Determination of Glycan Structure from Tandem Mass Spectra

Sebastian Böcker[1,2], Birte Kehr[1], and Florian Rasche[1]

[1] Lehrstuhl für Bioinformatik, Friedrich-Schiller-Universität Jena, 07743 Jena, Germany
{sebastian.boecker,florian.rasche}@uni-jena.de, BirteKehr@aol.com
[2] Jena Centre for Bioinformatics, Jena, Germany

Abstract. Glycans are molecules made from simple sugars that form complex tree structures. Glycans constitute one of the most important protein modifications, and identification of glycans remains a pressing problem in biology. Unfortunately, the structure of glycans is hard to predict from the genome sequence of an organism.

We consider the problem of deriving the topology of a glycan solely from tandem mass spectrometry data. We want to generate glycan tree candidates that sufficiently match the sample mass spectrum. Unfortunately, the resulting problem is known to be computationally hard. We present an efficient exact algorithm for this problem based on fixed-parameter algorithmics, that can process a spectrum in a matter of seconds. We also report some preliminary results of our method on experimental data. We show that our approach is fast enough in applications, and that we can reach very good *de novo* identification results. Finally, we show how to compute the number of glycan topologies of a given size.

1 Introduction

Glycans are – besides nucleic acids and proteins – the third major class of biopolymers, and are build from simple sugars. Since simple sugars can have up to four linkage sites, glycans are assembled in a tree-like structure. The elucidation of glycan structure remains one of the most challenging tasks in biochemistry, yet the proteomics field cannot be completely understood without these important post-translational modifications [1].

One of the most powerful tools for glycan structure elucidation is tandem mass spectrometry [5,15]. Mass spectrometry (MS) is a technology which, in essence, allows to determine the molecular mass of input molecules. Put in a simplified way, the input of the experiment is a molecular mixture and the output a peak list: a list of masses and their intensities. In tandem mass spectrometry, we select one type of molecules in the sample, fragment these parent molecules, and measure the masses of all fragments. Ideally, each peak should correspond to the mass of some sample molecule fragment, and its intensity to the frequency of that fragment in the mixture. The situation is, in fact, more blurred, due to noise and other factors. Tandem MS can provide general structural information

H.Q. Ngo (Ed.): COCOON 2009, LNCS 5609, pp. 258–267, 2009.

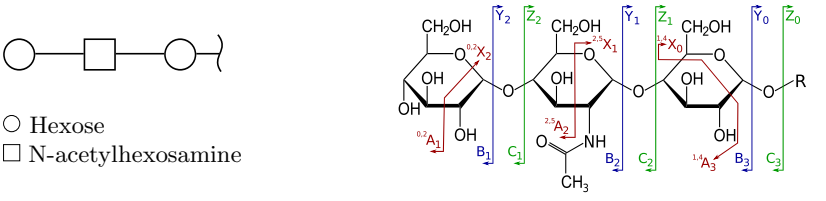

Fig. 1. Topology of a glycan made from three monosaccharides (left), and fragments resulting from tandem mass spectrometry analysis (right)

about the glycan, in particular its *topology* that can be represented as a labeled tree, see Fig. 1. Glycan mass spectra can be interpreted by searching a database of reference spectra [4], but such databases are vastly incomplete.

Recent approaches for *de novo* interpretation of tandem MS data usually build on two analysis steps, the first step being *candidate generation* (filtering) and the second step being *candidate evaluation*. A good candidate generation algorithm will generate a small set of candidates, but will not miss the correct interpretation. For glycans, a naïve approach to generate candidates is to decompose the parent mass of the glycan over the alphabet of monosaccharides [3], and then to enumerate all topologies that have the correct multiplicities of monosaccharides. Obviously, this is not feasible for large glycans.

Both candidate generation and candidate evaluation rely on certain scoring schemes, that are usually less sophisticated for candidate generation because of running time constraints: During candidate evaluation we only have to consider a small set of candidates. Recent approaches for tandem MS interpretation typically use scoring schemes that are elaborate modifications of the peak counting score, where one simply counts the number of peaks that are common to sample spectrum and candidate spectrum. Shan *et al.* [13] recently established that generating glycan topology candidates while avoiding peak double counts is an NP-hard problem. Existing approaches for glycan candidate generation can be subdivided into three categories: Some approaches enumerate all possible glycan topologies [7,8] and use strict biological rules to cut down on the number of candidates. Other tools use dynamic programming but simply ignore the problem of multiple peak counting [14]. Finally, Shan *et al.* [13] present a heuristic that avoids peak double counting.

Our contributions. We present a method that solves the candidate generation problem while at the same time, avoiding multiple peak counting. Although the corresponding problem is NP-hard, we present an exact method that allows to process a glycan tandem mass spectrum in a matter of seconds, and guarantees that all top-scoring candidate topologies are found. Our algorithm is fixed-parameter tractable [10] with respect to the parameter "number of peaks in the sample spectrum". We report some preliminary results on experimental data, showing that our candidate generation performs well in practice. We also show that solving the simpler candidate generation problem where one allows multiple peak counting, usually leads to poor results. Additionally, we present a recurrence for counting glycan topologies of a given size.

2 Preliminaries

For glycans, collision-induced dissociation (CID) is often used as fragmentation technique in the tandem MS experiment. There are three types of fragmentation that break the glycan topology, resulting in six types of ions [6], see Fig. 1: X, Y, and Z-ions correspond to fragments that contain the monosaccharide attached to the peptide (the glycan's reducing end) and are called *precursor ions* or *precursor fragments*. A, B, C-ions, in contrast, do not contain this monosaccharide. Using low collision energies in the fragmentation step, we predominantly generate B and Y ions, so we concentrate on these two types in our presentation.

Masses of molecules are measured in "Dalton" (Da), where 1 Dalton is approximately the mass of a neutron. We will often assume *integer masses* in our presentation for the sake of clarity. Accurate masses will be used in the scoring scheme.

We model a glycan topology as a rooted tree $T = (V, E)$, where the root is the monosaccharide attached to the peptide. Tree vertices are labeled with monosaccharides from a fixed alphabet Σ. Every vertex has an out-degree of at most four, because each monosaccharide has at most five linkages. Every element $g \in \Sigma$ and, hence, every vertex in the tree T is assigned an integer mass $\mu(g)$: this is the mass of the monosaccharide g, minus 18 Dalton for the mass of H_2O removed in binding. A fragment T' of T is a connected subtree, and the mass of T' is the sum of masses of the constituting vertices. Let $M := \mu(T)$ be the *parent mass* of the glycan structure. If we restrict ourselves to simple fragmentation events, then fragmentation of the tree means removing a single edge. Hence, we can represent each simple fragmentation event by a vertex $v \in V$, where the subtree $T(v)$ induced by v represents the non-precursor fragment, and the remainder of the tree is the precursor fragment. The resulting non-precursor fragments have the mass of a subtree of T induced by a vertex v, denoted $\mu(v)$. For precursor fragments we subtract $\mu(v)$ from the parent mass M. We ignore other mass modifications to simplify our presentation, such as the final H_2O, precursor fragment modifications through amino acids residues. These modifications can be easily incorporated into the presented methods.

Our method will take into account all possible glycan topologies, deliberately ignoring all biological restrictions. It is well known that certain branching types are observed seldom in biological samples: For example, most monosaccharides show only one to three linkages. But instead of completely forbidding such structures, we incorporate biological restrictions into our scoring model through penalties. In this way, we do not impede that rare structures can be found.

Our algorithm uses the concept of *fixed-parameter tractability* (FPT) [10]. This technique delivers exact solutions for an NP-hard problem in acceptable running time if the problem can be *parameterized*: In addition to the problem size n, we introduce a parameter k of the problem instance where typically $k \ll n$. A parameterized algorithm then restricts the exponential growth of its running time to the parameter k, whereas the running time is polynomial in n. Here, the problem size is the parent mass M whereas k is the number of (intense) peaks in the sample spectrum.

3 Candidate Generation

Assume we are given a candidate glycan tree T, and we want to evaluate T against the sample mass spectrum. We can use some simple fragmentation model to generate a hypothetical *candidate spectrum*, and use an additive scoring scheme to rate the candidate spectrum against the sample spectrum. Let $f(m)$ be the score we want to assign if a peak at mass m is present in our candidate spectrum. In its simplest incorporation, f is the characteristic function telling us if a peak is present in the *sample* mass spectrum at mass m. Then, summing $f(m)$ over all peak masses m in the candidate spectrum, we count all peaks that are common to both the sample spectrum and the candidate spectrum. We can also take into account expected peaks that are not present in the sample spectrum, by defining $f(m) = +1$ if a peak at mass m is present in the sample spectrum, and $f(m) = -1$ otherwise. Of course, for experimental data we use more involved scoring taking into account, say, peak intensities.

To simplify our presentation, let us assume for the moment that all our mass spectra consist of non-precursor ions only. Let $T = (V, E)$ be a labeled tree. Following Shan *et al.* [13] we define the scoring model $S'(T) := \sum_{v \in V} f(\mu(v))$. Unfortunately, scoring $S'(T)$ is not a peak counting score. Instead, for *every* subtree T' of T with mass $m' = \mu(T')$ we add $f(m')$ to the score: A tree that contains many subtrees of identical mass m' receives a high score if $f(m')$ is large even if it ignores all other peaks. We will show below how computations for this model can be transferred over to peak counting scores, though.

To find T that maximizes $S'(T)$, we define $S'[m]$ to be the maximal score of any labeled tree with total mass m. We use the simple recurrence from [13],

$$S'[m] = f(m) + \max_{m_1+m_2+m_3+m_4+\mu(g)=m} S'[m_1] + S'[m_2] + S'[m_3] + S'[m_4], \quad (1)$$

where the maximum is taken over all $g \in \Sigma$ and $0 \leq m_1 \leq m_2 \leq m_3 \leq m_4 \leq m$. We initialize $S'[0] = 0$. If one of the m_j in (1) equals zero, then the monosaccharide at the root of the subtree has less than four bonds. The maximal score of any glycan tree then is $S'[M]$, and we can backtrace through the array S' to find the optimal labeled tree. Shan *et al.* [13] simplify (1) to:

$$\begin{aligned} S'[m] &= f(m) + \max_{g \in \Sigma} \max_{m_1=0,\ldots,\lfloor \frac{m-\mu(g)}{2} \rfloor} S'_2[m_1] + S'_2\big[m - \mu(g) - m_1\big] \\ S'_2[m] &= \max_{m_1=0,\ldots,\lfloor \frac{m}{2} \rfloor} S'[m_1] + S'[m - m_1] \end{aligned} \quad (2)$$

The term $S'_2[m]$ corresponds to a "headless" subtree without a monosaccharide at its root. Using (2) we can compute $S'[M]$ in time $O(|\Sigma| \cdot M^2)$. Equations (1) and (2) can easily be modified to take into account properties of the monosaccharide g, such as the number of links of g for the scoring.

An exact algorithm for the peak counting problem. Recall that we actually want to compute the peak counting score $S(T) := \sum_{m=0,\ldots,M} f(m) \cdot g(m)$ where $g(m)$ is the characteristic function of the labeled tree T: If T contains one

or more subtrees $T(v)$ with mass $\mu(v) = m$ then $g(m) = 1$, and $g(m) = 0$ if no such subtree exists. Unfortunately, finding the labeled tree T that maximizes $S(T)$ is an NP-hard problem [13].

We now modify recurrences (2) to find the labeled tree T that maximizes $S(T)$. To this end, note that the complexity of the problem only holds for mass spectra that contain a "large" number of peaks. But sample spectra are relatively sparse and contain only tens of peaks that have significant intensity: The number of simple fragments of a given glycan topology is only linear to its number of monosaccharides. Let k be the number of peaks in the sample spectrum: k is the parameter of our problem, and we limit the running time explosion to this parameter, while maintaining a polynomial running time with respect to M. For the moment, we assume that k is small; we will show below how to deal with mass spectra that contain a larger number of peaks.

In order to avoid multiple peak counting we incorporate the set of explained peaks into the dynamic programming. Scott *et al.* [12] introduced such an approach as part of their color-coding strategy. Let C^* be the set of peak masses in the sample spectrum, where $|C^*| = k$. For every mass $m \leq M$ and every subset $C \subseteq C^*$ we define $S[C, m]$ to be the maximal score of any labeled tree T with total mass $\mu(T) = m$ and only the peaks from C are used to compute this score. At the end of our computations, $S[C^*, M]$ holds the maximal score of any labeled tree where all peaks from C^* are taken into account for scoring. We now modify (2) for our purpose: We define $S_2[C, m]$ to be the score of a "headless" labeled tree with mass m using peaks in C. Using S_2 we can restrict the branching in the tree to bifurcations. We limit the recurrence of $S[C, m]$ to two subtrees with disjoint peak sets $C_1, C_2 \subseteq C$, where C_1 is the subset of peaks explained by the first subtree and C_2 is the set of peaks explained by the second subtree. We require $C_1 \cap C_2 = \emptyset$ what guarantees that every peak is scored only once. Additionally, we demand $C_1 \cup C_2 = C \setminus \{m\}$. Clearly, sets C that contain masses bigger than m need not be considered. We obtain the following recurrences:

$$\begin{aligned} S[C,m] = \max_{g\in\Sigma} \max_{m_1=0,\ldots,\left\lfloor\frac{m-\mu(g)}{2}\right\rfloor} \max_{C_1\subseteq C\setminus\{m\}} \Big\{ & f(C,m) + S_2[C_1, m_1] \\ & + S_2\big[C \setminus (C_1 \cup \{m\}), m - \mu(g) - m_1\big] \Big\} \qquad (3) \\ S_2[C,m] = \max_{m_1=0,\ldots,\left\lfloor\frac{m}{2}\right\rfloor} \max_{C_1\subseteq C} & S[C_1, m_1] + S\big[C \setminus C_1, m - m_1\big] \end{aligned}$$

Note that we delay the scoring of a peak at mass m if m is not in C by extending the scoring function to $f(C, m)$. If $m \notin C$ but $m \in C^*$ then $f(C, m) = 0$. Otherwise, set $f(C, m) = f(m)$. So, both peaks not in C^* and peaks in C are scored, whereas scoring of peaks in $C^* \setminus C$ is delayed.

We now analyze time and space requirements of recurrence (3). One can easily see that the space required to store $S[C, m]$ is $O(2^k \cdot M)$. Time complexity for calculating the optimal solution increases by a factor of 3^k reaching $O(3^k \cdot |\Sigma| \cdot M^2)$, as there are 3^k possibilities to partition k peaks into the three sets C_1, C_2,

and $C^* \setminus (C_1 \cup C_2)$. The exponential running time factor can be reduced to 2^k [2], but the practical use seems to be limited due to the required overhead.

Recall that we have limited our computations to the case where only non-precursor ions are present in the mass spectra. We handle precursor ion by "folding" the spectrum onto itself, details will be given in the full paper.

To recover an optimal solution, we backtrace through the dynamic programming matrix starting from entry $S[C^*, M]$. Backtracking usually generates many isomorphic trees, which we remove from the final output, we defer the details. We can also compute all solutions that deviate at most δ from the score of the optimal solution, we omit the details. Running time of backtracking is $O(out \cdot 2^k \cdot Mn)$ where n is the maximal size of a glycan tree in the output, and *out* is the number of generated trees including isomorphic trees.

We note that several algorithm engineering techniques had to be used so that running times of our algorithm are as low as reported in Sec. 4. Due to space constraints, we defer all details to the full paper.

Scoring for candidate generation. The scoring presented above is overly simplified, what was done to ease the presentation. We now describe some modifications that are needed to achieve good results on real-world data. As noted above, our scoring has to be a simple additive scoring, and we only score fragments that stem from simple fragmentation events.

For our scoring we use real masses of fragments. All the presented recurrences iterate over integer masses m, but monosaccharide and subtree masses are non-integer. To deal with real masses, we define $S[m]$ to be the maximal score of any labeled tree whose exact mass falls into the interval $[m - 0.5, m + 0.5)$, and additionally store the exact mass of the subtree with optimal score in this interval. We update a matrix entry $S[m]$ only if the new subtree mass (the sum of $\mu(g)$, and masses of two headless subtrees) falls in the current interval. Note that for integer mass m_1, we may have to consider the neighboring entries $\{m_1 - 1, m_1, m_1 + 1\}$ in the maximum (3), since the sum of corresponding exact masses might fall into the current interval.

Note that the optimal solution of this problem might no longer be the optimal solution of the original problem. But since we are interested in all solutions up to some δ away from optimality, chances are that the optimal solution of the original problem will be part of the set of candidates we generate. A similar reasoning applies if we limit exact computations to the k most intense peaks in the spectrum, because only peaks of low intensity and small score will be used multiple times.

The basis of our peak score is its normalized intensity, assuming that a high intensity indicates a high probability that the peak is not noise. Mass spectrometrists assume that the mass error of a device roughly is normal distributed. To account for mass deviation $\Delta = |m_{\text{peak}} - m_{\text{fragment}}|$, we multiply the peak intensity with $\text{erfc}(\Delta/(\sigma\sqrt{2}))$, where erfc is the error function complement and σ the standard deviation of the measurement error, typically set to $\frac{1}{3}$ or $\frac{1}{2}$ of the mass accuracy.

We do not apply a penalty if a peak in the sample spectrum is not explained by our candidate spectrum. This may be justified by the fact that our scoring ignores fragments not resulting from a simple fragmentation event. This scoring leads to good results as long as the intensity of peaks from simple fragmentation events is higher than that of non-simple ones.

4 Results on Experimental Data

We implemented our algorithm in Java 1.5. Running times were measured on an Intel Core 2 Duo, 2.5 GHz with 3 GB memory. A set of batroxobin carbohydrate side chains from Bothrops moojeni venom served us to test the program [9]. We used 24 spectra of N-glycans from recent investigations, where the compound was ionized by a single proton. The spectra were measured using a Bruker Daltonics ultraFlex TOF/TOF instrument with a MALDI ion source. Glycans are composed of fucoses (F, mass 146.06 Da), hexoses (H, 162.05 Da), and N-acetylhexosamines (N, 203.08 Da). Glycans were detached from the protein, and the reducing end was marked by a 2-aminopyridine modification resulting in a mass increase of 78 Da for precursor ions. The raw data was baseline corrected and peaks were picked using the SNAP method provided by Bruker. We use the naming convention from [9] for reporting the analyzed glycans.

We used the following parameters for analyzing the spectra: We set $k = 10$, avoiding multiple peak counting for the ten most intense peaks. We allowed a mass deviation of 1.0 Da and chose $\sigma = 0.5$ Da as standard deviation of the measurement error. After normalizing the sum of peak intensities we discarded all peaks with an intensity lower than 0.02. We chose the penalty for missing peaks as the average of the smallest intensity and the mean value of all peak intensities. We iteratively adjusted the score deviation δ for backtracking to obtain a candidate set of about 100 to 200 topologies. Results are shown in Table 1. In all but two cases (F4-4-H3N6F2, F6-1-H2N4F) the correct topology was part of this candidate set without any parameter tuning. We defer further details to the full paper. Average running time for generation of the candidates was 2.5 s without and 4.0 s including traceback.

To test if avoiding peak double-counting is needed for candidate generation, we set $k = 0$, so every peak could be counted an arbitrary number of times. Doing so, candidate generation produced the correct topology only for eight of the 24 spectra even if the candidate set was chosen to contain at least 500 structures. This shows that avoiding multiple peak counting is essential for the analysis. Certain glycan topologies do in fact create the same fragment mass several times: It must be understood that our approach does not *penalize* such topologies, but it also does not *reward* them.

Finally, we tested if further increasing k could improve the results of candidate generation. But as it turned out, increasing k to the 15 most intense peaks did not improve the results. So, computations can be carried out with a moderate k such as $k = 10$, without loosing specificity.

Once we have reduced the set of potential glycan topologies from the exponential number of initial candidates, to a manageable set of tens or hundreds

Table 1. Results of the algorithm for 24 N-glycans of batroxobin [9]. Running times including traceback. *Due to some special properties of these two spectra, the parameter set had to be slightly changed to generate the correct candidate. See full version for details.

glycan	parent mass	# peaks	# B/Y ion peaks	δ	# cand.	running time	rank eval.
H3N5F	1744.0	36	10	15%	126	0.74 s	1
H3N6F	1947.0	41	8	10%	184	1.20 s	3
H3N6	1801.0	28	8	7%	122	0.93 s	1
H4N4F	1703.0	66	15	22%	115	4.69s	1
H5N4F	1865.0	34	11	15%	154	5.60 s	1
F1-H3N5	1598.6	26	7	25%	121	0.61 s	2
F2-1-H3N4F	1541.0	16	6	10%	133	0.43 s	2
F2-2-H4N4	1557.6	21	2	20%	128	1.75 s	1
F2-4-H4N4F	1703.6	33	12	20%	150	1.84 s	1
F2-5-H5N4F	1865.7	29	11	11%	282	0.92 s	1
F2-5-H5N4	1719.6	23	7	10%	147	1.09 s	2
F3-H3N8F	2354.0	18	6	12%	383	5.21 s	2
F4-1-H3N5	1598.0	35	12	20%	154	1.37 s	1
F4-3-H3N4F2	1687.6	20	6	17%	145	2.65 s	1
F4-3-H3N5F	1744.0	41	13	20%	113	3.03 s	1
F4-3-H4N2F2	1849.0	24	8	4.5%	164	5.40 s	4
F4-4-H3N6F2*	2092.0	11	8	9%	450*	4.04 s	2
F5-1-H3N6F	1947.0	92	11	6%	171	1.47 s	2
F5-1-H5N4F	1865.0	37	10	15%	169	3.14 s	3
F6-1-H2N4F*	1379.0	38	11	60%	103*	0.48 s	1
F6-2-H5N4F	1865.0	34	11	15%	161	6.48 s	1
F7-2-H4N4F	1703.0	66	15	23%	146	4.66 s	1
F7-3-H3N6	1801.0	28	8	7%	122	0.91 s	1
F7-3-H4N5F	1907.0	29	9	9%	143	0.87 s	2

of structures, we can now evaluate each candidate glycan topology using an in-depth comparison between its theoretical spectrum and the sample spectrum. This comparison can also take into account peculiarities of the mass spectrometry analysis, such as multiple-cleaved fragment trees, other ion series such as A/X and C/Z ions, or those X-ions that have lost parts of a monosaccharide. Evaluation of candidate glycan structures is not the focus of this work, we just report some identification rates we were able to obtain after evaluation. Our scoring generalizes ideas of Goldberg *et al.* [8], details are deferred to the full paper. The evaluation step ranked the true topology in all except four cases in the TOP 20, for 12 structures even on the first rank. In many cases, top-scoring topologies are biologically impossible. Using biological knowledge on the structure of the analyzed glycans, we always find the true solution in the TOP 4, and 14 true topologies reach rank one, see the right column of Table 1.

5 Number of Glycan Trees

We now show how to calculate the number $N\big[n, |\Sigma|\big]$ of different glycan topologies with n vertices, where vertices are labeled with elements from Σ. There exists a very rich and diverse treatment of similar tree counting problems in the literature, in areas such as mathematical chemistry or phylogenetics. To the best of our knowledge, neither the recurrence reported below, nor anything at least as fast has previously been reported in the literature.

Recall that a glycan topology corresponds to a rooted tree such that every vertex has out-degree at most four. For $|\Sigma| = 1$, Otter [11] analyzed the asymptotical behavior of this number which, in turn, allows us to approximate the number of glycan trees over an arbitrary alphabet Σ. In practice, this approximation is rather crude as we do not take into account isomorphic trees.

We now present a method for the exact computation of $N[n] := N\big[n, |\Sigma|\big]$, for alphabets Σ of arbitrary size. Due to space constraints, we only report the recurrence and leave all details and the formal proof of Lemma 1 to the full version of the paper. We claim

$$N[n+1] = |\Sigma| \cdot \big(N[n] + N_2[n,n] + N_3[n,n] + N_4[n,n]\big). \tag{4}$$

where $N[0] = 0$. The $N_i[n,k]$ for $i = 0, \ldots, 4$ and $k \leq n_{\max}$ can be computed as

$$N_i[n,k] = \sum_{j=1}^{i} \sum_{m=\lceil n/i \rceil}^{\min\{k, \lfloor n/j \rfloor\}} \binom{N[m]+j-1}{j} \cdot N_{i-j}\big[n-jm, m-1\big] \tag{5}$$

We initialize $N_i[n,k]$ depending on i: For $i = 0$, we set $N_0[0,k] := 1$, and $N_0[n,k] := 0$ for all $n \geq 1$. For $i = 1$, we set $N_1[n,k] := N[n]$ for $n \leq k$, and $N_1[n,k] := 0$ otherwise. For $i \geq 2$, we set $N_i[n,k] := 0$ in case $i > n$ or $k \cdot i < n$ or ($k = 0$ and $i > 0$) holds. All other values can be computed from recurrences (4) and (5). For example, assume $|\Sigma| = 3$, then there exist $2.09 \cdot 10^7$ glycan trees with $n = 10$ vertices, and $2.15 \cdot 10^{18}$ glycan trees with $n = 30$ vertices. We have derived a similar recurrence for the number of glycan trees of a given mass m, we defer the details to the full paper.

Lemma 1. *Using recurrences* (4) *and* (5)*, the number of glycan trees with n vertices over an alphabet Σ can be computed in time $O(n^3)$ and space $O(n^2)$.*

6 Conclusion

We have presented an approach for the automated analysis of glycan tandem mass spectra. We focused on the problem of candidate generation needed to reduce the search space of glycan structures. Despite the computational complexity of the candidate generation problem, our approach avoids peak double counting and solves the problem exactly using fixed-parameter techniques. We present a simple scoring scheme for candidate generation. Evaluation using experimental data shows that our method achieves swift running times and very good identification results.

Acknowledgments. We thank Kai Maaß from the biochemistry department of the Justus-Liebig-Universität Gießen for providing the glycan mass spectra.

References

1. Apweiler, R., Hermjakob, H., Sharon, N.: On the frequency of protein glycosylation, as deduced from analysis of the SWISS-PROT database. Biochim. Biophys. Acta 1473(1), 4–8 (1999)
2. Björklund, A., Husfeldt, T., Kaski, P., Koivisto, M.: Fourier meets Möbius: fast subset convolution. In: Proc. ACM Theor. Comp., pp. 67–74. ACM Press, New York (2007)
3. Böcker, S., Lipták, Z.: A fast and simple algorithm for the Money Changing Problem. Algorithmica 48(4), 413–432 (2007)
4. Cooper, C.A., Gasteiger, E., Packer, N.H.: GlycoMod – a software tool for determining glycosylation compositions from mass spectrometric data. Proteomics 1(2), 340–349 (2001)
5. Dell, A., Morris, H.R.: Glycoprotein structure determination by mass spectrometry. Science 291(5512), 2351–2356 (2001)
6. Domon, B., Costello, C.E.: A systematic nomenclature for carbohydrate fragmentations in FAB-MS/MS spectra of glycoconjugates. Glycoconjugate J. 5, 397–409 (1988)
7. Gaucher, S.P., Morrow, J., Leary, J.A.: STAT: a saccharide topology analysis tool used in combination with tandem mass spectrometry. Anal. Chem. 72(11), 2331–2336 (2000)
8. Goldberg, D., Bern, M., Li, B., Lebrilla, C.B.: Automatic determination of O-glycan structure from fragmentation spectra. J. Proteome Res. 5(6), 1429–1434 (2006)
9. Lochnit, G., Geyer, R.: Carbohydrate structure analysis of batroxobin, a thrombin-like serine protease from bothrops moojeni venom. Eur. J. Biochem. 228(3), 805–816 (1995)
10. Niedermeier, R.: Invitation to Fixed-Parameter Algorithms. Oxford University Press, Oxford (2006)
11. Otter, R.: The number of trees. The Annals of Mathematics 49(3), 583–599 (1948)
12. Scott, J., Ideker, T., Karp, M., Sharan, R.: Efficient algorithms for detecting signaling pathways in protein interaction networks. J. Comput. Biol. 13(2), 133–144 (2006)
13. Shan, B., Ma, B., Zhang, K., Lajoie, G.: Complexities and algorithms for glycan structure sequencing using tandem mass spectrometry. In: Proc. of Asia Pacific Bioinformatics Conference (APBC 2007), Advances in Bioinformatics and Computational Biology, pp. 297–306. Imperial College Press (2007)
14. Tang, H., Mechref, Y., Novotny, M.V.: Automated interpretation of MS/MS spectra of oligosaccharides. Bioinformatics 21(suppl. 1), i431–i439 (2005); Proc. of Intelligent Systems for Molecular Biology (ISMB 2005)
15. Zaia, J.: Mass spectrometry of oligosaccharides. Mass Spectrom. Rev. 23(3), 161–227 (2004)