Published in final edited form as:

J Proteome Res. 2019 September 06; 18(9): 3532–3537. doi:10.1021/acs.jproteome.9b00367.

glypy: An Open Source Glycoinformatics Library

Joshua Klein^{*,†}, Joseph Zaia^{*,‡,†}

[†]Program for Bioinformatics, Boston University, Boston, Massachusetts 02215, United States

[‡]Department of Biochemistry, Boston University, Boston, Massachusetts 02215, United States

Abstract

Glycoinformatics is a critical resource for the study of glycobiology, and glycobiology is a necessary component for understanding the complex interface between intra- and extracellular spaces. Despite this, there is limited software available to scientists studying these topics, requiring each to create fundamental data structures and representations anew for each of their applications. This leads to poor uptake of standardization and loss of focus on the real problems. We present glypy, a library written in Python for reading, writing, manipulating, and transforming glycans at several levels of precision. In addition to understanding several common formats for textual representation of glycans, the library also provides application programming interfaces (APIs) for major community databases, including GlyTouCan and UnicarbKB. The library is freely available under the Apache 2 common license with source code available at <https://github.com/mobiusklein/> and documentation at <https://glypy.readthedocs.io/>.

Graphical Abstract

Keywords

glycomics; Python; software libraries

^{*}Corresponding Authors: jaklein@bu.edu, jzaia@bu.edu.

The authors declare no competing financial interest.

INTRODUCTION

Glycobiology is an essential component of our understanding of modern system-level biology. Glycans and glycoconjugates are essential to all life forms and physiological processes.¹ Glycosylation is one of the most complex and varied post-translational modifications found on proteins, modulating folding, trafficking, binding, and function.² Glycoconjugates are found on the cell's surface and extracellular matrixes, participating in signal transduction at many levels and impacting downstream biological processes, including immunity, cancer, extracellular architecture, and differentiation.^{1,3–7}

Glycoinformatics is necessary for the field to schematize, model, and study glycomics and glycobiology to meet emerging high-throughput analytical methods.^{8–10} Glycoinformatics tools require a complex, tiered representation of glycan structures to reflect the different levels of detail at which glycans are studied. Glycans are composed of monosaccharides connected by linkage and branching defined by biosynthetic enzymes. The structures may contain substituents that include sulfate, phosphate, and others depending upon the biological system.¹¹ Bioinformatics tools for analyzing liquid chromatography-coupled mass spectrometry (LC-MS) data often represent glycans at a composition level,^{12–14} where all that is necessary is the ability to calculate an aggregate mass or chemical composition. To represent glycan two- and three-dimensional structure, methods use a graph or tree representation.^{8,9} When a glycan structure is not completely defined by a given experiment, it may be annotated with alternative connections or repeated subgraphs.

Handling these representations can create a barrier for creating glycoinformatics software. As in other domains of bioinformatics, glycoinformatics tools need to read, manipulate, and write glycans. Glycan structures have been formalized using several machine-readable formats^{9,15,16} with differing levels of generality. To date, the canonical implementations of these formats and their structure representations have been released in Java^{9,16,17} with limited support for other programming languages beyond bindings to one of these implementations in MATLAB.¹⁸ This limits the community of glycoinformatics contributors to those who use those languages and those whose problem can be solved with the representations available in those libraries. The Python programming language is used extensively in bioinformatics where many of these resources are unavailable.

Recent work to develop glycomics databases of glycan structures has made such data^{19–22} available and searchable by both humans and software clients. Great effort has been put into semantic modeling of glycans and glycomics-related topics.²³ GlyTouCan²⁰ provides a public SPARQL query service, making the content of their data store available using a Resource Description Framework (RDF). These databases include both general information such as taxonomy assignments, motif associations, linked databases, and publications as well as structural information, much like databases for genomics and proteomics.

We present glypy, a Python library for describing and manipulating monosaccharides, substituents, glycan compositions, and glycan structures. We use Python for its ease of use, acceptance in the wider scientific computing community, large collection of supporting open source libraries, and for its speed of development. We believe it provides a strong foundation

of features necessary for writing more sophisticated glycomics software. For example, the Python community already has several mass spectrometry-related libraries,^{24–26} which make accessing experimental data practical, and can in turn be combined with the mass and fragmentation calculation features of glypy to identify experimental evidence explainable by glycans.^{12,27}

USE CASES

Structure Definitions

In genomics and proteomics, the building blocks of structures, nucleic acids and amino acids, are treated as discrete entities. For example, it is usually assumed that when one refers to cysteine, it is L-cysteine, and that the position of its connection to the next monomer in the sequence is fixed at carbon 1. This simplifying assumption is absent in glycomics, where stereo-chemistry, linkage, and positional connectivity are all variable, even for monomers with the same formula found in the same glycan, with functional consequences. glypy's structure module describes these facets and defines their basic behaviors, such as chemical composition and mass calculation or graph traversal, so that other modules and other programs can go do interesting things with them. glypy also represents glycan compositions where the types and counts of monosaccharides are known, but not their connectivity as well as glycan structures with fully connected monosaccharide graphs or partially connected subgraphs. glypy treats monosaccharides and substituent groups as nodes, discrete entities which may be connected to each other; those connections represent bond relationships, and collections of connected nodes compose graphs.

One common task when writing programs for analyzing data from mass spectrometry involves just calculating the mass of a molecule. Every structural type in glypy can calculate its elemental composition and mass, drawing on a library of known names mapped to chemical formulas and rules for combining them. These reusable components simplified the programming done in our previous work^{12,28} at the composition level. Remoroza et al.²⁷ used glypy's glycan structure representation to calculate theoretical glycan fragment masses to match against tandem mass spectra to assist in annotating a glycan spectral library. For an example of fragment generation, please see Code Sample 2 and an example of spectrum annotation in Figure 1 with implementation in the Supporting Information.

Structure Parsing

In other domains of bioinformatics, the primary structures of the biological entities of interest are linear sequences. The elements of these sequences are governed by a virtually universally accepted IUPAC nomenclature^{31,32} embedded in a plain text format like FASTA that maps individual letters to residues. Glycomics has a profusion of formats for describing

Supporting Information

The Supporting Information is available free of charge on the [ACS Publications website](https://doi.org/10.1021/acs.jproteo-me.9b00367) at DOI: [10.1021/acs.jproteo-me.9b00367](https://doi.org/10.1021/acs.jproteo-me.9b00367).

Code: `annotate_spectrum.py` code for generating Figure 1, including input mass spectrum and structure (`example.mgf` and `biantennary.glycoct`), and `digest.py` code for generating Figure 2 ([ZIP](#))

Code samples: Code Sample 1, parsing glycan structures from text (`format_readwrite.py`); Code Sample 2, theoretical glycan structure fragment generation (`fragmentation_example.py`); Code Sample 3, glySpace interaction with GlyTouCan20 and RDF (`glyspace_example.py`); Code Sample 4, demonstration of `similarity.monosaccharide.similarity.identity.is_a` for comparing different monomer variants (`is_a_query.py`); Code Sample 5, simple structure drawing code (`plot_example.py`) example scripts ([PDF](#))

glycan structures⁸ with varying degrees of precision, compactness, and completeness. The two most recent formats, GlycoCT¹⁵ and WURCS 2,¹⁶ are among the most expressive and complex, while some dialects of the IUPAC encoding³³ and LinearCode³⁴ are capable of concisely describing structures composed of common monosaccharides. GlycoCT and WURCS are able to represent arbitrary monosaccharides in a standardized and easily machine-readable manner, but they have many rules that must be followed and are difficult for humans to read or write themselves. Though it is theoretically possible to describe any monosaccharide with IUPAC, no standard exists for explicitly defining how IUPAC names and linkages should be written, leading to a plethora of dialects from different authors. Despite this, the naming conventions are familiar to biochemists, keeping the programmers guessing at the importance of a letter prefix or suffix. LinearCode cannot represent arbitrary monosaccharides, relying on a limited dictionary mapping specific monosaccharides to one or two letters. A glycoinformatician may need to deal with many data sources with structures encoded in multiple formats. The trivial-seeming task of parsing just one of these formats quickly becomes challenging when the programmer must account for the sheer variety of parts in a monosaccharide's definition and the different ways in which they may be connected. glypy's `io` module was designed to obviate this problem by providing readers and writers for GlycoCT_{condensed}, a limited subset of GlycoCT_{XML}, IUPAC three letter encoding, WURCS 2, and LinearCode. glypy also includes an implementation of GlycoCT's canonicalization algorithm to detect equivalent structures.¹⁵ For an example of structure parsing, please see Code Sample 1.

Database Access

While the ability to read, write, and represent these complex structures may be useful, they require that we already have the structure in some form. For some problems, structures may be learned *de novo* from empirical data, but most require we explore the space of possible structures to find the best-fitting solution. In other domains of bioinformatics, we consult databases like UniProt³⁵ or GenBank³⁶ to obtain reference structures to compare against, sometimes using a programmatic client like the one provided by BioPython³⁷ to accomplish the task. We include a module for communicating with GlyTouCan²⁰ and UnicarbKB.¹⁹ These databases organize data semantically using RDF. glypy communicates with the databases using SPARQL, a protocol and query language for RDF data, implemented with the RDFLib library.³⁸ These database application programming interfaces (APIs) let glycoinformaticians search for reference structures by predicate such as glycan motif classification, taxonomic origin, or known protein conjugate. For an example of accessing GlyTouCan, please see Code Sample 3.

Pattern Matching

In every domain of biology, recurring structural patterns are an essential concept. In glycobiology, we regularly deal with class-defining motifs like the *N*-glycan core, the various *O*-GalNAc glycan cores, or functional motifs like Lewis epitopes.^{2,39} Often, databases will include annotations for core motifs,⁴⁰ but not all motifs may be covered. Such motifs are often not linear and have defined linkage as well as topology, requiring a detailed node and subgraph comparison method to find these motifs in larger structures. glypy's rich monomer representation lends itself to comparing nodes expressively and includes an

implementation of both topological and linkage-specific traversals for substructure matching. This includes an implementation of the maximum common substructure algorithm from ref 41. More recent work has been done to develop a similar method over a semantic graphical representation which may be useful for future work.⁴²

An instance where motifs would not be annotated ahead of time are enzymatic recognition sites.^{18,43} These sites depend upon specific substructures being present and others being absent, forming a list of terminal or nonterminal patterns to match. This has several applications for generating glycan structure spaces either by simulating the biosynthesis process similar to the method in refs 18 and 44 or for generating perturbations of an existing glycan structure database with an exoglycosidase like a sialidase or fucosidase. Additional information may be captured by keeping track of parent $\rightarrow$ *enzyme* $\rightarrow$ child relationships. glypy includes a module dedicated to this type of enzyme simulation and for building these enzymatic graphs. See Figure 2 for a visual example of this graph with implementation in the Supporting Information.

Given the number of different properties a single monosaccharide can have, we can also define relationships among sets of monomers. For example, the monosaccharides fucose, rhamnose, and quinose are all 6-deoxy-hexoses, and being able to recognize this relationship may be useful. Similarly, being able to recognize that glucose, *N*-acetyl glucosamine, and glucuronic acid are all forms of hexose, but that while glucose is common to both of the other monosaccharides, this does not hold for any other combination of the three monosaccharides. glypy includes a method for testing these relationships, implemented using a distance function over monosaccharides. This distance function treats “unspecified” attributes like wildcards, matching any value. All pattern matching methods in glypy are based upon this method. For an example of monosaccharide pattern matching, please see Code Sample 4.

UTILITIES

Plotting

Because pictures are often worth more than words and are essential for publishing glycoinformatics results, glypy also includes some utilities for drawing glycan structures with the widely used CFG⁴⁵ and SNFG²⁹ symbol nomenclatures as well as simpler text-based residue labels with a simple example of SNFG rendering shown in Code Sample 5. These graphics are constructed with Matplotlib,⁴⁶ making them integrable with any other arbitrary plot built with Matplotlib. The rendered glycan structures can be laid out using the popular balanced tree layout or a more structurally descriptive topological layout. In addition to basic drawing, we also include logic for annotating bonds and MS/MS fragmentation events, as shown in Figure 1 and in the script “annotate_spectrum.py” in the Supporting Information of this article.

Semantic Graph Object Mapping

While executing SPARQL queries may be sufficient for well-defined tasks, it is often cumbersome for exploratory work. glypy therefore includes an object mapping layer that lets

the programmer treat entities in an RDF graph as if they were objects with predicates as attributes whose values are other mapped entities in the graph. This mapping layer is built into the GlyTouCan²⁰ and UnicarbKB¹⁹ clients. Because these semantic graphs are built upon a shared namespace, GlycoRDF,²³ we include a copy of that vocabulary compiled into a Python object that supports automatic completion, a feature absent from otherwise purely remote namespaces.

CONCLUSION

We developed glypy, a Python library for representing and manipulating glycans. We demonstrated its ability to translate glycan structures to and from text, calculate properties of glycan structures and components, describe and search for patterns within and among structures, and communicate with publicly available databases. glypy has already been used in published work on low molecular weight heparin,²⁸ CE-MS and LC-MS glycomics,^{12,27,47} and glycoproteomics.^{48,49} We assert that glypy takes care of the computational bookkeeping of what a glycan is so that other code can be concerned with the more important task of connecting a glycan's theoretical properties to empirical phenomena to solve real problems. The source code is readable and well-documented, making it accessible to new users with source and compiled builds freely available under the Apache2 License.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

ACKNOWLEDGMENTS

This work was supported by National Institute of Health Grant U01CA221234.

REFERENCES

- (1). Varki A Biological roles of glycans. *Glycobiology* 2017, 27, 3–49. [PubMed: 27558841]
- (2). Varki A; Cummings RD; Esko JD; Freeze HH; Stanley P; Bertozzi CR; Hart GW; Etzler ME *Essentials of Glycobiology*; Cold Spring Harbor Laboratory Press, 2009; p 784.
- (3). Maverakis E; Kim K; Shimoda M; Gershwin ME; Patel F; Wilken R; Raychaudhuri S; Ruhaak LR; Lebrilla CB Glycans in the immune system and The Altered Glycan Theory of Autoimmunity: A critical review. *J. Autoimmun* 2015, 57, 1–13. [PubMed: 25578468]
- (4). Glavey SV; Huynh D; Reagan MR; Manier S; Moschetta M; Kawano Y; Roccaro AM; Ghobrial IM; Joshi L; O'Dwyer ME The cancer glycome: Carbohydrates as mediators of metastasis. *Blood Rev.* 2015, 29, 269–279. [PubMed: 25636501]
- (5). Stowell SR; Ju T; Cummings RD Protein glycosylation in cancer. *Annu. Rev. Pathol.: Mech. Dis* 2015, 10, 473–510.
- (6). Bagdonaite I; Wandall HH Global aspects of viral glycosylation. *Glycobiology* 2018, 28, 443–467. [PubMed: 29579213]
- (7). Chandran PL; Dimitriadis EK; Mertz EL; Horkay F Microscale mapping of extracellular matrix elasticity of mouse joint cartilage: an approach to extracting bulk elasticity of soft matter with surface roughness. *Soft Matter* 2018, 14, 2879–2892. [PubMed: 29582024]
- (8). Campbell MP; et al. Toolboxes for a standardised and systematic study of glycans. *BMC Bioinf.* 2014, 15 (Suppl 1), S9.
- (9). von der Lieth C-WC-W; et al. EUROCarbDB: An open-access platform for glycoinformatics. *Glycobiology* 2011, 21, 493–502. [PubMed: 21106561]

- (10). Frank M; Schloissnig S Bioinformatics and molecular modeling in glycobiology. *Cell. Mol. Life Sci* 2010, 67, 2749–2772. [PubMed: 20364395]
- (11). Perez S; Aoki-kinoshita KF In *Guide to Using Glycomics Databases*; Aoki-kinoshita KF, Ed.; Springer Japan: Tokyo, Japan, 2017; Chapter 2.
- (12). Klein J; Carvalho L; Zaia J Application of network smoothing to glycan LC-MS profiling. *Bioinformatics* 2018, 34, 3511–3518. [PubMed: 29790907]
- (13). Artimo P; et al. ExPASy: SIB bioinformatics resource portal. *Nucleic Acids Res.* 2012, 40, W597–W603. [PubMed: 22661580]
- (14). Veillon L; Zhou S; Mechref Y In *Methods in Enzymology*; Shukla AK, Ed.; Academic Press, 2017; Vol. 585; Chapter 22, pp 431–477.
- (15). Herget S; Ranzinger R; Maass K; Lieth C-WVD GlycoCT-a unifying sequence format for carbohydrates. *Carbohydr. Res* 2008, 343, 2162–71. [PubMed: 18436199]
- (16). Matsubara M; Aoki-Kinoshita KF; Aoki NP; Yamada I; Narimatsu H WURCS 2.0 Update To Encapsulate Ambiguous Carbohydrate Structures. *J. Chem. Inf. Model* 2017, 57, 632. [PubMed: 28263066]
- (17). Horlacher O; Nikitin F; Alocchi D; Mariethoz J; Müller M; Lisacek F MzJava: An open source library for mass spectrometry data processing. *J. Proteomics* 2015, 129, 63–70. [PubMed: 26141507]
- (18). Liu G; Neelamegham S A Computational Framework for the Automated Construction of Glycosylation Reaction Networks. *PLoS One* 2014, 9, No. e100939.
- (19). Campbell MP; Peterson R; Mariethoz J; Gasteiger E; Akune Y; Aoki-Kinoshita KF; Lisacek F; Packer NH UniCarbKB: Building a knowledge platform for glycoproteomics. *Nucleic Acids Res.* 2014, 42, D215–21. [PubMed: 24234447]
- (20). Tiemeyer M; et al. GlyTouCan: an accessible glycan structure repository. *Glycobiology* 2017, 27, 915–919. [PubMed: 28922742]
- (21). Lütke T In *Guide to Using Glycomics Databases*; Aoki-kinoshita KF, Ed.; Springer Japan: Tokyo, Japan, 2017; Chapter 3.
- (22). Aoki-kinoshita KF In *A Practical Guide to Using Glycomics Databases*; AokiKinoshita KF, Ed.; Springer Japan: Tokyo, 2017.
- (23). Ranzinger R; et al. GlycoRDF: an ontology to standardize glycomics data in RDF. *Bioinformatics* 2015, 31, 919–25. [PubMed: 25388145]
- (24). Kösters M; Leufken J; Schulze S; Sugimoto K; Klein J; Zahedi RP; Hippler M; Leidel SA; Fufezan C pymzML v2.0: introducing a highly compressed and seekable gzip format. *Bioinformatics* 2018, 34, 2513–2514. [PubMed: 29394323]
- (25). Röst HL; Schmitt U; Aebersold R; Malmström L pyOpenMS: A Python-based interface to the OpenMS mass-spectrometry algorithm library. *Proteomics* 2014, 14, 74–77. [PubMed: 24420968]
- (26). Levitsky LI; Klein JA; Ivanov MV; Gorshkov MV Pyteomics 4.0: Five Years of Development of a Python Proteomics Framework. *J. Proteome Res* 2019, 18, 709–714. [PubMed: 30576148]
- (27). Remorosa CA; Mak TD; De Leoz ML; Mirokhin YA; Stein SE Creating a Mass Spectral Reference Library for Oligosaccharides in Human Milk. *Anal. Chem* 2018, 90, 8977–8988. [PubMed: 29969231]
- (28). Zaia J; Khatri K; Klein J; Shao C; Sheng Y; Viner R Complete Molecular Weight Profiling of Low-Molecular Weight Heparins Using Size Exclusion ChromatographyIon Suppressor-High-Resolution Mass Spectrometry. *Anal. Chem* 2016, 88, 10654–10660. [PubMed: 27709905]
- (29). Varki A; et al. Symbol nomenclature for graphical representations of glycans. *Glycobiology* 2015, 25, 1323–1324. [PubMed: 26543186]
- (30). Domon B; Costello CE A Systematic Nomenclature for Carbohydrate Fragmentation in FAB-MS/MS Spectra of Glycoconjugates. *Glycoconjugate J.* 1988, 5, 397–409.
- (31). IUPAC-IUB Commission on Biochemic. Abbreviations and Symbols for nucleic acids, polynucleotides and their constituents. *J. Mol. Biol* 1971, 55, 299–310. [PubMed: 5551389]
- (32). Union IU o. P. Nomenclature and symbolism for amino acids and peptides (Recommendations 1983). *Pure Appl. Chem* 1984, 56, 595–624.

- (33). Mcnaught AD NOMENCLATURE OF CARBOHYDRATES (Recommendations 1996) Prepared. Pure Appl. Chem 1996, 68, 1919–2008.
- (34). Nir D; Dukler A GLYCOFORUM A Novel Linear Code ® Nomenclature for Complex Carbohydrates. Trends in Glycoscience and Glycotechnology 2002, 14, 127–137.
- (35). The UniProt Consortium. UniProt: a hub for protein information. Nucleic Acids Res. 2014, 43, D204–212. [PubMed: 25348405]
- (36). Clark K; Karsch-Mizrachi I; Lipman DJ; Ostell J; Sayers EW GenBank. Nucleic Acids Res. 2016, 44, D67–D72. [PubMed: 26590407]
- (37). Cock PJ; Antao T; Chang JT; Chapman BA; Cox CJ; Dalke A; Friedberg I; Hamelryck T; Kauff F; Wilczynski B; De Hoon MJ Biopython: Freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics 2009, 25, 1422–1423. [PubMed: 19304878]
- (38). The RDFLib Development Team. RDFLib, 2005 <https://github.com/RDFLib/rdfliib> (accessed August 6, 2017).
- (39). Cummings RD The repertoire of glycan determinants in the human glycome. Mol. BioSyst 2009, 5, 1087–104. [PubMed: 19756298]
- (40). Aoki-Kinoshita K; et al. GlyTouCan 1.0 The international glycan structure repository. Nucleic Acids Res. 2016, 44, D1237– D1242. [PubMed: 26476458]
- (41). Aoki KF; Yamaguchi A; Okuno Y; Akutsu T; Ueda N; Kanehisa M; Mamitsuka H Efficient tree-matching methods for accurate carbohydrate database queries. Genome informatics. International Conference on Genome Informatics 2003, 14, 134–43. [PubMed: 15706528]
- (42). Alocci D; Mariethoz J; Horlacher O; Bolleman JT; Campbell MP; Lisacek F Property Graph vs RDF triple store: A comparison on glycan substructure search. PLoS One 2015, 10, 1–17.
- (43). Liu G; Neelamegham S Integration of systems glycobiology with bioinformatics toolboxes, glycoinformatics resources, and glycoproteomics data. Wiley Interdisciplinary Reviews: Systems Biology and Medicine 2015, 7, 163–181. [PubMed: 25871730]
- (44). Akune Y; Lin C-H; Abrahams JL; Zhang J; Packer NH; Aoki-Kinoshita KF; Campbell MP Comprehensive analysis of the N-glycan biosynthetic pathway using bioinformatics to generate UniCorn: A theoretical N-glycan structure database. Carbohydr. Res 2016, 431, 56–63. [PubMed: 27318307]
- (45). Varki A; Cummings RD; Esko JD; Freeze HH; Stanley P; Marth JD; Bertozzi CR; Hart GW; Etzler ME Symbol nomenclature for glycan representation. Proteomics 2009, 9, 5398– 5399. [PubMed: 19902428]
- (46). Hunter JD Matplotlib: A 2D graphics environment. Comput. Sci. Eng 2007, 9, 99–104.
- (47). Khatri K; Klein JA; Haserick JR; Leon DR; Costello CE; McComb ME; Zaia J Microfluidic Capillary Electro-phoresisMass Spectrometry for Analysis of Monosaccharides, Oligosaccharides, and Glycopeptides. Anal. Chem 2017, 89, 6645– 6655. [PubMed: 28530388]
- (48). Khatri K; Klein JA; White MR; Grant OC; Leymarie N; Woods RJ; Hartshorn KL; Zaia J Integrated Omics and Computational Glycobiology Reveal Structural Basis for Influenza A Virus Glycan Microheterogeneity and Host Interactions. Mol. Cell. Proteomics 2016, 15, 1895–1912. [PubMed: 26984886]
- (49). Klein JA; Meng L; Zaia J Deep Sequencing of Complex Proteoglycans: A Novel Strategy for High Coverage and Site-specific Identification of Glycosaminoglycan-linked Peptides. Mol. Cell. Proteomics 2018, 17, 1578–1590. [PubMed: 29773674]

Figure 1.

An annotated collisional dissociation tandem mass spectrum from a deuterio reduced and permethylated sialylated biantennary *N*-glycan. The structure was drawn using the SNFG symbol nomenclature,²⁹ and fragment names were derived from the Domon and Costello nomenclature for glycan fragments.³⁰ Red peaks are match theoretical fragments that contain at least one reducing end cleavage event, while blue peaks match fragments containing only nonreducing end cleavage event. Red cleavage lines indicate single bond cleavage events, while the short blue lines indicate a multiple bond cleavage event. Bond cleavages were not annotated if they were only assigned by a fragment with a neutral loss.

Figure 2.
A subset of a *N*-glycan biosynthetic enzyme graph.