Weighted Sequencing from Compomers: DNA de-novo sequencing from mass spectrometry data in the presence of false negative peaks

Sebastian Böcker

AG Genominformatik, Technische Fakultät
Universität Bielefeld
PF 100 131
D-33501 Bielefeld
boecker@CeBiTec.uni-bielefeld.de

Abstract: One of the main endeavors in today's Life Science remains the efficient sequencing of long DNA molecules. Today, most de-novo sequencing of DNA is still performed using electrophoresis-based Sanger Sequencing introduced in 1977, in spite of certain restrictions of this method. Recently, we proposed a new method for DNA sequencing using base-specific cleavage and mass spectrometry, that appears to be a promising alternative to classical DNA sequencing approaches: Among its benefits is the extremely fast data acquisition of mass spectrometry. This leads to the combinatorial problem of Sequencing From Compomers (SFC), and to the definition of sequencing graphs. Simulations indicate that this method may allow for de-novo sequencing of DNA molecules with 200+ nt.

An open problem in the context of SFC is that it does not take into account *false negative peaks* (missing peaks) that are common for real-world mass spectra. Here, we present a natural generalization of SFC, the Weighted Sequencing from Compomers (WSC) Problem, that allows us to cope with false negative peaks. We also show that the family of graphs introduced to solve SFC, can be generalized to capture the new aspects of WSC. Finally, we present a branch-and-bound algorithm to find all sequences that agree with the sample mass spectra with the exception of some missing peaks.

1 Introduction

Today, most de-novo sequencing of DNA without any *a priori* information regarding the sample sequence under examination, is still performed based on the Sanger concept from 1977, see [SNC77]. Typically, gel or capillary electrophoresis is used to acquire the sample data. Many other methods were proposed during the last decades [FCK02], but none was able to compete with Sanger Sequencing regarding sequencing length, cost, and reliability. It shall be understood that despite the dominance of Sanger Sequencing, this technique — just like any other sequencing technique — has certain shortcomings, such as: base-calling errors, heterozygous samples, or the time consuming data acquisition by electrophoresis, to name just a few (see for instance [APC+00]).

In [Bö03, Bö04] we propose a new approach to DNA de-novo sequencing not based on the Sanger concept, using MALDI-TOF mass spectrometry to acquire the experimental data. It has the advantages of fast data acquisition (about 4 seconds per sequence) and reliability, among others. Furthermore, we introduce the Sequencing From Compomers (SFC) Problem as an abstraction of the resulting data analysis issues. Simulations indicate that this method may enable de-novo sequencing of DNA molecules with 200+ nt, so sequencing lengths have the same order of magnitude as for Sanger Sequencing.

An open problem in the context of SFC is how to cope with false negative peaks in the mass spectra: A *false negative peak* (or *missing peak*) is a peak that an *in silico* simulation predicts to be present in a mass spectrum — assuming "error-free" biochemistry and mass spectrometry — but that cannot be detected in the measured mass spectrum. Unfortunately, a *single* false negative peak is usually sufficient to prohibit reconstruction of the correct DNA sequence by SFC.

In this paper, we extend the Sequencing From Compomer Problem to deal with false negative peaks in the sample mass spectrum: We introduce the Weighted Sequencing from Compomers (WSC) Problem and weighted sequencing graphs, and show how the latter can be used to solve WSC.

2 Experimental setup and data acquisition

Suppose that we are given an amplified, single stranded target DNA molecule (or *sample DNA*) of length 100–500 nt.[1] We cleave the sample sequence with a base-specific chemical or biochemical cleavage reaction: Such reactions cleave at exactly those positions where a specific base can be found. Several methods to achieve base-specific cleavage such as RNAse A, have been described in the literature [RDPS$^+$02, vBS$^+$02]. We modify the cleavage reaction by offering a mixture of cleavable versus non-cleavable "cut bases," such that not all cut bases but only a certain percentage will be cleaved. The resulting mixture contains in principle all fragments that can be obtained from the sample DNA by removing two cut bases, cf. Fig. 1 for an example. We call such cleavage reactions *partial*.

MALDI TOF mass spectrometry (MS for short) is then applied to the products of the cleavage reaction, resulting in a sample spectrum that correlates mass and signal intensity of sample particles [KH88]. The sample spectrum is analyzed to extract a list of signal peaks with masses and intensities. We repeat the above procedure, as well as the following analysis steps, using cleavage reactions specific to each of the four bases. For examples of experimental mass spectrometry data of base specific cleavage, we refer the reader to the literature, for instance [HSB$^+$03].

If the sample sequence is known, then exact chemical results of the employed cleavage reactions and, in particular, the masses of all resulting fragments are known in advance, and the subsequent mass spectrometry measurement can be simulated *in silico*. Clearly, this holds up to a certain extent only, see below.

[1] We will talk about sample DNA even though a cleavage reaction might force us to transcribe the sample to RNA.

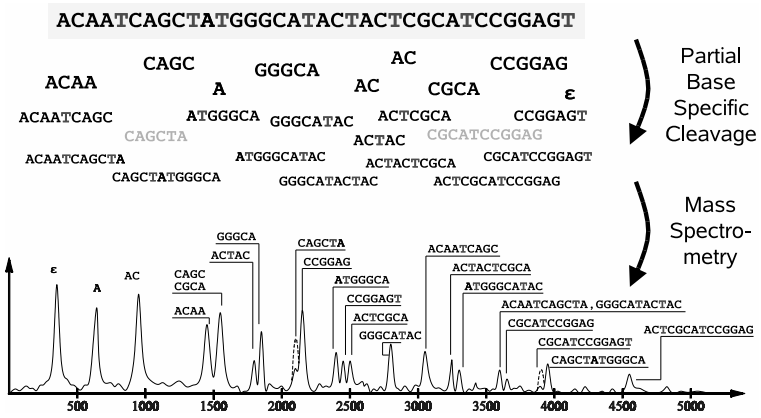

Figure 1: Partial cleavage using RNAse A with dCTP, rUTP, and dTTP. Gray fragments indicate that corresponding peaks cannot be detected in the sample mass spectrum. See text for details.

Having said that, we can also solve the inverse problem: For every peak detected in the sample mass spectrum, we can compute one or more base compositions (that is, DNA molecules with unknown order but known multiplicity of bases) that could have created the detected peak, taking into account the inaccuracy of the mass spectrometry read. Therefore, we obtain a list of base compositions and their intensities, for every incorporated cleavage method.

In real life, several limitations characteristic for mass spectrometry and partial cleavage make the problem of de-novo sequencing from mass spectrometry data more challenging, see [Bö03] for details. In particular, using partial cleavage results in an *exponential decay* (in the number of uncleaved cut bases) of signal intensities in the mass spectrum, so peaks from fragments containing many uncleaved cut bases will be difficult or impossible to detect.

This leads us to the following unexpected situation: In the setting of the classical Partial Digestion Problem, one uses restriction enzymes and incomplete cleavage in a way such that *long* fragments that contain many uncleaved restriction sites are *likely* to be detected, while inner fragments are more likely to be lost. In contrast, incorporating a mixture of cleavable and uncleavable cut bases produces many copies of fragments containing no uncleaved cut base and hence, intense peaks in the mass spectrum. But for fragments containing one, two, or more uncleaved cut bases, peak intensities decrease rapidly.

Second, peak intensities vary strongly and are comparatively hard to predict. Potentially, the intensity of a peak in a sample mass spectrum is so weak that this peak cannot be detected in the "noise" of the mass spectrum. A sensitive peak detection algorithm can reduce the number of missing peaks, but it cannot completely eliminate them in all cases.

We want to stress that there exists no overlap between our approach, and de-novo sequencing of peptides using Tandem Mass Spectrometry (MS/MS): There, the sample peptide is unspecifically fragmented at any position, so that all prefixes and suffixes of the sample string are present in the mass spectrum. Put simply, one has to assign every peak in the

mass spectrum to either a prefix or a suffix of the unknown string, and this can be efficiently done using Dynamic Programming [CKT$^+$01].

3 Methods

Mostly we will follow the notation of [Bö03] and refer the reader there for a more detailed discussion.

3.1 The compomer spectrum

Let $s = s_1 \dots s_n$ be a string over the alphabet Σ where $|s| = n$ denotes the *length* of s. We denote the maximal number of non-overlapping occurrences of a string x in s by $\mathrm{ord}_x(s)$.

For a string $s \in \Sigma^*$ and $x \in \Sigma$, we define the *string spectrum* $\mathcal{S}(s, x)$ of s, x by:

$$\mathcal{S}(s, x) := \{y \in \Sigma^* \,:\, xyx \text{ is a substring of } xsx\} \tag{1}$$

Thus, the string spectrum $\mathcal{S}(s, x)$ consists of those substrings of s that are bounded by x or by the ends of s. In this context, we call s *sample string* and x *cut character*, while the elements $y \in \mathcal{S}(s, x)$ will be called *fragments* of s (under x).

We use special characters **0**, **1** to uniquely denote start and end of the sample string. For an alphabet Σ we consider the set of all strings in Σ^* with attached prefix **0** and suffix **1**, $\mathbf{0}\Sigma^*\mathbf{1} := \{\mathbf{0}s\mathbf{1} \,:\, s \in \Sigma^*\}$.

We use the following mathematical representation of base compositions: We define a *natural compomer* (or *compomer* for short) to be a map $c : \Sigma \to \mathbb{N}$, where $\mathbb{N}$ denotes the set of natural numbers *including* 0. Let $\mathcal{C}_+(\Sigma)$ be the set of all natural compomers over the alphabet Σ. We denote the canonical partial order on $\mathcal{C}_+(\Sigma)$ by $\preceq$, that is, $c \preceq c'$ if and only if $c(\sigma) \leq c'(\sigma)$ for all $\sigma \in \Sigma$. We write 0 for the *empty compomer* $c \equiv 0$.

For $\Sigma = \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$ we use the notation $c = \mathrm{A}_i\mathrm{C}_j\mathrm{G}_k\mathrm{T}_l$ to represent the compomer $c(\mathrm{A}) = i, \dots, c(\mathrm{T}) = l$, omitting those characters with index zero. The function $\mathrm{comp} : \Sigma^* \to \mathcal{C}_+(\Sigma)$ maps a string $s \in \Sigma^*$ to the compomer of s by counting the number of characters of each type in s. For example, set $c := \mathrm{comp}(\mathrm{ACCTA})$ then $c(\mathrm{A}) = 2$, $c(\mathrm{C}) = 2$, $c(\mathrm{G}) = 0$, and $c(\mathrm{T}) = 1$ or, equivalently, $c = \mathrm{A}_2\mathrm{C}_2\mathrm{T}_1$. Compomers $\mathrm{comp}(\cdot)$ are also referred to as frequency vectors or Parikh-vectors.

Recall that due to the experimental setup, signals from fragments y with $\mathrm{ord}_x(y)$ above a certain threshold will be lost in the noise of the mass spectrum. Hence, for $s \in \Sigma^*, x \in \Sigma$, and $k \in \mathbb{N} \cup \{\infty\}$ we define the k-string spectrum of s (under x) by:

$$\mathcal{S}_k(s, x) := \{y \in \mathcal{S}(s, x) \,:\, \mathrm{ord}_x(y) \leq k\} \tag{2}$$

The integer k is called the *order* of the string spectrum. The *k-compomer spectrum* $\mathcal{C}_k(s, x)$ of s consists of the compomers of all fragments in the k-string spectrum $\mathcal{S}_k(s, x)$:

$$\mathcal{C}_k(s, x) := \mathrm{comp}\left(\mathcal{S}_k(s, x)\right) = \left\{\mathrm{comp}(y) \,:\, y \in \mathcal{S}(s, x),\ \ \mathrm{ord}_x(y) \leq k\right\} \tag{3}$$

In [Bö03] we define the Sequencing From Compomers (SFC) Problem to find all strings $s \in S$ that satisfy $\mathcal{C}_k(s,x) \subseteq \mathcal{C}_x$ for all $x \in \Sigma$. Here, $\mathcal{C}_x$ denotes the set of compomers corresponding to the measured mass spectrum with cleaved base x. The inclusion condition reflects the presence of additional peaks in the mass spectrum, as well as misinterpreted peaks due to measurement inaccuracies of the mass spectrometry data [Bö03]. Clearly, this formulation does not capture the problem of false negative peaks: The set of "measured" compomers $\mathcal{C}_x \subseteq \mathcal{C}_+(\Sigma)$ might be missing a compomer that corresponds to a false negative peak or, formally: the set $\mathcal{C}_k(s,x) \setminus \mathcal{C}_x$ is non-empty. Then the correct sample string is not a solution of this instance of SFC.

3.2 Weighted compomers

Let us concentrate on a *fixed* sample mass spectrum corresponding to cleaved base x: We want to penalize our method for assuming peaks that cannot be found in the sample mass spectrum. To this end, we define a *characteristic compomer weight* (CCW) as a function $w_x : \mathcal{C}_+(\Sigma) \to \mathbb{R}_{\geq 0}$. In its simplest incorporation, we set $w_x(c) := 0$ if the peak corresponding to compomer c can be found in the sample mass spectrum, and $w_x(c) := 1$ otherwise: With this w_x we can count missing peaks. Note that w_x is the characteristic function of the set $\mathcal{C}_x \subseteq \mathcal{C}_+(\Sigma)$ of observed compomers [Bö03]. This w_x is called *trivial* characteristic compomer weight in the following.

In general, w_x may also consider the "chances" that some peak is missing in any such measured mass spectrum, as well as peak intensities and peak masses in the sample mass spectrum.

A straightforward way to define a "false negative peak penalty" for a sample string candidate s, is to sum up the weights $w_x(c)$ of all compomers $c \in \mathcal{C}_k(s,x)$. For the trivial CCW, this is exactly the cardinality of $\mathcal{C}_k(s,x) \setminus \mathcal{C}_x$ and, hence, we count missing peaks. Unfortunately, this does not capture the multiplicity of compomers in the compomer spectrum $\mathcal{C}_k(s,x)$: One string s might "generate" some compomer $c \in \mathcal{C}_k(s,x) \setminus \mathcal{C}_x$ from only one fragment y with $\mathrm{comp}(y) = c$, while another generates this compomer from multiple fragments. As intensities in a mass spectrum are additive, the second case is less likely to happen by chance than the first.

To this end, we define the *multiplicity* of some compomer $c \in \mathcal{C}_+(\Sigma)$ with respect to $s \in \Sigma^*$ and $x \in \Sigma$ by

$$\mathrm{mult}_{s,x}(c) := \left| \left\{ (a,y,b) \in (\Sigma^*)^3 \ : \ c = \mathrm{comp}(y) \text{ and } xsx = axyxb \right\} \right| \tag{4}$$

Informally, $\mathrm{mult}_{s,x}(c)$ counts the multiplicity of fragments y in $\mathcal{S}(s,x)$ such that $c = \mathrm{comp}(y)$ holds. So, $\mathrm{mult}_{s,x}(c) \geq |\{y \in \mathcal{S}(s,x) \ : \ \mathrm{comp}(y) = c\}|$ must hold.

This enables us to define a sensible "false negative peak penalty" $w_{k,x}$:

$$w_{k,x}(s) := \sum_{c \in \mathcal{C}_k(s,x)} \mathrm{mult}_{s,x}(c) \cdot w_x(c) \tag{5}$$

We use (5) to establish a weighted version of SFC that takes into account false negative peaks. We do not need the compomer sets $\mathcal{C}_x$ for this, because their "information" is included in the characteristic compomer weights.

Weighted Sequencing from Compomers (WSC) Problem. Let $k \in \mathbb{N} \cup \{\infty\}$ be the fixed spectrum order. For all $x \in \Sigma$, let $w_x : \mathcal{C}_+(\Sigma) \to \mathbb{R}_{\geq 0}$ be the characteristic compomer weight for cut character x. Finally, let $S \subseteq \mathbf{0}\Sigma^*\mathbf{1}$ be the set of sample string candidates. Now, find all strings $s \in S$ minimizing

$$\varphi(s) := \sum_{x \in \Sigma} w_{k,x}(s) \tag{6}$$

where $w_{k,x}$ is defined in (5).

It is clear that SFC can be seen as a special case of WSC: For an instance of SFC, we use the corresponding trivial CCWs w_x for $x \in \Sigma$. Then, a string $s \in S$ is a solution of SFC if and only if it is a solution of WSC with zero weight. So, the WSC decision problem is at least as hard as SFC, which is NP-hard [Bö04].

3.3 The de Bruijn graph

A *directed graph* consists of a set V of vertices and a set $E \subseteq V^2 = V \times V$ of edges. An edge (v, v) for $v \in V$ is called a *loop*. We limit our attention to finite directed graphs with finite vertex sets. A *walk* in G is a finite sequence $p = (p_0, p_1, \ldots, p_n)$ of elements from V with $(p_{i-1}, p_i) \in E$ for all $i = 1, \ldots, n$, and $|p| := n$ denotes the *length* of p. An *edge weighting* of a directed graph with edge set E is a function $\tilde{w} : E \to \mathbb{R}$; in the following, we concentrate on edge weightings such that $\tilde{w}(e) \geq 0$ holds for all edges $e \in E$.

For an alphabet Σ and an spectrum order $k \geq 1$, the *de Bruijn graph* $B_k(\Sigma)$ is a directed graph with vertex set $V_k := \Sigma^k$ and edge set

$$E_k := \big\{(u, v) \in V_k^2 \,:\, u_{j+1} = v_j \text{ for all } j = 1, \ldots, k-1\big\}$$

where $u = (u_1, \ldots, u_k)$ and $v = (v_1, \ldots, v_k)$. We use the vector notation $v = (v_1, \ldots, v_k)$ instead of the string notation $v = v_1 \ldots v_k$ for the sake of lucidity. We denote an edge $\big((e_1, \ldots, e_k), (e_2, \ldots, e_{k+1})\big)$ of $B_k(\Sigma)$ by $(e_1, \ldots, e_{k+1})$ for short.

For a cut character $x \in \Sigma$, a *compomer alphabet* over (Σ, x) is a subset

$$\Sigma_x \subseteq \{c \in \mathcal{C}_+(\Sigma) \,:\, c(x) = 0\} \cup \{*\} \tag{7}$$

where $* \in \Sigma_x$ denotes a special source character we require to be an element of every compomer alphabet. Note that we can *add* compomer characters $c, c' \in \Sigma_x$: For the source character $* \in \Sigma_x$, we formally define $c + * = * + c = *$ for every compomer c.

The edges of the de Bruijn graph $B_k\big(\Sigma_x \setminus \{*\}\big)$ are $(k+1)$-tuples of compomers over the alphabet Σ. We use the notation

$$e_{[i,j]} := e_i + \mathrm{comp}(x) + e_{i+1} + \mathrm{comp}(x) + \cdots + e_{j-1} + \mathrm{comp}(x) + e_j \tag{8}$$

for $1 \leq i \leq j \leq k+1$ to denote the compomer corresponding to parts of an edge $e = (e_1, \ldots, e_{k+1})$ of $B_k(\Sigma_x)$, if the reference to the cut character x is clear. Now, $e_{[i,j]} = *$ holds if and only if there exists an index $i' \in [i,j]$ such that $e_{i'} = *$. Otherwise, we have $e_{[i,j]}(x) = j - i$.

For sample string $s \in \Sigma^*$ and cut character $x \in \Sigma$, we call strings $s_0, \ldots, s_l \in \Sigma^*$ satisfying $s = s_0 x s_1 x s_2 x \ldots x s_l$ and $\mathrm{ord}_x(s_j) = 0$ for all $j = 0, \ldots, l$ an *x-partitioning* of s. Clearly, there exists exactly one x-partitioning of s.

Let Σ be an alphabet, $x \in \Sigma$ a cut character, and Σ_x a compomer alphabet over (Σ, x). A string $s \in \Sigma^*$ is called *compatible* with a walk $p = p_0 \ldots p_{|p|}$ in the de Bruijn graph $B_k(\Sigma_x)$ if the x-partitioning $s_0, \ldots, s_l \in \Sigma^*$ of s satisfies $l = |p|$ and

$$p_j = \big(c_{j-k+1}, c_{j-k+2}, \ldots, c_j\big) \quad \text{for } j = 0, \ldots, l\,, \tag{9}$$

where $c_j := \mathrm{comp}(s_j)$ for $j = 0, \ldots, l$, and $c_{-j} := *$ for all integers $j > 0$. We have modified the definition of compatibility from [Bö03] to take into account the source character $*$.

3.4 Weighted sequencing graphs

We generalize the concept of directed sequencing graphs [Bö03] to take into account compomer weights of false negative peaks. For a cut character x, a characteristic compomer weight $w_x : \mathcal{C}_+(\Sigma) \to \mathbb{R}$, and a compomer alphabet $\Sigma_x \subseteq \{c \in \mathcal{C}_+(\Sigma) : c(x) = 0\} \cup \{*\}$, we define the *weighted sequencing graph* $G_k(x, \Sigma_x; w_x)$ of order $k \geq 1$ as follows: This is an edge-weighted directed graph, consisting of the de Bruijn graph $B_k(\Sigma_x) = (V_k, E_k)$ of order k, together with an edge weighting $\tilde{w}_x : E_k \to \mathbb{R}$ defined by

$$\tilde{w}_x(e_1, \ldots, e_{k+1}) := \sum_{i=1}^{k+1} w_x\left(e_{[i,k+1]}\right) \tag{10}$$

where we assume $w_x(*) = 0$ here and in the following.

Given a walk $p = (p_0, \ldots, p_l)$ in a directed graph G with edge weighting $\tilde{w}_x$, we define the *weight* of p as the sum of weights of its edges: $\tilde{w}_x(p) := \sum_{j=1}^{l} \tilde{w}_x\big((p_{j-1}, p_j)\big)$.

The following theorem is the main result of this paper, and it allows us to tackle WSC by "walking" weighted sequencing graphs. We omit the proof for the sake of brevity.

Theorem 1. *Let $s \in \Sigma^*$ be a string, $x \in \Sigma$ a cut character, and $w_x : \mathcal{C}_+(\Sigma) \to \mathbb{R}$ a characteristic compomer weight. Suppose we are given a walk p in the weighted sequencing graph $G_k(x, \Sigma_x; w_x)$ where Σ_x is a compomer alphabet over (Σ, x). If s and p are compatible, then*

$$w_k(s, x) = \tilde{w}_x(p) \tag{11}$$

holds, where $w_k(s, x)$ is defined in (5) *and $\tilde{w}_x$ denotes the edge weighting of $G_k(x, \Sigma_x; w_x)$.*

4 Algorithm

The algorithm presented in this section generalizes that of [Bö03]. We suppose that we know a compomer alphabet Σ_x such that $\mathcal{C}_0(s,x) \subseteq \Sigma_x$ holds for the correct sample string s. We are given characteristic compomer weights $w_x : \mathcal{C}_+(\Sigma) \to \mathbb{R}_{\geq 0}$ for $x \in \Sigma$ that were generated from sample mass spectra. We want to solve the Weighted Sequencing from Compomers Problem in the form that we search for all strings $s \in S$ such that $\varphi(s)$ is minimal. We concentrate on the case that the set of string candidates $S \subseteq \mathbf{0}\Sigma^*\mathbf{1}$ contains all strings of length in a given interval, which is especially relevant for applications: that is, $l_{\text{min}} \leq |s| \leq l_{\text{max}}$ holds for all $s \in S$.

To solve WSC, we present a depth-first search that backtracks through sequence space, moving along the edges of the sequencing graphs in parallel. In this way, we implicitly build walks in the weighted sequencing graphs of order k that are compatible with the constructed strings. By Theorem 1, every such string s has the same weight $\varphi(s)$ as the sum of weights of the compatible walks. This allows us to do a branch-and-bound check by stopping the recursion as soon as the resulting string has weight above the threshold, because all edge weights are non-negative.

First, we have to build the sequencing graphs $G_x := G_k(x, \Sigma_x; w_x)$ for $x \in \Sigma$. This means that for every edge e of the de Bruijn graph $B_k(\Sigma_x)$, we have to calculate and store the edge weight $\tilde{w}_x(e)$. A fast method of generating $G_k(x, \Sigma_x; w_x)$ is to iteratively build the graphs $G_\kappa(x, \Sigma_x; w_x)$ for $\kappa = 1, \ldots, k$. This can be done in $O\big(|\Sigma_x|^{k+1}\big)$ time for $|\Sigma_x| \geq 2$.

For the depth-first search, we make use of the following notations: s is the current string that will be a prefix of all string candidates constructed in subsequent recursion steps. $\psi \in \mathbb{R}_{\geq 0}$ denotes the weight of the current prefix string s, and $\psi_{\min} \in \mathbb{R}_{\geq 0} \cup \{\infty\}$ denotes the weight of the best solution found so far. Clearly, $\psi_{\min} \geq \varphi_{\min}$ always holds. As we want to construct only strings s satisfying $\varphi(s) \leq \varphi_{\min}$, we can stop the recursion as soon as ψ is too large. Let h_x denote the weight change that is added to ψ if we append the character $x \in \Sigma \cup \{\mathbf{1}\}$ to s. For $x \neq \mathbf{1}$, h_x equals the weight of some edge in G_x. Next, $\tilde{h}_x \geq h_x$ denotes the induced weight change if we append the character $x \in \Sigma$: Appending x will force edge transitions in G_σ for $\sigma \neq x$ in subsequent recursion steps. Finally, v_x denotes the active vertex in G_x.

Now, we start the recursion with $s \leftarrow \mathbf{0}$, $\psi \leftarrow 0$, and $\psi_{\min} \leftarrow \infty$. We initialize the current vertices $v_x \leftarrow (*, \ldots, *)$ for all $x \in \Sigma$.

The *recursion step* takes as input: the current prefix string s, its weight ψ, the best solution weight $\psi_{\min}$, and the current active vertices v_x for $x \in \Sigma$. Let s_x be the unique string satisfying $\text{ord}_x(s_x) = 0$ such that either xs_x is a suffix of s, or $s_x = s$ if $\text{ord}_x(s) = 0$. Set $c_x := \text{comp}(s_x)$.

- If $|s| + 1 \geq l_{\text{min}}$ then calculate $h_{\mathbf{1}}$. If $\psi + h_{\mathbf{1}} \leq \psi_{\min}$ then **output** $s\mathbf{1}$ with weight $(\psi + h_{\mathbf{1}})$, and set $\psi_{\min} \leftarrow \psi + h_{\mathbf{1}}$.
- If $|s| < l_{\text{max}}$, then calculate h_x and $\tilde{h}_x$ for all $x \in \Sigma$. For every character x satisfying

$\psi + \tilde{h}_x \leq \psi_{\min}$ do a recursion step: Replace s by sx; replace ψ by $\psi + h_x$; and in G_x, replace the active vertex $v_x = (v_1, v_2, \ldots, v_k)$ by $(v_2, \ldots, v_k, c_x)$.

- Return to previous level of recursion.

Note that h_x, $\tilde{h}_x$, and in particular, $h_{\mathbf{1}}$ can be computed as sums of edge weights in the sequencing graphs, we omit the details. As a post-processing step of the algorithm, we can sort out all string candidates s with weight $\varphi(s) > \psi_{\min}$. We omit the proof of the following theorem for the sake of brevity.

Theorem 2. *For all $x \in \Sigma$, let w_x be characteristic compomer weights satisfying $w_x(c) \geq 0$ for all compomers c. Let Σ_x be a compomer alphabet over (Σ, x). For a fixed spectrum order k and S as defined above, the algorithm of this section will return all strings $s \in S$ and their weights $\varphi(s)$ that are solutions of* WSC *and satisfy $\mathcal{C}_0(s, x) \subseteq \Sigma_x$.*

Our algorithm is a runtime heuristic and, as such, has exponential worst-case runtime. Also, there may be exponentially many solutions to WSC. But usually, we can find the correct answer much faster than the worst case analysis suggests. For $n := \max\{|s| : s \in S\}$ we need $O(n)$ memory in the recursion part of the algorithm. The critical factor is obviously storing the sequencing graphs and in general prohibits the use of orders $k > 2$.

A simple implementation of the complete process of de-novo sequencing from mass spectrometry data is now as follows: Firstly, we generate detected compomer sets $\mathcal{C}_x$ for all $x \in \Sigma$ as described in [Bö03]. These sets are used to define the trivial characteristic compomer weights w_x that, in turn, allow us to build weighted sequencing graphs G_x. We use our algorithm to generate all sample string candidates s that are solutions to WSC satisfying $\mathcal{C}_0(s, x) \subseteq \Sigma_x$. Clearly, we can further evaluate the generated sample string candidates by, say, an appropriate likelihood measure, taking into account MS data from all cleavage reactions.

We want to stress that a heuristic used to analyze the MS data which *cannot guarantee* to find the correct sample string, is not acceptable in the setting of DNA de-novo sequencing. So, there is no way to circumvent the computational complexity of WSC.

5 Discussion

We have introduced the Weighted Sequencing from Compomers Problem that stems from the analysis of mass spectrometry data from partial cleavage experiments. WSC extends the Sequencing From Compomers Problem introduced in [Bö03] by taking into account false negative peaks in the sample mass spectra. Although WSC is computationally difficult in general, we have introduced an approach to perform de-novo sequencing from such data. The introduced method uses weighted de Bruijn graphs to construct all DNA sequences that are "compatible" with the observed mass spectra.

We have tested the performance of our approach on simulated mass spectrometry data from random and biological sequences (data not shown). Simulation results indicate that the

presented approach is capable of reconstructing the correct sequence in many cases if the ratio of false negative peaks is small, and ambiguities are often limited to a small number of bases. So, this approach may enable de-novo sequencing from mass spectrometry data, even when false negative peaks must be taken into account. Application of the method to "real-world" mass spectrometry data is in progress.

Acknowledgments

Sebastian Böcker is currently supported by "Deutsche Forschungsgemeinschaft" (BO 1910/1-1) within the Computer Science Action Program. Additional programming provided by Matthias Steinrücken. I thank Zsuzsanna Lipták, Hans-Michael Kaltenbach, and Jens Stoye for proofreading earlier versions of this manuscript.

References

[APC+00] Altshuler, D., Pollara, V. J., Cowles, C. R., Etten, W. J. V., Baldwin, J., Linton, L., and Lander, E. S.: An SNP map of the human genome generated by reduced representation shotgun sequencing. *Nature*. 407:513–516. 2000.

[Bö03] Böcker, S.: Sequencing from compomers: Using mass spectrometry for DNA de-novo sequencing of 200+ nt. Extended abstract. In: *Proc. of WABI 2003, Budapest, Hungary*. Volume 2812 of *Lect. Notes Comput. Sc.* pp. 476–497. Springer. 2003.

[Bö04] Böcker, S.: Sequencing from compomers: Using mass spectrometry for DNA de-novo sequencing of 200+ nt. To appear in *J. Comput. Biol.* 2004.

[CKT+01] Chen, T., Kao, M.-Y., Tepel, M., Rush, J., and Church, G. M.: A dynamic programming approach to de novo peptide sequencing via tandem mass spectrometry. *J. Comput. Biol.* 8(3):325–337. 2001.

[FCK02] França, L. T. C., Carrilho, E., and Kist, T. B. L.: A review of DNA sequencing techniques. *Q. Rev. Biophys.* 35(2):169–200. May 2002.

[HSB+03] Hartmer, R., Storm, N., Böcker, S., Rodi, C. P., Hillenkamp, F., Jurinke, C., and van den Boom, D.: RNAse T1 mediated base-specific cleavage and MALDI-TOF MS for high-throughput comparative sequence analysis. *Nucleic Acids Res.* 31(9):e47. 2003.

[KH88] Karas, M. and Hillenkamp, F.: Laser desorption ionization of proteins with molecular masses exceeding 10,000 Daltons. *Anal. Chem.* 60:2299–2301. 1988.

[RDPS+02] Rodi, C. P., Darnhofer-Patel, B., Stanssens, P., Zabeau, M., and van den Boom, D.: A strategy for the rapid discovery of disease markers using the MassARRAY system. *BioTechniques*. 32:S62–S69. 2002.

[SNC77] Sanger, F., Nicklen, S., and Coulson, A. R.: DNA sequencing with chain-terminating inhibitors. *Proc. Natl. Acad. Sci. USA*. 74(12):5463–5467. 1977.

[vBS+02] von Wintzingerode, F., Böcker, S., Schlötelburg, C., Chiu, N. H., Storm, N., Jurinke, C., Cantor, C. R., Göbel, U. B., and van den Boom, D.: Base-specific fragmentation

of amplified 16S rRNA genes and mass spectrometry analysis: A novel tool for rapid bacterial identification. *Proc. Natl. Acad. Sci. USA*. 99(10):7039–7044. 2002.