Chen et al. BMC Systems Biology 2012, 6(Suppl 2):S5
http://www.biomedcentral.com/1752-0509/6/S2/S5

BMC Systems Biology

PROCEEDINGS Open Access

Two combinatorial optimization problems for SNP discovery using base-specific cleavage and mass spectrometry

Xin Chen[1*], Qiong Wu[1,2], Ruimin Sun[1], Louxin Zhang[3]

From 23rd International Conference on Genome Informatics (GIW 2012)
Tainan, Taiwan. 12-14 December 2012

Abstract

Background: The discovery of single-nucleotide polymorphisms (SNPs) has important implications in a variety of genetic studies on human diseases and biological functions. One valuable approach proposed for SNP discovery is based on base-specific cleavage and mass spectrometry. However, it is still very challenging to achieve the full potential of this SNP discovery approach.

Results: In this study, we formulate two new combinatorial optimization problems. While both problems are aimed at reconstructing the sample sequence that would attain the minimum number of SNPs, they search over different candidate sequence spaces. The first problem, denoted as $SNP{-}MS_{\mathcal{P}}$, limits its search to sequences whose *in silico* predicted mass spectra have all their signals contained in the measured mass spectra. In contrast, the second problem, denoted as $\text{SNP - MS}_{\mathcal{Q}}$, limits its search to sequences whose *in silico* predicted mass spectra instead contain all the signals of the measured mass spectra. We present an exact dynamic programming algorithm for solving the $SNP{-}MS_{\mathcal{P}}$ problem and also show that the $\text{SNP - MS}_{\mathcal{Q}}$ problem is NP-hard by a reduction from a restricted variation of the 3-partition problem.

Conclusions: We believe that an efficient solution to either problem above could offer a seamless integration of information in four complementary base-specific cleavage reactions, thereby improving the capability of the underlying biotechnology for sensitive and accurate SNP discovery.

Background

Single nucleotide polymorphisms (SNPs) is a common type of DNA sequence variations that occur when a single nucleotide base is altered at a specific locus. They are among the most important genetic factors that contribute to human disease and biological functions. However, discovering novel SNPs is a scientifically challenging task. Among others, one valuable approach proposed for SNP discovery is based on base-specific cleavage and mass spectrometry [1-3].

The SNP discovery approach based on base-specific cleavage and mass spectrometry usually adopts a data-acquisition procedure as summarized below. First, a target sample DNA sequence is PCR-amplified using primers that incorporate the T7 promoter sequences. Then, the PCR products are in-vitro transcribed and subsequently digested with the endonuclease RNase A in four base-specific cleavage reactions. Each reaction can cleave the sample sequence to completion at all loci wherever a specific base is found. Finally, the matrix-assisted laser desorption/ionization time-of-flight mass spectrometry (MALDI-TOF MS) is applied to the cleavage products, resulting in four measured mass spectra, each corresponding to one base-specific cleavage reaction.

Since each cleavage product is expected to be made of three non-cleavage bases, it is fairly straightforward to calculate the base composition from its measured mass signal. With all these base compositions in hand, the task

* Correspondence: chenxin@ntu.edu.sg
[1]School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore
Full list of author information is available at the end of the article

of discovering SNPs in the sample sequence is now left to a computational solution. In principle, this computational solution shall find a way to integrate the four complementary base-specific mass spectra, and then identify those SNPs that necessarily account for the unanticipated base compositions (i.e., corresponding to the measured mass signal changes as compared with an *in-silico* predicted mass spectra from a reference sequence). See Figure 1 for schematic outline of the SNP discovery approach using base-specific cleavage and mass spectrometry.

The early proof-of-concept studies on the above SNP discovery approach using base-specific cleavage and mass spectrometry were presented in [3-5], where the identification of SNPs however was done by visual inspection. Shortly afterwards, two automated computational solutions were developed [1,2]: one was implemented in the proprietary MassARRAY™ SNP Discovery software package from Sequenom, Inc. and the other implemented in the software package called RNaseCut which is instead freely available online [6]. In particular, the solution in [1] mainly comprises of two separate procedures. It first computes all potential SNPs that give rise to each unanticipated based composition and then score them by taking into account the mass spectrometry data from the four base-specific cleavage reactions. Thus, the integration of the four base-specific cleavage reactions was done only in the second step. Apparently, such an integration strategy is far from being optimal, as at least it assumes that the occurrences of potential SNPs are independent in the first step.

In this paper, we study two new combinatorial optimization problems to exploit the full potential of the above SNP discovery approach. While both problems are aimed at reconstructing the sample sequence that would attain the minimum number of SNPs, they search over different candidate sequence spaces. The first problem, denoted as $SNP{-}MS_{\mathcal{P}}$, limits its search to sequences whose *in silico* predicted mass spectra have all their signals contained in the measured mass spectra. In contrast, the second problem, denoted as $\text{SNP - MS}_{\mathcal{Q}}$, limits its search to sequences whose *in silico* predicted mass spectra instead contain all the signals of the measured mass spectra. Then, we present an exact dynamic programming algorithm for solving the $SNP{-}MS_{\mathcal{P}}$ problem and also show that the $\text{SNP - MS}_{\mathcal{Q}}$ problem is NP-hard by a reduction from the restricted variation of the 3-partition problem [7,8].

Methods

Preliminaries

Let $s \in \Sigma^*$ denote a string over the four-base alphabet $\sum = \{A, C, G, T\}$. The length of s is denoted by $|s|$, the i-th base of s by $s[i]$, and the substring of s from the i-th base to the j-th base by $s[i, j]$, for $1 \le i \le j \le |s|$. We use $\lfloor$ to denote the empty string so that $|\lfloor| = 0$. The concatenation of two strings s and t is denoted by $s \cdot t$, and the concatenation of l copies of a string s is denoted by s^l.

Given a string s and a cut base $x \in \sum$, a *cleavage fragment* refers to a substring of s that does not contain x and that cannot be extended in either side without crossing a base x. Formally, the substring $s[i, j]$ is a cleavage fragment with respect to the cut base x if the following three conditions are satisfied: (i) $s[i - 1] = x$ if $i \ne 1$, (ii) $s[j + 1] = x$ if $j \ne |s|$, and (iii) $s[k] \ne x$, $\forall k \in [i, j]$. In addition, the empty string ε is a cleavage fragment if there exits $i \in [1, |s| - 1]$ such that $s[i] = s[i + 1] = x$. Given a cleavage fragment, we use $A_iC_jG_kT_l$ to denote its base composition of i As, j Cs, k Gs, and l Ts. In [1], this base composition is termed as a *compomer* of the string s with respect to the cut base x. The whole set of compomers is hence called the *compomer spectrum* of the string

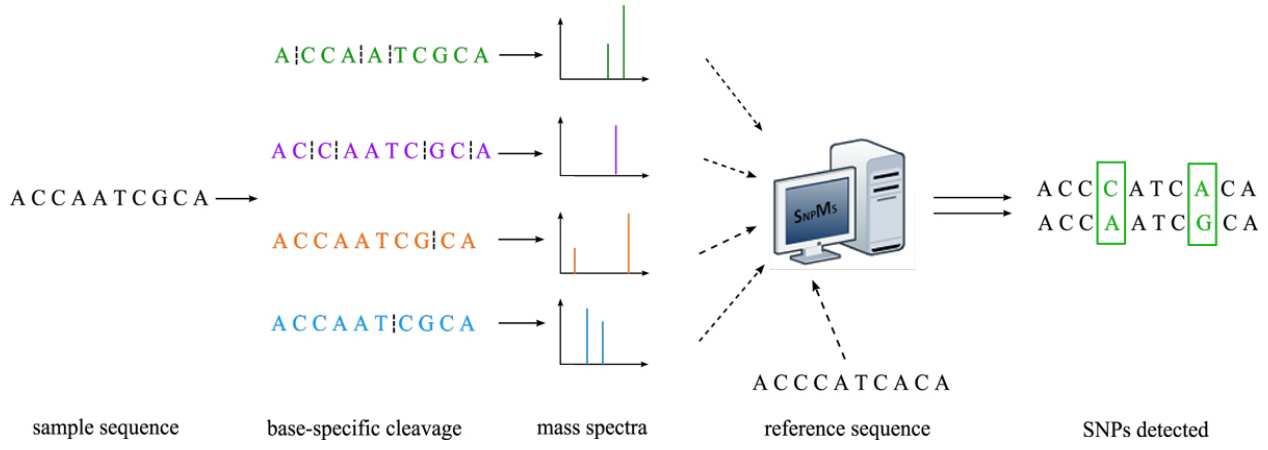

Figure 1 Schematic outline. The SNP discovery approach using base-specific cleavage and mass spectrometry.

s with respect to the cut base x, and denoted by Finally, let $\mathcal{C}_{\sum}(s) = \{\mathcal{C}_x(s) : x \in \sum\} = \{\mathcal{C}_A(s), \mathcal{C}_C(s), \mathcal{C}_G(s), \mathcal{C}_T(s)\}$, a collection of four compomer spectra of the string s where each is generated with one cut base.

Example 1 *Let* s := ACATGCTACATTA. *Then, the string s contains four cleavage fragments with respect to the cut base* A: C, TGCT, C, *and* TT. *With respect to the cut base* T, *it instead contains five cleavage fragments:* ACA, GC, ACA, $\lfloor$, *and* A. *Their respective compomer spectra are* $\mathcal{C}_A(s) = \{A_0C_1G_0T_0, A_0C_1G_1T_2, A_0C_0G_0T_2\}$ *and* $\mathcal{C}_T(s) = \{A_2C_1G_0T_0, A_0C_1G_1T_0, A_0C_0G_0T_0, A_1C_0G_0T_0\}$. *Note that each compomer appears in a compomer spectrum at most once.*

Problem formulation

Let $d_H(s, s')$ denote the Hamming distance between two strings s and s' of equal length. It measures the minimum number of substitutions required to transform one string into the other. Given a collection of compomer spectra $\mathcal{C}_\Sigma = \{\mathcal{C}_x : x \in \Sigma\}$ of an unknown string s' (i.e., the sample DNA sequence experimented) which can in principle be generated from a mass spectrometry experiment, and a string s (i.e., the reference DNA sequence) which is believed to differ from the unknown string s' by a number of substitutions only, we formulate below two combinatorial optimization problems for SNP discovery.

Definition 2 (**The SNP** $-$ $\mathbf{MS}_\mathcal{P}$ **problem**) *Given a string s and a collection of compomer spectra* $\mathcal{C}_\Sigma = \{\mathcal{C}_x : x \in \Sigma\}$, *find a string* s' *such that* $\mathcal{C}_x(s') \subseteq \mathcal{C}_x$, *for all* $x \in \sum$ *and* $d_H(s, s')$ *is minimized.*

Definition 3 (**The SNP** $-$ $\mathbf{MS}_\mathcal{Q}$ **problem**) *Given a string s and a collection of compomer spectra* $\mathcal{C}_\Sigma = \{\mathcal{C}_x : x \in \Sigma\}$, *find a string* s' *such that* $\mathcal{C}_x \subseteq \mathcal{C}_x(s')$, *for all* $x \in \sum$ *and* $d_H(s, s')$ *is minimized.*

The only difference between the above two problem formulations is that one requires $\mathcal{C}_x(s') \subseteq \mathcal{C}_x$ and the other requires $\mathcal{C}_x \subseteq \mathcal{C}_x(s')$, for all the cut bases. Once the string s' is found, it is easy to identify the SNPs in s', i.e., those base substitutions that transform s' into s.

Example 4 *In this example, we let* $\sum$:= {A, T} *for simplicity. Given the string s*:= ATAAT *and the set* $\mathcal{C} = \{\mathcal{C}_A, \mathcal{C}_T\}$ *of compomer spectra (of an unknown string) where*

$$\mathcal{C}_A = \{A_0T_1, A_0T_2\} \quad \text{and} \quad \mathcal{C}_T = \{A_0T_0, A_1T_0\}.$$

The feasible solutions to the $SNP{-}MS_\mathcal{P}$ *problem for the above instance include the strings such as* ATATA, TATAT, TTATT, ATATT, *and* ATTAT. *Their respective Hamming distances to the input string s are 2, 3, 2, 1, and 1. The string* s' = TTAAT *is not a feasible solution because the compomer* $A_2T_0 \in \mathcal{C}_T(s')$ *but* $A_2T_0 \notin \mathcal{C}_T$ *so that* $\mathcal{C}_T(s') \not\subseteq \mathcal{C}_T$.

The feasible solutions to the $SNP - MS_\mathcal{Q}$ *problem for the above instance include the strings such as* TTATA, TATTA, ATATT, *and* ATTAT. *Their respective Hamming distances to the input string s are 3, 5, 1, and 1. The string* s' = TTAAT *is not a feasible solution because the compomer* $A_2T_0 \in \mathcal{C}_T$ *but* $A_1T_0 \notin \mathcal{C}_T(s')$ *so that* $\mathcal{C}_T \not\subseteq \mathcal{C}_T(s')$.

The measured mass spectra of a sample sequence are rarely perfect in practice. Some peaks may actually represent noises, while some true signal peaks are missing. The problem $SNP{-}MS_\mathcal{P}$ is so formulated that its computational solution would be robust against noisy peaks but susceptible to missing peaks (i.e., there is a good chance to recover the sample sequence even if some noisy peaks are present in the measured mass spectra, but the chance would become much less if there are some true signal peaks missing). In contrast, the problem SNP - $MS_\mathcal{Q}$ is so formulated that its computational solution would be robust against missing peaks but susceptible to noisy peaks.

We noticed that several computational problems in the literature that are more or less related to our problems introduced above. In [9], a so-called *sequencing from compomers* problem was studied which, like the $SNP{-}MS_\mathcal{P}$ problem, also aimed to reconstruct the sample sequence from a given collection of compomer spectra, but without help of a reference sequence. In [10], the *spectral alignment* problem differs from the $SNP{-}MS_\mathcal{P}$ problem mainly by its exploration on short read sequencing data rather than the mass/compomer spectra data, which may lead to wide implications in the subsequent algorithm design and complexity analysis. Moreover, in [1], a so-called *SNP discovery from mass spectrometry* problem was defined in a similar way to the SNP - $MS_\mathcal{Q}$ problem. However, it has only a single compomer as input, as opposed to a collection of four complementary compomer spectra used in the SNP - $MS_\mathcal{Q}$ problem.

Results

An exact dynamic programming algorithm for $SNP{-}MS_\mathcal{P}$

In this subsection, we shall describe an exact dynamic programming algorithm for solving the $SNP{-}MS_\mathcal{P}$ problem. Without loss of generality, we may assume in the remaining of this section that every base of Σ will eventually occur in the optimal solution to a given instance of the $SNP{-}MS_\mathcal{P}$ problem. Consequently, only those feasible solutions that contains all the bases of Σ need to be considered when we search for the optimal solution. In case some base x would not occur in the optimal solution s' note that it becomes relatively easy to find s' since we would have $s' \in \mathcal{L}_x \cap \mathcal{R}_x$ and $|s'| = |s|$. See below for definitions of $\mathcal{L}_x$ and $\mathcal{R}_x$.

Let us start with some preliminary definitions and notations. For a string s, a cleavage fragment $s[i, j]$ is called *internal* if neither $i = 1$ nor $j = |s|$, *left-ended* if $i = 1$, or *right-ended* if $j = |s|$. In addition, a cleavage fragment $\lfloor$ is always considered internal. Given a collection of compomer spectra $\mathcal{C}_\Sigma$, we call a string is *I-compatible* if the compomers of its internal cleavage fragments are all contained in $\mathcal{C}_\Sigma$ (under the respective cut base). A string is called *L-compatible* (resp. *R-compatible*) if it is I-compatible and if the compomers of its left-ended (resp. right-ended) cleavage fragments are all contained in $\mathcal{C}_\Sigma$ as well.

Example 5 *Consider the string s given in Example 1. The four cleavage fragments of s with respect to the cut base* A *are all internal. Among the five cleavage fragments of s with respect to the base* T, *the first cleavage fragment* ACA *is left-ended, the last cleavage fragment* A *is right-ended, and the other three cleavage fragments in the middle are all internal.*

Example 6 *Let* $\mathcal{C}_\Sigma = \{\mathcal{C}_A, \mathcal{C}_C, \mathcal{C}_G, \mathcal{C}_T\}$ *be a collection of compomer spectra where*

$$\mathcal{C}_A = \{A_0C_1G_0T_0, A_0C_1G_1T_2, A_0C_0G_0T_2\},$$
$$\mathcal{C}_C = \{A_1C_0G_0T_0, A_1C_0G_1T_1, A_1C_0G_0T_1, A_2C_0G_0T_2\},$$
$$\mathcal{C}_G = \{A_2C_1G_0T_1, A_3C_2G_0T_3\}, \quad \text{and}$$
$$\mathcal{C}_T = \{A_2C_1G_0T_0, A_0C_1G_1T_0, A_0C_0G_0T_0, A_1C_0G_0T_0\}.$$

We show in Table 1 *whether each of the given strings is I-compatible, L-compatible, or R-compatible with* $\mathcal{C}_\Sigma$.

For each compomer $A_iC_jG_kT_l \in \mathcal{C}_x$ in a given collection of compomer spectra $\mathcal{C}_\Sigma$, we use $\mathcal{I}_x(A_iC_jG_kT_l)$ to denote the set of strings that (i) consist of i As, j Cs, k Gs, l Ts, (ii) contain exactly three distinct bases (i.e., three bases in the set $\Sigma \setminus \{x\}$), and (iii) are I-compatible with $\mathcal{C}_\Sigma$. It is easy to check that $|\mathcal{I}_x(A_iC_jG_kT_l)| \leq \frac{(i+j+k+l)!}{i!j!k!l!}$. In particular, if there exists in $A_iC_j\ G_kT_l$ a non-cut base whose composition value is zero, then we have $|\mathcal{I}_x(A_iC_jG_kT_l)| = \emptyset$ so that $|\mathcal{I}_x(A_iC_jG_kT_l)| = 0$. Furthermore, we may define the following set

$$\mathcal{I}_x = \bigcup_{A_iC_jG_kT_l \in C_x} \mathcal{I}_x(A_iC_jG_kT_l), \quad \forall x \in \Sigma.$$

Table 1 Examples.

strings	I-compatible	L-compatible	R-compatible
ATGATAC	✗	✗	✗
ATGCTAC	✓	✗	✗
ACATGCT	✓	✓	✗
TACATTA	✓	✗	✗
CTACATTA	✓	✗	✓

This table shows whether each of the given strings is I-compatible, L-compatible, or R-compatible with $\mathcal{C}_\Sigma$.

Then, let $\mathcal{I}_\Sigma = \{\mathcal{I}_A, \mathcal{I}_C, \mathcal{I}_G, \mathcal{I}_T\}$. Analogously, we may define $\mathcal{L}_x(A_iC_jG_kT_l)$, $\mathcal{R}_x(A_iC_jG_kT_l)$, $\mathcal{L}_\Sigma = \{\mathcal{L}_A, \mathcal{L}_C, \mathcal{L}_G, \mathcal{L}_T\}$ and $\mathcal{R}_\Sigma = \{\mathcal{R}_A, \mathcal{R}_C, \mathcal{R}_G, \mathcal{R}_T\}$for the L-compatible strings and the R-compatible strings, respectively. Clearly, $L_x \subseteq \mathcal{I}_x$ and $\mathcal{R}_x \subseteq \mathcal{I}_x$, for all $x \in \Sigma$.

Example 7 *Consider the collection of compomer spectra* $\mathcal{C}_\Sigma$ *given in Example 6. For the compomer* $A_0C_1G_1T_2 \in \mathcal{C}_A$, *we have* $\mathcal{I}_A(A_0C_1G_1T_2) = \{CGTT, CTTG, GCTT, GTTC, TCGT, TGCT, TTCG, TTGC\}$, *and* $\mathcal{L}_A(A_0C_1G_1T_2) = \mathcal{R}_A(A_0C_1G_1T_2) = \emptyset$. For the compomer $A_0C_1G_1T_0 \in \mathcal{C}_T$, *we have* $\mathcal{I}_T(A_0C_1G_1T_0) = \mathcal{L}_T(A_0C_1G_1T_0) = \mathcal{R}_T(A_0C_1G_1T_0) = \emptyset$.

Given a string t which could be a potential cleavage fragment with respect to the cut base x (i.e., the string t does not contain any base x), we say a string s begins with the string t if $t \cdot x$ is a prefix of $s \cdot x$, or say a string s ends with the string t if $x \cdot t$ is the suffix of $x \cdot s$. The following lemma is useful to design a dynamic programming algorithm for solving the $SNP{-}MS_\mathcal{P}$ problem. Its easy proof is omitted. Recall that our discussions in this section are limited only to the feasible solutions containing all the bases of Σ.

Lemma 8 *A string s' of length* $|s|$ *is a feasible solution to the* $SNP{-}MS_\mathcal{P}$ *problem if and only if*

- *all the substrings of* s' *are I-compatible with* $\mathcal{C}_\Sigma$,
- s' *begins with a string in* $\mathcal{L}_x$ *for some* $x \in \sum$, *and*
- s' *ends with a string in* $\mathcal{R}_x$ *for somex* $\in \sum$.

Suppose we have an input instance $\langle s, \mathcal{C}_\Sigma \rangle$ of the $SNP{-}MS_\mathcal{P}$ problem. Given a string $t \in \mathcal{I}_x$ where $x \in \sum$, we define $\mathcal{H}(i, t)$ to be the minimum Hamming distance between the prefix of s of length i and a string which is such that

- all its substrings are I-compatible with $\mathcal{C}_\Sigma$,
- it begins with a string from $\mathcal{L}_y$ for some $y \in \Sigma$, and
- it ends with the given string t.

To compute $\mathcal{H}(i, t)$, we first find in the string $x \cdot t$ the rightmost position k at which the base $(x \cdot t)[k]$ is its first occurrence. Formally, we may write

$$k = \max\{j : \forall i,\ 1 \leq i < j \leq |x \cdot t|,\ (x \cdot t)[i] \neq (x \cdot t)[j]\}.$$

Then, let x':= $(x \cdot t)[k]$, $p := (x \cdot t)[1, k - 1]$, and $q := (x \cdot t)[k, |x \cdot t|]$. Note that $x' \neq x$ and the string p contains all the bases of Σ except x'.

Example 9 *Let* $t :=$ CGTT $\lfloor$ I_A. *Then,* $x \cdot t =$ ACGTT, $k = 4$, $x' =$ T, $p =$ ACG, *and* $q =$ TT.

To compute $\mathcal{H}(i, t)$, we now use the following recurrence relation

$$\mathcal{H}(i, t) = \min_{t' \in \mathcal{I}_{x'}} \{\mathcal{H}(i - |q|, t') + d_H(s[i - |q| + 1, i], q)\}.$$

$$\exists t'', t' = t'' \cdot p$$

Note that the minimization in the above is taken over all those strings t' in $\mathcal{I}_{x'}$ which have p as the suffix. If there is no such a string in $\mathcal{I}_{x'}$, then we let $\mathcal{H}(i, t) = \infty$. The initial conditions for the recurrence relation are given as follows:

$$\mathcal{H}(i,\ t) = \begin{cases} \infty & \text{if } i < |t| \text{ and } t \in \mathcal{I}_x \\ d_H(s[1,\ i],\ t) & \text{if } i = |t| \text{ and } t \in \mathcal{L}_x \\ \infty & \text{if } i = |t| \text{ and } t \in \mathcal{I}_x \backslash \mathcal{L}_x. \end{cases}$$

Theorem 10 *Let s' be the string that leads to*

$$d_H(s,\ s') = \min_{\forall t \in \mathcal{R}_x, x \in \Sigma} \mathcal{H}(|s|,\ t),$$

then s' would be an optimal solution to the input instance $\langle s, \mathcal{C}_{\sum} \rangle$ *of the SNP - MS$_{\mathcal{P}}$ problem.*

Proof: For the correctness of the above dynamic programming algorithm, we need to show that (i) every feasible solution of the $SNP{-}MS_{\mathcal{P}}$ problem would be essentially evaluated by the dynamic programming algorithm, and (ii) every string evaluated by the dynamic programming algorithm must be a feasible solution of the $SNP{-}MS_{\mathcal{P}}$ problem.

Let the string s' be a feasible solution. Consider a cleavage fragment t of s' that contains all the bases of Σ except its corresponding cut base x. Clearly, $t \in \mathcal{I}_x$ and t is the suffix of a substring $s'[1, i]$ for some integer i. Without loss of generality, we can further suppose that $t \neq s'[1, i]$. To show (i), what we mainly need to show is that there exists a string $t' \in \mathcal{I}_{x'}$ such that p is the suffix of t' and t' is the suffix of the substring $s'[1, i - |q|]$, where x', p, and q are computed for the string t as described earlier. Indeed, we can find the string t' as follows. First, let $(i' - 1)$ be the position of the last occurrence of the base x' in the substring $s'[1, i - |t|]$; if there is no such occurrence, we let $i' = 1$. Then, we assign $t' := s'[i', i - |q|]$. Obviously, t' is the suffix of $s'[1, i\ |q|]$. Because $s'[i - |t|] = x$ and $x \neq x$, we have $i' \leq i - |t|$. It then follows from $p = s'[i - |t|, i - |q|]$ that p shall be the suffix of t'. Since p contains all the bases of $\sum$ except x' so, does t'. Moreover, t' is a cleavage fragment of s' with respect to the cut base x' because we have either $s'[i' - 1] = x'$ or $i' = 1$ on the left end of t' and $s'[i - |q| + 1] = x'$ on the right end of t'. By Lemma 8, we can see that $t' \in \mathcal{I}_A$. For the reader's convenience, we demonstrate in the following example how to find t' from t. Let s' = ACATGCTACATTA, $t = s'$ [4,7] = TGCT, $i = 7$, x = A, and $\mathcal{C}_{\sum}$ be the one as given in Example 6. Note that$t \in \mathcal{I}_A$. Further, for the given string t = TGCT, we have x' = C, p = ATG, and q = CT. Then, we obtain that $i' = 3$ and then $t' = s'$ [3, 7 − 2] = s' [3,5] = ATG. It is easy to check that p is the suffix of t', t' is the suffix of the substring $s'[1, i - |q|]$, and $t' \in \mathcal{I}_{x'}$.

On the other hand, let s' be a string evaluated by the dynamic programming algorithm. So, the string s' must begin with a string in $\mathcal{L}_x$ for some $x \in \sum$ and end with a string in $\mathcal{R}_y$ for some $y \in \sum$. Consider a cleavage fragment t of s' that was used to construct the string s' during the backtracking procedure of the algorithm. Clearly, the string t contains all the bases of $\sum$ except its corresponding cut base x. Moreover, $t \in \mathcal{I}_x$ and t is the suffix of a substring $s'[1, i]$ for some integer i. Without loss of generality, we can further suppose $t \neq s'[1, i]$ and $i \neq |s'|$, so that $s'[i - |t|] = s'[i + 1] = x$. Let t' be the string considered next to the string t during the backtracking procedure of the algorithm. Thus, we have $t' \in \mathcal{I}_{x'}$ such that p is the suffix of t' and t' is the suffix of the substring $s'[1, i - |q|]$, where x', p, and q are computed for the string t as described earlier. More specifically, there exists i' such that $t' = s'[i', i - |q|]$ and $s'[i' - 1] = s'\ [i - |q| + 1] = x'$ if $i' \neq 1$. To show (ii), by Lemma 8 and also by backward induction, what we mainly need to show is that the extended substring $s'[i', |s'|]$ is I-compatible with $\mathcal{C}_{\sum}$, given that the substring $s'[i - |t| + 1, |s'|]$ is already I-compatible with $\mathcal{C}_{\sum}$. To this end, we consider any internal cleavage fragment $s'[j, k]$ of s' $[i', |s'|]$ with respect to the cut base $x'' = s'[j - 1] = s'[k + 1]$. By definition of the internal cleavage fragment, we have $j \geq i' + 1$ and $k \leq |s'| - 1$. In the following we distinguish four cases:

- If $j \geq i - |t| + 2$, then $s'[j, k]$ is an internal cleavage fragment of $s'[i - |t| + 1, |s'|]$. Since $s'[i - |t| + 1, |s'|]$ is already assumed to be I-compatible with $\mathcal{C}_{\sum}$, the base composition of $s'[j, k]$ shall be also contained in $\mathcal{C}_{x''}$.
- If $j = i - |t| + 1$, then $x'' = x$, which further implies that $k = i$ and $s'\ [j, k] = t$. Since $t \in \mathcal{I}_x$, the base composition of $s'[j, k]$ shall be contained in $\mathcal{C}_{x''}$.
- If $j \leq i - |t|$ and $k \geq i - |q|$, then $s'[i - |t|, i - |q|]$ is a substring of $s'[j, k]$. Since $s'[i - |t|, i - |q|]$ contains all the bases of Σ, the string $s'[j, k]$ can not be a cleavage fragment (as a cleavage fragment must not contain its corresponding cut base). Therefore, there shall not have the case where $j \leq i - |t|$ and $k \geq i - |q|$.
- If $k \leq i - |q| - 1$, then $s'[j, k]$ is an internal cleavage fragment of $t' = s'[i', i - |q|]$. Since $t \in \mathcal{I}_{x'}$, the base composition of $s'[j, k]$ shall be contained in $\mathcal{C}_{x''}$.

In conclusion, for every internal cleavage fragment of $s'[i', |s'|]$, its base composition is contained in $\mathcal{C}_{\sum}$ under the respective cut base. Therefore, the extended substring $s'[i', |s'|]$ is still I-compatible with $\mathcal{C}_{\sum}$.

Note that computing each entry $\mathcal{H}(i, t)$ of the dynamic programming table may take time $O(|s| \cdot |\mathcal{I}_{\sum}|)$, where $|\mathcal{I}_{\sum}| = |\mathcal{I}_A| + |\mathcal{I}_C| + |\mathcal{I}_G| + |\mathcal{I}_T|$. Hence, the above

dynamic programming algorithm can be done in time $O(|s|^2 \cdot |\mathcal{I}_{\sum}|^2)$. In the worst case, we may have $|\mathcal{I}_{\sum}| = O(|s|!)$, that is, $|\mathcal{I}_{\sum}|$ is in the factorial order of the input problem size. In practice, however, we would expect $|\mathcal{I}_{\sum}|$ not too large to be manageable, because cleavage fragments are usually of small size. Therefore, the above dynamic programming algorithm could be a practically feasible solution to the problem $SNP{-}MS_{\mathcal{P}}$, especially when compared to the brute-force algorithm which needs to examine all the possible strings s'. For the special case where $|\sum| = 2$, $SNP{-}MS_{\mathcal{P}}$ is actually an easy problem, as we can see from the above that $|\mathcal{I}_{\sum}| = O(|s|)$.

Corollary 11 *The above dynamic programming algorithm can solve the* $SNP - MS_{\mathcal{P}}$ problem in polynomial time when $|\sum| = 2$.

The NP-hardness of SNP - $MS_{\mathcal{Q}}$

This subsection is dedicated to prove that the SNP - $MS_{\mathcal{Q}}$ problem is NP-hard. We begin with a brief introduction of the 3-partition problem.

Definition 12 (The general form of the 3-partition problem) *Given a multiset of positive integers* $\mathcal{A} = \{a_1, a_2, \cdots, a_n\}$ *where* $n = 3m$ *and* $\sum_{i=1}^{n} a_i = mB$, *can we partition the multiset* $\mathcal{A}$ *into* m *multisets* $\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_m$, *such that the sum of each multiset is equal to* B?

The 3-partition problem is strongly NP-complete [7]. Therefore, it remains NP-complete even when the integers in $\mathcal{A}$ and the integer B are encoded in unary. In this case, the size of a problem instance is $\Theta(nB)$. In contrast, it becomes $O(n \log B)$ when using the binary encoding of integers.

Definition 13 (The restricted variation of the 3-partition problem) *Given a set of positive integers* $\mathcal{A} = \{a_1, a_2, \cdots, a_n\}$ *where* $n = 3m$, $\sum_{i=1}^{n} a_i = mB$, *and* $\frac{B}{4} < a_i < \frac{B}{2}, \forall 1 \le i \le n$, *can we partition the set* $\mathcal{A}$ *into* m *subsets* $\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_m$, *such that the sum of each subset is equal to* B?

There are two constraints imposed in the above restricted variation of the 3-partition problem. The first one limits $\mathcal{A}$ to be a set so that all the integers in $\mathcal{A}$ are distinct. The second one limits all the integers in $\mathcal{A}$ strictly between $\frac{B}{4}$ and $\frac{B}{2}$, which subsequently enforces every subset $\mathcal{A}_i$ to consist of exactly three elements. Interestingly, this restricted variation of the 3-partition problem remains strongly NP-complete [8], just like the general form of the 3-partition problem. Note that the second constraint $\frac{B}{4} < a_i < \frac{B}{2}$ was actually not imposed in [8]. But, it can be easily done by adding B to each a_i and then multiplying B by 4.

Theorem 14 *The* $SNP - MS_{\mathcal{Q}}$ *problem is NP-hard, even when* $|\sum| = 2$.

Proof: We prove it by a reduction from the above restricted variation of the 3-partition problem. As an input for 3-partition, we are given a set of distinct positive integers $\mathcal{A} = \{a_1, a_2, \cdots, a_n\}$ where $n = 3m$, $\sum_{i=1}^{n} a_i = mB$, and $\frac{B}{4} < a_i < \frac{B}{2}, \forall 1 \le i \le n$. Then, we construct an instance $\langle s, \mathcal{C}_{\sum} \rangle$ of the SNP - $MS_{\mathcal{Q}}$ problem as follows:

- Let $\Sigma = \{G, T\}$.
- Let s be the string such that $s \cdot T = (G^{B+2}T)^m$. That is, let $s \cdot T$ be the concatenation of m copies of the fragment $GG \cdots GT$, where each fragment consists of $(B + 2)$ consecutive base Gs followed by one base T. Note that $|s| = m(B + 3) - 1 = mB + 3m - 1$.
- Let $\mathcal{C}_G = \{G_0T_0, G_0T_1\}$ and $\mathcal{C}_T = \{G_{a_i}T_0 : 1 \le i \le n\}$ so that $\mathcal{C}_{\sum} = \{\mathcal{C}_G, \mathcal{C}_T\}$.

First, we check whether this construction can be done in polynomial time in the size of the input instance of the 3-partition problem. Since the restricted variation of the 3-partition problem is strongly NP-complete, we may encode the integers in unary so that the size of the input instance is $\Theta(nB)$. In the above reduction, we can easily see that the first step can be done in constant time, the second step in time $O(mB)$, and the third step in time $O(n \log B)$. Therefore, the total time needed for construction is $O(nB)$, no more than time polynomial in the size of the input instance of the 3-partition problem.

Next, we show that every feasible solution s'' to the reduced instance $\langle s, \mathcal{C}_{\sum} \rangle$ of the SNP - $MS_{\mathcal{Q}}$ problem is such that (i) $\mathcal{C}_T(s'') = \mathcal{C}_T$, (ii) s'' contains exactly $3m - 1$ base Ts, and (iii) $d_H(s, s'') \ge 2m$. For each compomer $G_{a_i}T_0 \in \mathcal{C}_T \subseteq \mathcal{C}_T(s'')$, there exists at least one cleavage fragment G^{a_i} in s'' that is obtained with respect to the cut base T. Since all the integers a_i are distinct, all such cleavage fragments shall be pairwise non-overlapping. Thus, the string s'' contains at least $\sum_{i=1}^{n} a_i = mB$ base Gs and at least $n - 1 = 3m - 1$ base Ts. On the other hand, since $|s| = mB + 3m - 1$, the string s'' hence consists of exactly $mB + 3m - 1$ bases. Therefore, we can deduce that s'' contains exactly $3m - 1$ base Ts and further that $\mathcal{C}_T(s'')$ cannot have any other compomer than those in C_T. By construction, we also know that the string s contains exactly $m - 1$ base Ts, which hence implies that $d_H(s, s'') \ge 2m$.

Now, we are going to show that there exists a valid partition for the input instance of the 3-partition problem if and only if there exists an optimal solution s' for the reduced instance of the SNP - $MS_{\mathcal{Q}}$ problem such that $d_H(s, s') = 2m$.

Suppose that $\mathcal{A}$ can be partitioned into m subsets $\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_m$ such that, for each subset $\mathcal{A}_i = \{a_{i_1}, a_{i_2}, a_{i_3}\}$, its size is three and its integer elements adds up to exactly B, that is, $|\mathcal{A}_i = 3|$and $\sum_{j=1}^{3} a_{i_j} = B, \forall 1 \leq i \leq m$. Then, we use the following procedure to find the string s':

1. $s' := \emptyset$;
2. **for** i = 1 to m
3. **for** j = 1 to 3
4. s'+ = $G^{a_{i_j}}T$; // append the string $G^{a_{i_j}}T$ to s'
5. **end**
6. **end**
7. s':= $s'[1, |s'| - 1]$; // remove the last base T

As one can easily check, the resulting string s' is such that $|s'| = mB + 3m - 1$, $\mathcal{C}_G \subseteq \mathcal{C}_G(s')$, and $\mathcal{C}_T \subseteq \mathcal{C}_T(s')$. Therefore, s' is a feasible solution to the reduced instance $\langle s, \mathcal{C}_\sum \rangle$ of the SNP - $MS_\mathcal{Q}$ problem. On the other hand, since $\sum_{j=1}^{3} a_{i_j} = B, \forall 1 \leq i \leq m$, we can deduce that $s'[k] = s[k]$ if $s'[k] = G$ or $s[k] = T$; otherwise, $s'[k] \neq s[k]$, $\forall k \in [1, mB + 3m - 1]$. Therefore, $d_H(s, s') = |[k : s'[k] \neq s[k]\}| = |s| - |\{k : s'[k] = s[k]\}| = mB + 3m - 1 - |\{k : s'[k] = G\}| - |\{k : s[k] = T\}| = mB + 3m - 1 - mB - m + 1 = 2m$. It hence follows that s' is indeed an optimal solution to the reduced instance $\langle s, \mathcal{C}_\sum \rangle$ of the SNP - $MS_\mathcal{Q}$ problem.

Conversely, suppose that the string s' is an optimal solution to the reduced instance $\langle s, \mathcal{C}_\sum \rangle$ of the SNP - $MS_\mathcal{Q}$ problem such that $d_H(s, s') = 2m$. Then, we use the following procedure to find a partition $\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_m$ of A:

1. $s := s \cdot T$; s':= $s' \cdot T$;
2. $i := 1$; $j := 1$;
3. $\mathcal{A}_i := \emptyset$; $a_{i_j} := 0$;
4. **for** k = 1 to $mB + 3m$
5. **if** $s'[k] = T$
6. $\mathcal{A}_i := \mathcal{A}_i \cup \{a_{i_j}\}$;
7. j + +;
8. **if** $s[k] = T$
9. i + +; $j := 1$;
10. $\mathcal{A}_i := \emptyset$;
11. **end**
12. $a_{i_j} := 0$;
13. **else**
14. a_{i_j} + +;
15. **end**
16. **end**

It follows from the earlier discussions that $\mathcal{C}_T(s') = \mathcal{C}_T = \{G_{a_i}T_0 : 1 \leq i \leq n\}$ and also that s' contains exactly $3m - 1$ base Ts. Furthermore, since $d_H(s, s') = 2m$, we can deduce that $s'[k] = s[k]$ if $s[k] = T$, $\forall k \in [1, mB + 3m - 1]$. Notice that $s[k] = T$ if and only if k can be written as a multiple of $(B + 3)$, that is, $k = i(B + 3) \in [1, mB + 3m - 1]$, $\forall i$. Therefore, $s'[k] = T$ if $k = i(B + 3) \in [1, mB + 3m - 1]$, $\forall i$, which subsequently implies that $\mathcal{C}_T(s'[(i-1)(B+3)+1, i(B+3)-1]) \subseteq \mathcal{C}_T(s')$, for each $i \in [1, m]$. Note that $s[(i - 1)(B + 3) + 1, i(B + 3) - 1]$ is a substring of s that consists of $(B + 2)$ base Gs; it is located either strictly between two consecutive base Ts or strictly between one base T and one end of the string s. Since $C_T(s'[(i - 1)(B + 3) + 1, i(B + 3) - 1]) \subseteq C_T(s')$, we can let $\mathcal{C}_T(s'[(i-1)(B+3)+1, i(B+3)-1]) = \{G_{a_{i_1}}T_0, G_{a_{i_2}}T_0, \ldots, G_{a_{i_j}}T_0\}$ such that $a_{i_1} + a_{i_2} + \cdots + a_{i_j} + j - 1 = B + 2$. Since $\frac{B}{4} < a_{i_j} < \frac{B}{2}$, we can deduce that $j = 3$; hence $a_{i_1} + a_{i_2} + a_{i_3} = B$. Let $\mathcal{A}_i = \{a_{i_1}, a_{i_2}, a_{i_3}\}$, for all $i \in [1, m]$. Then, we can see that $\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_m$ is a partition of $\mathcal{A}$ such that the sum of integers in each subset is equal to B.

Extensions to edit distance

Naturally we may extend our previous problem formulations to the edit distance (i.e., Levenshtein distance). The resulting two new problems are formally defined as follows.

Definition 15 (The $SNP{-}MS_\mathcal{P}$ problem) *Given a string s and a collection of compomer spectra $\mathcal{C}_\sum = \{\mathcal{C}_x : x \in \Sigma\}$, find a string s' such that $\mathcal{C}_x(s') \in \mathcal{C}_x$, for all $\times \in \Sigma$ and $d_E(s, s')$ is minimized.*

Definition 16 (The SNP - $MS_\mathcal{Q}$ problem) *Given a string s and a collection of compomer spectra $\mathcal{C}_\sum = \{\mathcal{C}_x : x \in \Sigma\}$, find a string s' such that $\mathcal{C}_x \subseteq \mathcal{C}_x(s')$, for all $x \in \sum$ and $d_E(s, s')$ is minimized.*

These extensions make it possible to detect not only base substitutions but also base insertions and deletions. Hence, they would permit the mutation discovery in DNA sequences (see [1]). In the Additional file 1, we show that both $SNP{-}MS_\mathcal{P}$ and SNP - $MS_\mathcal{Q}$ are theoretically NP-hard, together with an exact dynamic programming algorithm for solving the $SNP{-}MS_\mathcal{P}$ problem.

Conclusions

To exploit the full potential of the SNP discovery approach using base-specific cleavage and mass spectrometry, in this paper we have studied two new combinatorial optimization problems, called $SNP{-}MS_\mathcal{P}$ and SNP - $MS_\mathcal{Q}$, respectively. We believe that any efficient solution to either problem could offer a more seamless integration of information in four complementary base-specific reactions than previously done in [1,2], thereby improving the capability of the underlying biotechnology (i.e., base-specific cleavage and mass spectrometry) for sensitive and accurate SNP discovery.

Although we cannot change the inherent complexity of our proposed dynamic programming algorithm for the $SNP{-}MS_\mathcal{P}$ problem, we believe that by improving and optimizing its implementation, the compute

runtime can be significantly reduced to the extent suitable for practical use. On the other hand, the NP-hardness result indicates that in the most general situation, solving the SNP - $MS_{\mathcal{Q}}$ problem exactly in polynomial time is impossible unless P = NP. In more realistic situations where only a very few SNPs (e.g., two or three SNPs) occur in a target sample sequence, however, the problem can be quite easily tackled, e.g., using an exhaustive search approach. In the future work, we shall try to prove that the $SNP{-}MS_{\mathcal{P}}$ problem is NP-hard and develop an efficient heuristic algorithm for the SNP - $MS_{\mathcal{Q}}$ problem for practical use.

Additional material

Additional file 1: Extensions to edit distance. The analysis results for the problems SNP - $MS_{\mathcal{P}_e}$ and SNP - $MS_{\mathcal{Q}_e}$ are presented. See "Additional file 1.pdf".

Acknowledgements
We would like to thank Yuguang Mu and Kai Tang for introducing us the problem of SNP discovery using base-specific cleavage and mass spectrometry. X.C.'s research was supported by the Singapore National Medical Research Council grant (CBRG11nov091) and a College of Science Collaborative Research Award at NTU. Q.W.'s research was supported by National Science Foundation for Young Scientists of China (61103066). L.Z.'s research was supported by the Singapore MOE AcRF Tier 2 grant (R-146-000-134-112).
This article has been published as part of *BMC Systems Biology* Volume 6 Supplement 2, 2012: Proceedings of the 23rd International Conference on Genome Informatics (GIW 2012). The full contents of the supplement are available online at http://www.biomedcentral.com/bmcsystbiol/supplements/6/S2.

Author details
[1]School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore. [2]The Key Laboratory of Embedded System and Service Computing, Ministry of Education; Tongji University, Shanghai 200092, China. [3]Department of Mathematics, National University of Singapore, Singapore.

Authors' contributions
XC conceived the study. All authors contributed to the problem analysis, read and approved the final manuscript.

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

Published: 12 December 2012

References

1. Bocker S: **SNP and mutation discovery using base-specific cleavage and MALDI-TOF mass spectrometry.** *Bioinformatics* 2003, **19(Suppl 1)**:i44-53.
2. Krebs S, Medugorac I, Seichter D, Forster M: **RNaseCut: a MALDI mass spectrometry-based method for SNP discovery.** *Nucleic Acids Research* 2003, **31(7)**.
3. Stanssens P, Zabeau M, Meersseman G, Remes G, Gansemans Y, Storm N, Hartmer R, Honisch C, Rodi CP, Bocker S, van den Boom D: **High-throughput MALDI-TOF discovery of genomic sequence polymorphisms.** *Genome Research* 2004, **14**:126-133.
4. Hartmer R, Storm N, Bocker S, Rodi CP, Hillenkamp F, Jurinke C, van den Boom D: **RNase T1 mediated base-specific cleavage and MALDI-TOF MS for high-throughput comparative sequence analysis.** *Nucleic Acids Research* 2003, **31(9)**.
5. Honisch C, Raghunathan A, Cantor CR, Palsson BO, van den Boom D: **High-throughput mutation detection underlying adaptive evolution of Escherichia coli-K12.** *Genome Research* 2004, **14(12)**:2495-2502.
6. **RNaseCut webpage link.** [http://www.vetmed.uni-muenchen.de/gen/forschung.html].
7. Garey MR, Johnson DS: **Complexity results for multiprocessor scheduling under resource constraints.** *Siam Journal on Computing* 1975, **4**:397-411.
8. Hulett H, Will TG, Woeginger GJ: **Multigraph realizations of degree sequences: Maximization is easy, minimization is hard.** *Operations Research Letters* 2008, **36(5)**:594-596.
9. Bocker S: **Sequencing from compomers: Using mass spectrometry for DNA de novo sequencing of 200+ nt.** *Journal of Computational Biology* 2004, **11(6)**:1110-1134.
10. Pevzner PA, Tang HX, Waterman MS: **An Eulerian path approach to DNA fragment assembly.** *Proceedings of the National Academy of Sciences of the United States of America* 2001, **98(17)**:9748-9753.

doi:10.1186/1752-0509-6-S2-S5
Cite this article as: Chen *et al.*: Two combinatorial optimization problems for SNP discovery using base-specific cleavage and mass spectrometry. *BMC Systems Biology* 2012 **6**(Suppl 2):S5.