouri Bigvand, "Speed: fast computation of sensitive spaced seeds," *Bioinformatics*, vol. 27, no. 17, p. 2433, 2011. [Online]. Available: +http://dx.doi.org/10.1093/bioinformatics/btr368

[11] C.-A. Leimeister, M. Boden, S. Horwege, S. Lindner, and B. Morgenstern, "Fast alignment-free sequence comparison using spaced-word frequencies," *Bioinformatics*, vol. 30, no. 14, p. 1991, 2014. [Online]. Available: +http://dx.doi.org/10.1093/bioinformatics/btu177

[12] R. A. Lippert, H. Huang, and M. S. Waterman, "Distributional regimes for the number of k-word matches between two random sequences," *PNAS*, vol. 99, no. 22, pp. 13 980–13 989, 2002. [Online]. Available: http://www.pnas.org/content/99/22/13980.abstract

[13] B. Ma and M. Li, "On the complexity of the spaced seeds," *Journal of Computer and System Sciences*, vol. 73, no. 7, pp. 1024 – 1034, 2007, bioinformatics {III}. [Online]. Available: //www.sciencedirect.com/science/article/pii/S0022000007000268

[14] B. Ma, J. Tromp, and M. Li, "Patternhunter: faster and more sensitive homology search," *Bioinformatics*, vol. 18, no. 3, p. 440, 2002. [Online]. Available: +http://dx.doi.org/10.1093/bioinformatics/18.3.440

[15] S. S. Mande, M. H. Mohammed, and T. S. Ghosh, "Classification of metagenomic sequences: methods and challenges," *Briefings in Bioinformatics*, vol. 13, no. 6, pp. 669–681, Nov. 2012. [Online]. Available: http://bib.oxfordjournals.org/content/13/6/669

[16] E. Myers, "The fragment assembly string graph," *Bioinformatics*, vol. 21, no. suppl 2, pp. ii79–ii85, 2005.

[17] T. Onodera and T. Shibuya, "The gapped spectrum kernel for support vector machines," in *Proceedings of the 9th International Conference on Machine Learning and Data Mining in Pattern Recognition*, ser. MLDM'13. Berlin, Heidelberg: Springer-Verlag, 2013, pp. 1–15. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-39712-7_1

[18] R. Ounit and S. Lonardi, "Higher classification sensitivity of short metagenomic reads with clark-s," *Bioinformatics*, vol. 32, no. 24, p. 3823, 2016. [Online]. Available: +http://dx.doi.org/10.1093/bioinformatics/btw542

[19] R. Ounit, S. Wanamaker, T. J. Close, and S. Lonardi, "CLARK: fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers," *BMC Genomics*, vol. 16, p. 236, 2015. [Online]. Available: http://dx.doi.org/10.1186/s12864-015-1419-2

[20] J. Qin, R. Li, J. Raes, and et al., "A human gut microbial gene catalogue established by metagenomic sequencing." *Nature*, no. 464, p. 5965, 2010.

[21] D. C. Richter, F. Ott, A. F. Auch, R. Schmid, and D. H. Huson, "MetaSim: a sequencing simulator for genomics and metagenomics," *PloS One*, vol. 3, no. 10, p. e3373, 2008.

[22] S. M. Rumble, P. Lacroute, A. V. Dalca, M. Fiume, A. Sidow, and M. Brudno, "Shrimp: Accurate mapping of short color-space reads," *PLOS Computational Biology*, vol. 5, no. 5, pp. 1–11, 05 2009. [Online]. Available: http://dx.doi.org/10.1371%2Fjournal.pcbi.1000386

[23] K. Song, J. Ren, G. Reinert, M. Deng, M. S. Waterman, and F. Sun, "New developments of alignment-free sequence comparison: measures, statistics and next-generation sequencing," *Briefings in Bioinformatics*, vol. 15, no. 3, pp. 343–353, 2014. [Online]. Available: http://bib.oxfordjournals.org/content/15/3/343.abstract

[24] L. V. Vinh, T. V. Lang, L. T. Binh, and T. V. Hoai, "A two-phase binning algorithm using l-mer frequency on groups of non-overlapping reads," *Algorithms for Molecular Biology*, vol. 10, no. 1, pp. 1–12, 2015. [Online]. Available: http://dx.doi.org/10.1186/s13015-014-0030-4

[25] Y. Wang, H. C. Leung, S. M. Yiu, and F. Y. Chin, "Metacluster 5.0: a two-round binning approach for metagenomic data for low-abundance species in a noisy sample," *Bioinformatics.*, vol. 28, 2012. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/bts397

[26] Y.-W. Wu and Y. Ye, "A Novel Abundance-Based Algorithm for Binning Metagenomic Sequences Using l-tuples," *Journal of Computational Biology*, vol. 18, no. 3, pp. 523–534, Mar. 2011. [Online]. Available: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3123841/

[27] G. Zeller, J. Tap, A. Y. Voigt, S. Sunagawa, J. R. Kultima, P. I. Costea, A. Amiot, J. Böhm, F. Brunetti, N. Habermann, R. Hercog, M. Koch, A. Luciani, D. R. Mende, M. A. Schneider, P. Schrotz-King, C. Tournigand, J. Tran Van Nhieu, T. Yamada, J. Zimmermann, V. Benes, M. Kloor, C. M. Ulrich, M. von Knebel Doeberitz, I. Sobhani, and P. Bork, "Potential of fecal microbiota for early-stage detection of colorectal cancer," *Molecular Systems Biology*, vol. 10, no. 11, 2014. [Online]. Available: http://msb.embopress.org/content/10/11/766

V. APPENDIX

Table IV shows the details of the datasets used in the experiments.

TABLE IV
DESCRIPTION TABLE OF SHORT READ DATASETS S

DataSet	N.Specie	Phylogenetic Distance	Abundance ratio	Tot. read
S1	2	Species	1:1	192734
S2	2	Species	1:1	390678
S3	2	Order	1:1	677450
S4	2	Phylum	1:1	750578
S5	3	Species and Family	1:1:1	650800
S6	3	Phylum and Kingdom	3:2:1	1426764
S7	5	Order, Order, Genus, Order	1:1:1:4:4	3307100
S8	5	Genus, Order, Order, Order	3:5:7:9:11	912448
S9	15	varius distances	1:1:1:1:1: 2:2:2:2:2: 3:3:3:3:3	4468336
S10_S	30	varius distances	4:4:4:4:4: 6:6:6:6:6: 7:7:7:7:7: 8:8:8:8:8: 9:9:9:9:9: 10:10:10:10:10	3000000