IsoSpec : hyperfast fine structure calculator

IsoSpec: Hyperfast Fine Structure Calculator

Mateusz K.Łącki,[*,†,‖] Michał Startek ,[†,‖] Dirk Valkenborg,[‡,¶,§] and Anna Gambin[†]

[†]*Department of Mathematics, Informatics, and Mechanics, University of Warsaw, 02-097 Warsaw, Poland.*

[‡]*Center for Proteomics, University of Antwerp, 2000 Antwerp, Belgium*

[¶]*Flemish Institute for Technological Research (VITO), 2400 Mol, Belgium*

[§]*Interuniversity Institute for Biostatistics and Statistical Bioinformatics, Hasselt University, 3500 Hasselt, Belgium*

[‖]Contributed equally to this work

E-mail: mateusz.lacki@biol.uw.edu.pl

Abstract

As high-resolution mass spectrometry (HRMS) becomes increasingly available, the need of software tools capable of handling more complex data is surging. The complexity of the HRMS data stems partly from the presence of isotopes that give rise to more peaks to interpret compared to lower resolution instruments.

However, a new generation of fine isotope calculators is on the rise. They calculate the smallest possible sets of isotopologues. However, none of these calculators lets the user specify the joint probability of the revealed envelope in advance. Instead, the user must provide a lower limit on the probability of isotopologues of interest, i.e. provide minimal *peak height*. The choice of such threshold is far from obvious. In particular, it is impossible to *a priori* balance the tradeoff between the algorithm speed and the portion of the revealed theoretical spectrum. We show that this leads to considerable inefficiencies.

Here, we present IsoSpec: an algorithm for fast computation of isotopologues of chemical substances that can alternate between joint probability and peak height threshold. We prove that IsoSpec is optimal in terms of time complexity. Its implementation is freely available under a 2-clause BSD license, with bindings for C++, C, R, and Python.

Introduction

Until fairly recently, detection of the fine structure isotopic distribution was generally beyond the capability of any mass spectrometer. However, as both FT-ICR MS and Orbitrap instruments continue to be improved, obtaining higher resolution and sensitivity, the detection of fine structure is becoming routine.[1–3] As much as 20M FWHM has already been recorded.[4] The rise of high-resolution (HRMS) and high-throughput mass spectrometry leads to more informative data providing valuable insights into, e.g., molecular identity. Experiments confirm superior identification powers of HRMS, enabling, for instance, correct recognition of metabolites[5] and lipids.[6]

However, more information is more data to analyze: a low resolution full scan mass spectrum of a single molecule consists of only a few peaks, where each peak counts ions that have roughly the same nominal mass. HRMS can resolve these clusters of ions into finer ones. Ideally, with high enough resolution, one could resolve individual isotopologues[7], i.e. molecules with the same isotopic composition. For instance, using HRMS one can discern water isotopologues $HD^{16}O$ and $H_2{}^{17}O$, both with a nominal mass equal to 19 Da. In consequence, more peaks need to be interpreted.

Regardless of the resolution reached by modern instruments and its theoretical limits resulting from thermodynamics[8], it is instructive to consider the unrealizable case of infinite resolution. In such a setting, the full isotopic distribution of Bovine Insulin, $C_{254}H_{377}N_{65}O_{75}S_{6}$, would be composed of more than 1.5 trillion different isotopologues. This number can be massively reduced if one introduces the probabilistic concept of the chance of finding a given type of isotopologue. Assuming statistical independence of the isotopic variants of atoms[9], 414 configurations are enough to represent around 99% of the overall probability. This phenomenon is known as probability *measure concentration.*[10]

Related Research. To bypass the problem of the rapid increase in the number of isotopologues traditional approaches to isotope calculations have mostly assumed nominal mass approximation[11–14], binning isotopologues with the same mass number; see Valkenborg *et al.*[15] In this approach isotopologues with the same nominal mass are indistinguishable: the theoretical distribution is centroided so that highly resolved peaks are represented together with their mass averaged out. The Fourier transform method proposed by Rockwood[16] exempts this rule: it relies on probing the Fourier transform of the mass distribution and offers, in principle, extremely high levels of resolutions. Still, one cannot expect to know *a priori* where to probe the transform and has to resolve to a meticulous search over a grid of mass values, which raises the task's computational complexity.

Recently, the interest shifted towards direct calculation of fine isotopic peaks, giving rise to elegant algorithms, such as ECIPEX[17] or enviPat.[18] ECIPEX generalizes the Fourier transform approach investigated by Rockwood to higher dimension. enviPat has recently bested ECIPEX in terms of runtime, which can be attributed to direct inspection of the problem on the level of counts of isotopes and by performing pruning of the so called *transition trees.* Both approaches do harness the probability *measure concentration* we exposed on the Bovine Insulin example. However, they specify their outcome in terms of heights of the reported peaks. For instance, they let one neglect all peaks below a given percentage of the highest peak, which is a heuristics first developed by Yergey.[19] A different approach to fine structure calculations, presented by Li[20], does not present such a disadvantage and the user can specify some joint probability p of the fine structure to be revealed. However, the output of that approach might not be the smallest possible set of isotopologues that is p probable. Together these peaks might be jointly p probable, but there are smaller sets of peaks with this quality.

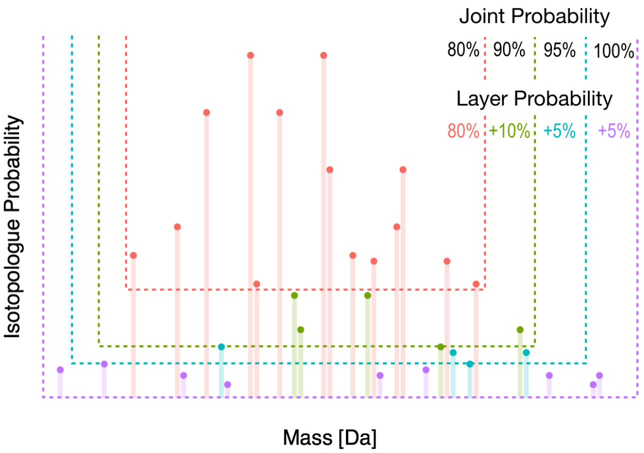


Figure 1: Division of isotopic envelope into optimal p-sets, $p \in \{80\%, 90\%, 95\%, 100\%\}$, for a toy molecule. Red peaks correspond to the smallest set of isotopologues that is at least 80% probable; in green we show the minimal additional *layer* of peaks that together with all previous ones are at least 90% probable; in cyan – 95%, in violet - 100%. IsoSpec finds minimal sets with a given joint probability without requiring a threshold on peak height, i.e. without a cut-off on the y-axis.

To our best knowledge, the question of how the choice of the threshold relates to the joint probability of the envelope has not yet been investigated. As demonstrated in Fig. S.4 in Supporting Information, this relation is far from trivial, potentially leading to calculations involving isotopologues that are altogether not so important for the analysis. In the case of Bovine Insulin, the smallest set that is 99.99% probable contains 6196 isotopologues in addition to the 414 contained in the smallest 99.9% probable set. On average, these 6196 isotopologues will amount to one per mille of all of the observed ions, making it impractical to consider them. The effect of *overrepresenting an improbable set* is more pronounced for bigger compounds, especially with many atoms of elements that have more than one abundant isotope, such as selenium or sulfur. This under-lines the role of precision in the choice of proper pruning threshold.

Our Approach. In this paper we study in detail the relationship between the threshold and the joint probability. We present an algorithm for retrieving the smallest possible set of isotopologues with a given probability that the user wishes to unveil. Our algorithm bridges the apparent gap between algorithms such as enviPat or ecipex and the recursive approach developed by Li.[20] In contrast to many other approaches, we also analyze the computational complexity of the presented solutions. We prove that our algorithm is optimal in terms of time complexity. Finally, we present an implementation of IsoSpec that is superior to the fastest fine structure calculator to date, enviPat, as tested on a set of more than 800,000 chemical formulas obtained by *in silico* fragmentation of 1000 human proteins.

The infinitely resolved spectrum can comprise thousands of peaks for just one molecule. One could doubt the usefulness of this concept arguing that this is experimentally unachievable. However, isotopologues can be aggregated based on the similarity of their masses so as to match the resolution of the used instrument, see Li.[20] Our approach guarantees that this can be achieved quickly and with control over the error of the approximation.

In the rest of the article we describe the theoretical gains from any strategy resulting in optimal pruning. Then, we describe the IsoSpec algorithm. Finally, we compare its runtime with the enviPat algorithm. In our presentation we focus on proteins; however, the implementation and the analysis both apply to any known compounds, even those containing other elements than carbon, hydrogen, nitrogen, oxygen, and sulfur.

The Complexity of Pruning

Consider a protein with c atoms of carbon, h hydrogen, n nitrogen, o oxygen, and s sulfur, $C_cH_hO_oN_nS_s$. Denote by $\mathcal{E}$ the set of the chemical elements the protein is composed of and by n_e the number of atoms of a given element e composing the protein, i.e. $n_e \in \{\mathrm{c,h,n,o,s}\}$. Finally, denote by i_e the number of stable isotopes of that element. The total number of different isotopic compositions assumed by $C_cH_hO_oN_nS_s$, i.e. the total number of *isotopologues*[7], equals $\prod_{e\in\mathcal{E}}\binom{n_e+i_e-1}{n_e}$, which is asymptotically polynomial in the numbers of atoms, $\mathcal{O}(\prod_{e\in\mathcal{E}} n_e^{i_e-1})$, see SI, Section 4. Section 1 provides an example deciphering the above notation. Carbon, nitrogen and hydrogen have two stable isotopes each, resulting roughly in a linear increase in isotopologues with the number of atoms of these elements. With respectively three and four stable isotopes the relation for oxygen becomes quadratic, and cubic for sulfur. This quantifies the extent of *combinatorial explosion* of the direct enumeration of all isotopologues. We would like to avoid finding unlikely isotopologues. Assuming that the isotopic variants of atoms composing $C_cH_hO_oN_nS_s$ are independent and drawn with the same abundances across elements[9], one pinpoints the probability of an isotopologue to be a product of multinomial distributions, equal to

$$\prod_{e\in\mathcal{E}} \binom{n_e}{n_{e0},\dots,n_{e,i_e-1}} p_{e,0}^{n_{e,0}} \cdots p_{e,i_e-1}^{n_{e,i_e-1}}, \tag{1}$$

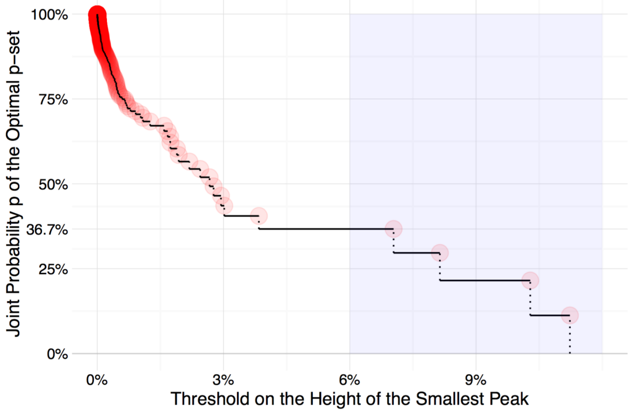


Figure 2: The *threshold function* obtained for Bovine Insulin. The function relates the choice of peak height threshold τ with the joint probability p of the resulting set of isotopologues, i.e. the ones with peak height at least τ. It usually happens that there is no peak with height exactly τ: the *effective* configuration (in red) is then to be found to the right on the same level. Trimming peaks less than 6%-probable (height below 0.06) one gets a set of 4 isotopologues (red dots on the blue background) with joint probability 36.7%. Higher intensity of red in top-left corner indicates that lower thresholds rapidly increase the number of resulting isotopologues.

and mass to $\sum_{e\in\mathcal{E}}\sum_{i=0}^{i_e-1} m_{e,i}n_{e,i}$, where n_{ej} is the count of element e's j^{th} isotope, and p_{ej} and m_{ej} are respectively its abundance and mass in daltons, both reported by IUPAC.[21] With Eq. (1) at hand, it is natural to search for sets of isotopologues that jointly surpass some limiting value of probability that is close to 100%, say p. Many such sets exist, so it seems reasonable to limit one's attention to the smallest one. We call such a set an optimal p-set – Fig. 1 explores that concept; see also SI, Section 2 for a discussion on uniqueness of the optimal p-set.

Observe that the optimal p-sets in Fig. 1 are separated by horizontal dashed lines up to configurations with the same probability. To obtain an optimal p-set one can choose a threshold on peak height and then discard some of the low probable peaks of the same height, see SI, Section 3. Usually there is only one peak with minimal height, so that the output of both the envi<span style="font-variant:small-caps"></span>Pat and ecipex algorithms coincides with an optimal p-set, for some joint probability p. However, to get p one has to establish a set of isotopologues first.

The relationship between the input threshold and the joint probability of the output p is presented in Fig. 2 on the example of Bovine Insulin. The resulting *threshold function* is locally flat, non-increasing, and right-continuous. The input threshold will usually be smaller than the actual minimal probability observed in the output p-set: we call isotopologues with that probability *effective.* They are depicted as red, semi-transparent circles in Fig. 2, and correspond to right ends of the intervals that make up the curve. High concentration of the *effective isotopologues* in the top left region suggests high sensitivity of the number of configurations in the optimal p-set to the choice of the input threshold. The idea behind the IsoSpec algorithm is to reach the input joint probability p by moving along the graph of the *threshold function*, from bottom-right to upper-left.

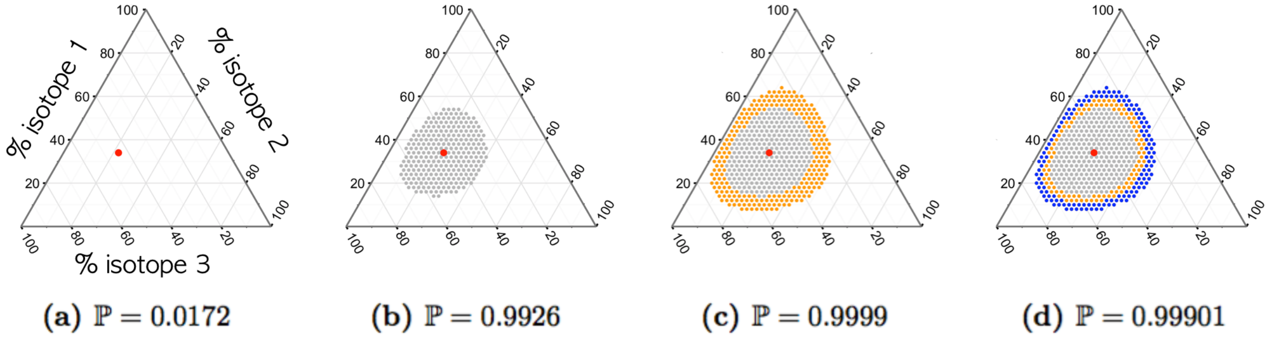
Before describing in detail the IsoSpec algorithm, let us briefly elaborate on the potential gains resulting from peak height thresholding. An isotopologue of $\mathrm{C_cH_hO_oN_nS_s}$ can be fully described by the numbers of isotopes of different elements that compose it, called *subisotopologues*[18], as in

$$\overbrace{{}^{12}\mathrm{C}_{\mathrm{c}_0}\;{}^{13}\mathrm{C}_{\mathrm{c}_1}}^{\text{subisotopologue}}\;\overbrace{{}^{1}\mathrm{H}_{\mathrm{h}_0}\;{}^{2}\mathrm{H}_{\mathrm{h}_1}}^{\text{subisotopologue}}\;\overbrace{{}^{14}\mathrm{N}_{\mathrm{n}_0}\;{}^{15}\mathrm{N}_{\mathrm{n}_1}}^{\text{subisotopologue}}$$
$$\underbrace{{}^{16}\mathrm{O}_{\mathrm{o}_0}\;{}^{17}\mathrm{O}_{\mathrm{o}_1}\;{}^{18}\mathrm{O}_{\mathrm{o}_2}}_{\text{subisotopologue}}\;\underbrace{{}^{32}\mathrm{S}_{\mathrm{s}_0}\;{}^{33}\mathrm{S}_{\mathrm{s}_1}\;{}^{34}\mathrm{S}_{\mathrm{s}_2}\;{}^{36}\mathrm{S}_{\mathrm{s}_3}}_{\text{subisotopologue}}.$$

A subisotopologue corresponding to element e can be thus represented as a tuple $\boldsymbol{n}_e = (n_{e,0},\ldots,n_{e,i_e-1})$ of specific isotope counts, where $\sum_{j=0}^{i_e-1} n_{e,j} = n_e$. The inspection of the probability of an isotopologue described by equation (1) further reveals that each multinomial distribution present in the product corresponds to the probability of exactly one subisotopologue. If e has three isotopes, then one can depict subisotopologues on a ternary plot, as in any subplot of Fig. 3. In general, subisotopologues constitute a discrete grid on a structure called *simplex.* With a growing number of atoms of each element in a chemical compound, the multinomial distributions in Equation (1) can be individually approximated by multivariate Gaussian distributions with the same mean and covariance matrix, see SI, Section 6. The asymptotical behavior results from the *Central Limit Theorem.* Considered together, these form yet a higher dimensional multivariate Gaussian distribution, with a mean μ and covariance matrix Σ, specified in SI. That normal distribution approximates the product of multinomials. By inspecting its smallest p-probable sets we can asymptotically investigate the behavior of the corresponding optimal p-set.

It is well known that ellipsoid of radius R defined by inequality $(x-\mu)^t\Sigma^{-1}(x-\mu) \leq R^2$ constitutes the smallest region with a given probability for the Gaussian distribution. Moreover, its probability can be easily obtained by evaluating the cumulative χ^2 distribution function with $k = \sum_{e\in\mathcal{E}} i_e - |\mathcal{E}|$ degrees of freedom in point R^2, where $|\mathcal{E}|$ is the number of elements in the compound. The number of isotopologues in the optimal p-set is asymptotically proportional to the volume of a p-probable ellipsoid, which is in turn proportional to

$$q_{\chi^2(k)}(p)^{\frac{k}{2}} \prod_{e\in\mathcal{E}} n_e^{\frac{i_e-1}{2}}, \tag{2}$$



(a) $\mathbb{P} = 0.0172$ **(b)** $\mathbb{P} = 0.9926$ **(c)** $\mathbb{P} = 0.9999$ **(d)** $\mathbb{P} = 0.99901$

Figure 3: The principle behind the IsoSpec algorithm. Consider a $n_e = 50$ atoms molecule made up entirely out of one fictitious element with three isotopes. The concepts of subistopologue and isotopologue coincide. Isotope content of isotopologues is represented as points in the above ternary plots. In general, isotopologues correspond to tuples of points on different simplices. To find the optimal 99.9%-set, one first establishes the most probable isotopologue, like in **(a)** in red. Then, one finds the first optimal p_1-set, as in **(b)** by choosing some threshold τ_1, like in **(b)** in grey. One then sums all peaks heights to see that $p_1 = 99.26\%$, smaller than 99.9%. One gets another threshold τ_2, establishes new layer of isotopologues, **(c)** in orange, and finds that $p_2 = 99.99\%$. This set is too big and one trims out the isotopologues in blue in **(d)**. Then, $p > 99.99\%$, but removing more isotopologues would bring joint probability below 99.99%.

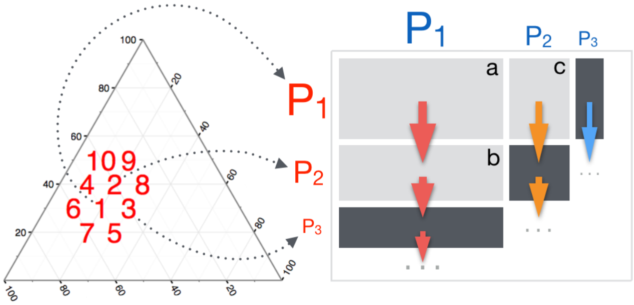


Figure 4: Merging subisotopologues into isotopologues on a toy example of a two element molecule. The lengths of the edges of rectangles correspond to probabilities of subisotopologues: these are decreasing for both the **red** and the **blue** element, and correspond to subisotopologues that concentrate around the most probable subisotopologue, as in the ternary plot. Isotopologues are visited lexicographically: first, one travels down the red pathway (column with rectangles **a** and **b**) till reaching a dark rectangle with area below threshold τ. Then, one travels down the orange pathway (column with rectangle **c**); and so on. Dark rectangles form the *fringe*: a set of neighbors of isotopologues more probable than τ. Having obtained another threshold $\upsilon < \tau$, one continues the lexicographic descent starting from the fringe until first isotopologues less probable than υ are reached, forming a new *fringe*.

where $q_{\chi^2(k)}(p)$ is the p-th quantile of the aforementioned χ^2-distribution, see SI, Section 6.

The overall number of isotopologues above a given probability behaves therefore asymptotically like $\mathcal{O}\left(\sqrt{\prod_{e\in\mathcal{E}} n_e^{i_e-1}}\right)$ – roughly a square root of the order of the total number of isotopologues obtained before. This both explains the sublinear growth of the optimal p-sets and provides asymptotic bounds on memory usage for any peak height trimming algorithm. Therefore, trimming truly effectively averts the *combinatorial explosion.*

The IsoSpec Algorithm

We will describe IsoSpec in four steps: how to generate subisotopologues; how to merge subisotopologues into sets of isotopologues above a given threshold; how to generate a sequence of consecutive thresholds; and how to trim the output into the final shape. The first two steps are interwoven and describe a fully operational *peak height trimming* algorithm we call IsoSpec Threshold. With these four procedures, IsoSpec first generates the top probable subisotopologues. Eq. (1) indicates that together they form the top probable isotopologue. Then, IsoSpec iteratively produces optimal p-sets of isotopologues, each corresponding to some threshold τ from the sequence of thresholds. Every time a p-set is obtained, its joint probability p is established and compared with the target value $\mathbb{P}$. This is repeated until p gets larger than $\mathbb{P}$. Finally, the last *layer* of peaks is trimmed leaving the required optimal $\mathbb{P}$-set. Fig. 3 visualizes this approach on a simplified molecule composed of exactly one element.

Calculating subsequent subisotopologues corresponds to reporting configurations of a given multinomial distribution with decreasing probability. This is easy thanks to its unimodality. To define what we mean by unimodality, we first relate subisotopologues of element e spatially: let two subisotopologues $\boldsymbol{n}_e^1$ and $\boldsymbol{n}_e^2$ be neighbors, $\boldsymbol{n}_e^1 \sim \boldsymbol{n}_e^2$, iff one is obtainable from the other by changing the isotopic variant of exactly one atom. For instance, ${}^{16}\mathrm{O}_3 \sim {}^{17}\mathrm{O}{}^{16}\mathrm{O}_2$ as one atom changed from ${}^{16}\mathrm{O}$ to ${}^{17}\mathrm{O}$. However, ${}^{16}\mathrm{O}_3 \not\sim {}^{17}\mathrm{O}_2{}^{16}\mathrm{O}$, as two atoms would have to change from ${}^{16}\mathrm{O}$ to ${}^{17}\mathrm{O}$. Two neighbors are also close on the simplex in the geometric sense, like dots in Fig. 3. A discrete distribution is unimodal, iff the set of global maxima is connected. Consequently, every configuration not top probable has an equally or more probable a neighbor. The multinomial distribution is unimodal in that sense.[22]

Unimodality simplifies the task of reporting subisotopologues sorted by decreasing probability for a given element e. Call such procedure a *subgenerator*; see SI, Algorithm 1. A *subgenerator* starts from top probable subisotopologue. It gets there by a simple *hill climbing* algorithm: it starts with a subisotopologue close to the mean of the multinomial distribution and follows the direction of increasing probability until the maximum is reached. By unimodality, it must be a global one. It then enlists it in an empty priority queue PQ, with priorities set to probabilities of subisotopologues; check SI Table S3 for PQ properties. Then, it iteratively extracts the top probable element from PQ and inserts its yet unvisited neighbors. By unimodality one can only insert subisotopologues less probable than those popped out. Each configuration has a limited number of neighbors, so the size of PQ is of the order of the number of already visited subisotopologues, n. Using the standard heap implementation of the PQ, calculations involving n configurations take up $O(n \log(n))$ time.

We store the results of previous calls as well as the state of the subgenerator to avoid unnecessary recomputations. This way the retrieval of the already calculated probability, e.g. while passing from red pathway to orange pathway in Fig. 4, can be done faster. Multiple visits to subisotopologues can be avoided through hashing. The computational complexity of operations on subisotopologues is negligible compared to subisotopologue merger.

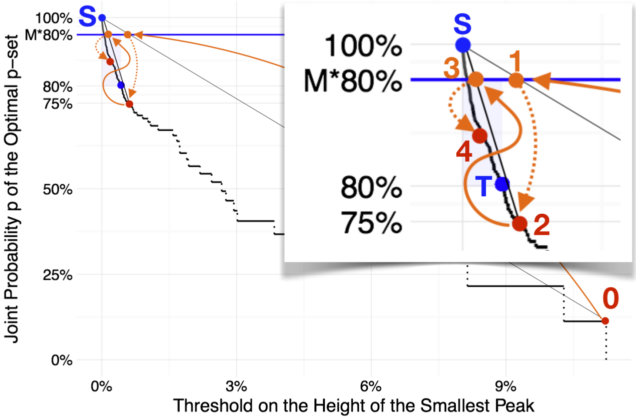


Figure 5: Adaptive linear approximation to the *threshold* function. It starts at point (P,P) – the top probable isotopologue, **0**, and aims at finding the optimal 80%-set, point **T**. Point **1** on line **0-S** is where we would get if our approximation using multiplier M was perfect. Instead, it leads to only 75% of the joint probability, as indicated by point **2**. Line **2-S** provides another approximation, and suggests point **3**. In reality, we move to **4** – already above the target 80%. The *effective isotopologues* on the *threshold function* between points **4** and **T** can to be trimmed.

A *subgenerator* provides the k-th most probable subisotopologue and its probability. To get an isotopologue, one considers a tuple of $|\mathcal{E}|$ different subisotopologues, each obtained with a different *subgenerator*. The probability of an isotopologue is the product of probabilities of its constituent subisotopologues. IsoSpec uses a series of thresholds to obtain *layers* of isotopologues. It starts by merging top probable subisotopologues. Given any isotopologue γ, it uses *subgenerators* to establish its less probable neighbors, the *successors*. A *successor* of γ has precisely one subisotopologue changed to the next one in line. For instance, isotopologues **b** and **c** are successors of **a** in Fig. 4. To generate isotopologues above a threshold τ consists in inserting and popping elements from a queue, see Algorithm 2 in SI for details. In comparison to *subgenerator*, sorting elements is redundant, and so a priority queue can be replaced with a simple FIFO queue, see SI for definition. To avoid repeated visits to the same configurations, IsoSpec follows a lexicographic visiting schedule, as shown by colored arrows in Fig. 4. Each *popped out* isotopologue qualifies to a given *layer* if its probability is above τ. Otherwise, it is stored in a so-called *fringe*, and used in the next iteration with a new threshold. The procedure is repeated until the joint probability exceeds that required by the user.

Successive threshold values result from an adaptive linear approximation to the *threshold function*, see Fig. 5. Given the top probable isotopologue with probability P we can draw a

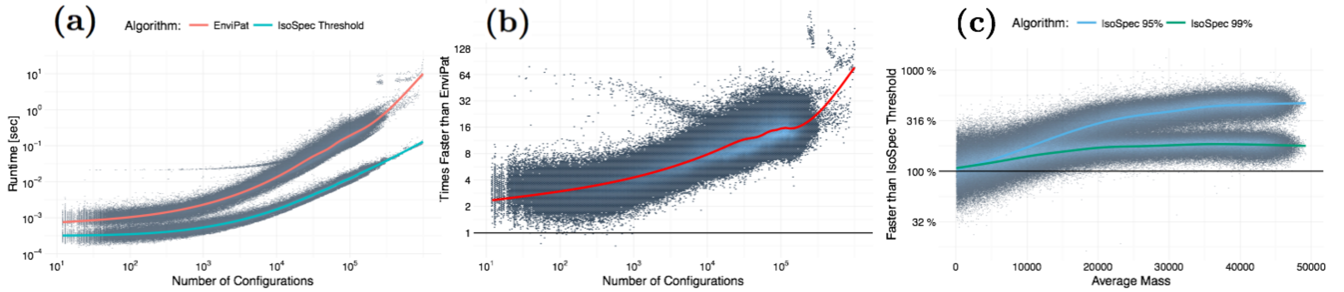


Figure 6: Comparison of envіPat and IsoSpec Threshold **(a,b)** and of IsoSpec Threshold with IsoSpec calculating the optimal 99% and 95% sets. Absolute peak height threshold was set to equal one ten-thousandth of the highest peak height (enviPat default). Fig. **(a)** shows the absolute runtime as a function of the overall number of calculated configuration. In Fig. **(b)**, we express the relative runtime of enviPat in the runtime of IsoSpec Threshold, showing how much faster is our approach. Both axis in **(a,b)** are in logarithmic scales. Fig. **(c)** shows how much faster is the calculation of the optimal 99% and 95% sets (with IsoSpec) than obtaining the set of isotopologues more probable than 10^{-4} HP (with IsoSpec-threshold). In contrast to **(a)** and **(b)**, the abscissa states the average mass of a compound, as the number of configurations (isotopologues) is variable for the different sets. Smooth lines represent fitted polynomial trend lines in all plots. The analysis is based on 805,367 compounds.

line between point (P, P) and point $S = (0, 1)$. Point S lies on the *threshold function*, as the choice of a 0 threshold on peak height results in a full set of isotopologues, i.e. a 100% probable set. On that line we find a point slightly above the required value $\mathbb{P}$, say $M\mathbb{P}$ where $M > 1$ is chosen heuristically. The x coordinate of that point provides the first threshold, τ_1. Applying the previous procedure on τ_1 we get the optimal probability p_1. A new line is drawn between point S and (τ_1, p_1) and the procedure is iteratively repeated until $p_k > \mathbb{P}$, where k is the number of the last iteration. The slight overestimate is needed for the algorithm to converge.

Finally, the trimming of the last *layer* of isotopologues can be performed in a linear time with its size; see SI, Algorithm 3. Section 8 provides a proof of the algorithm's optimality.

Results

We perform runtime analysis on a set of more than 800 000 ions' formulas generated from a list of 1000 human proteins from Uniprot. This set of formulas contains 1000 precursors and all derived b and y ions. This computational experiment therefore simulates the spectra preparation step for a tandem MS database driven identification procedure.

Both enviPat and IsoSpec are implemented in C++, however while enviPat can only be called from R, IsoSpec can be called from C++, C, R and Python. We have used the Python interface in our simulations.

In Fig. 6 **(a,b)** we compare runtimes of enviPat and IsoSpec Threshold on individual fragments. Both tools aim at calculating the same set of isotopologues defined by a common threshold on peak height, equal to one ten-thousandth of the highest peak, 10^{-4}HP for short. Fig. 6 **(a)** reports absolute runtimes in seconds. Fig. 6 **(b)** expresses enviPat's runtime in that of IsoSpec Threshold to show directly how much faster is the latter, which is roughly 2 to more than 100 fold, the gap widening with the size of a molecule. The optimal 99% and 95% sets are always smaller than the set of isotopologues more probable than 10^{-4}HP and can be usually obtained faster using IsoSpec, as can be seen in Fig. 6 **(c)**. The advantage clearly increases with compound size; consider SI Section 6 and Fig. S3. for the asymptotic dependence between runtimes and the choice of the joint probability threshold. This opens way for various *rapid scan* procedures that could compare the actual spectrum with a relatively small optimal p-set to rule out that a given compound is there.

The overall time to compute spectra for a CID identification procedure for a given substance is the sum of runtimes needed to obtain the spectra of the precursor and all fragments. We report these total runtimes in Fig. 7, which simply aggregates information conveyed in Fig. 6. In particular, subfigure **(a)** confirms that a procedure based on IsoSpec will be at least an order of magnitude faster as compared to enviPat.

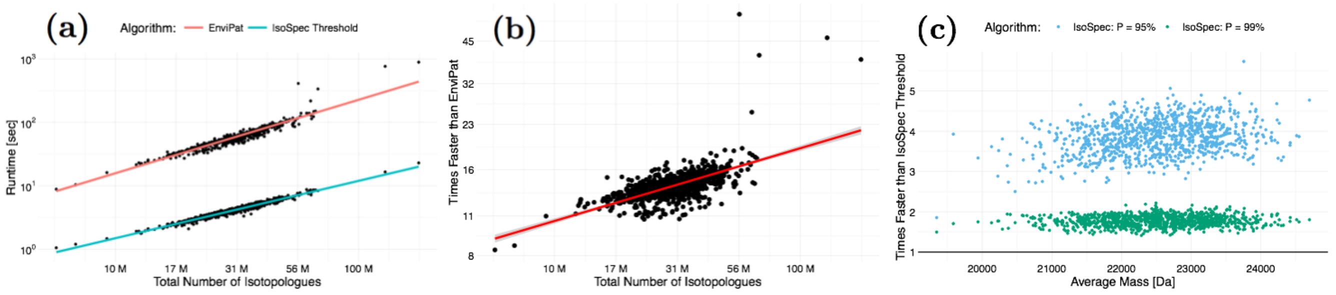


Figure 7: Comparison of ENVIPAT with ISOSPEC THRESHOLD **(a,b)** and ISOSPEC THRESHOLD with ISOSPEC aiming at joint probability equal to 99% and 95% **(c)** on *fragment identification* problem (1000 compounds). In **(a)** we see the absolute runtimes of ENVIPAT and ISOSPEC THRESHOLD: **(b)** specifies how much faster is the second approach in terms of the runtimes of the first one. On the x-axis of the **(a,b)** plots we show the total number of configurations generated in the tandem MS theoretical simulation for a given protein. Both axes are in logarithmic scales. In **(c)** one notices speedup resulting from a search for the optimal 95% and 99% sets (ISOSPEC) instead of the set of isotopologues more probable than 10^{-4} HP (ISOSPEC-THRESHOLD).

Concluding Remark

In this article we introduce the concept of the optimal P-set and show its relevance in the problem of simulating a theoretical infinitely resolved spectrum. This concept gives rise to an optimal algorithm, ISOSPEC, that efficiently uses available computational resources. The main strengths of the method are: (1) an increase in runtime speed in-between one and two orders of magnitude compared to other approaches (2) mathematically proven asymptotically optimal (linear) runtime of the presented method (3) the ability to solve the problem in terms of joint probability rather than the peak height threshold, if desired, (4) bindings for four mainstream computer languages.

ISOSPEC can be freely downloaded under a 2-clause BSD license from `http://matteolacki.github.io/IsoSpec/`.

Acknowledgement Work supported by Polish NCN grants 2014/12/W/ST5/00592, 2015/17/N/ST6/03565 and partially by Flemish SBO grant *InSPECtor*, 120025, IWT.

Supporting Information Available: This material is available free of charge via the Internet at `http://pubs.acs.org/`.

References

(1) Nikolaev, E. N.; Jertz, R.; Grigoryev, A.; Baykut, G. *Anal. Chem.* **2012**, *84*, 2275–2283.

(2) G. Marshall, A.; T. Blakney, G.; Chen, T.; K. Kaiser, N.; M. McKenna, A.; P. Rodgers, R.; M. Ruddy, B.; Xian, F. *Mass Spectrom.* **2013**, *2*, S0009.

(3) Michalski, A.; Damoc, E.; Lange, O.; Denisov, E.; Nolting, D.; Muller, M.; Viner, R.; Schwartz, J.; Remes, P.; Belford, M.; Dunyach, J.-J.; Cox, J.; Horning, S.; Mann, M.; Makarov, a. *Mol. Cell. Proteomics*

(4) Hendrickson, C. L.; Quinn, J. P.; Kaiser, N. K.; Smith, D. F.; Blakney, G. T.; Chen, T.; Marshall, A. G.; Weisbrod, C. R.; Beu, S. C. *J. Am. Soc. Mass Spectrom.* **2015**, *26*, 1626–1632.

(5) Nagao, T.; Yukihira, D.; Fujimura, Y.; Saito, K.; Takahashi, K.; Miura, D.; Wariishi, H. *Anal. Chim. Acta* **2014**, *813*, 70–76.

(6) Schwudke, D.; Schuhmann, K.; Herzog, R.; Bornstein, S. R.; Shevchenko, A. *Cold Spring Harbor Perspect. Biol.*

(7) McNaught, A. D.; Wilkinson, A. *IUPAC Gold Book*; Blackwell Scientific Publications: Oxford, 1997.

(8) Dittwald, P.; Valkenborg, D.; Claesen, J.; Rockwood, A. L.; Gambin, A. *J. Am. Soc. Mass Spectrom.* **2015**, *26*, 1732–1745.

(9) Kienitz, H. *Angew. Chem.* **1961**, *73*, 634.

(10) Talagrand, M. *Ann. Probab.* **1996**, *24*, 1–34.

(11) Rockwood, A. L. *Rapid Commun. Mass Spectrom.* **1995**, *9*, 103–105.

(12) Dittwald, P.; Claesen, J.; Burzykowski, T.; Valkenborg, D.; Gambin, A. *Anal. Chem.* **2013**, *85*, 1991–1994.

(13) Snider, R. K. *J. Am. Soc. Mass Spectrom.* **2007**, *18*, 1511–1515.

(14) Böcker, S.; Letzel, M. C.; Lipták, Z.; Pervukhin, A. *Bioinformatics* **2009**, *25*, 218–224.

(15) Valkenborg, D.; Mertens, I.; Lemière, F.; Witters, E.; Burzykowski, T. *Mass Spectrom. Rev.* **2012**, *31*, 96–109.

(16) Rockwood, A. L.; Van Orden, S. L.; Smith, R. D. *Rapid Commun. Mass Spectrom.* **1996**, *10*, 54–59.

(17) Ipsen, A. *Anal. Chem.* **2014**, *86*, 5316–5322.

(18) Loos, M.; Gerber, C.; Corona, F.; Hollender, J.; Singer, H. *Anal. Chem.* **2015**, *87*, 5738–5744.

(19) Yergey, J. a. *Int. J. Mass Spectrom. Ion Phys.* **1983**, *52*, 337–349.

(20) Li, L.; Murat Karabacak, N.; Cobb, J. S.; Wang, Q.; Hong, P.; Agar, J. N. *Rapid Commun. Mass Spectrom.* **2010**, *24*, 2689–2696.

(21) Brand, W. A.; Coplen, T. B.; Vogl, J.; Rosner, M.; Prohaska, T. *Pure Appl. Chem.* **2014**, *86*, 425–467.

(22) Finucan, H. M. *Biometrika* **1964**, *51*, 513–517.

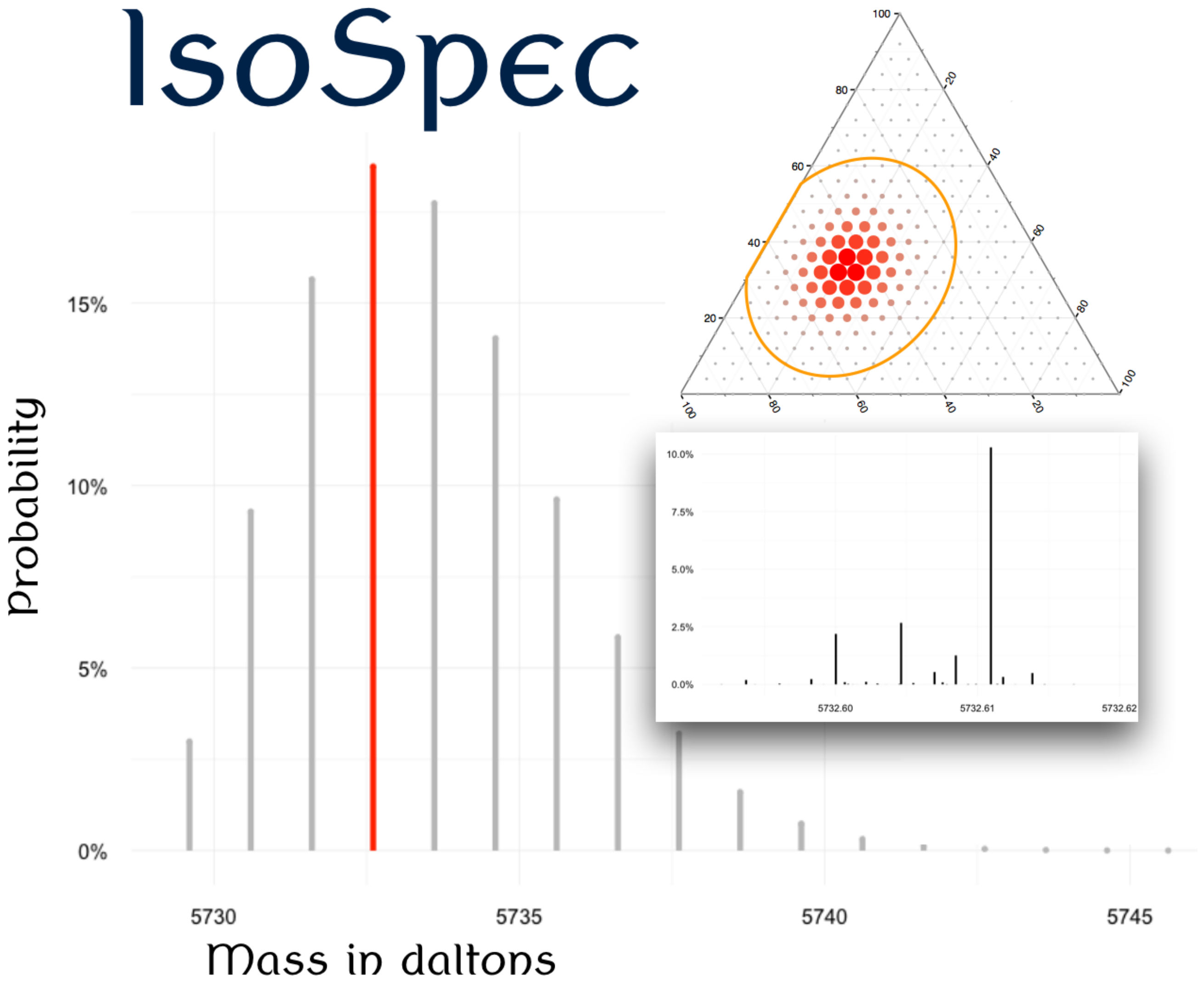


Figure 8: For TOC only.