Kracht and Schober *BMC Bioinformatics* (2015) 16:50
DOI 10.1186/s12859-015-0482-7

BMC Bioinformatics

METHODOLOGY ARTICLE Open Access

Insertion and deletion correcting DNA barcodes based on watermarks

David Kracht* and Steffen Schober

Abstract

Background: Barcode multiplexing is a key strategy for sharing the rising capacity of next-generation sequencing devices: Synthetic DNA tags, called barcodes, are attached to natural DNA fragments within the library preparation procedure. Different libraries, can individually be labeled with barcodes for a joint sequencing procedure. A post-processing step is needed to sort the sequencing data according to their origin, utilizing these DNA labels. The final separation step is called demultiplexing and is mainly determined by the characteristics of the DNA code words used as labels.
Currently, we are facing two different strategies for barcoding: One is based on the Hamming distance, the other uses the edit metric to measure distances of code words. The theory of channel coding provides well-known code constructions for Hamming metric. They provide a large number of code words with variable lengths and maximal correction capability regarding substitution errors. However, some sequencing platforms are known to have exceptional high numbers of insertion or deletion errors. Barcodes based on the edit distance can take insertion and deletion errors into account in the decoding process. Unfortunately, there is no explicit code-construction known that gives optimal codes for edit metric.

Results: In the present work we focus on an entirely different perspective to obtain DNA barcodes. We consider a concatenated code construction, producing so-called watermark codes, which were first proposed by Davey and Mackay, to communicate via binary channels with synchronization errors. We adapt and extend the concepts of watermark codes to use them for DNA sequencing. Moreover, we provide an exemplary set of barcodes that are experimentally compatible with common next-generation sequencing platforms. Finally, a realistic simulation scenario is use to evaluate the proposed codes to show that the watermark concept is suitable for DNA sequencing applications.

Conclusion: Our adaption of watermark codes enables the construction of barcodes that are capable of correcting substitutions, insertion and deletion errors. The presented approach has the advantage of not needing any markers or technical sequences to recover the position of the barcode in the sequencing reads, which poses a significant restriction with other approaches.

Keywords: Next generation sequencing, Barcodes, Multiplexing, Edit distance, Watermark codes, Insertions, Deletions, Sequence embedding

Background

Due to the steadily increasing throughput on platforms for next-generation sequencing and dropping prices of commercially available devices, DNA sequencing becomes broadly accessible for researchers. Since the output of bases in each sequencing run has reached giga to tera orders within the last few years, strategies for efficiently sharing the sequencing capacity has become of particular interest. Multiplexed sequencing is a major key technique, that makes sequencing devices accessible in parallel: DNA samples from different experiments can be pooled into batches and sequenced in parallel in a single sequencing run. Before joining different samples it is mandatory to uniquely label the DNA fragments. DNA barcodes, artificially synthesized sequences of nucleic acids, are used as

*Correspondence: david.kracht@uni-ulm.de
Institute of Communications Engineering, Ulm University, Albert-Einstein-Allee 43, 89081 Ulm, Germany

labels to tag the fragments and to separate the output of the sequencers according to the input samples.

The robustness of multiplexing in general relies on the properties of the used barcodes and how well they are adapted to the underlying sequencing protocol and platform. Essential experimental pre-processing steps, which are needed to prepare the DNA material can cause errors on the target genomic sequence and the barcodes as well. Physical and chemical sequence modifications, e.g. fragmentation, ligation, or copy procedures are known sources of such errors. These errors lead to cross-talk during demultiplexing, i.e., sequences from different batches can not clearly be distinguished, which is of course highly undesirable.

Different constructions for barcodes have been proposed, for example those of Hamady et al. [1] and Bystrykh [2] are based on Hamming codes [3] or the approach of Krishnan et al. [4] based on BCH codes [5], to name just a few. For short lengths it is even feasible to apply brute-force search techniques, e.g. Frank [6] or Mir et al. [7], where some of the resulting codes of the latter approach even reach fundamental bounds. The constructions mentioned so far are designed to correct substitution errors only. From a conceptional point of view all of them try to provide codes that maximize the so-called Hamming distance between the individual code words. The Hamming distance between a pair of sequences measure the minimal number of symbol-wise substitution that are needed to transform them into each other. But, some specific sequencing platforms are known to have exceptional high numbers of insertion and deletion errors as reported for Roche 454 Pyrosequencing [8], PacBio sequencers [9] or Ion Torrent PGM [10]. See [11-13] for a comparison of these sequencing techniques. Hence, especially for insertion and deletion prone devices one has to consider barcodes that are capable to correct indels.

Promising attempts to find barcodes that are robust to indels have been considered in [14,15] using the so-called *edit* or *Levenshtein* distance (see [16] for an overview). For calculating the distance between code words the Levenshtein distance takes insertion and deletion operations into account and is therefore better suited for applications where decoding based on Hamming distance fails. Unfortunately, there is no code-construction known that directly gives optimal codes in edit metric. Some greedy (later evolutionary) algorithms has been proposed in [17,18] to find sets of barcodes of moderate size with high minimal edit distance, additionally fulfilling biological constraints. However, a practical decoding step for the obtained barcodes has not been addressed in the mentioned papers. This was later done by [19], where it is stated that maximizing the edit distance for barcodes (within a sequence context) is a sub-optimal or even wrong strategy. The context of a code words, which is simply the sequence that contains the DNA tag plays an important role. Due to indels the exact boundaries of code words can not be correctly recovered. This leads to additional errors, if the sequence context was not included in the code construction. The DNA context at one end of a code word can be taken into account by using an adapted *Sequence-Levenshtein distance*, as proposed in [19].

In this manuscript, we provide an entire different perspective to obtain barcodes. We give codes based on concepts introduced by Davey and Mackay [20]. The original watermark approach is aimed to synchronize and decode a continuous stream of large binary data-blocks. In the domain of DNA codes we face additional constraints, for which the original concept is adapted. We finally give an exemplary set of barcodes and provide an in silico application, which shows that demultiplexing based on the watermark concept is applicable in the field of next-generation sequencing. Basic concepts of watermark coding has already been considered for data embedding in DNA [21,22], which is closely related to the barcoding approach for DNA sequencing. But, the transmission of biologically compatible sequences through an evolutionary channel (in living cells) is only slightly similar to the approach we consider in the present manuscript.

Note, that search approaches like [16,19] can be used to find better codes in terms of code rate and minimal (sequence) edit distance, but we see two striking advantages of the watermark concept for barcoding. First, the watermark concept contains an implicit synchronization technique, that does not need preambles or markers to find boundaries of code words within an unknown sequence. Embedding of barcodes in an unknown context is not generalized in approaches considering barcodes based on (sequence) edit distance. A two-ended embedding of sequences is not reflected in this metric. Furthermore, we are able to give an optimal decoding procedure, adapted to a specific error channel. In short, this enables a maximum degree of freedom for existing as well as future experimental settings. Decoding speed is the second important aspect of multiplexing approaches, as the number of barcodes and the available read-length dramatically increased within the last few years. The decoding of barcodes based on watermarks also provides an efficient method for fast decoding of large scale multiplexing approaches.

Methods

The main concepts presented in this section originates from the ideas first given by Davey and MacKay [20]. The fact, that quaternary watermark codes can be applied for DNA sequencing was preliminary outlined in [23], but sequence constraint on practical oligonucleotides can not be found in this conference paper. Let us denote some basic facts about barcodes first and postpone specific

sequence constraints to the section about reasonable encoder settings.

For applying the concepts of watermark codes on DNA barcoding we have to focus on the following constraints, with implications for the modifications we consider here. We highlight our contribution to the basic concepts with the following barcode constraints: A barcode is

- a quaternary sequence. Therefore we will propose generalized non-binary models and concepts here (the original channel was considers strictly binary).
- in general embedded in other sequences. Hence we give an adapted transmission scheme and a novel approach to detect barcode boundaries (in [20] the watermark pattern is repetitive and symbols are decoded as stream).
- length-limited. That consequently restricts the number of code-constructions for which an adaption is practical (long LDPC codes are use in the original paper, not plausible for short barcodes).

We will stress additional contributions to the work of Davey and MacKay, where needed. Let us start with a model for the sequencing process and continue with encoding and decoding based on watermarks.

Sequencing model

We will first define a communications theoretic model to formally describe the barcoding application.

Substitution model

A simplistic communication theoretic model for the sequencing of barcodes has been proposed in [7]. Namely, a fixed barcode word $\mathbf{b} \in \mathcal{A}^N$, with $\mathcal{A} = \{A, G, C, T\}$ as set of possible symbols (nucleic acids), is entering a *communication channel* with output $\mathbf{r} \in \mathcal{A}^N$, where the channel itself is described by the conditional probabilities to receive a string $\mathbf{y}$ if $\mathbf{x}$ was sent, for $\mathbf{x}, \mathbf{y} \in \mathcal{A}^N$.

For our purposes this model has to be extended into two directions: First, a barcode word $\mathbf{b}$ is assumed to be embedded into a randomly chosen context to obtain the sequence $\mathbf{t}$ (details will be given below). Second, the sequence $\mathbf{t}$ is sent over a so-called *Sequencing Channel* resulting in the received word $\mathbf{r}$. The channel not only substitutes symbols from $\mathbf{t}$, but is also able to delete or insert symbols, hence the length of $\mathbf{r}$ and $\mathbf{t}$ may differ.

Embedding of barcodes

Given a barcode word $\mathbf{b}$ of length N the sequence $\mathbf{t}$ is obtained as the concatenation of two random sequences $\mathbf{t}_{\text{pre}}$, $\mathbf{t}_{\text{post}}$ and the sequence $\mathbf{b}$ as follows

$$\mathbf{t} = \underbrace{t_1 \cdots}_{\mathbf{t}_{\text{pre}}} \underbrace{t_{\delta+1} \cdots t_{\delta+N}}_{\mathbf{b}=b_1 b_2 \cdots b_N} \underbrace{\cdots t_L}_{\mathbf{t}_{\text{post}}} \in \mathcal{A}^L,$$

with $\delta \in \{0, 1, .., L - N\}$. The sequences $\mathbf{t}_{\text{pre}}$ and $\mathbf{t}_{\text{post}}$ are assumed to have a random length and the symbols are chosen uniformly at random, hence the length L of $\mathbf{t}$ is a random variable. We will later choose the lengths of the embedding sequences according to a quasi-normal length-distribution on integers (see Section *Simulations*).

Barcoding and working hypotheses

In this paragraph we like to focus on two different paradigm for sequence embedding in our barcoding approach: First we like to address the total embedding of barcodes. Well, for most sequencing scenarios, we find the barcodes next to fixed technical sequences, which are more reliable due to sequence specific (biological) reactions, e.g. primer or adapter oligonucleotides. For sequencing experiments, where we can rely on the exact knowledge of the position of a barcode at one end, highly optimized code words has been proposed in [19]. As an extension of our work is possible to include $\mathbf{t}_{\text{pre}}$ or $\mathbf{t}_{\text{post}}$ as partially known, nevertheless we want to restrict ourselves to have no prior knowledge about the context. The main gain from this very strict premise is a striking freedom of experimental setups for which we can apply our concepts. For example on platforms with paired end sequencing we are able to decode barcodes independent of the direction of reads not restricting the reads to start with a barcode symbol.

The second aspect is the assumption of inherent barcoded sequences. In this manuscript we focus on the problem of decoding (discrimination of code word sequences), conditioned on the fact that a code word is present in the multiplexed sequences. The problem of detecting a barcode (if it is not guaranteed that every sequence contains one) is a more challenging problem, which we want to avoid in the present assay. On codes based on sequence edit distance this extended problem is addressed for, e.g. the PacBio SMRT platform in [24]. We conjecture that the detection problem of barcodes based on watermarks can be solved in future investigations. Such investigation might also lead beyond the barcoding for sequencing applications, which we will address at the end of this manuscript. Nevertheless, we rely on barcoded samples for the following considerations.

Sequencing channel model

We define a very simplified model for sequence errors and discuss some aspects of oversimplification in the next paragraph. Let us describe the processes involved in sequencing as a memoryless quaternary channel, i.e. each symbol is handled independently of others. This channel model is specified by a set of parameters $\mathbf{S}$, p_i and p_d that are integrated as follows: A transition matrix $\mathbf{S} \in \mathbb{R}^{4\times4}$ which describes the substitution probabilities, and p_i and p_d to specify insertion and deletion probabilities.

The channel is modeled as an infinite state-machine on symbol level, in which a symbol t_i is queued to pass the channel and therefore will undergo one of the following three events: With probability p_i, the symbol t_i remains in the queue and the received stream is prolonged with a random inserted symbols where we assume an uniform distribution on $\mathcal{A} = \{A, G, C, T\}$. With probability p_d, the actual queued symbol is deleted. With probability $p_t = 1-p_i-p_d$ the symbol t_i is passed to a substitution channel which substitutes the symbol t_i according to the transition matrix $\mathbf{S}$, with transition probabilities $S(r_j, t_i) = Pr(r_j|t_i)$. In order to downsize the number of parameters in our model, we will consider $\mathbf{S}$ as symmetric 4-ary substitution matrix later, i.e., we consider substitutions from a base into another with a single error parameter (similar to the model proposed by Jukes and Cantor [25]). Nevertheless, the model considered here would be able to mimic a refined channel, if exact empirical parameters can be considered.

Empirical parameters for a channel model

Empirical data about sequence errors can be seen as the crux of all HMM based approaches in the field of DNA sequencing, as predictions can only perform as good as the underlying assumptions. But, aside from advertising error rate of the big vendors of sequencing devices, real estimates are rather rare to find in literature. Further, there is no real agreement about a common technique to mine such data in a correct way, e.g, using sequence alignment to predict error count is recently in critic of over-fitting, due to predefined alignment costs with impact on the estimates. The commutability of substitutions, insertion and deletion events (a substitution is equal to a deletion followed by an insertion) made things even more difficult. Additionally, there are some indications, that the sequencing channel is more complex as the model we utilize in our approach: Sequencing errors seem to be highly depend on the sequence context, with extended implications on the distribution of symbols in sequenced reads. For Illumina this was shown, e.g. in [26]. We know that the DNA polymerase molecule is prone to bursts of insertions and deletions, if for example repetitive symbols (homopolymers) are present in the physical template sequence. Note, that the reliability of the approach presented here is sensitive to empirical channel parameters to obtain best estimates for demultiplexing samples. A stepwise refinement driven by the feedback of experimental studies is mandatory to adapt watermark barcodes for the demands of different sequencing platforms.

Demultiplexing problem

During the multiplexing step, different samples are labeled with different barcodes, for example $\mathbf{t}'$ is labeled with $\mathbf{b}'$ ($\mathbf{b}'$ embedded in $\mathbf{t}'$) and $\mathbf{t}''$ contains $\mathbf{b}''$ (see Figure 1). After passing the discussed sequencing channel the resulting sequences $\mathbf{r}'$ and $\mathbf{r}''$ can differ from $\mathbf{t}'$ and $\mathbf{t}''$. As the barcodes are possibly affected by errors, $\mathbf{r}'$ and $\mathbf{r}''$ might be associated to the wrong origin during the demultiplexing step, what is called crosstalk. The encoding and decoding based on watermarks is able to reduce such crosstalk.

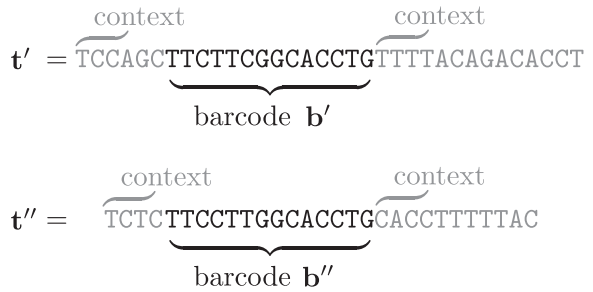

Figure 1 Sequence embedding of barcodes. Two example sequences $\mathbf{t}'$ and $\mathbf{t}''$ labeled with barcodes $\mathbf{b}'$ and $\mathbf{b}''$ embedded in an unknown DNA context.

Barcode construction on watermarks

For the construction of barcodes we use concatenated encoding similar to the scheme proposed by Davey and Mackay, which consists of the following two blocks (see Figure 2): An outer code $\mathcal{C}_1$ with parameters $[\mathbb{F}_{q_1}, n_1, k_1]$ is a code of length n_1, dimension k_1 and alphabet size q_1 (Galois field $\mathbb{F}_{q_1}$), that provides a set of $q_1^{k_1}$ code words. Note, that long LDPC outer codes are used in [20]. In order to avoid the constructions of short LDPC codes for barcoding, we will consider different outer codes later. It is worth to mention, that minimal distance and the ability for soft decoding is the only demands on outer codes. We consider information words $\mathbf{c} \in \mathbb{F}_{q_1}^{k_1}$, which are mapped to inner code words $\mathbf{d} \in \mathcal{C}_1 \subset \mathbb{F}_{q_1}^{n_1}$. The code $\mathcal{C}_1$ provides a set of code words with a high minimal Hamming distance. The $n_1 - k_1$ redundant symbols are used to arrange outer code words as distant as possible. Such a code can give a code rate of $R_1 = \frac{k_1}{n_1}$.

The second block consist of an inner encoder, which works in a complete inverse direction. An inner code $\mathcal{C}_2$ is used to create barcode words that have a low Hamming distance to a watermark sequence. The similarity of all barcodes to this watermark pattern is utilized to gain synchronization as explained below. The inner code adds redundancy to the code words by mapping each outer symbols $d_j \in \mathbb{F}_{q_1}$ to a sparse sequence representation $\mathbf{s}^{(j)} \in \mathbb{Z}_4^{n_2}$. The set of sequences $\mathbf{s}^{(j)}$ with low mean Hamming weight (number of non-zero symbols) can be seen as inner code $\mathcal{C}_2$. The cardinality of this inner code set is q_1 and the rate of the inner code can be stated as $R_2 = \frac{\log_2(q_1)}{n_2 \log_2(4)}$. By joining n_1 of these inner code words, a

sparse inner block $\mathbf{s} = \mathbf{s}^{(1)}\mathbf{s}^{(2)} \cdots \mathbf{s}^{(n_1)}$ of length $N = n_1 n_2$ is generated.

A barcode word $\mathbf{b} = \mathbf{s} \oplus \mathbf{w} \in \mathcal{A}^N$ is obtained via a symbol-wise *adding* of an arbitrary watermark sequence $\mathbf{w}$ to the sparse inner block $\mathbf{s}$ using a fixed mapping of $\mathcal{A} = \{A, G, C, T\}$ onto $\mathbb{Z}_4$, where addition is defined modulo 4 (explicit mappings in Additional file 1 section). The final set of barcodes is denoted as $\mathcal{C} = \mathcal{C}_2 \circ \mathcal{C}_1$ throughout this manuscript. The code $\mathcal{C}$ gives a code rate $R = \frac{k_1}{n_2} \frac{\log_2(q_1)}{\log_2(4)}$.

Decoding

The decoder consists of two blocks (see Figure 2): An inner decoder $\mathcal{D}_2$, which utilizes a hidden Markov model (HMM) and channel parameters $\mathcal{H}$ to provide symbol-wise likelihoods. These are fed into the outer decoder $\mathcal{D}_1$ that performs a maximum likelihood decoding to obtain an estimate $\hat{\mathbf{c}}$ of the sent code word. We will first define the HMM and explain how this can be used to find the likeliest transmitted sequence (optimal decoding). Afterwards we give a modified HMM, that enables to estimate the boundaries of embedded barcodes and end this section with a suboptimal symbol-wise decoding approach with lowered complexity.

HMM for decoding

The basic idea of the HMM presented in this paragraph refers to considerations in [20]. To explain the function of the HMM for decoding it is helpful to ignore the random context and the embedding of code words initially. Therefore we assume $\mathbf{t}_{\text{pre}}$ and $\mathbf{t}_{\text{post}}$ to be absent and the transmitted sequence is exactly one barcode, i.e. $\mathbf{t} = t_1 t_2 \cdot \cdot t_{M=N} = \mathbf{b}$.

If no insertions or deletions occur in a channel the received sequence $\mathbf{r} = r_1 r_2 \cdots r_L$ is as long as the sent sequence $\mathbf{t}$, i.e. $L = M$, but some symbols r_i might differ from t_i. Assuming that errors occur independent of the position and equally distributed, we can use fixed substitution probabilities to describe the channel.

For channels with insertion and deletion events the symbol-wise fixed association $t_i \leftrightsquigarrow r_i$ is usually lost. For example, for a single insertion event at the i-th symbol, t_i will be associated to r_{i+1}. A single deletion event before transmitting the i-the symbol, will shift t_i to the symbol r_{i-1} in the received sequence. Obviously, such errors accumulate during the transmission.

One of the main problems of decoding is to estimate the number of insertions and deletions given a received sequence $\mathbf{r}$. Therefore we define the *drift* x_i at the i-th transmit symbol as (# insertions) - (# deletions) that occurred in the received sequence before taking t_i into account. The drifts $\{x_i\}$ can be seen as the hidden states of an HMM.

Further, we assume the received sequence

$$\mathbf{r} = r_1 r_2 \cdot \cdot r_M = \mathbf{r}^{(1)}\mathbf{r}^{(2)} \cdot \cdot \mathbf{r}^{(L)}$$

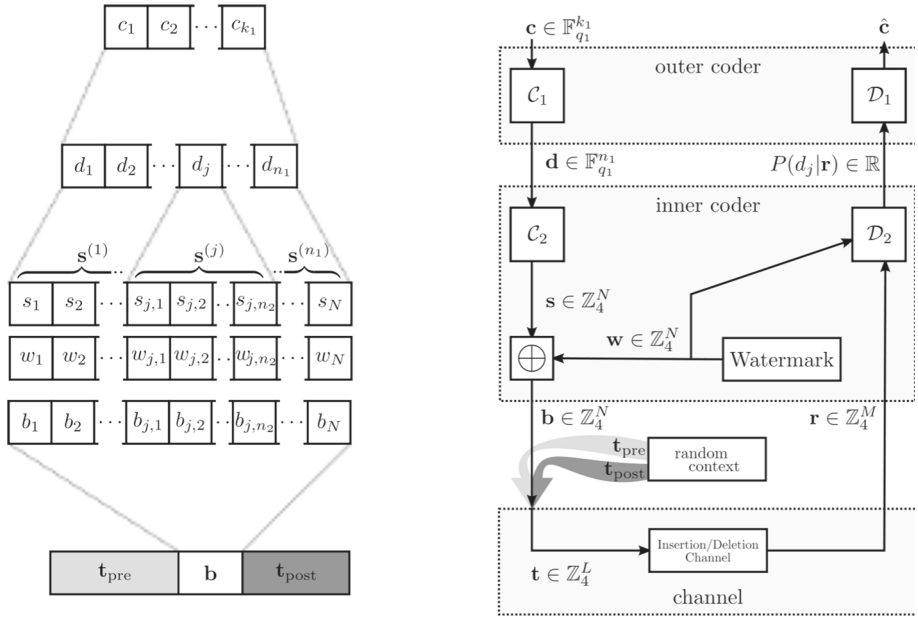

Figure 2 Transmission scheme for barcodes based on watermark codes. Consisting of three blocks: 1) outer coder, encoding index words $\mathbf{c}$ as code words of a linear code $\boldsymbol{\mathcal{C}}_1[\mathbb{F}_{q_1}, n_1, k_1]$ and soft-decoder $\boldsymbol{\mathcal{D}}_1$. 2) inner coder, with inner code $\boldsymbol{\mathcal{C}}_2$ (for sparse sequence representation), decoder $\boldsymbol{\mathcal{D}}_2$ (providing likelihood values) and watermark sequence $\mathbf{w}$ (known to encoder and decoder). 3) embedding of barcodes in random context and channel with insertion, deletion and substitution errors.

to be assembled of sub-sequences $\mathbf{r}^{(i)}$, as observables, based on the hidden state x_i and the transmit symbol t_i. Every state transition $x_{i-1} \rightarrow x_i$ causes an emission of a sub-sequences $\mathbf{r}^{(i)}$, that is associated to the position i in $\mathbf{t}$ (in general HMMs the emissions are associated to single states and not to transitions, compare with Figure 3). To characterize the transition probabilities among hidden states and the emission probabilities of observables in the HMM, we use the following set of parameters $\mathcal{H}$: $\{\mathbf{S}, p_i, p_d, I\}$. Although we used an identical notation for parameters as before (infinite state-machine in section *Sequencing Channel*), the channel model and the HMM discussed here are not equivalent.

The matrix $\mathbf{S}$ is parametrizing substitution events, with p_i and p_d we model the probabilities of insertion and deletions and I is used to limit the maximal length of observables. For computational reasons we suppose that only I consecutive (leading) insertions can be present in observables (the channel model is capable to insert an infinite number of symbols). We further assume that the last symbol of an observable $\mathbf{r}^{(i)}$ is created by either deleting the actual transmit symbol t_i or appending t_i at the end, substituted according to $\mathbf{S}$. If t_i is deleted, we can either just observe $\mathbf{r}^{(i)} = \epsilon$ (the empty symbol) or the last symbol of the observable will be a random (inserted) symbol, if leading insertion are present. This limits the set of observables in our model to

$$\mathbf{r}^{(i)} \in \left\{ \epsilon, t_i, \bar{t}_i, *t_i, *\bar{t}_i, **t_i, \ldots, \underbrace{**\cdots*}_{I}\bar{t}_i \right\},$$

where we use wild cards $*$ to symbolize random leading insertions. With $\bar{t}_i \in \mathbb{Z}_4 \setminus t_i$ we denote symbols differing from t_i.

Now we give the transition and emission probabilities of the stated HMM. Due to our non-binary adaption, we use a slightly different notations than used in the original approach in [20]. An elementary part for both probabilities is the length distribution of observables, that can be derived from

$$\alpha_l(l') = \begin{cases} 1 & \text{for } l' = 0 \\ \frac{(p_i)^{l'}}{1-(p_i)^I} & \text{for } 0 < l' < I \\ 0 & \text{else,} \end{cases}$$

the probability of observing l' random insertions, conditioned on an observable of length l. The probability for a sequence $\mathbf{r}^{(i)}$ of length l is given by

$$p(l) = \alpha_l(l)p_d + \alpha_l(l-1)p_t,$$

with $p_t = 1 - p_i - p_d$. The sequence can be obtained via l insertions and deleting t_i or via $l-1$ leading insertions and attaching t_i (or $\bar{t}_i$) as last symbol.

The transition probability of the HMM can easily be stated as $\Pr(x_i|x_{i-1}) = p(l)$ by choosing $l = x_i - x_{i-1} + 1$, i.e. two consecutive drift states determine the length of observables and vice versa. The joined probability $\Pr(\mathbf{r}^{(i)}, x_i|x_{i-1})$ of observing $\mathbf{r}^{(i)}$ with a state transition $x_{i-1} \rightarrow x_i$ is

$$Q^S(l, r_l^{(i)}, t_i) = \frac{\alpha_l(l)}{4^l}p_d + \frac{\alpha_l(l-1)}{4^{l-1}}p_t S(r_l^{(i)}, t_i),$$

with $0 \leq l \leq I+1$. Explicitly, $\mathbf{r}^{(i)}$ can consists of l random insertions and is not associated with t_i (as t_i is deleted with probability p_d). It is also possible to have $l-1$ leading insertions and t_i is substituted according to the substitution matrix $\mathbf{S}$. The probability Q^S can easily be used to calculate likelihoods $\Pr(\mathbf{r}|\mathcal{H}, \mathbf{t})$ as shown in the next section. Note, for the original binary perspective in [20] no matrix representation of the substitution probability is needed. Here we generalize the basic concepts.

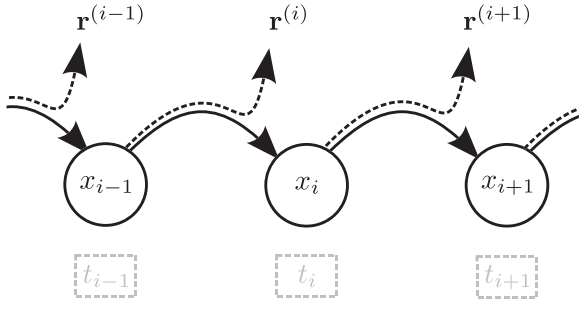

Figure 3 Illustration of the HMM. Exemplary series of transitions of the HMM and observables $\mathbf{r}^{(j)}$, based on state transition $x_{j-1} \rightarrow x_j$ and transmit symbols t_j for $j \in \{i-1, i, i+1\}$.

Optimal decoding

Given a received sequence $\mathbf{r} = r_1 r_2 \cdots r_L$, the set of possible barcodes $\mathbf{b} = b_1 b_2 \cdots b_N \in \mathcal{C}$ and model parameters $\mathcal{H}$ we can describe the decoding as the following maximization approach

$$\hat{\mathbf{b}} = \arg\max_{\mathbf{b}\in\mathcal{C}} \{\Pr(\mathbf{r}|\mathcal{H}, \mathbf{b})\}.$$

The sequence $\mathbf{r}$ is most likely to be captured, if $\hat{\mathbf{b}}$ is assumed to be the transmitted sequence and the channel exactly acts as described with the model parameters $\mathcal{H}$. Let us shortly explain how $\Pr(\mathbf{r}|\mathcal{H}, \mathbf{b})$ can be calculated for the given HMM. The following methods can be found in common text books or, e.g. in [27], but we recall the calculations as we focus on a very special kind of HMM here. Therefore we have to associate the observables of the HMM to received symbols, which is achieved with the drift states. If the barcode symbol b_i is assumed to be the i-th transmit symbol t_i, then b_i can be associated to the i-th observable $\mathbf{r}^{(i)}$ by the HMM. The drift state x_i can be used to associate the last symbol $r_l^{(i)}$ of an observable to

the symbol r_{i+x_i} in the received sequence $\mathbf{r}$. To calculate likelihood values we consider the so-called forward metric

$$F_i(y) = \Pr\left(r_1 \cdots r_{i-1+y}, x_{i-1} = y | \mathcal{H}, \mathbf{b}\right)$$

as probability of having reached the drift state y and received symbols $r_1 \cdot \cdot r_{i-1+y}$ before considering the i-th transmitted symbol. In a transmission setting where $\mathbf{t} = \mathbf{b}$ we can not have any non-zero drift state before considering the first symbol b_1. Thus we define $F_1(y) = \Pr(x_0 = y)$ as $F_1(y = 0) = 1$ and $F_1(y \neq 0) = 0$. The forward metric can easily be calculated as

$$F_{i+1}(y) = \sum_{l=0}^{I+1} F_i(y - \delta_l) Q^S(l, r_{i+y}, b_i),$$

with $\delta_l = l - 1$, for $1 \leq i \leq N$ and $1 \leq i + y \leq M$. The likelihood values $\Pr(\mathbf{r}|\mathcal{H}, \mathbf{b})$ can be obtained as $F_{N+1}(L - M)$. Using this optimal decoding approach there is no need for outer decoding and demultiplexing ends up with an estimated barcode as output of the inner decoder. For large sets of code words this maximum likelihood approach becomes computation expensive, as for an outer code of dimension k_1 and symbol alphabet of size q_1 there are $q_1^{k_1}$ calculations of the forward metric left for decoding. Before we give a suboptimal decoder with reduced complexity, we have to consider how sequence boundaries can be obtained for barcodes embedded in random context.

HMM to estimate barcode boundaries

Up to this paragraph we have not discussed the role of the watermark in the transmission. We have just illustrated an HMM able to give the likeliest received sequences $\mathbf{r}$ conditioned to a hidden transmit sequence $\mathbf{t}$ and parameters $\mathcal{H}$. We will now take a closer look at the watermark and how the inner encoding can be understood from a communication theoretic point of view (see Figure 2).

Recall that a barcode is constructed via addition of two quaternary sequences $\mathbf{s}, \mathbf{w} \in \mathbb{Z}_4^N$ as $\mathbf{b} = \mathbf{s} \oplus \mathbf{w}$. Due to the symmetry of the addition, there are two ways to perceive a data transmission: Apparently there is the transmission of $\mathbf{s} \overset{\mathbf{w}}{\rightarrow} \mathbf{b}$ with $\mathbf{w}$ causing some distortion. A further valid perspective of the transmission is $\mathbf{w} \overset{\mathbf{s}}{\rightarrow} \mathbf{b}$, which takes $\mathbf{w}$ as source with $\mathbf{s}$ causing substitution errors. Here we stick to the last concept treating the symbols s_i as independent and identically (iid) distributed errors on $\mathbf{w}$ (which is used to find the boundaries of barcodes).

Given an iid assumption for symbols s_i and the inner code $\mathcal{C}_2$ we can calculate an extended substitution matrix $\mathbf{S}'$, with probabilities $S'(b_i, w_i)$ for placing a symbol b_i in the barcode, when a w_i is present at position i. We use the matrix $\mathbf{S}'$ to define an upstream meta-channel that causes additional substitutions to those generated by the real sequencing channel. In analogy to the previous paragraph, we can state emission probabilities

$$Q_{w_i}^E\left(l, r_l^{(i)}\right) = \tfrac{\alpha_l(l)}{4^l} p_d + \tfrac{\alpha_l(l-1)}{4^{l-1}} p_t E\left(r_l^{(i)}, w_i\right),$$

for observing a certain $\mathbf{r}^{(i)}$ if the symbol w_i is present at position i. With

$$E\left(r_l^{(i)}, w_i\right) = \sum_b S\left(r_l^{(i)}, b\right) S'(b, w_i)$$

we denote the *effective* probability of having substitutions due to the sequences $\mathbf{s}$ and the substitution matrix $\mathbf{S}$. The task of estimating barcode boundaries can be reduced to the estimation of the likeliest sequence of drift states $\Pr(x_1 x_2 \cdot \cdot x_{N+1}|\mathcal{H}, \mathbf{r}, \mathbf{w})$ in the HMM using $Q_{w_i}^E$ as we show in the next section.

Finding barcode boundaries

For embedded code words we can understand the symbol b_i as shifted transmit symbol $t_{\delta+i}$ and thus b_i has to be linked to the observable $\mathbf{r}^{(\delta+i)}$ in the HMM (see section *Embedding of barcodes*). But we can easily integrate the sequence offset δ as initial drift x_0 . For the symbols b_i we therefore redefine the drift states $\{x_i\}$ for the HMMs according to embedded symbols.

To calculate likelihood values for received sequences, we now consider the forward metric

$$F_i(y) = \Pr(r_1 \cdots r_{i-1+y}, x_{i-1} = y | \mathcal{H}, \mathbf{w})$$

as probability of having reached the drift state y and received symbols $r_1 \cdot \cdot r_{i-1+y}$ before considering the i-th watermark symbol. We furthermore have to determine an initial distribution for the quantity $F_1(y)$ to be able to calculated the forward metric

$$F_{i+1}(y) = \sum_{l=0}^{I+1} F_i(y - \delta_l) Q_{w_i}^E\left(l, r_{i+y}\right),$$

with $\delta_l = l - 1$. We can furthermore calculate backward quantities

$$B_i(y) = \Pr(r_{i+1+y} \cdots | x_i = y, \mathcal{H}, \mathbf{w})$$

as probability of receiving a certain tail of symbols starting with r_{i+1+y} given a state y associated with the i-th watermark symbol, which can be calculated as

$$B_{i-1}(y) = \sum_{l=0}^{I+1} B_i(y + \delta_l) Q_{w_i}^E\left(l, r_{i+y+\delta_l}\right).$$

If we have a good guess for the distribution $F_1(y_1)$ and $B_N(y_N)$, i.e. an a priori distribution of having barcode boundaries at position y_1 respective $N + y_N$, we make use of it. To enable an alignment of the embedded barcodes, we have to introduce novel prior distributions, slightly different to those proposed by Davey and MacKay. Anyway, a conservative approach is setting non-zero probabilities for

$F_1(y) = \frac{1}{M}$, where $0 \leq y \leq M$ and $B_N(y) = 1$ on drift positions $-N \leq y \leq M - N$. Here we have to differ from the original concept [20], which is needed to detect a single non-repetitive watermark within an unknown context.

The likeliest drift associated with the i-th embedded symbol is finally inferred as

$$\hat{x}_i = \arg\max_y \left\{ \Pr(y|\mathcal{H}, \mathbf{r}, \mathbf{w}) \propto F_{i+1}(y) B_i(y) \right\}.$$

The estimates $\hat{x}_0$ and $\hat{x}_N$ are used to refer to the barcode boundaries in the received sequence.

Symbol-wise likelihoods

We can finally perform a symbol-wise decoding as follows: The forward and backward metric does not only provide estimates for the start and the end position of an entire barcode word, but also enables to calculate conditional likelihoods

$$P\left(\mathbf{r}|\mathcal{H}, \mathbf{w}, \mathbf{s}^{(j)}\right) = \sum_{y,z} F_u(y) \pi_u\left(y, z, \mathbf{s}^{(j)}\right) B_v(z)$$

based on inner code words $\mathbf{s}^{(j)} \in \mathcal{C}_2$. With the indexes $u = (j-1)n_2 + 1$ and $v = jn_2$ we denote delimiters of $\mathbf{s}^{(j)}$ (compare Figure 2). The drifts y and z define potential boundaries $u + y$ and $v + z$ of an emitted sub-sequence of $\mathbf{r}$ that is assumed to depend on $\mathbf{s}^{(j)}$. With $\pi_u(y, z, \mathbf{s}^{(j)})$ we symbolize a truncated version of the forward metric, starting at states y and ending at states z. For the evaluation of π we further consider emission probabilities $Q^S\left(l, r_l^{(i)}, w_i \oplus s_i\right)$. As the inner code words are determined by outer code symbols, i.e. $\mathcal{C}_2 : d_j \rightarrow \mathbf{s}^{(j)}$, we can easily derive symbol-wise marginal a posteriori probabilities $P(d_j|\mathcal{H}, \mathbf{r})$ from the conditional likelihoods. The symbol-wise marginals are finally utilized in the outer coder (see Figure 2).

Using this suboptimal inner decoding approach, we are able to decrease the computational costs to $q_1 n_1$ evaluations of the forward metric π (compare section *Optimal decoding*). As we need an additional soft decoding for the outer code, there are further operations needed: For a maximum likelihood approach we have to consider q_1^k code words and search for the likeliest solution.

Simulations

In order to perform an in silico application of barcodes based on the watermark concept, we first have to define some reasonable setting for encoding, which is already quite challenging.

Reasonable encoder settings

As stated before (see section *Barcode construction on watermark*), there are different parameters, which influences the concepts and for which we have to find a reasonable setup to run simulations. First there is an outer codes, which should in combination with the inner coding lead to short barcode words, because we do not want to produce exceptional overhead with multiplexing (tagging) target fragments. There is the minimum distance of outer code words and the sparsity of the inner encoder, which can independently be characterized. And finally we have the watermark sequence, which can randomly be chosen. We utilize the degree of freedom with the watermark to incorporate with additional sequence constraint, that barcodes should fulfill to be experimentally valid. Therefore we run a greedy search for the watermark pattern that maximizes the number of barcodes that meet all sequence constraint. But let us consider all particular setting one by one in the following paragraphs.

Suitable outer codes

For the construction of barcodes we focus on a target-length of $N = n_1 n_2 \in [12, .., 25]$ symbols with $n_1 \in \{3, 4, 5, 6\}$ and $n_2 \in \{4, 5, 6, 7, 8\}$. Further we limit the outer code $\mathcal{C}_1\left[\mathbb{F}_{q_1}, n_1, k_1, \mathrm{d}_H\right]$ to the best known linear codes listed in [28] for several cardinalities of Galois fields $\mathbb{F}_{q_1}$, for which we considered $q_1 \in \{2, 3, 4, 5, 7, 8, 9\}$. Long LDPC codes has been used in the original approach of Davey and MacKay (see [20]), but as the construction of short LDPC codes would be somehow confusing for readers involved in channel coding, we decided to use the best known linear codes. But we might note, that the hamming distance and the ability for soft decoding is the only demand on outer codes here and LDPC codes are likely to perform equivalent. The minimal Hamming distance d_H of the best known linear codes we used is either maximal or the highest known regarding given parameters. Additionally we bound the dimension k_1 to achieve code set sizes $48 < |\mathcal{C}_1| = q_1^{k_1} < 1000$. This guarantees a certain minimal code rate on one side and limit the computational effort of the outer decoding the other side. We end up at 263 possible code configurations, but most of the resulting codes perform disastrous with the watermark concept, because the watermark is heavily corrupted by inner code words.

Sparsity of the inner code

We consider the density (mean Hamming weight) as

$$\overline{\mathrm{w}}_H(\mathcal{C}_2) = \frac{1}{\mathrm{n}(\mathcal{C}_2)|\mathcal{C}_2|} \sum_{\mathbf{s} \in \mathcal{C}_2} \mathrm{w}_H(\mathbf{s})$$

for the inner code, with $\mathrm{w}_H(\cdot)$ as Hamming weight, $\mathrm{n}(\cdot)$ as length and $|\cdot|$ as cardinality of the code. The inner code $\mathcal{C}_2$ needs to exhibit a low density, to keep the watermark shining through the barcodes, when inner code words are added. For large densities there is no ability left to detect the barcode boundaries and consequently decoding will fail completely. Inspired by the approach in [29] we avoid

the all-zeros code word in $\mathcal{C}_2$, but further bound the density to $\overline{\mathrm{w}}_H(\mathcal{C}_2) < 0.3$, to keep the watermark present. Finally, we end up in a set of 73 parameter configurations for q_1, k_1, n_1 and n_2. For each of the 73 different parameter sets we run a brute-force search (10^7 trials), where we iteratively selected one inner code and watermark randomly to produce barcode sets. From the evaluated settings we kept the one, which met the following sequence constraints.

Sequence constraints

We filtered for code words with unbalanced counts of symbols, to respect limitations on the *GC content* of barcodes. Such filtering can be seen as a de facto standard in the construction of barcodes (see for example [1,7,19]) and is related to technical constraints due to the preparation and the sequencing of genomic material. The relative frequency of a subset of two symbols should not be below 40% and above 60% in each barcode, otherwise we excluded the barcode. We furthermore exclude barcodes with *prefect self-complementation* and more than two *sequential repetitions* of the same base (homopolymer length), similar to the restrictions stated in [24]. We consider this settings as sufficient and strict enough to avoid experimental problems during the preparation in real sequencing tasks. Discarding such inappropriate code words means an additional loss in code rate.

Increasing the mean edit distance

For decoding based on the HMM approach, edit distance implicitly matters, thus we try to increase the mean edit distance of code word with a simple strategy. For two sets of barcodes with identical counts of remaining barcodes (after filtering) we keep the one maximizing the mean edit distance

$$\overline{\mathrm{d}}_E(\mathcal{C}) = \frac{1}{\mathrm{n}(\mathcal{C})(\mathrm{n}(\mathcal{C})-1)} \sum_{\mathbf{b}_i,\mathbf{b}_j \in \mathcal{C}: i \neq j} \mathrm{d}_E(\mathbf{b}_1, \mathbf{b}_2),$$

where $\mathrm{d}_E(\mathbf{b}_1, \mathbf{b}_2)$ denotes the edit distance [30] of barcodes $\mathbf{b}_1$ and $\mathbf{b}_2$, which can be understood as the number of single-symbol sequence operations to transform $\mathbf{b}_1$ into $\mathbf{b}_2$ or vice versa. But, as the edit distance is bounded by the hamming distance, we do not gain a lot with this final heuristic refinement step.

Estimating the decoding error

To evaluate the refined set of 73 codes, we give the following demultiplexing scenario. For each code we consider an error curve according the estimates of the decoding error probability on different channel settings. The estimation of a single point in the error curve is based on a set of 200,000 barcodes, which we refer to as batch. Each batch is processed with a certain symbol mutation probability $p_{mut} \in [.01, .16]$. This value determines a symbol-wise probability for an edit operation that can be caused by our channel model. Similar approach has been considered in [19], but we like to be a bit more precise with the description of the modifications of the channel model. Buschmann et al. [19] considered a minimal mutation probability of 10^{-1} for their evaluation, but others claim, that error rates are several orders lower [11]. We will take such indications into account.

Each barcode from a batch is embedded in a random context of variable length (compare section *Embedding of barcodes*). We use normal distributed random variables (with mean $\mu_l = 50$ and variance $\sigma_l = 5$) to determine the length of $\mathbf{t}_{\mathrm{pre}}$ as well as $\mathbf{t}_{\mathrm{post}}$. We simply take the nearest non-negative integer to define the length of the random context, which is uniformly distributed on $\{A, C, G, T\}$.

We further use a state machine to produce erroneous received sequences based on the following four events: correct transmission `C`, substitution `S`, insertion `I` and deletion `D`. In slightly different notation to the former model (see. *Sequencing Channel*) we assign probabilities to the events `C` and `S` and do not use conditional probability like a substitution matrix **S**.

The probability for correct transmission `C` or substitution `S` is equal to p_t in the former representation. The probability for `C` equals $1 - p_{mut}$ and the probability for any of the error events (`S`,`I` or `D`) is p_{mut}. Every error event `S`, `I` or `D` is equal likely. To obtain an equal distribution among the mentioned events, it is easier to use the present notation, but equivalent behavior can be approached with both versions of the channel state-machines.

To save decoding time we iteratively pass each transmit sequence through the state machine, until we have at least one error event within the barcode region. The probability of having a defective code word of length N is $\Pr(\mathtt{def}) = 1 - (1 - p_{mut})^N$. We further considered an error-free barcode to be decoded perfectly, i.e. $\Pr(\mathtt{err}|\overline{\mathtt{def}}) = 0$, with `err` denoting the event of decoding error. Please note that this oversimplifies the false positive rate introduced by sporadic similarities of the context with dedicated barcode words. The rate is supposed to grow linear with the context length and the size of barcodes set. Nevertheless, the probability for false positive events exponentially tend to zeros with the length N of the code words. For barcodes of length ≥ 12 embedded in 100 random symbols we consider this error as marginal offset for the estimated decoding errors. Our evaluation finally end up in estimating the conditional error $\Pr(\mathtt{err}|\mathtt{def})$, which gives an estimate for the unconditioned error probability as $P_e = \Pr(\mathtt{err}|\mathtt{def})\Pr(\mathtt{def})$.

Results and discussion

First we give a rough overview on meaningful properties of watermark-based barcodes with the considered 73 parameterizations. Furthermore, we provide a refined

analysis and evaluation for certain codes in the already defined decoding scenario. To minimize the textual elements in the figures, we use the following notation: $q_1|k_1|n_1|n_2$ indicates the concatenated coding using an outer code $\mathcal{C}_1\left[\mathbb{F}_{q_1}, n_1, k_1\right]$ and an inner code $\mathcal{C}_2 : \mathbb{F}_{q_1} \rightarrow \mathbb{Z}_4^{n_2}$. The particular codes can be found in the Additional file 1 section of this paper.

Properties of barcodes based on watermarks

In Figure 4 we link the principal characteristics as mean edit distance $\overline{\mathrm{d}}_E$, mean density $\overline{\mathrm{w}}_H$ and the cardinality $|\mathcal{C}_2 \circ \mathcal{C}_1|$ of barcodes in a comprehensive illustration. We use star-symbols to indicated the mean density (several levels) and two dimensional coordinates to link mean edit distance and the size of code sets.

There are distinct blocks, where the influence of the inner code can be separately examined. With fixing the outer code, e.g. $q_1|2|3|n_2$ or $q_1|3|4|n_2$ for $q_1 \in \{7, 8, 9\}$ and increasing the length n_2 of inner code words, one can deduce how code rate is exchanged for lowered mean density and increased mean distance. However, we see configurations, for example $4|3|4|n_2$ for $n_2 \in \{4, 5, 6\}$, where we have not been able to increase the mean distance by prolonging the inner code. We have observed several clusters, where similar effects can be found.

For the codes $q_1|k_1|4|n_2$ or $q_1|k_1|5|n_2$ and $q_1|k_1|6|n_2$ the effect of the outer code can be separated. We find clusters of star symbols at different mean distances (see darkened areas in Figure 4). These levels can be explained through the different minimum Hamming distance of the outer codes. We have Hamming distances 2,3 and 4 for the present outer codes at n_1 equals to 4,5 and 6. For concatenated coding it is known that the minimum distances of inner and outer codes are multiplicative [31]. As the edit metric is upper bounded by the Hamming distance, we anticipate the described levels for edit distances. The mentioned leveling can consistently be found for all outer codes.

Despite we have maximized the edit distance of barcodes on average, it is also interesting to focus on the pairwise distance of barcodes. To examine the distance in detail we utilize the so-called distance distribution. In [32] the average number of code words at a certain distance to a fixed code word is considered as an useful distance measure for non-linear codes based on Hamming metric. The edit distance distribution of a codes $\mathcal{C}$ consists of the numbers

$$D_e = \frac{1}{M}\left|\left\{(i,j) : \mathrm{d}_E(\mathbf{b}_i, \mathbf{b}_j) = e, \quad \mathbf{b}_i, \mathbf{b}_j \in \mathcal{C})\right\}\right|,$$

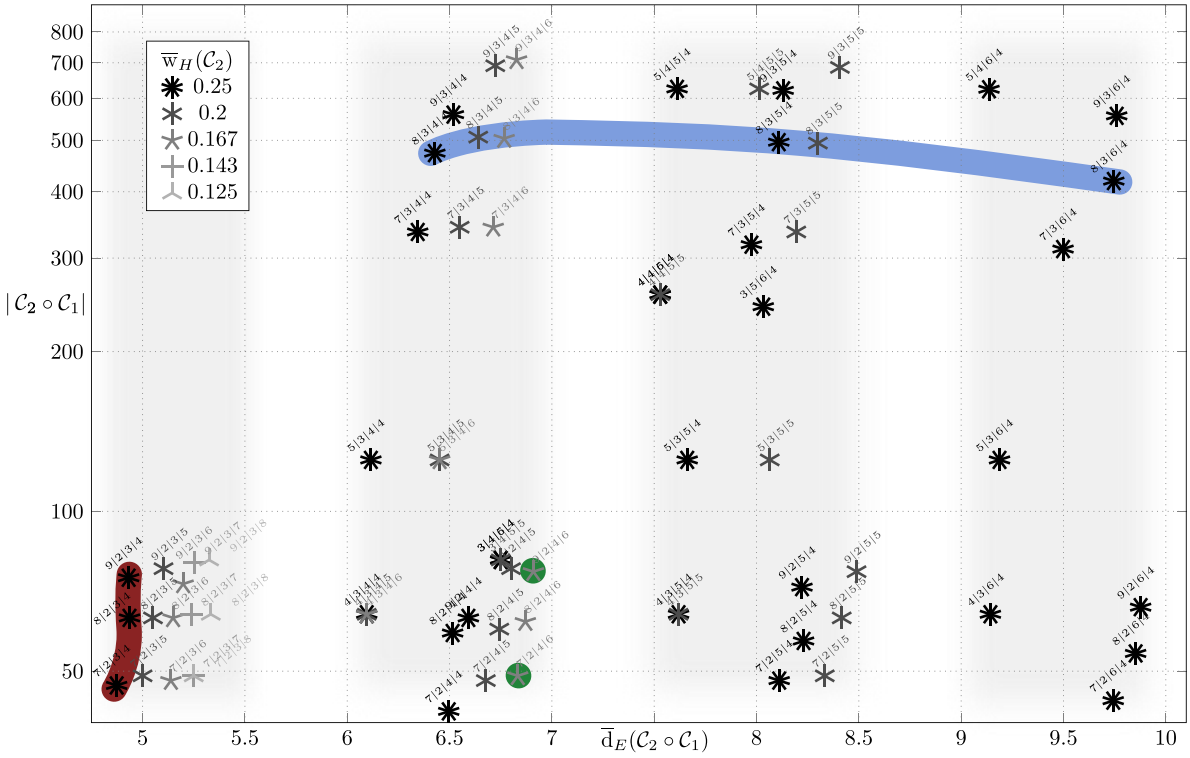

Figure 4 Properties of the examined barcodes based on watermarks. Star-symbols indicate the mean density and two dimensional coordinates link the mean edit distance $\overline{\mathrm{d}}_E$ and the size $|\mathcal{C}_2 \circ \mathcal{C}_1|$ of codes. Parameters of the barcodes are labeled as $q_1|k_1|n_1|n_2$, denoting an outer code $\mathcal{C}_1\left[\mathbb{F}_{q_1}, n_1, k_1\right]$ and inner code $\mathcal{C}_2 : \mathbb{F}_{q_1} \rightarrow \mathbb{Z}_4^{n_2}$. Codes evaluated in refined analysis/simulations (see. Figure 5 and 6) are colored.

where $M = |\mathcal{C}|(|\mathcal{C}| - 1)$ and d_E denotes the edit distance. In Figure 5 we illustrate the distance distribution of barcodes with parameters $8|3|n_1|n_2$ (Figure 4, in blue). Apparently, there are particular pairs of code words with very low edit distances, but as we recall the code construction based on inner code words with very low Hamming weight, this fact is not too surprising. Nevertheless, some longer codes show a negligible amount of such code words with small edit distances. For instance, in the code $8|3|6|4$ less than 1% of all possible pairs of barcodes show an edit distance $\mathrm{d}_E < 6$.

According to the very strict filtering (see *Sequence constraints*), we had to prune out the sets of $q_1^{k_1}$ outer code words, what additional lowers the code rate. A detailed description about the excluded barcodes can be found in the Additional file 1.

Evaluation of decoding

In Figure 6 we illustrate the estimates P_e for the decoding error probability of different codes settings. We ran simulations for all 73 barcode set and ranked the sets according the decoding behavior. To give a rough outline for the performance of our approach we show the barcodes, which performed best (Figure 4 and 6 in green) and worst (Figure 4 and 6 in red). A barcode length of 12 (codes $q_1|k_1|3|4$) seems to be insufficient to provide a good synchronization based on watermarks and thus the majority of decoding errors were found to be caused by synchronization issues (results not shown). The best performing sets of barcodes surprisingly have not occupied the maximal possible length, but 24 symbols. As there are only 49 sequences available, this set of code words provides a poor code rate of $\frac{\log_2(7^2)}{\log_2(4^{24})} = 0.117$.

A reasonable trade-off between error-correcting capability and cardinality is provided for example by codes with parameters $8|3|n_1|n_2$ (Figures 4, 5 and 6, in blue). Although we are facing relative low code rates (compared to approaches like [19]) ranging from 0.188 to 0.281, more than 400 sequences available with quite surprising decoding capability.

Decoding complexity

According to [20] we utilized a fast decoding approach with reduced complexity for our simulations. We bounded the maximal number of possible drift states $\{x_i\}$ in all HMMs to $\Delta \in \{5, .., 20\}$ according to the suggestion given by Davey and Mackay. The complexity for decoding a single embedded barcode is $\mathcal{O}(MLI + q_1 N \Delta I + q_1^{k_1})$, with N denoting the length of the barcode, that is embedded in a received sequence of length M. Decoding is based on the assumption that the channel can introduce maximal I inserted symbols and we consider the maximal drift between received and transmitted sequence to be limited by Δ. An order of MLI calculation are needed to estimate barcode boundaries, $n_2 \Delta I$ operations are needed to provide soft-values for each of the $q_1 n_1$ possible outer code words (result in $q_1 N \Delta I$) and $q_1^{k_1}$ final comparisons has to be spent for soft decoding the outer code in the most expensive case (maximum likelihood).

The prototype decoder with which we ran our simulations is implemented in MATLAB. We further parallelized the decoding procedure and created jobs of 10^6 received sequences, that were processed by single cores (Opteron, 2.6 GHz). The average length of received sequences was in a range of 112 and 150 symbols, resulting in an average processing time of 6 hours for the tasks with lowest calculation costs (code parameters $7|2|3|4$). The longest time we needed to complete demultiplexing of 10^6 sequences (code parameters $9|3|5|5$) was strictly below 24 hours (on a single core).

Future directions

Apart from the theoretical considerations we have given in this manuscript, there are lot of future direction starting from this initial point. Some of them are mandatory to enable an application in real biological experiments,

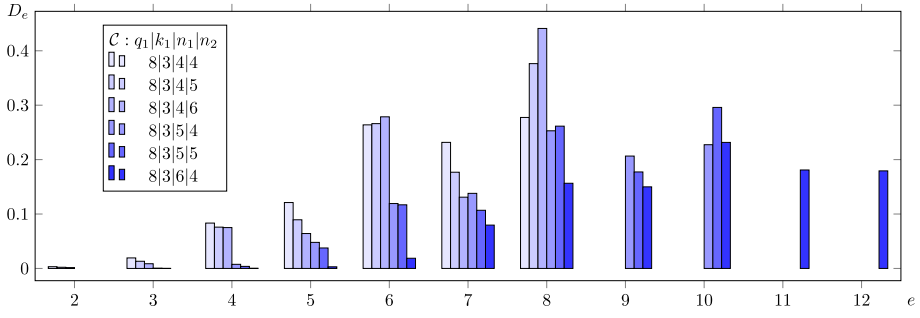

Figure 5 Exemplary distance distribution of a sets of barcode based on watermarks. D_e denotes the relative number of code-pairs with an edit distance equal to e. Each barcode set consists of maximal 512 code words, according to the examined outer codes $\boldsymbol{\mathcal{C}}_1$ $[\mathbb{F}_8, n_1, 3]$ (see also Figure 4, in blue). The distance distribution is normalized regarding the cardinality of code words after filtering.

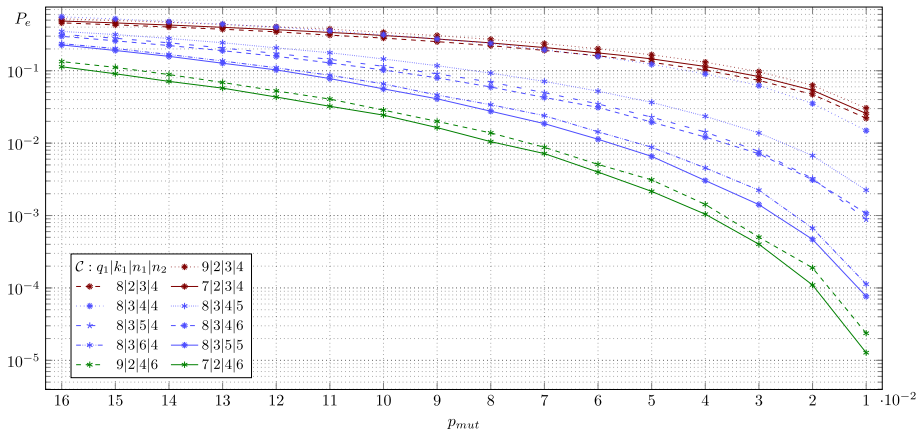

Figure 6 Simulation results for a realistic decoding scenario. Estimated decoding error probability P_e for barcode sets with parameters $q_1|k_1|n_1|n_2$, consisting of an outer codes $\boldsymbol{\mathcal{C}}_1\left[\mathbb{F}_{q_1}, n_1, k_1\right]$ and inner code $\boldsymbol{\mathcal{C}}_2 : \mathbb{F}_{q_1} \rightarrow \mathbb{Z}_4^{n_2}$. All evaluated codes are highlighted in Figure 4 with identical colors. Each barcode set is tested for different symbol mutation probabilities $p_{mut} \in [.01, .16]$. On average, each randomly drawn barcode is embedded in 100 random symbols before decoding.

others are modifications of the concepts for extended applications.

Let us first address the essential steps needed to establish an HMM based decoding in real experiments: The HMM, as core of the decoding system, is the most sensible part of the concept. It is mandatory to run experiments to gather reliable data about all channels, the concepts should be used for. From our point of view there is a lack of reliable data about insertions, deletions and substitution errors for possible channel models. For the sequencing application we assume that different platforms show a variety of *sequencing channels*, additionally affected by experimental parameters, e.g. the extend of PCR pre-processing of samples. To obtain an optimal suited decoder, the HMM should be adapted to the considered channels. As most of the channels show a correlation of errors, more complex HMMs should be considered, reflecting a channel model with memory. Finally, it might be possible to establish a self-adaptive algorithm to parametrize the HMMs without any prior knowledge about the ratio of errors in the channel. A suitable statistic and refined calibration steps should be invented. Another important point for estimating the error characteristics is the *construction of watermark codes*. Exact empirical parameters of the channel could be incorporated in the design of watermark codes to improve decoding steps, suited for special channels.

Further aspects that could be considered with the given concepts are the following: Aside from the synchronization aspect in this manuscript, it seems very promising to use the maximum likelihood decoding method for other sequences than watermark codes. Conditioned on good empirical parameters for an underlying HMM one could consider a reliable detection of barcodes based on the *Sequence-Levenshtein distance* in a probabilistic way rather than based on sequence alignment.

In the presented approach we focused on the discrimination of code words, assuming codes are always present in inspected sequences. The detection of code words within DNA context is another big issue that should be solved for future investigations using an HMM based decoding. Resent research shows that even for sequencing approaches the detection of barcodes is quite challenging. In [24] they focus on a specific problem with certain setups on the PacBio SMRT platform. Caused by technical reasons, sporadically barcodes are not present in the sequence data. Another interesting field of application of an HMM based sequence detection could be clonal studies, where the sequenced genome could or could not contain a predefined sequence, which was introduced in ancestor organisms.

Conclusion

We proposed an adaption of the watermark concept of [20] for DNA barcoding. A generalized channel model for sequencing and suitable modifications of the decoder were defined. Moreover, we investigated in a strategy to choose watermark sequences and inner codes in a reasonable way to enable barcoding in line with common experimental requirements. We provide a code construction, considering the best known linear codes as outer codes and biological sequence constraints to filter for suitable code words, resulting in an exemplary set of 73 different code sets ranging from 12 to 24 nucleotides. The codes are illustrated in a comprehensive scheme, highlighting watermark specific parameters as well as the mean edit distance, to give an impression how watermark based barcodes could be characterized. For a reduced set of codes

we finally evaluated the demultiplexing of sequences in a realistic simulation scenario. Within this in silico evaluation we could show that barcodes based on watermarks can theoretically be used for multiplexing. It is remarkable, that even with very short watermark patterns we are able to reliably find the barcodes boundaries in order to discriminate different code words with an HMM based decoder. The probability of decoding errors, which finally leads to the undesired cross-talk phenomenon was found to be very low. Other approaches that investigate barcodes with large (sequence) edit distance [16,19] show significant higher code rates for shorter barcodes, but we have given an entirely different concept that allows for large scale multiplexing approaches, also able to handle insertion and deletion errors.

Moreover, we can provide the marker-less synchronization based on watermarks, to recover the barcode boundaries. This synchronization concept provides an ultimate degree of freedom for experimental sequencing setups as well as for future applications, also apart from the sequencing context.

Additional file

Additional file 1: We provide 73 files containing all the examined barcodes. Further informations are given, like: The watermark in decimal and nucleic acid notation, as well as the mapping of the inner encoder and the used mapping from integers to nucleic acid $\mathbb{Z}_4 \rightarrow \{A, G, C, T\}$ are given in a preamble. We furthermore give the distance distribution of barcode words. Finally, all barcodes are listed (plus explicit construction via code concatenation). Dropped barcodes, which do not meet the sequence constraints are explicitly indicated.

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
ST initiated to work on a generalized channel model for the concepts in [20] and inspired an adaption for DNA barcoding. DK accomplished the main developments on general methods, the decoder and the simulation framework. DK was also assisted by the former master student Mahmoud Almarashli, who submitted a remarkable thesis. DK and ST wrote the manuscript. Both authors read and approved the final manuscript.

Acknowledgements
David Kracht is funded by the Deutsche Forschungsgemeinschaft (DFG) under the grant BO 867/30-1 in the priority program SPP 1395. Steffen Schober is supported by the DFG grant SCHO 1576/1.
The authors like to thank Mahmoud Almarashli for fruitful discussions and assistance in preliminary considerations and managing preparative simulations.

Received: 7 May 2014 Accepted: 29 January 2015
Published online: 18 February 2015

References
1. Hamady M, Walker JJ, Harris JK, Gold NJ, Knight R. Error-correcting barcoded primers for pyrosequencing hundreds of samples in multiplex. Nat Methods. 2008;5(3):235–7.
2. Bystrykh LV. Generalized dna barcode design based on hamming codes. PLoS One. 2012;7(5):36852.
3. Hamming RW. Error detecting and error correcting codes. Bell Syst Tech J. 1950;29:147–60.
4. Krishnan AR, Sweeney M, Vasic J, Galbraith DW, Vasic B. Barcodes for dna sequencing with guaranteed error correction capability. Electron Lett. 2011;47(4):236–7.
5. Lin S, Costello DJ. Error control coding, vol. 123. Englewood Cliffs, New Jersey: Prentice-hall; 2004.
6. Frank DN. Barcrawl and bartab: software tools for the design and implementation of barcoded primers for highly multiplexed dna sequencing. BMC Bioinformatics. 2009;10(1):362.
7. Mir K, Neuhaus K, Bossert M, Schober S. Short barcodes for next generation sequencing. PLoS One. 2013;8(12):82933.
8. Gilles A, Meglécz E, Pech N, Ferreira S, Malausa T, Martin JF. Accuracy and quality assessment of 454 gs-flx titanium pyrosequencing. Bmc Genomics. 2011;12(1):245.
9. Carneiro MO, Russ C, Ross MG, Gabriel SB, Nusbaum C, DePristo MA. Pacific biosciences sequencing technology for genotyping and variation discovery in human data. BMC Genomics. 2012;13(1):375.
10. Bragg LM, Stone G, Butler MK, Hugenholtz P, Tyson GW. Shining a light on dark sequencing: characterising errors in ion torrent pgm data. PLoS Comput Biol. 2013;9(4):1003031.
11. Loman NJ, Misra RV, Dallman TJ, Constantinidou C, Gharbia SE, Wain J, et al. Performance comparison of benchtop high-throughput sequencing platforms. Nat Biotechnol. 2012;30(5):434–9.
12. Shendure J, Ji H. Next-generation dna sequencing. Nat Biotechnol. 2008;26(10):1135–45.
13. Yang X, Chockalingam SP, Aluru S. A survey of error-correction methods for next-generation sequencing. Brief Bioinform. 2013;14(1):56–66.
14. Adey A, Morrison HG, Xun X, Kitzman JO, Turner EH, Stackhouse B, et al. Rapid, low-input, low-bias construction of shotgun fragment libraries by high-density in vitro transposition. Genome Biol. 2010;11(12):119.
15. Qiu F, Guo L, Wen TJ, Liu F, Ashlock DA, Schnable PS. Dna sequence-based "bar codes" for tracking the origins of expressed sequence tags from a maize cdna library constructed using multiple mrna sources. Plant Physiol. 2003;133(2):475–81.
16. Faircloth BC, Glenn TC. Not all sequence tags are created equal: designing and validating sequence identification tags robust to indels. PLoS One. 2012;7(8):42543.
17. Ashlock D, Guo L, Qiu F. Greedy closure evolutionary algorithms. In: Computational intelligence, proceedings of the world on congress on, vol. 2. Piscataway: IEEE; 2002. p. 1296–301.
18. Ashlock D, Houghten SK. A novel variation operator for more rapid evolution of dna error correcting codes. In: Computational intelligence in Bioinformatics and computational biology, 2005. CIBCB'05. Proceedings of the 2005 IEEE symposium on. Piscataway: IEEE; 2005. p. 1–8.
19. Buschmann T, Bystrykh LV. Levenshtein error-correcting barcodes for multiplexed dna sequencing. BMC Bioinformatics. 2013;14(1):272–73.
20. Davey MC, Mackay DJC. Reliable communication over channels with insertions, deletions, and substitutions. Inf Theory IEEE Trans. 2001;47(2):687–98.
21. Haughton D, Balado F. Biocode: Two biologically compatible algorithms for embedding data in non-coding and coding regions of dna. BMC Bioinformatics. 2013;14(1):121.
22. Haughton D, Balado F. A modified watermark synchronisation code for robust embedding of data in dna. In: Acoustics, speech and signal processing (ICASSP), 2013 IEEE international conference on. Piscataway: IEEE; 2013. p. 1148–52.
23. Kracht D, Schober S. Using the davey-mackay code construction for barcodes in dna sequencing. In: Turbo codes and iterative information processing (ISTC), 2014 8th international symposium on. Piscataway: IEEE; 2014. p. 142–6.
24. Buschmann T, Zhang R, Brash DE, Bystrykh LV. Enhancing the detection of barcoded reads in high throughput dna sequencing data by controlling the false discovery rate. BMC Bioinformatics. 2014;15(1):264.
25. Jukes TH, Cantor CR. Evolution of protein moleculese. Mamm Protein Metab. 1969;3:21–132.
26. Minoche AE, Dohm JC, Himmelbauer H. Evaluation of genomic high-throughput sequencing data generated on illumina hiseq and genome analyzer systems. Genome Biol. 2011;12(11):112.
27. Rabiner L, Juang BH. An introduction to hidden markov models. ASSP Mag IEEE. 1986;3(1):4–16.

28. Grassl M. Searching for linear codes with large minimum distance In: Bosma W, Cannon J, editors. Discovering mathematics with magma — reducing the abstract to the concrete. Algorithms and computation in mathematics, vol. 19. Heidelberg: Springer; 2006. p. 287–313.
29. Briffa JA, Schaathun HG. Improvement of the davey-mackay construction. In: Information theory and its applications, 2008. ISITA 2008. international symposium on. Piscataway: IEEE; 2008. p. 1–4.
30. Levenshtein VI. Binary codes capable of correcting deletions, insertions and reversals. Soviet Phys Doklady. 1966;10(8):707–10.
31. Forney GD. Concatenated codes, vol. 11. Cambridge: MIT Press; 1966.
32. MacWilliams FJ, Sloane NJA. The theory of error-correcting codes, vol. 16. Amsterdam, Netherlands: Elsevier; 1977.