SIRIUS: decomposing isotope patterns for metabolite identification†

Sebastian Böcker1, Matthias C. Letzel2, Zsuzsanna Lipták3 and Anton Pervukhin1,*

1Lehrstuhl für Bioinformatik, Friedrich-Schiller-Universität Jena, 07743 Jena, 2Organische Chemie I, Fakultät für Chemie, Universität Bielefeld, 33501 Bielefeld and 3AG Genominformatik, Technische Fakultät, Universität Bielefeld, 33501 Bielefeld, Germany

Received on August 28, 2008; revised on November 13, 2008; accepted on November 14, 2008

Advance Access publication November 17, 2008

Associate Editor: Olga Troyanskaya

1INTRODUCTION

High-resolution mass spectrometry (MS) allows determining the mass of sample molecules with very high accuracy (1–5 p.p.m.), and has become one preferred method of analyzing metabolites. The output of a mass spectrometer, after preprocessing, consists of peaks that ideally correspond to the masses of the sample molecules and their abundance. This brings into play the natural isotopic distributions of the elements: several peaks in the output correspond to the same type of sample molecule, reflecting its isotope pattern. In this article, we use this isotope pattern to identify the sample molecule by determining its molecular formula or sum formula, i.e. the number of atoms of each element.

The term “metabolite” is usually restricted to small molecules that are intermediates and products of the metabolism. These small molecules participate in almost all cellular processes such as signal transduction, stress response, catabolism, or anabolism. Today, databases mostly contain primary metabolites directly relevant for growth, development, and reproduction of a cell or an organism. In contrast, most of the metabolites not directly involved in the aforementioned functions are yet uncharacterized. The majority of known metabolites have mass <1000 Da: 96.5% of sum formulas in the KEGG COMPOUND database (Kanehisa et al., 2006) fall into this mass range. To identify a sample metabolite, its mass spectrum is compared to spectra in a reference database. This method is limited to identifying metabolites and chemical compounds that have been included in some reference mass spectra library. Hence, de novo interpretation of metabolite mass spectra is highly sought.

To resolve the sum formula, Kind and Fiehn (2006) suggest to proceed as follows: first, compute all sum formulas that share the monoisotopic mass with the input mass spectrum. For every such candidate molecule, simulate its isotope pattern, and match and rank it against the input isotope pattern. Several experimental studies using this setup have been reported in the literature, the most prominent by Iijima et al. (2008) who claim to have discovered almost 500 novel metabolites in Tomato (Solanum lycopersicum). Zhang et al. (2005) address a related problem for analyzing peptides, but this approach heavily builds on several ad hoc rules regarding admissible sum formulas and uses a heuristic search. Böcker and Rasche (2008) present a method for the automated interpretation of metabolite tandem mass spectra but ignore isotope patterns.

In this article, we present efficient algorithms for all steps of the analysis pipeline suggested above. We limit our presentation to the elements most abundant in living beings, but note that our methods also work for other sets of elements. We first show how to use integer decomposition techniques (Böcker and Lipták, 2007)
for decomposing real-valued masses, with large improvements over naïve approaches. Second, we present a method for rapid computation of isotope distributions and mean masses of isotope peaks, improving on previously best known results (Rockwood et al., 2004). Fast simulation of isotope patterns is vital due to the large search space. Third, we show how to rapidly match and rank such simulated spectra against the measured spectrum. We then report on the application of our method to high-resolution mass spectra. Finally, we present the software tool SIRIUS (Sum formula Identification by Ranking Isotope patterns Using mass Spectrometry) that implements all of the above algorithms and combines them with an easy-to-use graphical user interface.

2 ISOTOPES AND ISOTOPE PATTERNS

The elements most abundant in living beings are hydrogen (H), carbon (C), nitrogen (N), oxygen (O), phosphor (P) and sulfur (S). Atoms with the same number of protons but different number of neutrons are called isotopes of the element. Each of these isotopes occurs in nature with a certain abundance, and we limit our attention to these naturally occurring isotopes. The superscript preceding a symbol denotes the mass number of the atom: the number of protons plus the number of neutrons. The mass of an atom is measured in Dalton (Da). An atom’s mass is roughly but not exactly equal to its mass number, the difference being due to the binding energy in the atom’s nucleus. The masses of the different isotopes and their abundance are known up to very high precision (Audi et al., 2003, 2004). Fast simulation of isotope patterns is vital due to the resulting distribution as the molecule’s isotopic distribution.

Table 1. Natural isotopic distribution: relative abundance of isotopes and their masses in Dalton

| Element | Isotope | mass | Mass difference | Abundance (%) |
|---------|---------|------|----------------|---------------|
| Hydrogen | $^1$H | 1.007825 | +0.006277 | 99.985 |
|         | $^2$H | 2.014102 |         | 0.015 |
| Carbon  | $^{12}$C | 12.0 | | 98.890 |
|         | $^{13}$C | 13.003355 | +0.003355 | 1.110 |
| Nitrogen | $^{14}$N | 14.003074 |         | 99.634 |
|         | $^{15}$N | 15.000109 | +0.997035 | 0.366 |
| Oxygen  | $^{16}$O | 15.994915 | | 99.762 |
|         | $^{17}$O | 16.999132 | +1.004217 | 0.038 |
|         | $^{18}$O | 17.999161 | +2.004246 | 0.200 |
| Phosphor| $^{31}$P | 30.973762 |         | 100 |
|         | $^{32}$S | 31.972071 | | 95.020 |
|         | $^{33}$S | 32.971459 | +0.999388 | 0.750 |
|         | $^{34}$S | 33.967867 | +1.995796 | 4.210 |
|         | $^{36}$S | 35.967081 | +3.999510 | 0.020 |

For each element $E$ we define two discrete random variables, denoted $X_E$ and $Y_E$, representing the mass and the mass number, respectively. For example, $X_C$ with state space $\{12, 13\}$, $Y_C$ with state space $\{12, 13\}$ and $\mathbb{P}(X_C = 12) = \mathbb{P}(Y_C = 12) = 0.98890$, $\mathbb{P}(X_C = 13) = \mathbb{P}(Y_C = 13) = 0.01110$ are the random variables of carbon. Given a molecule consisting of $l$ atoms, we assign to the $l$-th atom, $i = 1, \ldots, l$, two random variables $X_i$ and $Y_i$, where $X_i \sim X_E$ and $Y_i \sim Y_E$, with $E$ being the corresponding element. Now we can represent the molecule’s mass distribution by the random variable $X := X_1 + \cdots + X_l$, and its nominal mass distribution, or isotopic distribution, by $Y := Y_1 + \cdots + Y_l$. In an ideal mass spectrum, normalized peak intensities correspond to the isotopic distribution of the molecule. Note that $X$ and $Y$ are correlated, since $X_E$ can be viewed as a function of $Y_E$ and $E$.

We refer to the peak at the monoisotopic mass as the monoisotopic peak, which is followed by the $+1$, $+2$, $\ldots$, peaks. What is the mass of the $+k$ peak, which is a superposition of several isotopoe species? It is reasonable to assume that its mass is the mean mass of all isotope species that add to its intensity (Rockwood et al., 2004): for a molecule with monoisotopic nominal mass $N$, let $X = X_1 + \cdots + X_l$ be the mass distribution and $Y = Y_1 + \cdots + Y_l$ be the isotopic distribution. The mean peak mass of the $+k$ peak is then $m_k = E(X | Y = N + k)$. We refer to the isotopic distribution together with the mean peak masses as the molecule’s isotopic pattern.

3 METHODS AND ALGORITHMS

3.1 Real-valued decompositions

We first concentrate on the problem of decomposing the monoisotopic mass $M_0$. We want to find all molecules with monoisotopic mass in the interval $[l, u] \subseteq \mathbb{R}$ where $l := M_0 - \epsilon$ and $u := M_0 + \epsilon$ for some measurement inaccuracy $\epsilon$. Formally, we search for all solutions of the integer knapsack equation (Kellerer et al., 2004)

$$a_1 c_1 + a_2 c_2 + \cdots + a_n c_n \in [l, u],$$

where $a_i, j = 1, \ldots, n$, are real-valued monoisotopic masses of elements satisfying $a_i > 0$. We search for all solution vectors $c = (c_1, \ldots, c_n)$ such that all $c_j$ are non-negative integers. We may assume $a_1 < a_2 < \cdots < a_n$. A straightforward solution is to enumerate all vectors $c$ with $c_1 = 1$ and $\sum a_i c_j \leq u$ and next to test if there is some $c_1 \geq 0$ such that $\sum a_i c_j \in [l, u]$. 

\[\text{Table 1. Natural isotopic distribution: relative abundance of isotopes and their masses in Dalton}\]

\[
\begin{array}{|c|c|c|c|}
\hline
\text{Element} & \text{Isotope} & \text{mass} & \text{Mass difference} & \text{Abundance (%)} \\
\hline
\text{Hydrogen} & ^1\text{H} & 1.007825 & & 99.985 \\
 & ^2\text{H} & 2.014102 & +0.006277 & 0.015 \\
\text{Carbon} & ^{12}\text{C} & 12.0 & & 98.890 \\
 & ^{13}\text{C} & 13.003355 & +0.003355 & 1.110 \\
\text{Nitrogen} & ^{14}\text{N} & 14.003074 & & 99.634 \\
 & ^{15}\text{N} & 15.000109 & +0.997035 & 0.366 \\
\text{Oxygen} & ^{16}\text{O} & 15.994915 & & 99.762 \\
 & ^{17}\text{O} & 16.999132 & +1.004217 & 0.038 \\
 & ^{18}\text{O} & 17.999161 & +2.004246 & 0.200 \\
\text{Phosphor} & ^{31}\text{P} & 30.973762 & & 100 \\
 & ^{32}\text{S} & 31.972071 & & 95.020 \\
 & ^{33}\text{S} & 32.971459 & +0.999388 & 0.750 \\
 & ^{34}\text{S} & 33.967867 & +1.995796 & 4.210 \\
 & ^{36}\text{S} & 35.967081 & +3.999510 & 0.020 \\
\hline
\end{array}
\]
This results in $\Theta(M_0n!)$ running time, for constant element masses. Alternatively, we can compute all potential decompositions up to some upper bound $U$ during preprocessing, sort them with respect to mass and use binary search; this results in $\Theta(U^n)$ space requirement. These approaches are unfavorable in theoretical complexity as well as in practice: for the elements CHNOPS there exist more than $7 \times 10^9$ molecular formulas with mass $\leq 1500 \text{Da}$.

In the remainder of this section, we transform the integer knapsack problem with real-valued coefficients into a problem instance with integer coefficients. We will show in the next section how to efficiently solve such instances. Choosing a blowup factor $b \in \mathbb{R}$, corresponding to precision $1/b$, we can round coefficients by $\phi(x):=\lfloor bx \rfloor$, so $a'_j:=\phi(a_j)$ and $l' := \phi(l)$, $u' := \phi(u)$ form an integer knapsack. Precision $1/b$ is merely a parameter of the decomposition algorithm and in principle independent of the measurement mass accuracy $\epsilon$. To avoid rounding error accumulation, precision is usually set one to two orders of magnitude smaller than the measurement accuracy.

Now, certain solutions $c$ of the integer coefficient knapsack are no solutions of the real-valued coefficient knapsack and vice versa. We can easily sort out false positive solutions by checking (1), resulting in additional running time. We now concentrate on the more intriguing problem of false negative solutions and thus have to refine (1) into a more discriminative criterion.

One can easily check that this bound is tight. So, we re-define the integer interval by $u' := [bu + \Delta u]$. Without rounding correction we have to decompose $(u-l)b$ integers, but rounding correction forces us to decompose an additional $\Delta u$ integers, independent of the interval size $u-l$. As an example, consider the elements CHNOPS and blowup factor $b=10^2$, then $\Delta(b)=\Delta(l)=0.492936$. So for $M_0=1000$, we have to decompose an additional 492 integers. Clearly, increasing $b$ usually decreases $\Delta(b)$. We stress that the running time of this approach is dominated by the number of decompositions of these integers, and not by the number of integers itself.

### 3.2 Integer decompositions

Assume that both the element masses $a_1, \ldots, a_n$ and the query mass $m$ are positive integers. We are looking for all non-negative integer vectors $(c_1, \ldots , c_n)$ satisfying (1) with $l=u=m$. This is a well-studied problem, referred to in its different variants as Coin Change Problem, Change Making Problem or Money Changing Problem, and can be solved with a simple dynamic programming algorithm in pseudo-polynomial time (Martello and Toth, 1990). The main disadvantage of this approach is rather large memory requirement, which again depends on the maximal mass $U$ we want to decompose.

Böcker and Lipták (2007) present an algorithm for determining all such decompositions with running time $O(m \cdot \gamma(m))$ and space $O(na_1)$, where $a_1$ is the smallest mass and $\gamma(m)$ the number of decompositions of mass $m$. We briefly sketch the algorithm. Given integer masses $a_1\leq \ldots \leq a_n$, a data structure of size $na_1$, referred to as Extended Residue Table (ER table), is computed in a preprocessing step. Entry $ER(r,i)$, for $r=0, \ldots, a_i-1$ and $i=1, \ldots, n$, is the smallest number congruent $r$ modulo $a_i$ which is decomposable over $a_1, \ldots, a_i$. Thus, the last column $ER(n, n)$ of the table gives, for each residue $r$, the smallest number congruent $r$ modulo $a_1$ that is decomposable over $a_1, \ldots, a_n$. Computation time is $O(na_1)$. All decompositions of the query $m$ are then recursively generated, limiting the number of unsuccessful paths by using information from the ER table. As a result, the running time of the algorithm is proportional only to the size of the table $na_1$ and the number of decompositions $\gamma(m)$, and does not depend directly on the input $m$. For decomposing molecule masses, this decomposition technique has several advantages over classical dynamic programming, such as improved running times and preprocessing independent of the largest mass we want to decompose in the future. Regarding the application of decomposing molecule masses, this approach uses only one fifteenth of memory and shows better running times.

The number of decompositions $\gamma(m)$ for an integer mass $m$ over coprime integers $a_1, \ldots, a_n$ asymptotically behaves like a polynomial of degree $n-1$ in $m$ (Wilf, 1990). Following Beck et al. (2001), we can approximate the number of molecules over the elements CHNOPS with real mass in the interval $[M, M+\epsilon]$ by

$$\hat{\gamma}(M, \epsilon) \approx 3.10657 \cdot 10^{-9} \epsilon M^3 + 8.22867 \cdot 10^{-7} \epsilon M^4 + 8.05088 \cdot 10^{-5} \epsilon M^3.$$  \hfill (2)

We can also approximate the number of molecules with mass up to $M$ by integrating (2). In Figure 1, we plot the number of decompositions for masses of up to $1500 \text{Da}$ over the elements CHNOPS.

### 3.3 Simulating isotope patterns

We first observe that for elements CHNOPS, all molecules have isotopic distributions that decrease rapidly with increasing mass. In particular, we can restrict ourselves to computing the first $K$ non-zero values of the distribution, for rather small $K$ such as $K=10$. For example, amongst 11 479 entries in the KEGG COMPOUND database with mass $\leq 3000 \text{Da}$, no molecule has intensity of the +10 peak larger than 0.00007.

The atoms hydrogen, carbon and nitrogen have only two natural isotopes. Thus, the isotopic distribution of a molecule $E_i$ consisting of $l$ identical atoms of type $E \in \{\text{H, C, N}\}$ follows a binomial distribution: let $q_k$ denote the probability that $E_i$ has nominal mass $N+k$, where $N$ is the monoisotopic nominal mass of $E_i$. Then, $q_k = \binom{l}{k} p^k (1-p)^{l-k}$, where $p$ is the probability of the monoisotopic isotope. The values of the $q_k$ can be computed iteratively, since $q_{k+1} = \frac{1}{1+p} q_k$ for $k \geq 0$, thus computation time is $O(K)$.

Where an element $E$ has $r \geq 2$ isotopes (such as oxygen and sulfur), the isotopic distribution of $E_i$ can in theory be computed as follows: let $p_i$ for...
\[ i = 0, \ldots, r - 1 \] denote the probability of occurrence of the \( i \)-th isotope. Then, the probability that \( E_{i} \) has nominal mass \( N + k \) is \( \sum_{l_{0}, l_{1}, \ldots, l_{r-1} \geq 0} (l_{i} + 1) \prod_{i=0}^{r-1} l_{i}! \), where the sum runs over all \( l_{0}, l_{1}, \ldots, l_{r-1} \geq 0 \) satisfying \( \sum_{i=0}^{r-1} i \cdot l_{i} = k \) and \( \prod_{i=0}^{r-1} i! \).

Given two discrete random variables \( Y \) and \( Y' \) with state spaces \( \Omega, \Omega' \subseteq \mathbb{N} \), we can compute the distribution of the random variable \( Z := Y + Y' \) by folding the distributions,

\[ P(Z = N) = \sum_{k} P(Y = k) \cdot P(Y' = N - k). \]

If we restrict ourselves to the first \( K \) values of this distribution, we can compute it in time \( O(K^2) \). Kubinyi (1991) suggests to compute the isotopic distributions of oxygen \( O \) and sulfur \( S \) by successive folding of the respective distribution: using a Russian multiplication scheme for the folding, this results in an algorithm with running time \( O(K^2 \log l) \). In applications, we do not compute these distributions on the fly but during preprocessing, for all \( l \leq L \) fixed. This results in \( O(KL) \) memory for every such element, where \( L \) is small: The 128 oxygen atoms already have mass of about 2048 Da, exceeding the relevant mass range. For molecules consisting of different elements, we first compute or look up the isotopic distributions of the individual elements, and then combine these distributions by folding in \( O(n^2 K^2) \) time.

Using Fourier transforms of isotope distributions, we can multiply Fourier transforms instead of folding these distributions (Rockwood and Van Orden, 1996). Doing so we can eventually replace the \( K^2 \) factor in the algorithm’s running time by a \( K \log K \) factor. As we limit our attention to small \( K \) such as \( K = 10 \), this will not result in a speedup of the algorithm in practice. Also, this approach may face the problem of numerical errors.

We now come to the more challenging problem of efficiently computing the mean peak masses of a distribution. Doing so using the definition \( m_{n} = E(X | Y = N + k) \) is highly inefficient, because we have to sum up over all isotope species. Pruning strategies have been developed to speed up computation (Yergey, 1983), but pruning leads to a loss of accuracy (Rockwood et al., 2004). We now present a simple recurrence for computing these masses analogous to the folding of distributions: let \( Y = Y_1 + \ldots + Y_r \) and \( Y' = Y'_1 + \ldots + Y'_r \) be isotopic distributions of two molecules with monoisotopic nominal masses \( N \) and \( N' \), respectively. Let \( p_{k} := P(Y = N + k) \) and \( q_{k} := P(Y' = N' + k) \) denote the corresponding probabilities, \( m_{n} \) and \( m'_{n} \) the peak masses of the \( +k \) peaks. Consider the random variable \( Z = Y + Y' \) with monoisotopic nominal mass \( N = N' + N'' \).

**Theorem 1.** The mean peak mass \( \bar{m}_{n} \) of the \( +k \) peak of the random variable \( Z = Y + Y' \) can be computed as:

\[ \bar{m}_{n} = \frac{1}{\sum_{j=0}^{N} P(Y = N + k)} \sum_{j=0}^{N} p_{k-j} \left( m_{n} + m'_{n+j} \right). \]

Note that \( \sum_{j=0}^{N} p_{k-j} := P(Z = N + k) \). Since by independence, \( P(Y_{1} = N_{1}, \ldots, Y_{r} = N_{r}) = \prod_{i=1}^{r} P(Y_{i} = N_{i}) \), the theorem follows by rearranging summands. We omit the formal proof.

The theorem allows us to ‘fold’ mean peak masses of two distributions to compute the mean peak masses of their sum. This implies that we can compute mean peak masses as efficiently as the distribution itself. This improves on the previously best known method (Rockwood et al., 2004), replacing the linear running time dependence on the number of atoms by its logarithm.

### 3.4 Scoring candidate molecules

We want to discriminate between (tens of thousands of) candidate molecules generated by decomposing the monoisotopic mass. To this end, we compare the simulated isotopic distribution with the measured peaks. Matching peak pairs between the spectra is trivial for this application.

Zhang and Chait (2000) and Zhang et al. (2002) suggest to use Bayesian Statistics to evaluate mass spectra matches:

\[ \mathbb{P}(M_{i} | D, B) = \frac{\mathbb{P}(M_{i} | B) \cdot \mathbb{P}(D | M_{i}, B)}{\sum_{M_{j} | B} \mathbb{P}(M_{j} | B) \cdot \mathbb{P}(D | M_{j}, B)} \]

where \( D \) is the data (the measured spectrum), \( M_{i} \) are the models (the candidate molecules) and \( B \) stands for any prior background information.

Regarding this background information, we set the prior probability \( \mathbb{P}(M_{i} | B) \) to zero for all molecules but the decompositions of the monoisotopic mass. We also assign prior probability zero to molecular formulas that cannot correspond to a molecule, because of chemical considerations: Senior’s third theorem states that the sum of valences has to be greater than or equal to twice the number of atoms minus one (Senior, 1951). Molecules violating Senior’s third theorem are rare, particularly for natural compounds: in the KEGG COMPOUND database, <0.16% of substances violate this rule. We also filter out radicals with odd sum of valences. We refrain from using further priors such as the hetero-to-carbon ratio (King and Fiehn, 2007) because this might rather reproduce what is already known.

Next, we assign probabilities to the observed masses and intensities. Assuming independence (in particular from background information) we calculate

\[ \mathbb{P}(D | M_{i}, B) = \prod_{j} \mathbb{P}(M_{i} | m_{j}) \prod_{j} P(f_j | p_j). \]

Here, \( \mathbb{P}(M_{i} | m_{j}) \) is the probability to observe peak \( j \) at mass \( M_{j} \) when its true mass is \( m_{j} \), and \( P(f_j | p_j) \) is the probability to observe peak \( j \) with intensity \( f_j \) when its true intensity is \( p_j \). Clearly, the independence of peak intensities is violated because these intensities sum to one, but this product can be seen as a rough estimate of the true probability.

Mass spectrometrists assume that the mass error of a device is roughly normally distributed with mean zero. If the mass accuracy \( \alpha \) of the measurement (in p.p.m.) is given, then we can set the standard deviation \( \frac{1}{2} \times 10^{-6} M_{j} \) for peak \( j \), assuming that >99.7% of measurements fall into the specified mass range. But we also observe that peaks of low intensity show less mass accuracy than those with high intensity, which can be attributed to the difficulties of separating a peak of low intensity from the background noise. Our data indicate a roughly linear dependence between peak intensity and mass accuracy. To this end, we use two mass accuracies \( \alpha_1 \) (at full intensity) and \( \alpha_0 \) (at minimal intensity) are provided by the user, and we set

\[ \sigma_{j} = \frac{\left( p_{j} \alpha_{1} + (1 - p_{j}) \alpha_{0} \right) \cdot 10^{-6} M_{j}}{2}. \]

We want to estimate the probability that, given a peak with true mass \( m_{j} \), a peak at mass \( M_{j} \) is observed in the measured spectrum: more precisely, the probability of observing a mass difference of \( |M_{j} - m_{j}| \) or larger. We can compute this probability using the complementary error function ‘erfc’:

\[ \mathbb{P}(M_{j} | m_{j}) = \text{erfc} \left( \frac{|M_{j} - m_{j}|}{\sqrt{2} \sigma_{j}} \right) = \frac{2}{\sqrt{\pi}} \int_{z \over \sigma_{j}}^{\infty} e^{-t^2} dt \]

with \( z = |M_{j} - m_{j}| \). \( \sigma_{j} \).

Even for high-resolution MS, spectra show a systematic mass shift due to calibration inaccuracies. We can easily eliminate this shift for all masses but the monoisotopic mass: we do not compare masses of the +1, . . . peaks directly but instead the difference to the monoisotopic peak, \( M_{j} - M_{0} \). We refer to subsequent re-normalization. We found that for both of our datasets, the same parameter \( \alpha = +0.02 \) leads to excellent results.

Computation of \( P(f_{j} | p_{j}) \) is done analogously to that of \( \mathbb{P}(M_{i} | m_{j}) \). Our data indicates that after correction, log ratios between measured and predicted peak intensity \( \log(f_{j} | p_{j}) \) roughly follow a normal distribution.
Again, precision parameters $\beta_1$ (at full intensity) and $\beta_0$ (at minimal intensity) are provided by the user (in percent). We compute $$\delta_j := \frac{1}{2} \log \left( 1 + \frac{p_j \beta_0}{\beta_1} + (1 - p_j) \frac{\beta_0}{\beta_1} \right)$$ as our precision interval, such that >99.7% of log intensity ratios $\log(f_j/p_j)$ fall into the range $[-\delta_j, +\delta_j]$. Now, $5f_j/p_j$ can be estimated analogously to (6).

4 EXPERIMENTAL RESULTS

4.1 Datasets

To evaluate our method we used two datasets measured on two instruments. Mass spectra with single charge were measured from several organic (macro)molecules, composed of the elements CHNOPS. For every such spectrum, the sum formula of the sample molecule is known. The spectra were acquired over a period of 2 years; the molecules range in mass from 117 Da to ~1000 Da. Peak detection and estimation of peak masses and intensities were conducted using vendor software.

The first dataset consists of 153 mass spectra. Electrospray ionization (ESI) experiments were performed using the Fourier Transform Ion Cyclotron Resonance (FT-ICR) mass spectrometer APEX III (Bruker Daltonik GmbH, Bremen, Germany). The FT-MS was equipped with a 7.0 T, 160 mm bore superconducting magnet, infinity cell and interfaced to an external (nano)ESI ion source. All mass spectra were externally mass calibrated. The five analysis parameters were chosen as $\alpha_1 = 3$, $\alpha_0 = 6$, $\beta_1 = 10$, $\beta_0 = 90$ and $off = +0.02$.

The second dataset consists of 86 mass spectra. ESI experiments were performed using the oa-TOF mass spectrometer MicrOTOF (Bruker Daltonik GmbH, Bremen, Germany). Quasi-internal mass calibration was used, by measurement of an infused calibrant prior to the compound of interest. For the oa-TOF analysis, we set $\alpha_1 = 5$, $\alpha_0 = 6.5$, $\beta_1 = 10$, $\beta_0 = 90$ and again $off = +0.02$.

4.2 Identification accuracy

Every input ‘mass spectrum’ consists of masses $M_0, \ldots, M_b$ and intensities $f_0, \ldots, f_b$. For every such spectrum, we computed all molecules such that the monoisotopic mass $m_0$ has relative mass difference of at most $\alpha_1$ p.p.m. Next, we discarded molecules violating Senior’s third theorem and radicals with odd sum of valences. For each remaining molecule, we computed its theoretical isotopic distribution (with $K = 10$) and compared it to the measured isotopic distribution. We ranked the molecules according to resulting probabilities. We did not use any other background information to identify the molecule.

For the 153 mass spectra in our FT-ICR dataset, 89 resulted in a correct identification; in 86% of the mass spectra, the correct interpretation was found in the TOP 10 explanations. There is a clear correlation between mass and identification accuracy, confer Table 2. For mass spectra $\leq$700 Da, the true interpretation was always found in the TOP 10 explanations, except in one case where it had rank 13. For 86 mass spectra in the oa-TOF dataset, the correct sum formula was found in the TOP 10 interpretations in all but two cases. Moreover, 79 out of 86 compounds were correctly identified, which correspond to an identification rate of >90%, (Table 2). Better identification results on the oa-TOF dataset with lower mass accuracy show the crucial importance of including intensity measurements into the candidate evaluation. We note that the intensity accuracy of the oa-TOF instrument is significantly higher than that of the FT-ICR. We have also tested the variation of identification rates with different scoring parameters: identification results are relatively stable for small disturbances of parameter values, see supplementary material. Parameter estimation could be automated using a small training set. We are planning to include this feature in future implementations.

4.3 Running times

We analyzed all 239 mass spectra on a Pentium M 1.5 GHz processor with blowup $b = 5 \times 10^4$, using only a few Megabyte of memory. This results in running times of <1.3 s per spectrum for the complete analysis of one mass spectrum. Clearly, running times depend on molecule masses, see again Table 2. Increasing the blowup beyond $5 \times 10^4$ increased running times, presumably because the smaller table can be kept in the processor cache.

5 IMPLEMENTATION

We have developed a java-based graphical tool called SIRIUS. At the SIRIUS core lie efficient algorithms for generating all elemental compositions for a given mass and error, calculating isotope patterns for all chemically relevant compositions, and matching and ranking the candidate molecules against the input spectrum. SIRIUS combines these algorithms with a powerful graphical user interface. An extensive management system allows simplified data handling and offers an easy way to integrate new
algorithms and data structures into the framework. Through a user-friendly interface, SIRIUS allows the user to import datasets in most common mass spectrometry file formats. It supports automatic recognition of molecular ion adducts present in the input spectrum, handy visualization of identified sum formulas and their isotope patterns and customizable export of identification results to common human-readable file formats. Finally, the software provides a basic functionality to search for sum formulas identified by the algorithm in NCBI PubChem Database.

Preparation of a new analysis run can be divided into the following steps: initializing input data and instrument parameters, setting up algorithm parameters and extracting isotope patterns from the input peaklist. Input peaklist and machine settings can be reused for multiple analyses on the same data. SIRIUS provides the user with reasonable default values for algorithm parameters. The program also offers to save all algorithm and mass spectrometer settings. To this end, SIRIUS creates a persistent workspace that can be used to store local settings and to automatically reload them on request.

We use the ProteomeCommons.org IO Framework (Falkner et al., 2007) to import mass spectrometry data, which allows reading most MS data formats including mzData and mzXML. We parse the peaklist and divide it into signal groups related to different compounds. A peaklist can also contain several signal groups belonging to the same compound, modified by different molecular ion adducts. Identifying modifications is done by calculating mass differences between monoisotopic peak masses. In view of the small number of adducts, we apply a simple exhaustive search to find all matching mass differences. If there is no prior knowledge on the source of modification, the user can choose one or more adduct types for an isotope pattern.

The output of the algorithm is a list of candidate sum formulas for each compound. Sum formulas are listed in the summary table, sorted in decreasing order of likelihoods. To view an entry in more detail, the user can select and compare theoretical and measured isotope patterns visually. Analysis results can be exported to the application workspace and opened for further evaluation. Export file formats include plain text, PDF and XML documents.

6 CONCLUSION
We presented an approach to determine the sum formula of an unknown metabolite solely from its high-resolution isotope pattern. Our approach allows us to reduce the number of potential sum formulas to only a few candidates; in many cases we were able to determine the correct molecular formula. The approach is time- and memory-efficient and can be executed on a regular desktop PC. We further presented methods for the efficient simulation of isotope patterns. This is vital for larger molecules where the search space increases rapidly.

Results on experimental data clearly show the potential of our approach, in particular for oa-TOF data. In our evaluation, we have deliberately ignored some information such as prior probability of the elements or hetero-to-carbon ratio (Kind and Fiehn, 2007). We believe that such information should rather be used in a ‘post-processing’ step by an expert, instead of automatically filtering out certain sum formulas a priori. Finally, we introduced a user-friendly

1http://pubchem.ncbi.nlm.nih.gov/
software called SIRIUS, which implements all of the methods presented.

ACKNOWLEDGEMENTS
Additional programming by Martin Engler. We thank Dr H. Luftmann, Universität Münster, Organisch-Chemisches Institut, for making available the oa-TOF dataset and an anonymous referee for helpful comments.

Funding: Deutsche Forschungsgemeinschaft (BO 1910/1 to A.P.); Alexander von Humboldt Foundation and the Bundesministerium für Bildung und Forschung, within the group ‘Combinatorial Search Algorithms in Bioinformatics’ (to Z.L.).

Conflict of Interest: none declared.

REFERENCES
Audi,G. et al. (2003) The AME2003 atomic mass evaluation (ii): Tables, graphs, and references. Nucl. Phys. A, 729, 129–336.
Beck,M. et al. (2001) The polynomial part of a restricted partition function related to the frobenius problem. Electron. J. Comb., 8, N7.
Böcker,S. and Lipták,Z. (2007) A fast and simple algorithm for the Money Changing Problem. Algorithmica, 48, 413–432.
Böcker,S. and Rasche,F. (2008) Towards de novo identification of metabolites by analyzing tandem mass spectra. Bioinformatics, 24, 149–155.
Falkner,J.A. et al. (2007) Proteomecommons.org io framework: reading and writing multiple proteomics data formats. Bioinformatics, 23, 262–263.
Hsu,C.S. (1984) Diophantine approach to isotopic abundance calculations. Anal. Chem., 56, 1356–1361.
Iijima,Y. et al. (2008) Metabolite annotations based on the integration of mass spectral information. Plant J., 54, 949–962.
Kanehisa,M. et al. (2006) From genomics to chemical genomics: new developments in KEGG. Nucl. Acids Res., 34, D354–D357.
Kellerer,H. et al. (2004) Knapsack Problems. Springer, Berlin, Heidelberg.
Kind,T. and Fiehn,O. (2006) Metabolomic database annotations via query of elemental compositions: mass accuracy is insufficient even at less than 1 ppm. BMC Bioinformatics, 7, 234.
Kind,T. and Fiehn,O. (2007) Seven golden rules for heuristic filtering of molecular formulas obtained by accurate mass spectrometry. BMC Bioinformatics, 8, 105.
Kubinyi,H. (1991) Calculation of isotope distributions in mass spectrometry: a trivial solution for a non-trivial problem. Anal. Chim. Acta, 247, 107–119.
Martello,S. and Toth,P. (1990) Knapsack Problems: Algorithms and Computer Implementations. John Wiley & Sons, Chichester.
Rockwood,A.L. and Van Orden,S.L. (1996) Ultrahigh-speed calculation of isotope distributions. Anal. Chem., 68, 2027–2030.
Rockwood,A.L. et al. (2004) Isotopic compositions and accurate masses of single isotopic peaks. J. Am. Soc. Mass Spectr., 15, 12–21.
Senior,J. (1951) Partitions and their representative graphs. Am. J. Math., 73, 663–689.
Wilf,H. (1990) Generating functionology. Academic Press, New York.
Yergey,J.A. (1983) A general approach to calculating isotopic distributions for mass spectrometry. Int. J. Mass Spectrom. Ion Phys., 52, 337–349.
Zhang,J. et al. (2005) Predicting molecular formulas of fragment ions with isotope patterns in tandem mass spectra. IEEE/ACM Trans. Comput. Biol. Bioinform., 2, 217–230.
Zhang,N. et al. (2002) ProbID: a probabilistic algorithm to identify peptides through sequence database searching using tandem mass spectral data. Proteomics, 2, 1406–1412.
Zhang,W. and Chait,B.T. (2000) ProFound: an expert system for protein identification using mass spectrometric peptide mapping information. Anal. Chem., 72, 2482–2489.