

Accurate Profiling of Microbial Communities from Massively Parallel Sequencing using Convex Optimization

Or Zuk^{1,2}, Amnon Amir³, Amit Zeisel³, Ohad Shamir⁴, and Noam Shental⁵

¹ Broad Institute of MIT and Harvard; ² Toyota Technological Institute at Chicago

³ Department of Physics of Complex Systems, Weizmann Institute of Science

⁴ Microsoft Research, New England

⁵ Department of Computer Science, The Open University of Israel

Abstract. We describe the Microbial Community Reconstruction (**MCR**) Problem, which is fundamental for microbiome analysis. In this problem, the goal is to reconstruct the identity and frequency of species comprising a microbial community, using short sequence reads from Massively Parallel Sequencing (MPS) data obtained for specified genomic regions. We formulate the problem mathematically as a convex optimization problem and provide sufficient conditions for identifiability, namely the ability to reconstruct species identity and frequency correctly when the data size (number of reads) grows to infinity. We discuss different metrics for assessing the quality of the reconstructed solution, including a novel phylogenetically-aware metric based on the Mahalanobis distance, and give upper-bounds on the reconstruction error for a finite number of reads under different metrics. We propose a scalable divide-and-conquer algorithm for the problem using convex optimization, which enables us to handle large problems (with $\sim 10^6$ species). We show using numerical simulations that for realistic scenarios, where the microbial communities are sparse, our algorithm gives solutions with high accuracy, both in terms of obtaining accurate frequency, and in terms of species phylogenetic resolution.

Keywords: Microbial Community Reconstruction, Massively Parallel Sequencing, Short Reads

1 Introduction

Characterization of the micro-organisms present in a microbial community is of major biological and clinical importance. Since different micro-organisms have different genomes, it is possible to identify species based on their DNA sequences, using either whole-genome sequencing, or sequencing of pre-specified regions. The 16S ribosomal RNA gene (16S rRNA) is of particular interest for identifying microbial communities via sequencing. It has both highly conserved regions, present in almost all microbial species, together with variable regions. The conserved regions allow sequence amplification using universal PCR primers, while the variable regions provide information used to distinguish between different species. Large databases [3, 4] with millions of 16S rRNA sequences may enable species identification by querying sequencing results in a database.

Previous methods aiming to characterize microbial communities using microarrays [7] and Sanger sequencing [2] have shown that, in principle, it is possible to identify species present in a sample, yet it is not clear how to get accurate estimation of species frequencies from the analog measurements provided by these technologies. Massively Parallel Sequencing (MPS) [16], also known as Next-Generation Sequencing (NGS), provides high-throughput digital sequence data and can allow a more detailed and accurate picture of the species in the mixture. In this method, one obtains a large number of short sequence reads from the mixture, and the goal is to reconstruct the identities and quantities of the species present. Many studies have used short reads to characterize microbial communities [10], yet they did not demonstrate an ability to identify the specific species present and quantify their abundance in the mixture - reliable recognition was typically achieved only at coarse genus level [12]. The main drawback of MPS is the relatively short read length (typically around 50-400 base-pairs in current technologies), which poses a problem for species reconstruction; short reads do not provide unambiguous evidence in support of the presence of a specific species, as typically the same read may originate from multiple different species, and cannot be uniquely aligned to the reference database.

Recently, more sophisticated methods for quantifying species abundance were developed, for 16S rRNA [6, 17] and whole metagenome shot-gun sequencing data [25]. These methods take into account read-assignment ambiguity and enable increased species resolution, but the question of maximal reconstruction resolution achieved was not systematically studied.

In this paper, we study mathematically the Microbial Community Reconstruction problem (**MCR**) - in which we use MPS data to characterize a microbial community. In a nutshell, the computational and statistical problem we face is as follows: given a large collection of short MPS reads (strings) sampled from a known database of species' sequences (longer strings) according to a certain unknown distribution, our goal is to estimate the sampling frequencies for each species in the database, and specifically recover the support of the distribution, i.e. the list of species with non-zero sampling probabilities. We model the sequencing process statistically, providing a probabilistic generative model for the short read data at hand. We prove conditions for *identifiability* - namely the ability to reconstruct precisely the identity and frequency of species present in the mixture from the short read data as the number of reads is increased. We prove upper-bounds on reconstruction errors for a finite number of reads. We propose a divide-and-conquer algorithm, handling large scale problems with hundreds of thousands of species, which is particularly appealing for sparse microbial communities - that is, realistic scenarios where only hundreds or a few thousands of species are present in the mixture, out of the possible millions of species in the database (see e.g. [5, 18]). We study the reconstruction performance in these realistic settings by simulating reads from the Greengenes 16S rRNA database [4].

Our goal here is to formulate and study the problem mathematically. Practical considerations (e.g. amplification and sequencing biases, restrictions on primers, paired-end reads) together with experimental results for real sequencing data are described in a separate publication [1].

In the spirit of reproducible research, we have implemented all of our algorithms in the Matlab package **COMPASS** (Convex Optimization for Microbial Profiling by Aggregating Short Sequences reads), which is freely available at github: <https://github.com/orzuk/COMPASS>.

2 The MCR Problem Formulation

We describe informally and briefly the biological settings. Our goal is to identify the species present in a given sample. We extract DNA, use 16S rRNA universal primers and amplify the DNA in this region. We then assume that DNA is sheared randomly and sequence it using MPS. We assume that the sequences database contains 16S rRNA sequences for *all* species present in the mixture, then we reconstruct the species in the mixture *in silico*. A schematic representation of the **MCR** method is shown in Figure 1.

We denote by N the number of species in the database. The species' 16S rRNA sequences are marked S_1, \dots, S_N , represented as strings over the alphabet $\mathcal{Y} = \{ 'A', 'C', 'G', 'T' \}$. We assume that the S_i 's are *distinct* sequences. The sequences may have *different* lengths n_1, \dots, n_N , with n_i the length of the i -th species' sequence, i.e. $S_i \in \mathcal{Y}^{n_i}$. For the 16S rRNA gene, the lengths n_i are roughly 1500 base-pairs. We define the maximum sequence length as $n_{MAX} \equiv \max_i n_i$. We denote by $s_{i,j}$ the j -th nucleotide in the i -th species' sequence S_i , and by $s_{i,j:k}$ the substring containing nucleotides $j, j+1, \dots, k$ in the sequence S_i .

We represent the proportion of each species in the mixture using a vector \mathbf{x} of length N , with x_i the frequency of species i . We have $\mathbf{x} \in \Delta_N$, where Δ_N is the N -dimensional simplex, $\Delta_N = \{ \mathbf{x} : x_i \geq 0, \sum_{i=1}^N x_i = 1 \}$. We represent the interior of a set \mathcal{A} as $int(\mathcal{A})$. In particular, $int(\Delta_N)$ is the subset of Δ_N containing vectors with positive entries, $int(\Delta_N) = \{ \mathbf{x} : x_i > 0, \sum_{i=1}^N x_i = 1 \}$.

We observe data in the form of R reads of length L , $r_1, \dots, r_R \in \mathcal{Y}^L$, with L typically around $\sim 50-100$, as in the Illumina sequencing technology. We represent the data by a vector of read frequencies, $\mathbf{y} \in \Delta_{4^L}$, with the j -th coordinate given by $y_j = \frac{1}{R} \sum_{i=1}^R 1_{\{lex(r_i)=j\}}$, $\forall j = 1, \dots, 4^L$. Here $lex(r)$ is the index of r in the lexicographic ordering of all 4^L possible reads (i.e. $lex('AAA \dots A') = 1, \dots, lex('TTT \dots T') = 4^L$). We also define the inverse lexicographic ordering transformation, lex^{-1} , which for a given index j , gives the corresponding sequence (e.g. $lex^{-1}(18) = 'AAA \dots ATC'$).

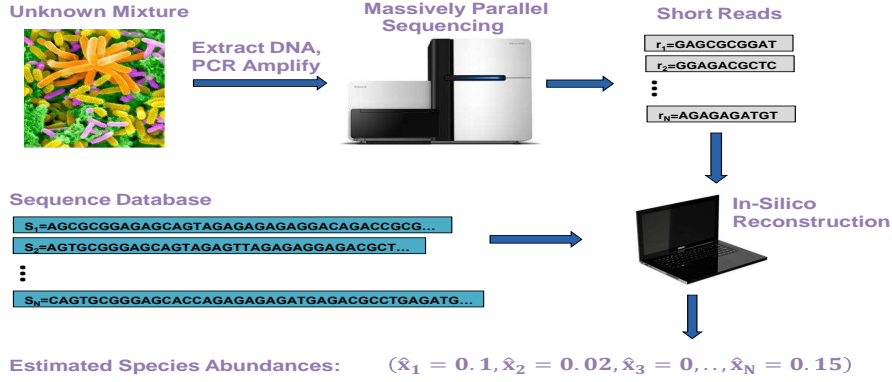


Fig. 1. The steps performed for species reconstruction using the **MCR** method. First, DNA is extracted and amplified using PCR with universal primers matching the 16S rRNA gene. The DNA is then sheared and sequenced using MPS, producing millions of short sequence reads. The sequencing data (reads), together with a database of 16S rRNA sequences, are entered into the computational pipeline providing estimated species abundances as output.

In the **MCR** problem, the data vector \mathbf{y} and the database sequences S_1, \dots, S_N are given as input. Our goal is to reconstruct the species frequencies vector \mathbf{x} from this information. The vector \mathbf{y} is of exponential length (4^L) but very sparse, with only $M \leq R$ non-zero coordinates, where M is the number of *unique* sequence reads. We store and manipulate only the non-zero part of \mathbf{y} - therefore the computational complexity of all of our algorithms will depend on M , and not the exponentially large 4^L (see Section 5). In typical MPS experiments with current technologies R may be on the order of $\sim 10^5 - 10^8$.

2.1 Probabilistic Model

We formulate a probabilistic generative model capturing the sequencing process. We assume that the R reads are sampled identically and independently (i.i.d.) from the set of amplified regions in two steps,

1. First, sample a microbial species b from the set of possible species $\{1, \dots, N\}$, with the probability of species j being sampled proportional to the amount of DNA from this species, $x'_j \equiv Pr(b = j) = \frac{x_j n_j}{\sum_{i=1}^N x_i n_i}$.
2. Next, sample a read r from a distribution given by the species b . We represent sampling probabilities using a $4^L \times N$ read-sampling matrix $A = A(S, L)$ whose (i, j) -th entry is the probability to observe read i given that we know it came from species j , $A_{ij} = Pr(r = i | b = j)$.

Remark 1. The vector of sampling probabilities \mathbf{x}' from step 1 is obtained by re-weighting the frequency vector \mathbf{x} according to the sequence lengths. For ease of notation, we disregard this re-weighting, and denote both vectors as \mathbf{x} . When all sequences lengths n_j are identical we have indeed $\mathbf{x}' = \mathbf{x}$. More generally, the vectors are different but we can easily convert \mathbf{x} to \mathbf{x}' or \mathbf{x}' to \mathbf{x} using the above relation $x'_j = \frac{x_j n_j}{\sum_{i=1}^N x_i n_i}$

Together, the sampling process defines a probability distribution $P_{\mathbf{x}} = P_{\mathbf{x}}(\mathbf{y}; A, L)$ on the space of possible frequencies Δ_{4^L} ,

$$P_{\mathbf{x}}(\mathbf{y}; A, L) = \begin{cases} \sum_{j=1}^N A_{ij} x_j & \mathbf{y} = \mathbf{e}^{(i)} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $\mathbf{e}^{(i)} \in \Delta_{4^L}$ is the i -th vector in the standard basis, $e_i^{(i)} = 1, e_j^{(i)} = 0, \forall j \neq i$. The data can be represented as R i.i.d. random variables, $\mathbf{y}^{(1)}, \dots, \mathbf{y}^{(R)} \sim P_{\mathbf{x}}(\mathbf{y}; A, L)$, with the sample frequency \mathbf{y} represented

as, $\mathbf{y} = \frac{1}{R} \sum_{i=1}^R \mathbf{y}^{(i)}$. We denote the **MCR** problem with read sampling matrix A by **MCR**(L, A). In its simplest form, A can be constructed as follows,

$$A_{ij} = \frac{\sum_{k=1}^{n_j-L+1} 1_{\{lex^{-1}(i)=s_{j,k:k+L-1}\}}}{n_j - L + 1} \quad (2)$$

This matrix represents *uniform* sampling of *error-free* reads along the sequence of the chosen species j , assuming $L < n_j \forall j$. A non-zero element A_{ij} means that read i appears in the sequence of species j . If $L > n_i$ we assume that the ‘tail’ of each read is sampled uniformly from \mathcal{T} (see Appendix Section A.1).

Remark 2. The above construction of A assumes no read errors and no biases. Incorporating more realistic sequencing models with non-uniform read density due to amplification biases, read errors (substitutions and indels), alignments errors etc. can be done by changing the definition of A from eq. (2). The same database S may thus yield different matrices A , and the statistical and algorithmic properties of a certain **MCR** problem depend on the database S only through the matrix A . The assumption in step 1 is that species DNA fragments are sampled according to their DNA frequencies out of the total DNA present in a sample. The model cannot accommodate deviations from this assumption which may arise from amplification biases and limited library complexity, which may distort the species frequencies - that is, the fraction of reads originating from a certain species may not represent the species’ true frequency in the mixture. Accounting and correcting for such biases require analyzing multiple samples together.

In similar to the read frequencies vector \mathbf{y} , the matrix A is also huge ($4^L \times N$) but very sparse. In particular, the number of non-zero rows in A , denoted K is much smaller than 4^L , as most of the rows in A are zero and need not be stored. In the simple model above, $K \leq \sum_{j=1}^N (n_j - L + 1)$, which is roughly equal to the database size in nucleotides. In more complicated models involving read error, K will be larger, but still much smaller than 4^L . The computational complexity of our algorithms depends on K (see Section 5).

An estimator $\hat{\mathbf{x}}$ of the frequency vector \mathbf{x} is simply a function from the set of all reads and database, to the n -dimensional simplex $\hat{\mathbf{x}} : \Delta_{4^L} \times \mathcal{S} \rightarrow \Delta_N$, $\hat{\mathbf{x}} = \hat{\mathbf{x}}(\mathbf{y}, S)$ (here \mathcal{S} is the set of all possible sequences databases, i.e. the space of all ordered finite collections of strings over \mathcal{T}).

We can solve the **MCR** problem by finding an estimator $\hat{\mathbf{x}}$ minimizing an empirical loss function. That is, define $\hat{\mathbf{y}} = A\hat{\mathbf{x}}$, the empirical reads distribution given the estimator $\hat{\mathbf{x}}$. We would like to minimize the loss $l(\hat{\mathbf{y}}, \mathbf{y})$. Thus, one can define the following estimator,

$$\hat{\mathbf{x}} = \operatorname{argmin}_{\mathbf{x} \in \Delta_N} l(A\mathbf{x}, \mathbf{y}) \quad (3)$$

A natural loss function is the Kullback-Leibler divergence $l_{KL}(\mathbf{y}, \hat{\mathbf{y}}) = D(P_{\mathbf{y}} || P_{\hat{\mathbf{y}}})$. This formulation is equivalent to maximizing the likelihood of the data \mathbf{y} , according to the probabilistic model in eq. (1). Maximizing the likelihood using the EM algorithm was proposed in [13] for a very similar likelihood formulation - this approach, however, is currently not scalable to a large number of species. We choose instead the l_2 loss $l_2(\mathbf{y}, \hat{\mathbf{y}}) = \|\mathbf{y} - \hat{\mathbf{y}}\|_2$, mainly for computational considerations. The l_2 loss leads to a standard optimization problem and many off-the-shelf solvers can be used.

We expect real mixtures to be sparse, with only a few hundreds to a few thousands species present (out of hundreds of thousands). it is therefore appealing to use a sparsity-promoting loss in the cost function in eq. (3), for example by penalizing l_0 norm of \mathbf{x} . This is especially important when the number of reads is limited, to avoid over-fitting of the solution to the randomly sampled reads. The l_0 norm is not convex, leading to an intractable computational problems. The most common remedy of replacing the l_0 norm by the convex l_1 norm does not work in our problem since for probability vectors in the simplex $\mathbf{x} \in \Delta_N$ the constraint $\|\mathbf{x}\|_1 = 1$ trivially holds. Promoting sparsity for probability distributions in the simplex by convex relaxation was recently proposed [20], but the approach does not scale to our problem’s size. Instead, we developed a scalable divide-and-conquer thresholding algorithm (see Section 5) which minimizes the l_2 error, while enforcing sparsity implicitly, by a repeated truncation of non-zero frequencies. The resulting solution is guaranteed to be sparse, while still keeping the l_2 error low as desired.

2.2 Evaluating the Solution: Metrics

To evaluate reconstruction accuracy, we need a measure comparing the reconstructed solution $\hat{\mathbf{x}}$, with the correct solution \mathbf{x} . Different applications may require different metrics - for example, in some applications we may be interested only in the *identity* of the species, while in other applications one would want to detect changes in *frequencies*. It may be important to identify the particular species or strain, or one may be satisfied with coarser reconstruction at the genus or family level. There are two major groups of performance metrics:

1. Phylogenetically-Unaware criteria: These metrics take into account only the species identities and frequencies. Examples include the l_p norm between the two vectors, recall-precision and Jaccard index. We use the simple l_2 norm as a representative of this group. This metric measures the deviation in species *frequencies* between the true and reconstructed solutions, $\mathcal{D}_{l_2}(\mathbf{x}, \hat{\mathbf{x}}) = \sqrt{\sum_{i=1}^M (x_i - \hat{x}_i)^2}$.
2. Phylogenetically-Aware criteria: These metrics take into account the phylogenetic relationship between species. The main intuition here is that identifying a species close to the true species is in fact almost as good as reconstructing the correct species. Examples include unifrac [14], weighted unifrac [15], and DPCoA [19]. We propose a novel Phylogenetically-aware criterion, using a Mahalanobis distance,

$$\mathcal{D}_{MA}(\mathbf{x}, \hat{\mathbf{x}}; D) = \sqrt{(\mathbf{x} - \hat{\mathbf{x}})^T D (\mathbf{x} - \hat{\mathbf{x}})} = \sqrt{\sum_{i,j} D_{ij} (x_i - \hat{x}_i)(x_j - \hat{x}_j)}.$$

The matrix D is constructed to capture the phylogenetic distance between species (for example, from the species 16S rRNA sequences themselves). High (low) values of D_{ij} correspond to pairs of species (i, j) which are closely-related (remote). For concreteness, we choose specifically $D = A^T A$, which represent the similarity between species based on their 16S rRNA sequences. The resulting Mahalanobis distance measures the agreement between the true and reconstructed solutions, in terms of both the species identities and their frequencies, while taking into account the similarities between closely related species.

3 Species Identifiability

In this section we study species identifiability - that is, the ability to correctly identify the species and their frequencies as the number of reads, R , goes to infinity.

Definition 1. We say that the problem $\mathbf{MCR}(L, A)$ is identifiable, if for every $\mathbf{x}^{(1)} \neq \mathbf{x}^{(2)} \in \Delta_N$, there exists $\mathbf{y} \in \Delta_{4^L}$ such that $P_{\mathbf{x}^{(1)}}(\mathbf{y}; A; L) \neq P_{\mathbf{x}^{(2)}}(\mathbf{y}; A; L)$.

Species identifiability captures fundamental limits of our ability to reconstruct the species frequency vector from the observed reads data. If the problem $\mathbf{MCR}(L, A)$ is identifiable, then *in principle* it is possible to correctly reconstruct the species frequencies vector \mathbf{x} , since different vectors will generate different distributions on the observed reads. If the problem is not identifiable, recovering the correct frequencies vector \mathbf{x} may not be possible, regardless of the data size and computational resources available, since other (incorrect) frequency vectors give rise to an identical distribution on the observed reads data.

The identifiability question is not unique to the \mathbf{MCR} problem, and arises more generally when reconstructing the identity of long sequences in a mixture using short reads. For example, conditions for the identification of isoforms from RNA-seq data were given in [11]. The different Isoforms in [11] are analogous to the different species in our problem, yet the precise modeling assumptions and identifiability criteria are different in the two problems. Identifiability is determined by both the similarity between the sequences of different species, and the read length. Longer and more diverse sequenced regions provide more information on the DNA sequence of different species in the mixture, and allow to distinguish between the underlying species more easily. However, even when the sequenced regions are informative enough, short sequenced reads obtained from these region may map to multiple species, thus species identification can be hard when reads are too short. We next formalize this intuition mathematically, showing how identifiability is determined by the input sequence database (and the read length L) through the matrix A , which represents the relation between the unknown vector \mathbf{x} and the observed data \mathbf{y} (see Appendix for proofs of all Propositions),

Proposition 1. Let $A_{(1)}$ be the matrix constructed from A , concatenated with an all 1's row vector $\mathbf{1}_N$, $A_{(1)} \equiv \begin{pmatrix} A \\ \mathbf{1}_N \end{pmatrix}$. The reconstruction problem $\mathbf{MCR}(L, A)$ is identifiable if and only if $\text{rank}(A_{(1)}) = N$.

As the read length increases, it becomes increasingly easier to distinguish between species,

Proposition 2. Assume $N > 4$. Suppose that the database S is composed of N distinct sequences such that no sequence is a substring of another sequence, i.e. $s_{i,j:k} \neq s_{i'} \forall i \neq i' \in \{1, \dots, N\}, \forall j, k \in \{1, \dots, n_i\}$. Let $A^{(u,L)}$ be the sampling matrix obtained by uniform sampling of reads with read length L , according to eq. (2). Then there is a critical read length L_c , $1 < L_c \leq \max_i n_i$ such that the problem $\mathbf{MCR}(L, A^{(u,L)})$ is identifiable if and only if $L \geq L_c$.

Remark 3. We assume that no sequence in the database is a substring of another database sequence for mathematical convenience. This assumption usually holds in practice provided a long enough region is sequenced, and can be relaxed while still obtaining similar identifiability results. In addition, we demonstrated identifiability for a uniform read sampling distribution, but a similar result can be obtained for other read sampling distributions.

Species identifiability is a worst-case measure, as it requires *all* species to be identified correctly. In practice, we may settle for a weaker notion - for example we would still consider a reconstruction as successful if all species except a small minority were identified correctly. We next define *partial* identifiability, which is a weaker property characterizing our ability to correctly reconstruct identities and frequencies of specific species, while for other species the reconstruction may remain ambiguous.

Definition 2. We say that the problem $\mathbf{MCR}(L, A)$ is partially identifiable for species j , if for any $\mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \Delta_N$ such that $P_{\mathbf{x}^{(1)}}(\mathbf{y}; A, L) = P_{\mathbf{x}^{(2)}}(\mathbf{y}; A, L) \forall \mathbf{y} \in \Delta_{4^L}$, we have $x_j^{(1)} = x_j^{(2)}$.

We can check partial identifiability using the following proposition,

Proposition 3. Problem $\mathbf{MCR}(L, A)$ is partially identifiable for species j , if and only if the standard basis vector $\mathbf{e}^{(j)} \in \Delta_N$ is orthogonal to the null-space of $A_{(1)}$, that is $A_{(1)}\mathbf{x} = 0 \Rightarrow x_j = 0 \forall \mathbf{x} \in \mathbb{R}^N$.

We present the identifiability properties achieved for real 16S rRNA data in the Appendix (Section A.5).

4 Reconstruction Error

While identifiability ensures that one can *in principle* reconstruct correctly the species vector \mathbf{x} , it essentially assumes an unlimited number of reads and computational power. Here we study the reconstruction error in more realistic scenarios, with a finite number of reads. We next prove general rigorous upper-bounds on reconstruction error, in terms of the matrix A and the number of reads R . In the Appendix (Section A.8) we examine the actual error achieved in practice using simulations.

The next proposition gives bounds on the approximation error of the true frequency vector \mathbf{x}^* by the estimator $\hat{\mathbf{x}}$, which we obtain using the empirically-observed frequencies \mathbf{y} ,

Proposition 4. Consider the problem $\mathbf{MCR}(L, A)$ with R sequence reads, and let $\hat{\mathbf{x}}$ be the estimator minimizing the l_2 loss, $\hat{\mathbf{x}} = \text{argmin}_{\mathbf{x} \in \Delta_N} l_2(A\mathbf{x}, \mathbf{y})$. Then,

1. Let $\lambda_{\min}(A^\top A)$ be the smallest eigenvalue of $A^\top A$. The Euclidian l_2 distance satisfies:

$$\Pr\left(\mathcal{D}_{l_2}(\hat{\mathbf{x}}, \mathbf{x}^*) \leq \frac{2 + \sqrt{\log(1/\delta)}}{\sqrt{R\lambda_{\min}(A^\top A)}}\right) \geq 1 - \delta, \quad \forall \delta \in (0, 1) \quad (4)$$

2. The Mahalanobis distance with weight matrix $A^\top A$ satisfies:

$$\Pr\left(\mathcal{D}_{MA}(\mathbf{x}, \mathbf{x}^*; A^\top A) \leq \frac{2 + \sqrt{\log(1/\delta)}}{\sqrt{R}}\right) \geq 1 - \delta, \quad \forall \delta \in (0, 1). \quad (5)$$

The bound on the convergence rate of the \mathcal{D}_{l_2} error depends on spectral properties of the matrix $A^\top A$. This is related to the database coherence, or similarity between the sequences S_i , encoded as similarity between the rows of A . In particular, when the problem is non-identifiable, the matrix $A^\top A$ has a zero eigenvalue and the reconstruction error may be arbitrarily large.

In contrast, the Mahalanobis bound does not depend on the matrix A or even the dimension N . Even if the problem is non-identifiable, we still achieve convergence under the Mahalanobis distance - yet the entries in the solution vector will not converge to the corresponding entries in the true frequencies vector \mathbf{x} , i.e. the reconstruction may assign (part of) the abundance of a specific species to different, yet highly similar species.

5 Divide-and-Conquer Algorithm

Solving a large scale **MCR** problem with hundreds of thousands of species is computationally challenging. Even computing and storing the matrix A is not trivial, let alone minimizing the loss $l(A\mathbf{x}, \mathbf{y})$ in eq. (3). We developed a scalable divide-and-conquer thresholding approach to cope with large problems. In a nutshell, the algorithm divides the species into distinct blocks - solves a reduced-size problem within each block, setting species with low frequency in the solution for each block to zero, merges solutions from different blocks and iterates to reduce problem size. For the reduced size sub-problems we minimize the l_2 loss, resulting in a convex optimization problem in each block which we solve (exactly) using the CVX convex optimization software package [8, 21]. We describe the algorithm in more details in the Appendix (Section A.7).

We implemented the divide-and-conquer algorithm in the **COMPASS** Matlab package (with some computationally demanding parts implemented in C). For a problem of size $N \sim 5 \times 10^5$, running time is a few hours on a standard PC. The algorithm showed accurate reconstruction performance on simulated and real sequence data (see Section A.8 and [1]).

6 Discussion

We formulated the **MCR** problem mathematically, proposed an algorithm for solving large scale problems, and obtained results on reconstruction performance.

We applied our approach on the 16S rRNA gene. However, the approach is generic and could be applied to other genes or regions. The reconstruction performance is determined by properties of the genomic region used (in our case, 16S rRNA). Different genes or regions will provide different information allowing us to distinguish between different species or strains, for example using clade-specific markers [23].

Extending our method to genome-wide metagenomics sequencing is possible, although computationally challenging. Our approach relies on the presence of database of reference sequences, and cannot be used as is for *de novo* discovery of new species. Currently there are ~ 3000 whole-genome sequences in the NCBI database [9], compared to $\sim 10^6$ 16S rRNA sequences in the Greengenes database, thus the current utility of the whole-genome approach is limited, although it can be useful as a first filter before the remaining reads can be used for *de novo* discovery (assembly). More importantly, as these database are likely to grow in the near future, it will become increasingly appealing to use whole-genome sequencing, especially for identifying small variations in very close strains, or newly born alleles in present strains (where the 16S rRNA sequences may be identical and not allow identification).

Providing efficient algorithms for the **MCR** problem is important - solving the **MCR** problem directly for N in the order of hundreds of thousands is currently infeasible due to memory and time issues. We used a feasible divide-and-conquer approach to cope with this problem yet there is still room for algorithmic improvements, especially when coping with read errors, which effectively increase the size of the matrix A . Designing faster algorithms for handling larger databases will become crucial in light of the expected growth of microbial databases, in terms of both the number species and the regions (including whole-genomes) covered.

References

1. A. Amir, A. Zeisel, O. Zuk, M. Elgart, S. Stern, O. Shamir, P.J. Turnbaugh, Y. Soen, and N. Shental. High resolution microbial community reconstruction by integrating short reads from multiple 16S rRNA regions. *In Revision*, 2013.
2. A. Amir and O. Zuk. Bacterial community reconstruction using compressed sensing. *Journal of Computational Biology*, 18(11):1723–1741, 2011.
3. J.R. Cole, Q. Wang, E. Cardenas, J. Fish, B. Chai, R.J. Farris, A.S. Kulam-Syed-Mohideen, D.M. McGarrell, T. Marsh, G.M. Garrity, et al. The ribosomal database project: improved alignments and new tools for rRNA analysis. *Nucleic Acids Research*, 37(suppl 1):D141–D145, 2009.
4. T.Z. DeSantis, P. Hugenholtz, N. Larsen, M. Rojas, E.L. Brodie, K. Keller, T. Huber, D. Dalevi, P. Hu, and G.L. Andersen. Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with arb. *Applied and environmental microbiology*, 72(7):5069–5072, 2006.
5. P.B. Eckburg, E.M. Bik, C.N. Bernstein, E. Purdom, L. Dethlefsen, M. Sargent, S.R. Gill, K.E. Nelson, and D.A. Relman. Diversity of the human intestinal microbial flora. *Science*, 308(5728):1635–1638, 2005.
6. I. Eskin, F. Hormozdiari, L. Conde, J. Riby, C. Skibola, E. Eskin, and E. Halperin. eALPS: Estimating abundance levels in pooled sequencing using available genotyping data. In *Recomb*, pages 32–44. Springer, 2013.
7. T.J. Gentry, G.S. Wickham, C.W. Schadt, Z. He, and J. Zhou. Microarray applications in microbial ecology research. *Microbial Ecology*, 52(2):159–175, 2006.
8. M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008. http://stanford.edu/~boyd/graph_dcp.html.
9. D.H. Haft and A. Tovchigrechko. High-speed microbial community profiling. *Nature Methods*, 9(8):793–794, 2012.
10. M. Hamady and R. Knight. Microbial community profiling for human microbiome projects: Tools, techniques, and challenges. *Genome Research*, 19(7):1141–1152, 2009.
11. D. Hiller, H. Jiang, W. Xu, and W.H. Wong. Identifiability of isoform deconvolution from junction arrays and rna-seq. *Bioinformatics*, 25(23):3056–3059, 2009.
12. S.M. Huse, L. Dethlefsen, J.A. Huber, D.M. Welch, D.A. Relman, and M.L. Sogin. Exploring microbial diversity and taxonomy using SSU rRNA hypervariable tag sequencing. *PLoS Genetics*, 4(11):e1000255, 2008.
13. D. Kessner, T. Turner, and J. Novembre. Maximum likelihood estimation of frequencies of known haplotypes from pooled sequence data. *Molecular Biology and Evolution*, 30(5):1145–1158, 2013.
14. C. Lozupone and R. Knight. UniFrac: a new phylogenetic method for comparing microbial communities. *Applied and Environmental Microbiology*, 71(12):8228–8235, 2005.
15. C.A. Lozupone, M. Hamady, S.T. Kelley, and R. Knight. Quantitative and qualitative β diversity measures lead to different insights into factors that structure microbial communities. *Applied and Environmental Microbiology*, 73(5):1576–1585, 2007.
16. E.R. Mardis. The impact of next-generation sequencing technology on genetics. *Trends in Genetics*, 24(3):133–141, 2008.
17. P. Meinicke, K.P. Aßhauer, and T. Lingner. Mixture models for analysis of the taxonomic composition of metagenomes. *Bioinformatics*, 27(12):1618–1624, 2011.
18. B.J. Paster, S.K. Boches, J.L. Galvin, R.E. Ericson, C.N. Lau, V.A. Levanos, A. Sahasrabudhe, and F.E. Dewhirst. Bacterial diversity in human subgingival plaque. *Journal of Bacteriology*, 183(12):3770–3783, 2001.
19. S. Pavoine, A.B. Dufour, and D. Chessel. From dissimilarities among species to dissimilarities among communities: a double principal coordinate analysis. *Journal of Theoretical Biology*, 228(4):523–537, 2004.
20. M. Pilanci, L. El Ghaoui, and V. Chandrasekaran. Recovery of sparse probability measures via convex programming. In *NIPS*, 2012.
21. CVX Research. CVX: Matlab software for disciplined convex programming, ver. 2.0. <http://cvxr.com/cvx>, 2012.
22. R.T. Rockafellar. *Convex Analysis, volume 28 of Princeton Mathematics Series*. Princeton University Press, 1970.
23. N. Segata, L. Waldron, A. Ballarini, V. Narasimhan, O. Jousson, and C. Huttenhower. Metagenomic microbial community profiling using unique clade-specific marker genes. *Nature Methods*, 9(8):811–814, 2012.
24. J. Shawe-Taylor and N. Cristianini. *Kernel methods for pattern analysis*. Cambridge University Press, 2004.
25. L.C. Xia, J.A. Cram, T. Chen, J.A. Fuhrman, and F. Sun. Accurate genome relative abundance estimation based on shotgun metagenomic reads. *PloS One*, 6(12):e27992, 2011.

Appendix

A.1 Dealing with Sequences Shorter than the Read Length

In rare cases the read length L might be larger than the sequence length n_j for a particular species j . For completeness, we adopt a convention of a read having its first n_i nucleotides matching the sequence, and the next $n_i - L$ nucleotides distributed uniformly in \mathcal{R}^{L-n_i} . In this case eq. (2) generalizes to,

$$A_{ij} = \frac{4^{\min(0, n_j - L)} \sum_{k=1}^{\max(1, n_j - L + 1)} \mathbf{1}_{\{lex^{-1}(i)_{1:\min(n_j, L)} = s_{j, k:k+L-1}\}}}{\max(1, n_j - L + 1)} \quad (6)$$

where $lex^{-1}(i)_{1:k}$ denotes the first k nucleotides in the i -th read (in lexicographic ordering). One can adopt different conventions for this case, for example obtaining a shorter read (of length n_j), or using a ‘joker’ symbol for the tail (i.e. for example when sequencing the molecule ‘AACGCT’ a read of length 10 will be ‘AACGCTNNNN’). The choice of different conventions does not change our result significantly - we chose the above for mathematical convenience.

A.2 Proof of Proposition 1

Proof. From eq. (1), we have $P_{\mathbf{x}}(\mathbf{e}^{(i)}; A, L) = [A\mathbf{x}]_i, \forall i = 1, \dots, 4^L$. Therefore identifiability holds if and only if $A\mathbf{x}^{(1)} = A\mathbf{x}^{(2)} \Rightarrow \mathbf{x}^{(1)} = \mathbf{x}^{(2)}, \forall \mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \Delta_N$.

The vector $A_{(1)}\mathbf{x}$ is of size $4^L + 1$, obtained as a concatenation of $A\mathbf{x}$ with one additional entry, $[A_{(1)}\mathbf{x}]_{4^L+1} = \sum_{j=1}^N x_j$. For any $\mathbf{x} \in \Delta_N$ the last entry $[A_{(1)}\mathbf{x}]_{4^L+1}$ is equal to 1. Therefore $A_{(1)}\mathbf{x}^{(1)} = A_{(1)}\mathbf{x}^{(2)} \iff A\mathbf{x}^{(1)} = A\mathbf{x}^{(2)}, \forall \mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \Delta_N$.

If $rank(A_{(1)}) = N$, we have $A_{(1)}\mathbf{x}^{(1)} = A_{(1)}\mathbf{x}^{(2)} \Rightarrow \mathbf{x}^{(1)} = \mathbf{x}^{(2)}, \forall \mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \mathbb{R}^N$. Therefore in particular the relation is true for any $\mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \Delta_N \subset \mathbb{R}^N$ and identifiability holds.

Conversely, if $rank(A_{(1)}) < N$ then there exists a non-zero vector $\mathbf{x} \in \mathbb{R}^N, \mathbf{x} \neq \mathbf{0}_N$ in the null-space of $A_{(1)}$. Thus $A_{(1)}\mathbf{x} = 0$ and in particular $[A_{(1)}\mathbf{x}]_{4^L+1} = \sum_{j=1}^N x_j = 0$. Take a vector $\mathbf{x}^{(1)} \in int(\Delta_N)$. Then there exists $\epsilon > 0$ such that $\mathbf{x}^{(2)} \equiv \mathbf{x}^{(1)} + \epsilon\mathbf{x} \in \Delta_N$. But $A\mathbf{x}^{(1)} = A\mathbf{x}^{(2)}$ and $\mathbf{x}^{(1)} \neq \mathbf{x}^{(2)}$, therefore the problem $\mathbf{MCR}(L, S, A)$ is not identifiable. ■

A.3 Proof of Proposition 2

Proof. Take $L = 1$. Then the vector \mathbf{y} simply measures the fraction of ‘A’s, ‘C’s, ‘G’s and ‘T’s in the sample, and is of length 4. The matrix $A^{(u,1)}$ is of size $4 \times N$, and $rank(A^{(u,1)}) \leq 4$. Therefore, there exist a non-zero vector \mathbf{x} in the null-space of $A^{(u,L)}$, $A^{(u,L)}\mathbf{x} = \mathbf{0}$. Let $\mathbf{x}^{(1)} \in int(\Delta_N)$. Then there exists $\epsilon > 0$ such that $\mathbf{x}^{(2)} \equiv \mathbf{x}^{(1)} + \mathbf{x} \in \Delta_N$. But $P_x(\mathbf{x}^{(1)}) = P_x(\mathbf{x}^{(2)})$ for $\mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \Delta_N$. Hence the problem $\mathbf{MCR}(1, A^{(u,1)})$ is not identifiable.

Take $L = n_{MAX} (= \max_i n_i)$. For each species j define the read $r^{(j)} \equiv [S_j : 'A'^{(L-n_j)}]$ where $'A'^{(k)}$ is a string of k consecutive ‘A’s, and $[a : b]$ denotes the concatenation of the two strings a and b . The read $r^{(j)}$ contains the sequence S_j , followed by a string of ‘A’s. Since $S_{j'}$ is not a subsequence of S_j for any $j \neq j'$, the read $r^{(j)}$ cannot appear when sequencing any other sequence $j' \neq j$, so $A_{lex(r^{(j)})_{j'}} = 0 \forall j' \neq j$, and the $lex(r^{(j)})$ -th row of A is all zeros except for the j -th term. This means that A has N independent rows, indexed by $lex(r^{(1)}), \dots, lex(r^{(N)})$ and $rank(A) = N$. Therefore $rank(A_{(1)}) = N$ and the problem $\mathbf{MCR}(n_{MAX}, A^{(u, n_{MAX})})$ is identifiable.

Suppose that the problem is $\mathbf{MCR}(L, A^{(u,L)})$ is identifiable, and let $L' > L$. By definition, for every $\mathbf{x}^{(1)} \neq \mathbf{x}^{(2)} \in \Delta_N$, there exists $\mathbf{y} \in \Delta_{4^L}$ such that $P_{\mathbf{x}^{(1)}}(\mathbf{y}; A; L) \neq P_{\mathbf{x}^{(2)}}(\mathbf{y}; A; L)$. But the distribution

$P_{\mathbf{x}^{(i)}}(\cdot; A; L)$ is obtained by a projection of the distribution $P_{\mathbf{x}^{(i)}}(\cdot; A; L')$ (for $i = 1, 2$), with $P_{\mathbf{x}^{(i)}}(\cdot; A; L) = \sum_{y', y=y'_{1:L}} P_{\mathbf{x}^{(i)}}(\cdot; A; L')$. Therefore, there must exist $\mathbf{y} \in \Delta_{4L'}$ with $P_{\mathbf{x}^{(1)}}(\mathbf{y}'; A; L') \neq P_{\mathbf{x}^{(2)}}(\mathbf{y}'; A; L')$ and the problem $\mathbf{MCR}(L', A^{(u, L')})$ is also identifiable for L' . ■

A.4 Proof of Proposition 3

Proof. In similar to Proposition 1, since $P_{\mathbf{x}}(\mathbf{e}^{(i)}; A, L) = [A\mathbf{x}]_i, \forall i = 1, \dots, 4^L$ we have partial identifiability if and only if $A\mathbf{x}^{(1)} = A\mathbf{x}^{(2)} \Rightarrow \mathbf{x}_j^{(1)} = \mathbf{x}_j^{(2)}, \forall \mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \Delta_N$, which holds if and only if $A_{(1)}\mathbf{x}^{(1)} = A_{(1)}\mathbf{x}^{(2)} \Rightarrow \mathbf{x}_j^{(1)} = \mathbf{x}_j^{(2)}, \forall \mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \Delta_N$.

Assume that $A_{(1)}\mathbf{x} = 0 \Rightarrow x_j = 0 \forall \mathbf{x} \in \mathbb{R}^N$. Then, for any two vectors $\mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \Delta_N$ take $\mathbf{x} = \mathbf{x}^{(1)} - \mathbf{x}^{(2)}$ to get,

$$A_{(1)}\mathbf{x}^{(1)} = A_{(1)}\mathbf{x}^{(2)} \Rightarrow A_{(1)}(\mathbf{x}^{(1)} - \mathbf{x}^{(2)}) = 0 \Rightarrow [\mathbf{x}^{(1)} - \mathbf{x}^{(2)}]_j = 0 \Rightarrow x_j^{(1)} = x_j^{(2)}. \quad (7)$$

Therefore, $\mathbf{MCR}(L, A)$ is partially identifiable for species j . For the other direction, assume that $\mathbf{MCR}(L, A)$ is partially identifiable for species j . Let $\mathbf{x} \in \mathbb{R}^N$. Take some $\mathbf{x}^{(1)} \in \text{int}(\Delta_N)$ and set $\mathbf{x}^{(2)} = \mathbf{x}^{(1)} + \alpha\mathbf{x}$ with $\alpha > 0$ small enough such that $\mathbf{x}^{(2)} \in \Delta_N$. Then,

$$A_{(1)}\mathbf{x} = 0 \Rightarrow A_{(1)}\mathbf{x}^{(1)} = A_{(1)}\mathbf{x}^{(2)} = 0 \Rightarrow x_j^{(1)} = x_j^{(2)} \Rightarrow x_j = 0. \quad (8)$$
■

A.5 Identifiability in the 16S rRNA Database

We checked the ability to identify species based on their 16S rRNA sequences. We downloaded the 16S rRNA Greengenes database from greengenes.1b1.gov [4] (file ‘current_prokMSA_unaligned.fasta.gz’, version dated 2010). After collecting species with identical 16S rRNA sequences, we were left with $N = 455,055$ unique sequences of the 16S rRNA gene, with mean sequence length 1401 - we refer to these N unique sequences as the species. We assume that the entire 16S rRNA gene is available - this can be achieved for example by shot-gun or RNA sequencing (In practice, the choice of primers is restricted due to biochemical considerations. This will affect the region sequenced and therefore all aspects of the reconstruction performance including identifiability - see [1]). Although the sequences are all distinct when considering the entire 16S rRNA sequences, identifiability is not guaranteed since we only observe short reads covering possibly non-unique portions of the 16S rRNA gene, which may cause ambiguities. We plot in Figure 2 the number of uniquely identifiable species as a function of the read length L . Even for very short L , we can identify most species, since the short reads aggregate information from the entire 16S rRNA gene. However, even when L is long ($L = 100$), there is still a small subset of species which are not identifiable.

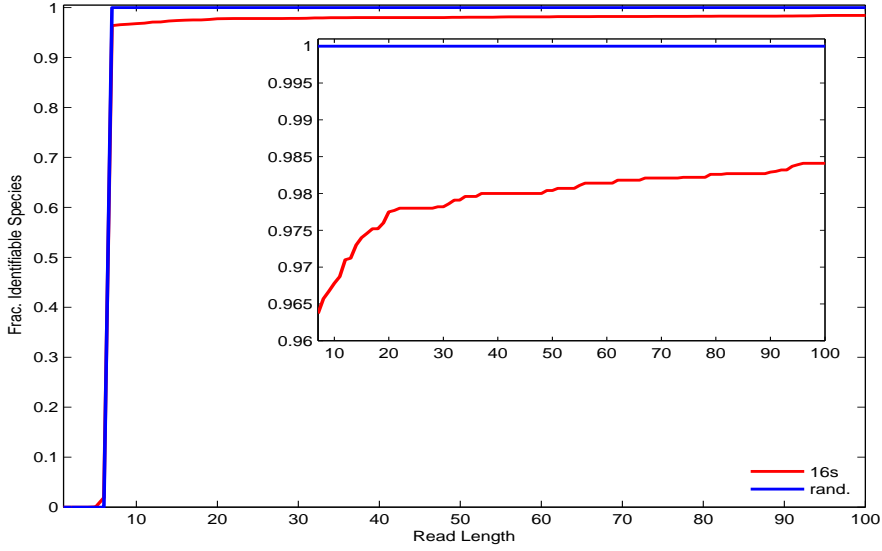


Fig. 2. Partial identifiability as a function of the read length. The red line shows results for a set of $N = 10,000$ similar species from the Greengenes database. For comparison, the blue line shows results for $N = 10,000$ sequences of the same length, with uniformly drawn i.i.d. characters. (i.e. $Pr('A') = Pr('C') = Pr('G') = Pr('T') = 0.25$ for each base). The X-axis is read length used. The y-axis shows the fraction of identifiable species. At $L = 7$ we see a big jump in identifiability, as expected, since this is the point where the number of equations 4^L passes the number of species N . For random sequences the problem is identifiable for $L \geq 7$ (i.e., 100% of species are partially identifiable). For the sequences from the 16S rRNA database, the vast majority ($\sim 96.5\%$) of species are partially identifiable for $L = 7$. The number of partially identifiable species then increases slowly with read length (see inset). Even at $L = 100$ the problem is still not identifiable, but $\sim 98.5\%$ of species can be identified. The remaining un-identified species contain groups of species with very close sequences, which can be resolved only by increasing read length even further.

A.6 Proof of Proposition 4

Proof. Eq. (3) with a l_2 loss implies that $A\mathbf{x}$ is the Euclidean projection of \mathbf{y} on the convex set $A(\Delta_N) \equiv \{\mathbf{z} : \exists \mathbf{x} \in \Delta_N, \mathbf{z} = A\mathbf{x}\}$ (namely, it is the closest point to \mathbf{y} in $A(\Delta_N)$). Similarly, $A\mathbf{x}^*$ is the Euclidean projection of \mathbf{y}^* on $A(\Delta_N)$. Since projections on convex sets can only reduce distances [22], we have,

$$\|A\mathbf{x} - A\mathbf{x}^*\|_2 = \|A\mathbf{x} - \mathbf{y}^*\|_2 \leq \|\mathbf{y} - \mathbf{y}^*\|_2. \quad (9)$$

The left hand side above is equal to the Mahalanobis distance, since

$$\mathcal{D}_{MA}(\mathbf{x}, \mathbf{x}^*; A^\top A) = \sqrt{(\mathbf{x} - \mathbf{x}^*)^\top (A^\top A) (\mathbf{x} - \mathbf{x}^*)} = \|A\mathbf{x} - A\mathbf{x}^*\|_2. \quad (10)$$

Therefore we get

$$\mathcal{D}_{MA}(\mathbf{x}, \mathbf{x}^*; A^\top A) \leq \|\mathbf{y} - \mathbf{y}^*\|_2. \quad (11)$$

Recall that $\mathbf{y} = \frac{1}{R} \sum_{i=1}^R \mathbf{y}^{(i)}$ where the $\mathbf{y}^{(i)}$ are i.i.d. vectors with $E[\mathbf{y}^{(i)}] = \mathbf{y}^*$. Using large-deviation bounds on vectors [24] we get,

$$Pr\left(\|\mathbf{y} - \mathbf{y}^*\|_2 \leq \frac{2}{\sqrt{R}} + \sqrt{\frac{\log(1/\delta)}{R}}\right) \geq 1 - \delta, \quad \forall 0 < \delta < 1 \quad (12)$$

Combining eqs. (11,12), we get part 2 of the proposition.

To prove part 1, we need to convert this result to a bound on the Euclidian distance between \mathbf{x} and \mathbf{x}^* . The conversion is performed by first writing an eigen-decomposition of $A^\top A$, $A^\top A = U\Lambda U^\top$ where U is an orthogonal matrix and Λ a diagonal matrix with the eigenvalues of $A^\top A$. This gives,

$$\begin{aligned} \mathcal{D}_{MA}(\mathbf{x}, \mathbf{x}^*; A^\top A)^2 &= (\mathbf{x} - \mathbf{x}^*)^\top (U\Lambda U^\top)(\mathbf{x} - \mathbf{x}^*) \\ &\geq \|U^\top(\mathbf{x} - \mathbf{x}^*)\|_2^2 \lambda_{\min}(A^\top A) \\ &= \|(\mathbf{x} - \mathbf{x}^*)\|_2^2 \lambda_{\min}(A^\top A) \\ &= \mathcal{D}_{l_2}(\mathbf{x}, \mathbf{x}^*)^2 \lambda_{\min}(A^\top A) \end{aligned} \quad (13)$$

Dividing both sides by $\lambda_{\min}(A^\top A)$, taking the square root and substituting in eq. (5) gives immediately part 1. ■

A.7 Details of Divide-and-Conquer Algorithm

Box 1: Divide-and-Conquer Reconstruction Algorithm

Input: S - Set of Sequences, \mathbf{y} - read measurements, Probabilistic model

Output: \mathbf{x} - vector of species frequencies

Parameters: B - block size. τ_B - frequency threshold for each block. $k_{B,j}$ - number of partitions into blocks in j -th iteration, k_F - final number of species allowed

1. Partition to blocks: Set \mathbf{v} as a binary vector with one entry per species. If this is the first partitioning, set iteration number $j = 1$. Repeat $k_{B,j}$ times:
 - (a) Partition species randomly into non-overlapping blocks of size B .
 - (b) In each block (B) compute the matrix $A^{(B)}$, (where $^{(B)}$ denotes the restriction of a vector or a matrix to a block B), and solve (exactly) the convex optimization problem (using CVX),

$$\min_{\mathbf{x}^{(B)}} \|A^{(B)}\mathbf{x}^{(B)} - \mathbf{y}\|_2 \quad \text{s.t.}, x_i^{(B)} \geq 0 \quad (14)$$

- (c) Collect all species with frequency above the threshold: if $x_i^{(B)} \geq \tau_B$, set $v_i = 1$. Set $j = j + 1$.
 - (d) Collect all linearly dependent species: For each i which is non-identifiable in the block (i.e. species i is orthogonal to the null space of $A^{(B)}$) set $v_i = 1$.
2. Collect results from blocks: Keep only indices i with $v_i = 1$, i.e. species with high enough frequency in at least one block reconstruction.
3. Reduce problem size: Keep only species i with $v_i = 1$. Set $V = \{i, v_i = 1\}$ and set $A = A^{(V)}$, $\mathbf{x} = \mathbf{x}^{(V)}$. If $|V| > k_F$, go back to step 1.
4. Solve for the last time the l_2 minimization problem for the reduced matrix,

$$\min_{\mathbf{x}^{(V)}} \|A^{(V)}\mathbf{x}^{(V)} - \mathbf{y}\|_2 \quad \text{s.t.}, x_i^{(V)} \geq 0 \quad (15)$$

Normalize $\mathbf{x}^{(V)}$ to sum to one, and output the normalized vector as the solution

A.8 Simulation Results

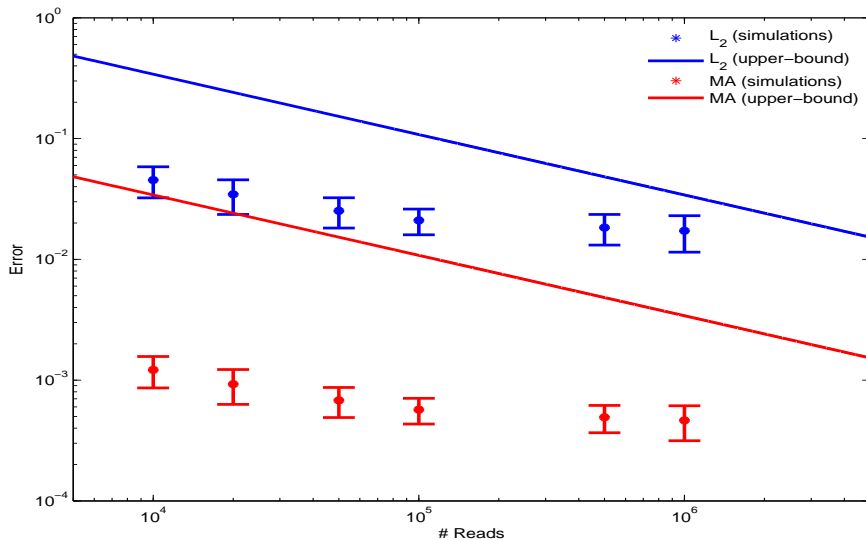


Fig. 3. The curves show the l_2 (blue) and Mahalanobis (red) errors in reconstruction for the example described in the text as function of sample size (number of reads used). Error-bars show mean and 1 standard deviation of error over 100 simulations. Solid curves show the theoretical upper-bounds, taken with $\delta = 1/2$, giving a bound on the median error. For both metrics, the performance achieved in practice is significantly better than the upper bound.

To evaluate the actual reconstruction performance in practice, we have performed a simulation study. In Figure 3 we compare the actual reconstruction performance using simulations to the general rigorous bounds obtained in Section 4.

In our simulations, we studied the performance as a function of the number of reads using the Greengenes 16S rRNA database, with $N = 455,055$ unique 16S rRNA sequences. In each simulation we sampled at random $k = 200$ species out of the total N . We sampled the species frequencies from a power-law distribution with parameter $\alpha = 1$, with frequencies normalized to sum to one. We then sampled sequence read according to the model in eq. (1). Read length was $L = 100$. The number of reads R was varied from 10^4 to 10^6 .

We performed reconstruction using Algorithm 1, with the following parameters: block size $B = 1000$, threshold frequency $\tau_B = 10^{-3}$. The parameter $k_{B,j}$ represents a trade-off between time complexity and accuracy. was initialized to 1 at $j = 1$, then set to 10 when total size was below 150,000. Then, set to 20 below 20,000. The final block size used was $k_F = 1000$.

Very low error ($\sim 2\%$) is achieved for $R > 500,000$, showing that accurate reconstruction is possible for a feasible number of reads. The error rate achieved in practice is much lower than the theoretical bounds, indicating that tighter bounds might be achieved. There are many reasons for the gap between our bounds and simulation results: the concentration inequalities we have used may not be tight, the particular frequency distribution chosen may perform better than the worst-case distribution, and most importantly, the small number of species present in the simulated mixture may enable accurate detection with a smaller sample size. Proving improved bounds on reconstruction performance which consider all these issues including the sparsity of the solution is interesting yet challenging. Standard techniques (e.g. from compressed sensing) would need to be modified to achieve improved bounds since they assume incoherence of the matrix A which does not hold in our case, and do not consider the poisson sampling model we use for the reads.