

Cover Page



Universiteit Leiden



The handle <http://hdl.handle.net/1887/28508> holds various files of this Leiden University dissertation.

Author: Peironcely Miguel, Julio Eduardo

Title: Automated de novo metabolite identification with mass spectrometry and cheminformatics

Issue Date: 2014-09-03

Automated de novo metabolite
identification with mass spectrometry and
cheminformatics

Julio E. Peironcely

Automated de novo metabolite identification with mass spectrometry and cheminformatics

Julio Eduardo Peironcely Miguel

PhD thesis with summary in Dutch

ISBN: 978-90-74538-85-5

© 2014 Julio E. Peironcely. All rights reserved. No part of this thesis may be reproduced or transmitted in any form, by any means, electronic or mechanical, without prior written permission from the author.

The printing of this thesis was financially supported by the Netherlands Organization for Applied Scientific Research (TNO)

Automated de novo metabolite identification with mass spectrometry and cheminformatics

Proefschrift

ter verkrijging van
de graad van Doctor aan de Universiteit Leiden,
op gezag van Rector Magnificus prof. mr. C. J. J. M. Stolker,
volgens besluit van het College voor Promoties
te verdedigen op woensdag, 3 september 2014
klokke 15.00 uur

door

Julio Eduardo Peironcely Miguel

geboren te Barcelona, Spanje

in 1982

Promotiecommissie

Promotor:

Prof. Dr. T. Hankemeier Leiden University

Copromotores:

Dr. T. Reijmers University of Amsterdam

Dr. L. Coulier DSM, Delft

Overige leden:

Prof. Dr. J-L. Faulon University of Evry, France

Dr. A. Bender University of Cambridge, United Kingdom

Prof. Dr. A. P. IJzerman Leiden University

Prof. Dr. M. Danhof Leiden University

Para Lin y Nina.

Para mis padres.

Table of Contents

Chapter 1

Introduction..... 9

Chapter 2

OMG: Open Molecule Generator..... 27

Chapter 3

Understanding and classifying metabolite space and metabolite-likeness..... 61

Chapter 4

An automated pipeline for de novo metabolite identification using mass spectrometry-based metabolomics 107

Chapter 5

De novo identification of metabolites with open molecule generator for metabolomics 137

Chapter 6

Conclusions and perspectives..... 163

Appendix

Samenvatting..... 175

Acknowledgements..... 181

Curriculum vitae 185

Publications 187

1 Introduction

Introduction

Metabolomics

The completion of the Human Genome Project [1] was considered at that moment as an important milestone for curing many diseases. With a new understanding of one's genes we should be able to understand better the underlying mechanisms of complex diseases. Ultimately, this should lead to better diagnosis and treatment of diseases[2]. The great expectations bestowed on genomics and the sequencing of the human genome were however not met, since (i) many genetic factors can contribute to a disease and (ii) most diseases have not a pure genetic cause. For example, individuals with the same genetic background, like monozygotic twins, can develop during the course of their lives different diseases. This shows the need for a holistic view where other biochemical levels, apart from genomics, are necessary to understand biological systems.[3]

Looking at an organism as a system, i.e. a collection of components like genes, enzymes or metabolites, which are all interconnected at different levels such as cellular, organ and overall system level, allows researchers to better understand the causes of disease and to develop better diagnostics and, ultimately personalized, treatments.[4] Such an approach does not only focus on genes, proteins and metabolites but also at the interactions between all of them. The study of these different building blocks of organism gave birth to systems biology and several omics

era, with a plethora of fields such as genomics, proteomics, metabolomics, peptidomics, transcriptomics, and many others.

Metabolomics is the science that studies metabolites, the small molecules (<1000Da) involved in metabolism, either as substrates or as products of metabolic reactions. Being closer to the phenotype than genes, metabolites are ideal read outs of alterations like disease, including metabolic disorders, diet, lifestyle, and medical interventions have on a biological system.[5, 6]

The aim of metabolomics is to measure qualitatively (which) and quantitatively (how much) metabolites are present in a biofluid, tissue, or cell in order to answer biological questions.[6] These measurements can be static, measured only once on a single moment in time, or dynamic, over multiple time points, so that the progress of a disease or the effect of a treatment can be monitored.

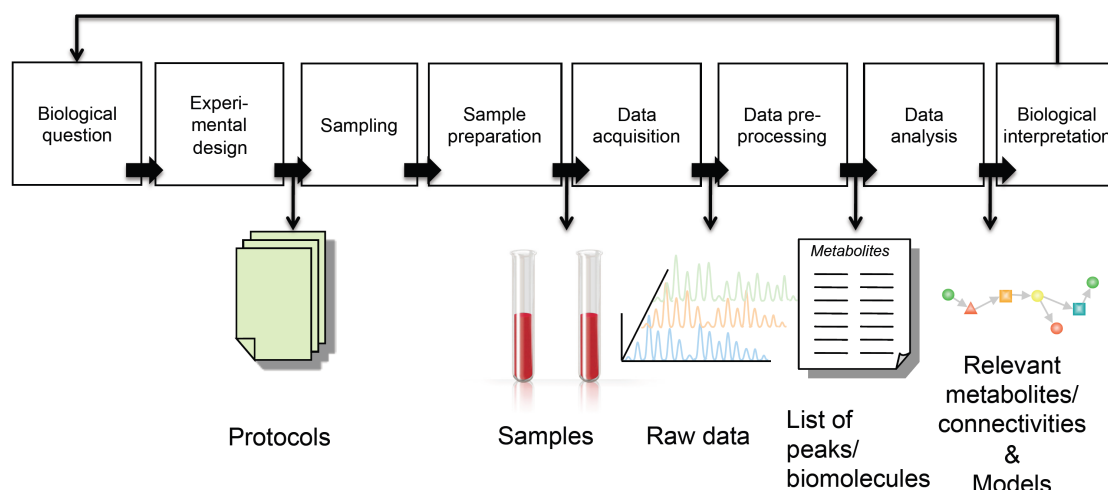


Figure 1 Workflow of a metabolomics pipeline. Start of any experiments should be a clear biological question. The different analytical and processing steps lead to biological data, which after interpretation should answer the original question or suggest follow up experiments. The contents of this thesis contribute to the data processing and data analysis steps.

Metabolites form a heterogeneous family of molecules for which there is not a single analytical strategy yet available that can measure them all in a single run. As a

consequence, different analytical platforms are being developed and used suitable for profiling specific classes of metabolites. One of the most often used analytical techniques in metabolomics is mass spectrometry, mostly after separation of metabolites by liquid chromatography, capillary electrophoresis or gas chromatography (MS).[7, 8]

Metabolite identification, the identification of the precise chemical structure of a metabolite, is one of the major challenges in metabolomics. The two most frequently analytical techniques used for this are nuclear magnetic resonance (NMR) and mass spectrometry. While NMR can provide for each atom of a molecule rich information about their neighbouring atoms and hence, about their structure, NMR requires a high concentration and a high sample purity of the metabolite that needs to be identified. This is not usually the case for, among others, human samples. Therefore, MS is widely used to measure human metabolites in complex samples.

Analytical Chemistry

LC-MS

Mass spectrometry is commonly used in metabolomics to detect metabolites in a qualitative manner, by measuring the mass of molecules, calculating their elemental composition and to elucidate their chemical structures by interpretation of their fragmentation spectra. And MS is used in a quantitative manner by determining the abundance of metabolites as absolute concentrations or relative to a reference compound. Usually MS instruments are composed of four modules: (i) an ionizer, which charges the molecules in the sample, prior or after transfer into the gas phase;

(ii) an extraction system, which transfers the ionized molecules (or their fragments) from the sample introduction unit to the analyzer; (iii) a mass analyzer, which separates the molecules according to mass; and (iv) a detector, which quantifies the abundance of the ions. The information that can be obtained from MS is the ratio of the mass of ions and their charge, the so-called mass-over-charge ratio (m/z). From these m/z values, the mass of the molecule can be derived after revealing whether the ions were formed by protonation, deprotonation, adduct formation, etc; from the mass of the molecule the elemental composition (EC) or a short-list of possible elemental compositions can be derived; from a EC one or multiple structures can be proposed. The more accurate the mass is determined the fewer candidate elemental compositions are obtained. The mass accuracy depends on the instrument employed, and even for high accuracies, such as in the low part per million (ppm) or sub-ppm range, unique elemental compositions cannot always be obtained. However, when including several constraints into accounts by including fragment ions and their relations among each other,[9] the number of possible elemental compositions for a certain mass can be reduced. Still, for a given mass many elemental compositions can be found. And, even worse, for a given EC millions or billions of candidate structures can be proposed.

Analyzing metabolites with mass spectrometry poses many challenges. Samples in metabolomics studies are complex and can contain thousands of metabolites. Some of these metabolites can have identical atomic mass, elemental formula and/or be structural isomers, which complicates their identification using mass spectrometry. Additionally, other compounds that are also present in the sample we measure, can

affect how well our analyte of interest ionizes. This process is known as ion suppression and is the reason why metabolites in complex samples are usually first separated using chromatography prior to detection with MS. Gas chromatography (GC) is used to separate compounds that are volatile or that via derivatization can become volatile. Non-volatile (but also volatile) compounds are separated according to their polarity using liquid chromatography (LC). In both cases the metabolites present in the mobile phase, either gas or liquid, pass through a column and interact with the stationary phase in the column. This interaction with the stationary phase delays the elution of some metabolites, and thus, achieving their separation. The diameter, length and characteristics of the stationary phase of a column determine how the metabolites will be separated. The time after which a metabolite elutes is called the retention time (RT).

Once metabolites have been separated they need to be detected and quantified, which is often achieved by a mass spectrometer. The combination of the information obtained from the LC and MS experiments, RT and m/z respectively, is also referred to as a metabolite feature. In metabolomics experiments, metabolites are often characterized by their RT and m/z values, when their identity is still unknown.

MSⁿ

Ion traps are mass analyzers that can trap an ion, the so-called parent ion, with a certain pre-specified mass/charge ratio, fragment it, trap the fragments and fragment them further (if needed). This type of MS fragmentation is known as multi-stage mass fragmentation (MSⁿ), where n indicates the number of consecutive

fragmentations performed. The resulting mass spectral tree contains information about the parent and fragment ions and the relationships between them. This type of data contains valuable structural information for metabolite identification, since two isomeric molecules, with the same elemental composition but different chemical structures will most likely produce different mass spectral trees. The depth (number of consecutive fragmentations) and width (number of fragment ions obtained at a given level) is determined by the available amount of the compound, by the size of the compound (a small compound cannot yield many fragments) and by its chemical composition (some bonds are more resistant to cleavage than others).

De Novo Metabolite Identification

During a metabolomics study, one or multiple metabolite features can be of interest for the biological question at hand. These features can be for example biomarkers that indicate significant difference between the metabolic profiles of healthy and diseased patients or they can be the metabolic end product of a gene knockout experiment, or the degradation products of a newly tested drug, for instance. In all these cases, knowing the chemical structure of those metabolite features is essential to interpret the results of the metabolomics study. De novo and non-de novo metabolite identification are regarded as one of the major bottlenecks facing metabolomics.[10–12]

A critical step in metabolite identification is the selection of candidate structures for an unknown metabolite. Standard metabolite identification methods retrieve the

chemical structures of a compound, that is present in a library, by matching its experimental mass, elemental composition and/or spectrum against those of compounds in a library. This approach only allows the identification of metabolites that have already been discovered, for which mass spectra are required in a comparable manner, and that are properly stored in a database. However, such a strategy will not allow to identify unknown compounds, which are not in a database yet, or also for some known metabolites because the proper database has not been selected, or the data acquisition conditions were different. As mentioned earlier, with the proper method one or multiple elemental compositions can be derived from the mass of a molecule, however, up to millions of structures can be obtained for a single EC. Therefore, querying masses and ECs in databases will return an incomplete list of candidate structures.

De novo identification deals with 'truly' unknown metabolites that are not present in compound libraries. In many cases this means that the unknown has not been identified before, which poses many challenges. Firstly, all possible candidate structures have to be generated, ensuring that the correct structure is not missing among the candidates. Secondly, this (usually very large) list of structures has to be reduced by filtering unwanted structures. Ideally, one or a handful of candidate structures will be left at the end, including the correct structure. Lastly, the proposed structures have to be validated in order to have a confident identity assignment.

Cheminformatics

Cheminformatics, also known as chemoinformatics, is the field of handling chemical information electronically. The use of computers to solve chemical problems can be traced back to 1946 when IBM accounting machines were used to construct the rotational spectra of asymmetric rotors. Cheminformatics has been applied extensively in drug discovery, analytical chemistry and structural biology, among others.

In drug discovery, chemists use cheminformatics to select suitable drug candidates, test *in silico* their activity against biological targets, build predictive models (for instance of drug-likeness) and calculate physicochemical properties. Other uses of cheminformatics include querying databases of small compounds (<1000Da) in search for compounds that are similar to known active molecules, or for those that have a high predictive activity against a protein of interest. Apart from searching compounds that are similar to known compounds, cheminformatics helps scientists to navigate the chemical space. The aim is designing molecules that are structurally different to existing ones but with properties that better match properties ideal in a certain situation, for instance having a good ADMET profile (administration, distribution, metabolism, excretion and toxicity) and easy to be synthesized.

Cheminformaticians have produced a plethora of predictive models. These models try to predict for new molecules a property or characteristic, which have been learnt from a set of molecules for which this property or characteristic is known. Examples of these models are drug-likeness or nature product-likeness. In simple words, these

models use machine-learning algorithms to establish the relationship between molecular properties (structural or physicochemical) and a desired property or behavior. All these models rely on the concept that “similar molecules will have similar activities”. An example of predicting behavior of molecules are quantitative structure activity-relationship (QSAR) models, which aim to predict the binding affinity of molecules to a certain target based on certain predictor properties (via a regression model), or which aim to predict whether a molecule is biologically active or not (via a classification model). Quantitative structure property-relationship (QSPR) models on the other hand predict properties of a molecule like solubility, mutagenicity or internal energy based on the structural characteristics of the molecule. Cheminformatics models can be used in principle in metabolite identification strategies to, e.g., reject candidate structures that do not have a predicted property value close to the experimental, like retention time, or that they do not exhibit a desired property, like metabolite-likeness.

In analytical chemistry, cheminformatics helps to store, process and compare all sorts of spectrometry data. One of the oldest applications of cheminformatics is to assist in the elucidation of new compounds. In other words, to use computers and algorithms to determine the chemical structure of a compound measured using analytical chemistry techniques.

Structure elucidation

In the 1960s databases were built to store and retrieve mass spectra. The same decade witnessed in analytical chemistry one of the most ambitious uses ever of

cheminformatics in any field, the DENDRAL project.[13] The aim of DENDRAL was to create the first expert system for Computer Assisted Structure Elucidation (CASE), which would automatically predict the chemical structure from the mass spectrum of an unknown compound. In the years after DENDRAL was introduced, databases of NMR spectra and tools to predict the structure of unknown molecules were also developed.

Structure elucidation is one of the oldest areas in cheminformatics, and aims to determine the chemical structure of a molecule based on its experimental data, usually MS or NMR.[14] At the core of a structure elucidation system lays a structure generator. This software tool takes as an input the elemental composition of the molecule and optionally some constraints, and generates possible chemical structures. Common constraints are prescribed substructures, forbidden substructures and desired properties of the generated molecules.

Most generators are deterministic and exhaustive: they generate all possible molecules for a given input. Since structure generation is a combinatorial problem, it can lead to an explosion in the number of generated molecules. To avoid this, one can provide constraints that limit the number of molecules, or use a stochastic generator, which will produce a subset of the possible results. While computationally affordable, stochastic generators do not guarantee that the correct structure will be included in the list of results. Therefore, scientists focused so far on the identification of small molecules using regularly deterministic generators.

Structure generators represent molecules as graphs given the resemblance of molecules and graphs. Molecular atoms can be seen as graph vertices and molecular bonds as graph edges. This allows structure generators to use graph theory, a sub-field of mathematics. Graph theory allows cheminformaticians to elaborate theorems and theoretical proofs of completeness and correctness. From these theorems algorithms can be developed to generate graphs (and molecules). Finally, these algorithms are programmed as software tools, namely structure generators. Several graph theory-based algorithms to generate molecules have been developed.[15] The choice of structure generator will depend on how complete one wants the results to be, how fast the calculation the structures should be, and how easily new constraints should be implemented.

After decades of research in CASE, several structure generators have been developed, MOLGEN[16] being the most advanced. Unfortunately, MOLGEN and others are commercial and the source code of these tools is not available and they are closed tools. In other words, they are not freely available neither can they be customized and extended to fulfill the needs of metabolomics researchers. However, all available structure generators so far do not fulfill all needs for identification of metabolites, such as using only atom types present in biological systems, applying constraints available from experimental data or using several substructures.

Scope and outline of the thesis

The goal of this thesis is to design, develop and integrate new methodologies and software tools such as a structure generators and chemoinformatic models in a

pipeline that enables de novo metabolite identification. The ultimate pipeline should propose candidate structures based on LC- MSⁿ for those unknown metabolites that are not present in any database. LC- MSⁿ was chosen because this analytical platform allows to detect a large number of metabolites in human samples also at lower concentrations and yields structural information of unknown metabolites via extensive fragmentation. The list of candidate structures for a unknown metabolites, or unknown compound in general, should be exhaustive, i.e. the actual structure may not be missed, but at the same time as short as possible due to elimination of candidates using multiple filtering criteria. A short list of candidate structures can then be checked by an expert to identify the final candidate structure. At the start of this thesis, software tools to process and analyze LC-MSⁿ were scarce, vendor dependent and not open source. Therefore, the first aim in this thesis was to develop a structure generator and to develop algorithms to filter the results obtained by the structure generator. Therefore a metabolite-likeness predictive method had to be developed. In addition, these algorithms should be open source to allow others to use them and improve them further and on the other hand to implement open source tools in our own pipeline. In a parallel project, another researcher developed algorithms for the preprocessing of spectral trees and algorithms to compare spectral trees. The second aim was therefore to integrate all tools developed in this thesis and in the parallel project into one pipeline to allow the nearly fully automated processing and interpretation of spectral trees of unknown metabolites and to obtain a short list of candidate structures.

In **Chapter 2**, a structure generator, the Open Molecule Generator (OMG), is developed. OMG is open source and at the heart of a de-novo identification pipeline, since it can generate all possible chemical structures for an unknown metabolite with a given elemental composition. The canonical augmentation approach, originally designed to generate graphs, was adapted to generate all possible molecules for a given elemental composition and optional fragments and implemented as open source.

The aim of the research reported in **Chapter 3** was to develop a method to constrain the number of molecules generated by OMG. The Metabolite-likeness filter has been developed to remove unwanted molecules that do not resemble human metabolites. Different classification models were developed, optimized and trained to discern between human metabolites and non-metabolite molecules based on their physicochemical and structural properties. These models were tested and the best performing one selected to reject molecules that would obtain a low metabolite-likeness score.

In **Chapter 4**, the tools developed in Chapter 2 and 3 of this thesis and in other related metabolite identification research projects are integrated into an metabolite identification pipeline (Figure 2), optimized and applied to identify metabolites detected in human urine, which could be known and unknown metabolites. MSⁿ spectra of human urine metabolites were acquired, processed, the elemental compositions assigned to fragment ions and neutral losses, and the spectral trees were compared using fragmentation tree similarity to a database of known

metabolites in order to obtain prescribed substructures. These substructures and the elemental composition were taken as input for generation of candidate structures with the OMG. These structures were filtered using Metabolite-likeness, internal energy and fragmentation prediction filters. The performance of the overall identification pipeline was discussed and possible future developments proposed.

In **Chapter 5**, the structure generation method was further improved based on the experiences obtained in Chapter 4, and the Parallel Molecule Generator (PMG) developed. These improvements included the use of a faster algorithm, the execution in parallel using multiple processors and the use of a bad list of substructures and bad rings, to reject while generating unwanted molecules that contained one or multiple unwanted moieties. The improvement resulted in significant reduction of time required to create a candidate list, which was 100-fold compared to OMG for unknown metabolites for which a list of prescribed substructures was provided.

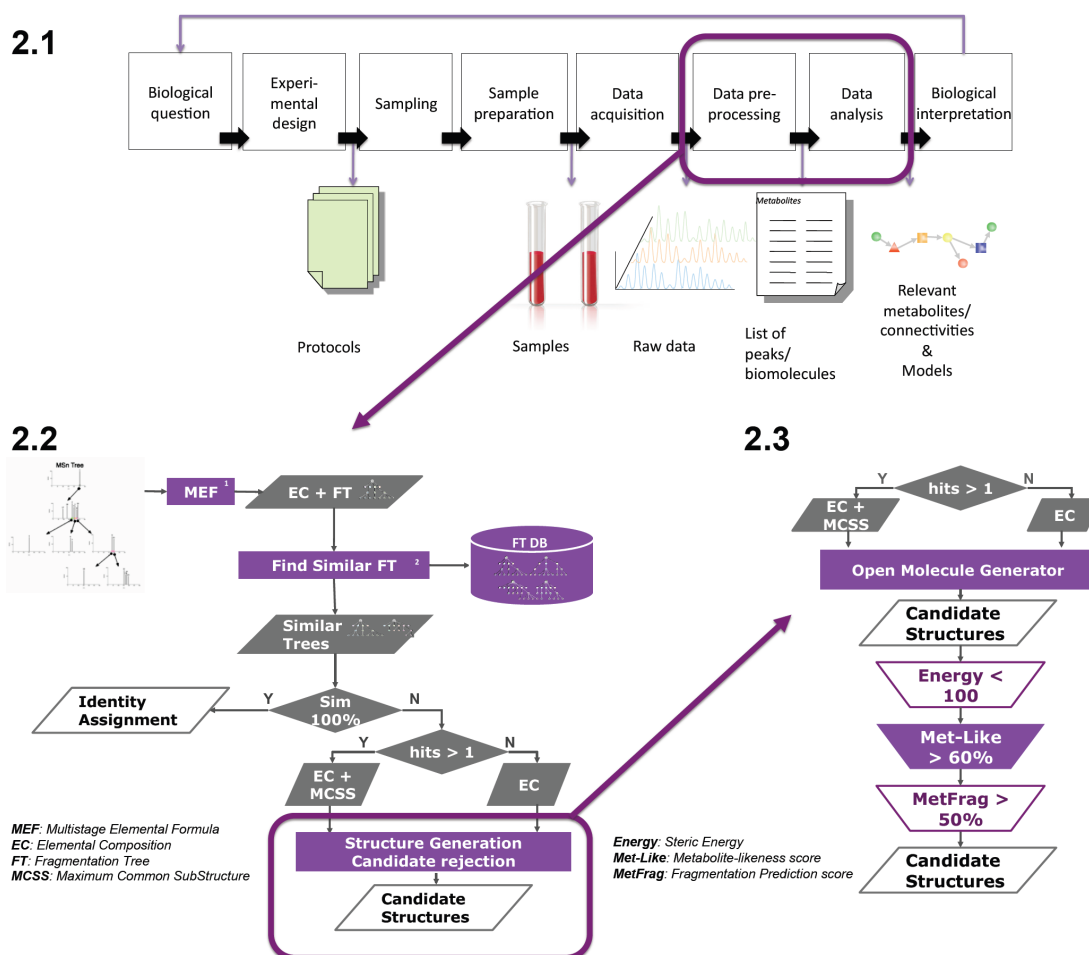


Figure 2 Integration in the metabolomics workflow (Figure 2.1) of the metabolite identification pipeline (this thesis, Figure 2.2) into the metabolomics workflow. Components developed during this thesis are Open Molecule Generator (OMG) and Metabolite-likeness (Figure 2.3). Lessons learned during the use of the metabolite identification pipeline motivated the improvement of OMG and the creation of the Parallel Molecule Generator (PMG, this thesis).

References

1. Venter JC, Adams MD, Myers EW, et al.: The Sequence of the Human Genome. *Science* 2001, 291 :1304–1351.
2. Lander ES: Initial impact of the sequencing of the human genome. *Nature* 2001, 409:187–197.
3. Kitano H: Systems biology: a brief overview. *Science* 2002, 295:1662–4.
4. Chuang H-Y, Hofree M, Ideker T: A Decade of Systems Biology. *Annual review of cell and developmental biology* 2010:1–24.
5. Hall R, Beale M, Fiehn O, Hardy N, Sumner L, Bino R: Plant metabolomics: the missing link in functional genomics strategies. *Plant Cell* 2002, 14:1437–40.
6. Fiehn O: Metabolomics--the link between genotypes and phenotypes. *Plant molecular biology* 2002, 48:155–71.

7. Werner E, Heilier J-F, Ducruix C, Ezan E, Junot C, Tabet J-C: Mass spectrometry for the identification of the discriminating signals from metabolomics: current status and future trends. *Journal of chromatography. B, Analytical technologies in the biomedical and life sciences* 2008, 871:143–63.
8. Villas-Bôas SG, Mas S, Åkesson M, Smedsgaard J, Nielsen J: Mass spectrometry in metabolome analysis. *Mass Spectrometry Reviews* 2005, 24:613–646.
9. Rojas-Chertó M, Kasper PT, Willighagen EL, Vreeken RJ, Hankemeier T, Reijmers TH: Elemental composition determination based on MS(n). *Bioinformatics* 2011, 27:2376–83.
10. Dunn WB, Erban A, Weber RMJM, Creek DJ, Brown M, Breitling R, Hankemeier T, Goodacre R, Neumann S, Kopka J, Viant MR: Mass appeal: metabolite identification in mass spectrometry-focused untargeted metabolomics. *Metabolomics* 2012:1–23.
11. Johnson CH, Gonzalez FJ: Challenges and opportunities of metabolomics. *Journal of Cellular Physiology* 2012, 227:2975–81.
12. Scalbert A, Brennan L, Fiehn O, Hankemeier T, Kristal BS, Van Ommen B, Pujos-Guillot E, Verheij E, Wishart D, Wopereis S: Mass-spectrometry-based metabolomics: limitations and recommendations for future progress with particular focus on nutrition research. *Metabolomics* 2009, 5:435–458.
13. Lindsay RK, Buchanan BG, Feigenbaum EA, Lederberg J: *Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project*. New York: McGraw-Hill Book; 1980.
14. Faulon J-L, Visco DP, Roe D: Enumerating Molecules. In *Reviews in Computational Chemistry*. John Wiley & Sons, Inc.; 2005:209–286.
15. Brinkmann G: Isomorphism rejection in structure generation programs. *Discrete Mathematical Chemistry* 2000, 51:25 – 38.
16. Kerber A, Laue R, Grüner T, Meringer M: MOLGEN 4.0. *Match* 1998, 37:205 – 208.

2 *OMG: Open Molecule Generator*

Published in *Journal of Cheminformatics* 2012, **4**:21.

OMG: Open Molecule Generator

Computer Assisted Structure Elucidation has been used for decades to discover the chemical structure of unknown compounds. In this work we introduce the first open source structure generator, Open Molecule Generator (OMG), which for a given elemental composition produces all non-isomorphic chemical structures that match that elemental composition. Furthermore, this structure generator can accept as additional input one or multiple non-overlapping prescribed substructures to drastically reduce the number of possible chemical structures. Being open source allows for customization and future extension of its functionality. OMG relies on a modified version of the Canonical Augmentation Path, which grows intermediate chemical structures by adding bonds and checks that at each step only unique molecules are produced. In order to benchmark the tool, we generated chemical structures for the elemental formulas and substructures of different metabolites and compared the results with a commercially available structure generator. The results obtained, i.e. the number of molecules generated, were identical for elemental compositions having only C, O and H. For elemental compositions containing C, O, H, N, P and S, OMG produces all the chemically valid molecules while the other generator produces more, yet chemically impossible, molecules. The chemical completeness of the OMG results comes at the expense of being slower than the commercial generator. In addition to being open source, OMG clearly showed the added value of constraining the solution space by using multiple prescribed substructures as input. We expect this

structure generator to be useful in many fields, but to be especially of great importance for metabolomics, where identifying unknown metabolites is still a major bottleneck.

Computer Assisted Structure Elucidation (CASE) of chemical compounds is one of the classical problems positioned at the intersection of informatics, chemistry, and mathematics. CASE tools have been employed during decades to elucidate the chemical structure of small organic molecules. In its most general definition, a structure elucidation system receives experimental chemistry data of an unknown molecule as input, and outputs a list of possible chemical structures. The input can be the elemental composition of the elusive molecule, nuclear magnetic resonance (NMR) and/or mass spectrometry (MS) spectra (provided the generator can simulate spectra and match it to the experimental ones) or information of prescribed substructures. The output is a list of candidate structures matching these conditions, ideally containing all possible structures without duplications. A small list of candidates is dependent on the number of constraints derived from experimental data; the higher the number of constraints we use the smaller the candidate list will be. The ultimate goal for such a system being fully automated and returning only one and correct molecule is not yet at our reach, despite decades of research[1].

The DENDRAL[2] project is widely regarded as the initiator of the use of these methods to provide a system for Computer Assisted Structure Elucidation (CASE). It involved the development of artificial intelligence algorithms that would extract heuristics from MS and NMR data and use them to constrain the output of a

structure generator. CONGEN was the structure generator developed within DENDRAL, which preceded a more advanced generator known as GENOA[3]. Many commercial structure generators were developed later, most renowned ones being CHEMICS[4], ASSEMBLE[5], SMOG[6], and the most widely used of all of them, the general purpose structure generator MOLGEN[7]. These closed source software tools work like a black box, where the user cannot, on the one hand, understand the functioning of the software and on the other hand, customize the tool to his needs. These drawbacks of closed source software (where the source code is not provided) can be circumvented by open source tools. Two open source structure generators have been developed that work with NMR data, the deterministic LSD[8] and the stochastic SENECA[9]. Implementation of open source stochastic and deterministic structure generators have been explored within the Chemistry Development Kit (CDK)[10, 11]. Unfortunately, these generators failed to generate all chemical structures possible and were discontinued in recent releases of CDK. Despite these efforts, no general purpose deterministic structure generator has been developed in an open source format so far.

The advance of “omics” sciences in the last decade, in particular of metabolomics[12], has renewed the interest of researchers in developing better structure generators. Metabolomics aims at detecting and identifying metabolites in an organism and has resulted in a large list of potential biomarkers for which the chemical structure is unknown[1, 13]. When trying to identify the structure of unknown molecules, scientists first perform an identity search by querying reference databases using their experimental information[1, 14–16]. In such case, they use the

elemental composition of the metabolite derived from mass spectrometry (MS) or the spectra of nuclear magnetic resonance (NMR). When the metabolite is a real unknown it is not present in any database, therefore the query returns no results. This forces scientists to propose candidate structures using a different approach, one of them is using a structure generator[17, 18] , which produces all possible molecules given an elemental composition and optional, other constraints. Examples of constraints are prescribed substructures that each output molecules should contain and that are derived from experimental NMR, MS², or MSⁿ data. Hence, the need for deterministic and flexible structure generators in the field of metabolomics presents should be met with new algorithms[1].

The majority of structure generators rely on graph theory to produce their desired output. Interestingly, compounds can be represented as molecular graphs where atoms and bonds are translated into vertices and edges, respectively, to which theorems and algorithms proposed by graph theory can be applied. This ensures that the output is correct, exhaustive, and free of isomorphs. Such methods can be the orderly enumeration proposed by Read[19] and Faradzev[20], a stochastic generator[21], the homomorphism principle[22] used by MOLGEN, or the “canonical augmentation path” proposed by McKay[23]. This last method, originally intended to generate simple graphs by adding vertices, has been applied to the generation of some families of graphs and also to generate the chemical universe of molecules up to 11 atoms[24] and recently to 13 atoms[25]. Despite the goal was to generate molecules, these two approaches initially employed canonical path augmentation to generate all possible simple graphs up to 11 and 13 vertices, respectively. Posterior

topological and ring system filter were used to remove unwanted graphs. Lastly, the vertices were colored with chemical elements and the edges with a bond order, which turned the graphs into molecules. Simple chemical constraints like connectivity and atom valence were applied to reduce the list of final molecules. This process, which relies on generating simple graph, is necessarily limited on the size of the molecules that can be generated because a linear increase in the number of atoms produces an exponential increase of both the number of graphs and molecules. Here we present the Open Molecule Generator (OMG), a structure generator based too on McKay augmentation algorithms, but rather than first generating graphs and secondly transforming these graphs into molecules, our implementation of McKay technique directly constructs molecules. In this way we can generate chemical structures much greater than 13 atoms. Essential concepts of graph theory will be introduced in the methods section.

Chen mentioned two future challenges facing CASE systems[26]. The first challenge for elucidating structures is to have a knowledge system of previously identified compounds, as well as mining tools for such data. In this direction, Rojas-Chertó et al.[27] developed a system to store spectral data and mine the database to extract substructure information that can be used as prescribed substructures in our structure generator. The second challenge is the need for filtering and selecting candidate structures. This is often performed by predicting a property of the candidate structures that is related to the field of research, for instance, predicting the spectra in analytical chemistry, the bioactivity in ligand design, or the Metabolite-Likeness[28] in metabolomics studies, to name a few. Furthermore, the

need of a structure generator tool that can be adapted to the requirements of the field in which it is going to be applied, demonstrates the usefulness of open source tools compared to commercial "black box" generators.

In this paper we present the first general purpose open source structure generator, Open Molecule Generator. OMG adapts methodologies from the field of graph theory and deterministic graph enumeration to the classical problem of chemical structure generation. In this sense, we have used the approach of "canonical path augmentation" to ensure that we exhaustively generate non-isomorphic chemical structures for a given elemental composition. This generation tool has been implemented using CDK[10, 11], a widely used open source library for the development of chemoinformatics software. It allowed the representation of entities such as molecules, atoms, and bonds in our program and the use of functions like removing hydrogen atoms, checking the saturation of a molecule, removing a bond, and many more. The resulting tool generates all possible non-duplicate chemical structures for a given elemental composition, with the option to generate only those that contain one or multiple non-overlapping substructures, which is the most important constrain to reduce the number of resulting candidate structures when a knowledge system is not available[18]. We have used OMG to generate molecules for the elemental composition of well known metabolites, also including one or more prescribed substructures as input. These results are compared to those obtained by MOLGEN.

Materials and methods

Chemical Elements and Atom Types

We would like to describe some concepts related to atoms that are necessary to understand the theory and algorithm behind OMG and the use of CDK to handle chemistry.

In nature, atoms of different chemical elements (carbon, nitrogen, oxygen, and others) are connected to each other by bonds in order to form molecules. The valence, to which we will also refer as degree, of these chemical elements determine how many bonds each element can have. Carbon has a valence of 4, oxygen of 2, nitrogen of 3 or 5, sulfur of 2,4 or 6, phosphor of 3 or 5. Thus a carbon atom becomes saturated when it has 4 bonds, where a single bond counts as one bond, a double as two bonds, and a triple as three bonds. Regarding molecules, we consider a molecule to be saturated when all its atoms are saturated. In some special occasions, atoms are charged, which makes them having a different valence. In the case of OMG, we only use neutral atoms and as a consequence only neutral molecules are produced, therefore all finished molecules will contain atoms with the valences mentioned before.

A chemical element can have multiple atom types, also for the same valence of an element, as defined by the dictionary of atom types in CDK. This dictionary defines for each atom the number of neighbors, pi bonds, charges, lone electron pairs, and hybridizations, in order to accommodate the different states a chemical element can have due to different bonds, number of neighboring atoms, charges and

hybridizations. These atom types are based on the chemical elements that have been observed in nature for saturated molecules. This is why we use the CDK atom dictionary to validate the atoms of our finished molecules.

OMG will output only molecules that are saturated and that contain the atoms specified in the elemental composition. Apart from finished molecules, OMG has to represent during the generation process intermediate chemical structures that are not finished yet. These might contain disconnected fragments and atoms that are not saturated. CDK atom types are not designed to represent atom types of unsaturated chemical elements; therefore we opted for implementing a simple atom dictionary. For each chemical element, this dictionary defines its valence, in other words, the maximum degree. Hence for intermediate chemical structures we only check that the current degree of each atom does not exceed the maximum degree.

MOLGEN can also produce molecules with multiple valences, but it handles them in a different way. While with OMG only the elemental composition needs to be provided to generate molecules with multiple valences, MOLGEN requires knowing a priori which one of the multiple valences has to be used. It uses by default the lowest valence, this is, N valence 3, P valence 3, and S valence 2, unless a different valence is specified. In Table 1 the atom types produced by OMG and MOLGEN for non-default valences are presented. Using sulfur as an example, OMG will output molecules with containing sulfur valence 2, 4 and 6. For the same chemical element, MOLGEN will produce by default molecules with sulfur valence 2. If one sets the valence of sulfur to 6, it will only produce sulfur valence 6 and not valence 2 and

valence 4. MOLGEN cannot generate molecules with atoms of different valences for the same chemical element, this is, if molecule has two sulfur atoms, one will not be of valence 4 and the other of valence 6, both will be either valence 2, 4 or 6.

Valence	MOLGEN	OMG
N valence 5		
P valence 5		
S valence 4		
S valence 6		

Table 1 Atom types produced by OMG and MOLGEN for non-default valences of N(5), P(5) and S(4 and 6).

The principle followed by CDK to build its atom dictionary is to allow atom types with valences for which there is a consensus agreement on their existence, this is, for which known molecules exist with such valences. Conversely, MOLGEN produces all theoretically possible combinations of bond orders for a given valence, as it can be observed in Table 1. For example, as it can be seen for P valence 5 OMG only produces one atom type with one double bond and three single bonds. In comparison, MOLGEN produces all the combinations of single, double, and triple bonds that add to 5. As a consequence, when the desired valence is unknown, which is usually the case in metabolite identification, molecules need to be generated with all possible valences. As a result, the number of output molecules by both generators is different for elemental compositions that contain chemical elements

with multiple valences. This deterministic generation of valences in MOLGEN comes at the expense of generating molecules having unrealistic structures.

Graph Theory and Chemistry

The chemical structure of molecules can be represented as a graph, where atoms and bonds in molecules correspond to vertices and edges, respectively, in graphs. In molecules, bonds connecting two atoms can have a degree depending on the number of electrons they share. Such a degree can also be assigned to the edges of a graph, which is called a multigraph. The different chemical elements present in the periodic table are represented in graphs as colors assigned to the vertices. We define a non-directed colored multigraph as $G = (V, E)$ where V is a set of vertices and E is a multiset of edges, where each edge is an unordered pair of vertices, and a function $Col: V \rightarrow colors$. In this multigraph, we say that $a, b \in V$ are n -connected if there are exactly n edges $(a, b) \in E$. Apart from the color function, a multigraph is characterized by the function $d: V \times V \rightarrow N$, which returns the degree of the edge connecting each couple of vertices. From now on we will indistinctively refer to graphs and multigraphs.

In chemistry, the valence rule determines the maximum number of bonds each chemical element has. In order to take this into account, we define which returns the number of edges of a given vertex and a *max-degree* function $md: V \rightarrow M$, which returns the maximum number of edges of a given vertex. We say that a multigraph is under-saturated if there is at least one vertex v' such that $d(v', v') < md(v')$. A multigraph is saturated if the equality $d(v', v') = md(v')$ holds for every vertex. In chemistry,

molecules correspond to saturated colored multigraphs and max-degree depends on the color, which is the chemical element. For instance, for a carbon element, $md(C) = 4$ and for an oxygen element, $md(O) = 2$.

We consider a multigraph to be connected if $\forall v, w \in V, \exists S_{\{v,w\}} = \{v_1, \dots, v_m\}$ such that v, v_1 and v_m, w are connected and for each $i < m$, v_i is connected to v_{i+1} . In other words, a multigraph is connected if for all pair of vertices, there exists at least one path $S_{\{v,w\}}$ connecting both vertices. This condition is necessary for chemistry, since intermediate chemical structures in the generation process can be composed of disconnected fragments, it ensures that the generated molecules are one fully connected structure and not made of disconnected substructures. Notice that hydrogen atoms (the most frequently found chemical elements with degree 1) are not considered in the generation process, since they are terminal elements of the molecule and they cannot connect two disconnected elements of the molecule. Hydrogen atoms are only used to validate the completeness of finished molecules. Halogen atoms like fluorine, chlorine, and iodine, also of degree 1, are considered during the generation process.

Graph Labeling

An isomorphism π is a function that for each vertex $v \in V$, $Col(\pi(v)) = Col(v)$ and for each pair of vertices $v \in V, v' \in V'$ $d(\pi(v), \pi(v')) = d(v, v')$. A labeling function $\sigma: V \rightarrow \{1, \dots, n\}$ is a bijective map from the vertices of a colored multigraph to an ordered list labels with a cardinality equal to the number of vertices. Put simple, σ assigns to each vertex a label. Let σ^{-1} be the inverse function of σ , which returns

the vertex corresponding to a label. We say a labeling function is canonical if given any two isomorphic colored multigraphs $G = (V, E)$ and $G' = (V', E')$, the bijective function $\pi: V \rightarrow V'$ defined as $\pi(a) = \sigma^{-1}(\sigma(a))$ is an isomorphism of V in V' . Therefore, a canonically labeled multigraph is a multigraph whose vertices are associated to an ordered list through a canonical labeling function. Furthermore, a canonical hash of the labeling is a bijective function between the space of the canonically labeled multigraphs and the value space and it is represented as a string of integers. It is interesting to note here that two isomorphic graphs have the same canonical hash, a fact that will be used to remove duplicated molecules during the generation process.

Using Fragments

A fragment or substructure of a molecule is equivalent to a fragment or subgraph of a graph. We define a fragment as a subset of a graph and it is characterized by the function $d_f: V \times V \rightarrow N$ where N is the number of edges connecting each pair of vertices in the subgraph. Such d_f has to fulfill the condition $d_f(a, b) \leq d(a, b), \forall a, b \in V$ and at least for one edge $d_f < d$, this is, the fragment should have fewer edges than the graph.

Canonical Augmentation

An augmentation of a multigraph $G = (V, E)$ is a multigraph $G' = (V, E')$, defined on the same set of vertices, such that $\forall a, b \in V, d_G(a, b) = d_{G'}(a, b)$, except for one and

only one pair where $d_{G'}(a,b) = d_G(a,b) + 1$. Let $e' \in E'$ be the edge which degree has been increased, $d(e') = d(e) + 1$.

Let e be the last edge of G and v_1, v_2 the vertices of e . Consider G' to vertices of G , a copy of G , to which a bond order decrease is performed. The resulting multigraph G' after this decrease in bond order, can be seen as the result of a canonical deletion on G , the reverse operation of a canonical augmentation. In our definition of canonical augmentation we consider a multigraph $G' = (V', E')$ to be canonically augmented from $G = (V, E)$ if it is an augmentation and G . In other words, we consider G' to be a canonical augmentation of G if a canonical deletion in G' results in G .

Description of the algorithm

The generation of structures can be seen as a tree of intermediate chemical structures that our tool explores. At the root of the tree we find a collection of fully isolated/disconnected atoms. One bond is added at each level of the tree, resulting in fully connected/finished molecules at the leaves. The canonical augmentation path is a depth-first backtracking algorithm, where the recursive function *generate* described in Algorithm 1, implements the addition of one bond in all possible ways for a given intermediate chemical structure, and evaluates for each extended molecule that this extension has been performed in a canonical way, as described before. Here adding one bond means increasing the degree of the bond between two atoms, hence a single bond becomes a double bond and a double bond becomes a triple bond. If there is no bond between two atoms, a single bond is created.

```

1: generate(M)
2:   If saturated(M) AND are_all_H_used(M)
3:     If connected_fragments(M) == 1
4:       store_to_file(M)
5:       Nmols = Nmols + 1
6:       If degree(M) < max_degree(M)
7:         generate(M)
8:       Endif
9:     Endif
10:  Else
11:    New Map
12:    List_of_bonds = extend(M)
13:    Foreach bond in list_of_bonds
14:      M' = add_bond(bond,M)
15:      canonM' = canonize(M')
16:      If not is_present(map,canonM')
17:        add(map,canonM')
18:        If is_canonical_augmentation(canonM',M',M)
19:          generate(M')
20:        Endif
21:      Endif
22:    End
23:  Endif
24: End
Algorithm 1

```

Between lines 2 and 9 of Algorithm 1, the molecule is stored if it is finished, which occurs when the molecule is saturated and all the atoms of the elemental composition, including the hydrogen atoms, have been used, all the atoms are validated by the CDK atom dictionary and are connected forming one single structure and not multiple disconnected fragments.

In the case the molecule is not finished, it would be extended in all possible ways by adding one bond. If there exists a bond between a pair of atoms function *extend*, in line 12 of Algorithm 1, will increase the multiplicity. The generation of new bonds is controlled by OMG atom type definitions for intermediate chemical structures,

which guarantee that the degree of the atoms does not exceed the maximum degree allowed for its chemical element.

Function *canonize*, in line 15 of Algorithm 1, returns the canonical version of the molecule. We modified the graph canonizer Nauty[23, 29] in order to allow multigraphs and not only simple graphs. Other canonizers for graphs exist like MOLGEN-CID[30] or the Signature Canonizer[31], but Nauty has been the most widely used for graphs as well as for chemistry problems, like InChI[32] codes. Nauty is the canonizer of choice because it is the fastest of all available canonizers for bounded valence graphs below 100 vertices[33] (molecules are examples of this class of graphs). Firstly, the function *canonize* translates the molecule into a colored multigraph. Secondly, it utilizes Nauty to calculate the canonical labeling of the multigraph. Thirdly, this canonical labeling is used to construct the canonical version of the input molecule. Lastly, the canonical hash string of each augmented molecule is stored in a hash map, lines 16 and 17, in order to remove duplicated extensions at each level of the tree. Each unique extension is checked for canonical augmentation, line 18, using Algorithm 2, or Algorithm 3 in case prescribed substructures were provided. If this extension is successful, the function *generate* is called, line 19 of Algorithm 1, and the molecule we want to continue extending is passed as a parameter. When a molecule cannot be extended any further, the recursion is terminated and the program backtracks in the search tree.

```
1: Is_canonical_augmentation(canonM', M', M)
2:   last_bond = get_last_bond(canonM')
3:   M'' = remove_bond(M', last_bond)
4:   return are_the_same(M'', M)
5: End
```

Algorithm 2

```
1: Is_canonical_augmentation_fragments(canonM', M', M)
2:   last_bond = get_last_bond(canonM')
3:   While bond_belongs_to_fragment(last_bond, canonM')
4:     last_bond = get_previous_bond(canonM')
5:   Endwhile
6:   M'' = remove_bond(M', last_bond)
7:   return are_the_same(M'', M)
8: End
```

Algorithm 3

Input and Output

The minimum input required is the elemental composition of the structures that have to be generated. Optionally, a structure-data file (SDF) can be provided containing one or more prescribed substructures that we want our output molecules to contain. Since OMG does not take hydrogen atoms into account during the generation of intermediate chemical structures, the hydrogen atoms present in the substructures will be removed before the generation process begins. These substructures should be non-overlapping, i.e. they should not share any atoms. This limitation is due to the fact that our algorithm grows molecules by adding bonds and, if two atoms in different fragments were in fact the same atom, our algorithm would create bonds between those atoms, which would clearly lead to incorrect results. In practice, multiple substructures can be available, but the user does not know if they overlap. This limitation can be circumvented by using the largest

substructure as constraint for the generation and the remaining substructures as a posterior filtering, only keeping the molecules with those substructures.

By default, the structure generator returns the count of molecules it generated. Optionally, it can store all the molecules in an SDF file. If prescribed fragments are provided, OMG outputs only the molecules containing such fragments. We have opted to use SDF as our input and output format, but via CDK, other formats can easily be implemented in OMG.

Data

As mentioned in the introduction, the identification of the chemical structure of metabolites is one of the current bottlenecks of metabolomics. In this sense, a structure generator can contribute to overcome this bottleneck, since it can provide candidate structures for an unknown metabolite. Therefore, metabolites appear to be a relevant family of compounds to test our structure generator. A list of metabolites was selected and their elemental composition was compiled to evaluate the performance of our structure generator on different inputs. The source of the compounds employed was the Human Metabolome Database (HMDB)[34], which contains almost 8,000 metabolites and is the most comprehensive database of human metabolites. A study of the human metabolite space and the properties of the metabolites that occupy it, has been previously reported[28]. The selection criteria were to include cyclic and acyclic compounds, of different molecular weights, and containing different chemical elements like C, O, N, P, and S. A first test set included metabolites with C, O, and H, chemical elements with one valence. A

second test set included metabolites with C,O,H and also chemical elements with multiple valences, like N, P, and S. Furthermore, for some of these metabolites, several substructures were drawn and provided to the structure generator as additional input. These substructures are easily identified by an expert from direct inspection of MS² or MSⁿ experimental data. The aim was to assess the importance of having fragment information to reduce the list of generated structures.

Results and discussion

Structure Generation from Elemental Formula

The algorithm presented in this work, the Open Molecule Generator, was tested and compared with the commercial structure generator, MOLGEN. Both generators take resonance into account producing all the contributing structures. As a result, the two resonant forms of benzene will be considered as different molecules. Both OMG and MOLGEN are not limited to acyclic structures[35, 36], thus the two structure generators tested can generate molecules with rings. Furthermore, both tools generate molecules containing common chemical elements present in metabolites, like C, O, N, H, P, and S, and are not limited to only 4 chemical elements[36]. Both structure generators generate molecules for a given elemental composition by exhaustively producing all non-redundant chemical structures.

The number of molecules produced after using the elemental compositions of a diverse selection of metabolites containing only C, O and H, is presented in Table 2. For all these metabolites, the same number of molecules is generated by both generators. While both generators produce complete results, MOLGEN does it in less

time. The time between initialization and finalization was measured using time functions in JAVA for OMG and equivalent functions in python for MOLGEN. We can observe in Table 2 the time in seconds to generate all the candidate structures and the time to generate each molecule in milliseconds. If we look at time per molecule, MOLGEN is 4 times faster than OMG for small molecules like pyruvic acid. For larger molecules MOLGEN obtains a constant time per molecule between 0.008 and 0.009 milliseconds, while OMG ranges from 18 to 45 milliseconds depending on the elemental composition. Lightweight profiling of OMG was performed using VisualVM (version 1.3.4), in order to have an understanding of the limiting points in the performance of OMG. The most relevant finding was that the canonization process, which uses Nauty, took half of the total running time.

We observed that MOLGEN stops the generation of molecules after two billion molecules, as it can be observed for a large molecule like cholic acid (Table 2). Since both generators produce the same molecules for elemental composition with C, O and H, we can only assume that more than two billion molecules could be generated. In the case of phenyllactic acid, MOLGEN produces more than 48 million molecules in 404 seconds. Due to excessive computational time, no results for this elemental composition are reported for OMG, though the same number of molecules is expected (if executed for enough time) as is the case for all the other elemental compositions in this subset.

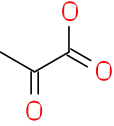
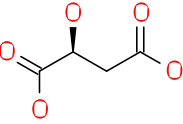
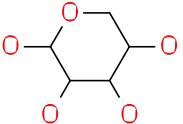
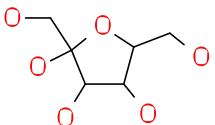
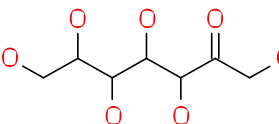
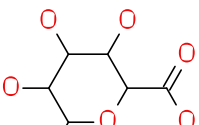
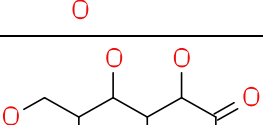
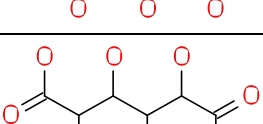
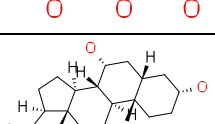
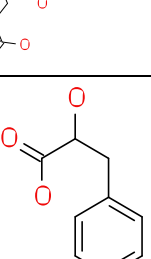
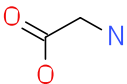
Structure	Name HMDB ID Elemental Composition	MOLGEN			OMG		
		# Candidate Structures	Time (s)	Time per molecule (ms)	# Candidate Structures	Time (s)	Time per molecule (ms)
	Pyruvic acid HMDB00243 C3H4O3	152	0.129	0.849	152	0.509	3.349
	Malic acid HMDB00156 C4H6O5	8,070	0.222	0.028	8,070	27.074	3.355
	D-Xylose HMDB00098 C5H10O5	18,092	0.332	0.018	18,092	125.783	6.952
	D-Fructose HMDB00660 C6H12O6	267,258	2.381	0.009	267,258	5,035.371	18.841
	Sedoheptulose HMDB03219 C7H14O7	4,106,823	38.945	0.009	4,106,823	186,248.085	45.351
	Pectin HMDB03402 C6H10O7	3,183,337	26.512	0.008	3,183,337	46,320.522	14.551
	Galactonic acid HMDB00565 C6H12O7	767,569	6.957	0.009	767,569	22,475.987	29.282
	Galactaric acid HMDB00639 C6H10O8	8,568,129	78.354	0.009	8,568,129	186,730.365	21.794
	Cholic acid HMDB00619 C24H40O5	* More than 2,147,483,646	* not available	* not available	* More than 2,147,483,646	* not available	* not available
	Phenylactic acid HMDB00779 C9H10O3	48,496,265	404.052	0.008	** More than 48,496,265	** not available	** not available

Table 2 Number of chemical structures generated by OMG and MOLGEN using as input only the elemental compositions of metabolites containing C, O and H elements.

* Results were not generated due to excessive computational time needed to generate all the candidate structures. However, we expect OMG to generate more molecules than MOLGEN, due to the larger amount of atom types produced by OMG.

** Results were not generated due to excessive computational time needed to generate all the candidate structures.

As stated in Methods, both generators treat atoms having multiple valences in different ways, this is the reason to use a second set of molecules containing also N, P and S. The default valences used by MOLGEN for N is 3, for P is 3, and for S is 2, unless stated otherwise. The results for these molecules are presented in Table 3. As expected, the number of candidate structures differs between both generators. For the elemental composition of glycine, MOLGEN produces 84 molecules only with N valence 3 and 162 molecules only with N valence 5. For the same elemental composition, OMG produces 97 molecules with valence 3 and 5 for N, which include the 84 of MOLGEN N valence 3 and 13 additional molecules with valence 5, containing N with the atom types depicted in Table 1 for OMG-CDK. The difference in the number of candidate structures is larger for elemental compositions containing many atoms with multiple valences, as is the case of creatinine. For this metabolite, MOLGEN generates 93,323 candidate structures with the default valence 3 for N. On the contrary, OMG produces 303,601 candidate structures, containing N valence 3 and 5.

Structure	Name HMDB ID Elemental Composition	MOLGEN			OMG		
		# Candidate Structures	Time (s)	Time per molecule (ms)	# Candidate Structures	Time (s)	Time per molecule (ms)
	Glycine HMDB00123 C2H5NO2	N_3 84 N_5 162	0.118 0.120	1.405 0.741	97	0.452	4.660

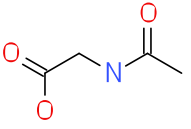
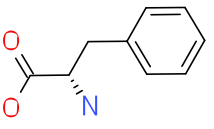
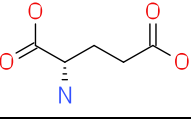
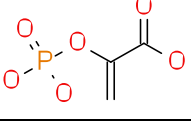
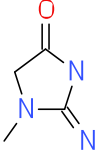
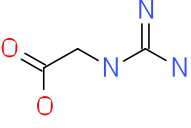
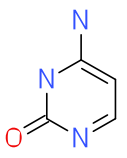
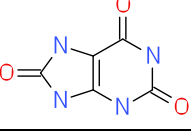
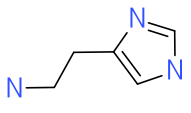

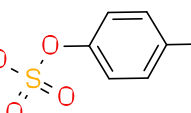
	Acetyl-glycine HMDB00532 C4H7NO3	18,469	0.282	0.015	26,530	126.117	4.754
	Phenylalanine HMDB00159 C9H11NO2	277,810,163	2227.796	0.008	* More than 277,810,163	* not available	* not available
	Glutamic acid HMDB00148 C5H9NO4	440,821	2.945	0.007	685,392	12,348.456	18.017
	Phosphoenolpyruvic acid HMDB00263 C3H5O6P	P_3 51,323 P_5 129,421	0.562 1.398	0.011 0.011	83,977	761.378	9.067
	Creatinine HMDB00562 C4H7N3O	93,323	0.933	0.010	303,601	3,921.157	12.915
	Guanidinoacetic acid HMDB00128 C3H7N3O2	45,626	0.585	0.013	124,808	1,962.532	15.724
	Cytosine HMDB00630 C4H5N3O	108,769	1.149	0.011	491,299	3,952.098	8.044
	Uric acid HMDB00289 C5H4N4O3	464,899,034	3488.097	0.008	* More than 464,899,034	* not available	* not available
	Histamine HMDB00870 C5H9N3	46,125	0.631	0.014	134,278	3,566.544	26.561
	D-Cysteine HMDB03417 C3H7NO2S	3,838	0.156	0.041	15,978	131.004	8.199
	p-Cresol sulfate HMDB11635 C7H8O4S	S_6 592,625,133	5078.132	0.009	* More than 82,000,000	* not available	* not available

Table 3 Number of chemical structures generated by OMG and MOLGEN using as input only the elemental compositions of metabolites containing C, O, H, N, P and S elements.

* Results were not generated due to excessive computational time needed to generate all the candidate structures. We expect OMG to generate more molecules than MOLGEN, due to the larger amount of atom types produced by OMG.

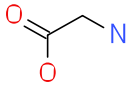

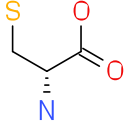

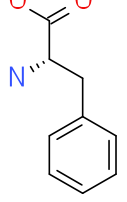


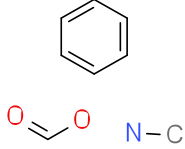
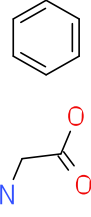
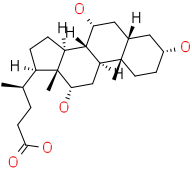
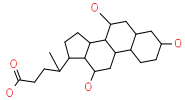
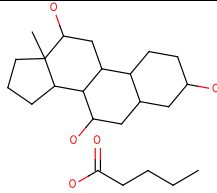
In the case of phosphoenolpyruvic acid, we require P valence 5 to be considered. On the one hand, running MOLGEN with the default valence for P yields 51,323 candidate structures but the correct molecule is missing. On the other hand, forcing the valence of P to be 5, returns 129,421 candidate structures, with the correct molecule also produced but also an excessive quantity of unrealistic molecules due to unrealistic atom types for P. Alternatively, OMG generates 83,977 candidate structures with P valence 3 and 5, including the desired molecule, where all of them are valid molecules as defined by the CDK atom dictionary.

We observe in Table 3 that the running time per generated molecule now ranges between 0.008 and 0.041 milliseconds, while OMG requires between 4.8 and 26.6 milliseconds. Such difference in execution speed between MOLGEN and OMG makes that for some large elemental compositions, only results are reported for MOLGEN. This is the case of phenylalanine, uric acid and p-cresol sulfate. However, for these metabolites, we assume that the number of candidate structures would have been higher with OMG than the one reported by MOLGEN using the default valences.

Structure Generation from Elemental Formula and Prescribed Substructures

Structure generation is a combinatorial problem where the number of output molecules grows exponentially with to the number of input atoms. When using one or more prescribed substructures as input to the generators in addition to elemental composition, less candidate structures are obtained (Table 4). Whereas MOLGEN can only accept one substructure, OMG can accept multiple substructures as input with the constraint that these do not overlap, i.e., they should not share any atom.

Phenylalanine is a good example how the number of generated structures can be reduced by using more prescribed substructures, as will be discussed below in more detail.

Structure	Name HMDB ID Elemental Composition	Prescribed substructure(s)	MOLGEN			OMG		
			# Candidate Structures	Time (s)	Time per molecule (ms)	# Candidate Structures	Time (s)	Time per molecule (ms)
	Glycine HMDB00123 C2H5NO2		6	0.167	27.833	6	0.539	89.833
	D-Cysteine HMDB03417 C3H7NO2S		100	0.193	1.930	210	3.177	15.129
	Phenylalanine HMDB00159 C9H11NO2		76,247	52.774	0.692	107,155	19386.019	180.916
			* not possible	* not possible	* not possible	595	271.809	456.822
			* not possible	* not possible	* not possible	289	172.655	597.422
			* not possible	* not possible	* not possible	26	25.147	967.192
	Cholic acid HMDB00619 C24H40O5		** not possible	** not possible	* not possible	334	120.519	360.835
			* not possible	* not possible	* not possible	2,505	119.418	47.672

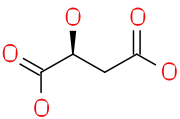

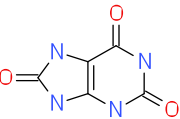
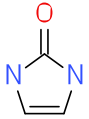
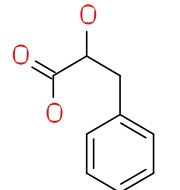

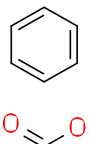
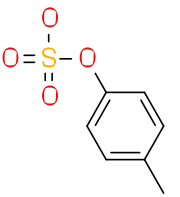
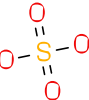

	Malic acid HMDB00156 C4H6O5		1,436	0.229	0.159	1,436	4.688	3.265
	Uric acid HMDB00289 C5H4N4O3		150,114	962.016	6.409	6,069,863	155828.437	25.672
	Phenyllactic acid HMDB00779 C9H10O3		21,040	15.674	0.745	26,164	163.904	6.264
			* not possible	* not possible	* not possible	525	3.973	7.568
	p-Cresol sulfate HMDB11635 C7H8O4S		S_6 13,177	65.667	4.983	13,177	63.047	4.785
			S_6 70,330	94.898	1.349	17,232	1204.357	69.891

Table 4 Number of chemical structures generated by OMG and MOLGEN using as input an elemental composition and one or more prescribed and non-overlapping fragments.

* MOLGEN can only accept one prescribed substructure, while OMG accepts multiple substructures, provided that these do not overlap, this is, they do not share any atom.

** MOLGEN is not able to generate molecules using this large substructure as input. The reason could not be found.

Substructure information is of great relevance for metabolomics experiments involving MSⁿ data, where often the only information available of an unknown metabolite that needs to be identified is the elemental composition and in some cases substructures. Provided that no database entries exist for this experimental information, one is forced to generate the structures via CASE. The inclusion of substructure information brings the list of candidate structures to a manageable size. For p-cresol sulfate, using the sulfate group with both generators as prescribed substructure, produces 13,177 molecules. When benzene is the prescribed

substructure, OMG generates 17,232 candidate structures and MOLGEN 70,330, all containing sulfur with valence 6, hence the difference between both generators.

Whereas only the elemental composition of phenylalanine as input generates 277 million structures with MOLGEN and for OMG an even higher number of candidate structures is expected as both nitrogen valences of 3 and 5 are taken into account (Table 3), using benzene as a substructure provides only 107,155 (OMG) and 76,247 (MOLGEN) candidate structures (Table 4). The number of generated molecules for the elemental composition of phenylalanine is even further reduced by prescribing multiple fragments as input: OMG outputs 595 molecules when provided with two fragments and 289 molecules for three fragments (Table 4). The use of large fragments yields the larger reduction in output molecules, as it can be seen for the last example of phenylalanine, where two big fragments describe most of its structure and return only 26 chemical structures.

For larger molecules containing ten or more carbon atoms, which is a common situation in chemistry, it is not practical for the identification of metabolites to exhaustively generate candidate structures without using substructure constraints, with MOLGEN and OMG, due to the large number of results. Using the elemental composition of a large metabolite like cholic acid, both structure generators cannot produce all possible candidate structures, which are expected in the order of billions. This was only possible using substructure information to reduce the size of the search tree: when providing a substructure that describes a large part of the molecule, OMG generates only 334 structures (Table 4). When using two

substructures, OMG returned 2,505 candidate structures. However, MOLGEN was unable to return results using the same large substructure or two substructures as an input and the reason could not be found by us.

The use of prescribed substructures affected the running time of both generators. For MOLGEN, the time per molecule ranged between 0.16 and 27.8 milliseconds, which represents in some cases a 10,000-fold increase in computation time compared to using only elemental compositions. Concerning OMG, the time per molecule ranged between 3.3 and 967.2 milliseconds, a 100-fold increase in running time. Despite this deterioration of execution time, the advantage of using one or ideally multiple prescribed substructures is clear: the number of candidate substructures is significantly reduced and the total time to calculate candidate structures is also reduced compared to not using any substructure.

The results here presented show that if we want MOLGEN to generate the correct molecule when the valence of some atoms is not the default one, like phosphoenolpyruvic acid or p-cresol sulfate, we need to know the valence in advance. Otherwise, MOLGEN should be executed using all possible valences for all atoms. This limitation is not present in OMG, which can produce different valences in the same execution. Unfortunately, the atom dictionary provided by CDK is not comprehensive concerning non-standard valences. On the positive side, the dynamic open source community of CDK keeps adding new atom types with each release of the library and we expect that this will improve the capabilities of OMG. This open source nature of CDK allows users to suggest or implement new atom types.

The generation of the molecules in the Open Molecule Generator has the shape of a tree. As stated by McKay[23], the check for canonical augmentation is branch-independent, which would allow to process branches of the generation trees in parallel. Theoretically the algorithm allows for parallelization, in practice this has not been implemented but it is one future extension of this work.

However, we have observed that OMG is in most of the cases slower than MOLGEN and this fact was more noticeable when generating millions of candidate molecules. The speed of OMG could be improved and we see several possibilities to achieve this, i.e. the use of a different canonizer or a less computationally demanding canonicity test for intermediate chemical structures, could significantly speed up the execution. Actually, obtaining millions of molecules as a result, quickly or slowly, is not desirable, but ideally, the goal of metabolite identification is to obtain a list of candidate structures that is short in order to examine it and find the structure belonging to the unknown metabolite. Exhaustive profiling, covering both on execution time and memory use, would be beneficial to discover improvement points for OMG. Fortunately, OMG allows multiple prescribed substructures and can handle large fragments, which reduced the number of generated molecules significantly. Handling multiple substructures allows OMG to provide a short list of candidate structures and additionally, its open source nature permits users to implement specific constraints to further reduce the candidate list, both during and after the generation process. Examples of such constraints would reject intermediate chemical structures with high steric energy values or other

physicochemical properties. Therefore we expect OMG to be useful in different application areas and its functionality to be extended in the near future.

Conclusion

In this work we have presented the Open Molecule Generator, to the best of our knowledge, the first implementation to chemical structure generation of the Canonical Path Augmentation approach, originally designed for simple graph enumeration adding vertices. We have adapted it to generate organic chemical structures and extended so that (i) it grows molecules by adding bonds, (ii) it can handle multigraphs, and (iii) accepts one or multiple non overlapping prescribed substructures. In addition, this is the first open source implementation of a deterministic structure generator. This will enable future developments like parallelization or the inclusion of constraints that are specific to the class of compounds being generated.

Our results show that the implementation of our algorithm generates all possible and valid chemical structures for a given elemental composition and optionally prescribed substructures. It is as complete as the best commercially available generator. Moreover, the current implementation of the OMG program presents an extra advantage over existing generators when large or multiple fragments are available to be used as constraints: we have demonstrated the benefit of incorporating constraints to reduce the number of output molecules significantly. The ability of OMG to generate multiple valences for an atom has proven to be useful as often no prior information is known on the desired chemical elements and

multiple valences of an element can be present in a molecule. When compared to MOLGEN, the only disadvantage of OMG is its speed, which is more severe when using only elemental compositions and less when including prescribed substructures. This issue will be addressed in future improvements of the program. We expect this tool to be used in various fields, one of them being metabolomics, where there is a clear need for flexible structure generators. We have successfully used OMG to propose candidate structures using prescribed substructures, in several on-going metabolite identification projects in our lab.

References

1. Kind T, Fiehn O: Advances in structure elucidation of small molecules using mass spectrometry. *Bioanalytical reviews* 2010, 2:23–60.
2. Lindsay RK, Buchanan BG, Feigenbaum EA, Lederberg J: Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project. New York: McGraw-Hill Book; 1980.
3. Carhart RE, Smith DH, Gray NAB, Nourse JG, Djerassi C: GENOA: A computer program for structure elucidation utilizing overlapping and alternative substructures. *Journal of Organic Chemistry* 1981, 46:1708 – 1718.
4. Funatsu K, Miyabayashi N, Sasaki S: Further development of structure generation in the automated structure elucidation system CHEMICS. *Journal of Chemical Information and Modeling* 1988, 28:18–28.
5. Badertscher M, Korytko A, Schulz K-P, Madison M, Munk ME, Portmann P, Junghans M, Fontana P, Pretsch E: Assemble 2.0: a structure generator. *Chemometrics and Intelligent Laboratory Systems* 2000, 51:73–79.
6. Molchanova MS, Shcherbukhin VV, Zefirov NS: Computer Generation of Molecular Structures by the SMOG Program. *Journal of Chemical Information and Modeling* 1996, 36:888–899.
7. Kerber A, Laue R, Grüner T, Meringer M: MOLGEN 4.0. *Match* 1998, 37:205 – 208.
8. Ley S V., Doherty K, Massiot G, Nuzillard JM: Connectivist approach to organic structure determination. LSD-program assisted NMR analysis of the insect antifeedant azadirachtin. *Tetrahedron* 1994, 50:12267–12280.
9. Steinbeck C: SENECA: A platform-independent, distributed, and parallel system for computer-assisted structure elucidation in organic chemistry. *Journal of chemical information and computer sciences* 2001, 41:1500–7.
10. Steinbeck C, Han Y, Kuhn S, Horlacher O, Luttmann E, Willighagen E: The Chemistry Development Kit (CDK): an open-source Java library for Chemo- and Bioinformatics. *Journal of Chemical Information and Computer Sciences* 2003, 43:493–500.
11. Steinbeck C, Hoppe C, Kuhn S, Floris M, Guha R, Willighagen EL: Recent Developments of the Chemistry Development Kit (CDK) - An Open-Source Java Library for Chemo- and Bioinformatics. *Current Pharmaceutical Design* 2006, 12:2111–2120.
12. Nielsen J, Oliver S: The next wave in metabolome analysis. *Trends in biotechnology* 2005, 23:544–6.

13. Scalbert A, Brennan L, Fiehn O, Hankemeier T, Kristal BS, Van Ommen B, Pujos-Guillot E, Verheij E, Wishart D, Wopereis S: Mass-spectrometry-based metabolomics: limitations and recommendations for future progress with particular focus on nutrition research. *Metabolomics* 2009, 5:435–458.
14. Dunn WB, Broadhurst D, Begley P, Zelena E, Francis-McIntyre S, Anderson N, Brown M, Knowles JD, Halsall A, Haselden JN, Nicholls AW, Wilson ID, Kell DB, Goodacre R: Procedures for large-scale metabolic profiling of serum and plasma using gas chromatography and liquid chromatography coupled to mass spectrometry. *Nature Protocols* 2011, 6:1060–1083.
15. Mohamed R, Varesio E, Ivosev G, Burton L, Bonner R, Hopfgartner G: Comprehensive analytical strategy for biomarker identification based on liquid chromatography coupled to mass spectrometry and new candidate confirmation tools. *Analytical chemistry* 2009, 81:7677–94.
16. Zhang T, Creek DJ, Barrett MP, Blackburn G, Watson DG: Evaluation of Coupling Reversed Phase, Aqueous Normal Phase, and Hydrophilic Interaction Liquid Chromatography with Orbitrap Mass Spectrometry for Metabolomic Studies of Human Urine. *Analytical Chemistry* 2012, 84:1994–2001.
17. Schymanski EL, Meinert C, Meringer M, Brack W: The use of MS classifiers and structure generation to assist in the identification of unknowns in effect-directed analysis. *Analytica Chimica Acta* 2008, 615:136–147.
18. Schymanski EL, Meringer M, Brack W: Automated Strategies To Identify Compounds on the Basis of GC/EI-MS and Calculated Properties. *Analytical Chemistry* 2011, 83:903–912.
19. Colbourn C, Read R: Orderly algorithms for graph generation. *International Journal of Computer Mathematics* 1979, 7:167–172.
20. Faradzev IA: Constructive Enumeration of Combinatorial Objects. in *Problèmes combinatoires et théorie des graphes*, University of Paris, Orsay 1978:131–135.
21. Faulon J-L: Stochastic Generator of Chemical Structure. 1. Application to the Structure Elucidation of Large Molecules. *Journal of Chemical Information and Modeling* 1994, 34:1204–1218.
22. Kerber A, Laue R: Group Actions, Double Cosets, and Homomorphisms: Unifying Concepts for the Constructive Theory of Discrete Structures. *Acta Applicandae Mathematicae* 1998, 52:63–90.
23. McKay B: Isomorph-Free Exhaustive Generation. *Journal of Algorithms* 1998, 26:306–324.
24. Fink T, Reymond J-L: Virtual exploration of the chemical universe up to 11 atoms of C, N, O, F: assembly of 26.4 million structures (110.9 million stereoisomers) and analysis for new ring systems, stereochemistry, physicochemical properties, compound classes, and drug discovery. *Journal of Chemical Information and Modeling* 2007, 47:342–53.
25. Blum LC, Reymond J-L: 970 Million Druglike Small Molecules for Virtual Screening in the Chemical Universe Database GDB-13. *Journal of the American Chemical Society* 2009, 131:8732–8733.
26. Chen WL: Chemoinformatics: past, present, and future. *Journal of Chemical Information and Modeling* 2006, 46:2230–55.
27. Rojas-Chertó M, Peironcely JE, Kasper PT, Van der Hoof JJJ, De Vos RCH, Vreeken R, Hankemeier T, Reijmers T: Metabolite Identification Using Automated Comparison of High-Resolution Multistage Mass Spectral Trees. *Analytical Chemistry* 2012, 84:5524–5534.
28. Peironcely JE, Reijmers T, Coulier L, Bender A, Hankemeier T: Understanding and Classifying Metabolite Space and Metabolite-Likeness. *PLoS ONE* 2011, 6:e28966.
29. McKay BD: *Nauty User's Guide (Version 2.4)*. 2009.
30. Braun J, Gugisch R, Kerber A, Laue R, Meringer M, Rücker C: MOLGEN-CID--A canonizer for molecules and graphs accessible through the Internet. *Journal of Chemical Information and Computer Sciences* 2004, 44:542–8.
31. Faulon J-L, Collins MJ, Carr RD: The signature molecular descriptor. 4. Canonizing molecules using extended valence sequences. *Journal of chemical information and computer sciences* 2004, 44:427–36.
32. IUPAC International Chemical Identifier (InChI), Technical Manual [http://www.inchi-trust.org/sites/default/files/inchi-1.04/InChI_TechMan.pdf].
33. Foggia P, Sansone C, Vento M: A Performance Comparison of Five Algorithms for Graph Isomorphism. In *3rd IAPR TC-15 Workshop on Graph-based Representations in Pattern Recognition*. 2001:188–199.
34. Wishart DS, Knox C, Guo AC, Eisner R, Young N, Gautam B, Hau DD, Psychogios N, Dong E, Bouatra S, Mandal R, Sinelnikov I, Xia J, Jia L, Cruz JA, Lim E, Sobsey CA, Shrivastava S, Huang P, Liu P, Fang L, Peng J, Fradette R, Cheng D, Tzur D, Clements M, Lewis A, Souza A De, Zuniga A, Dawe M, Xiong Y, Clive D, Greiner R, Nazzyrova A, Shaykhtudinov R, Li L, Vogel HJ, Forsythe I: HMDB: a knowledgebase for the human metabolome. *Nucleic Acids Research* 2009, 37:D603–610.

35. Fujiwara H, Wang J, Zhao L, Nagamochi H, Akutsu T: Enumerating treelike chemical graphs with given path frequency. *Journal of Chemical Information and Modeling* 2008, 48:1345–57.
36. Imada T, Ota S, Nagamochi H, Akutsu T: Efficient enumeration of stereoisomers of tree structured molecules using dynamic programming. *Journal of Mathematical Chemistry* 2011, 49:910–970.

3 *Understanding and classifying metabolite space and metabolite-likeness*

Published in *PLoS ONE* 2011, 6:e28966.

Understanding and classifying metabolite space and metabolite-likeness

While the entirety of 'Chemical Space' is huge (and assumed to contain between 10^{63} and 10^{200} 'small molecules'), distinct subsets of this space can nonetheless be defined according to certain structural parameters. An example of such a subspace is the chemical space spanned by endogenous metabolites, defined as 'naturally occurring' products of an organisms' metabolism. In order to understand this part of chemical space in more detail, we analyzed the chemical space populated by human metabolites in two ways. Firstly, in order to understand metabolite space better, we performed Principal Component Analysis (PCA), hierarchical clustering and scaffold analysis of metabolites and non-metabolites in order to analyze which chemical features are characteristic for both classes of compounds. Here we found that heteroatom (both oxygen and nitrogen) content, as well as the presence of particular ring systems was able to distinguish both groups of compounds. Secondly, we established which molecular descriptors and classifiers are capable of distinguishing metabolites from non-metabolites, by assigning a 'metabolite-likeness' score. It was found that the combination of MDL Public Keys and Random Forest exhibited best overall classification performance with an AUC value of 99.13%, a specificity of 99.84% and a selectivity of 88.79%. This performance is slightly better than previous classifiers; and interestingly we found that drugs occupy two distinct areas of metabolite-likeness, the one being more 'synthetic'

and the other being more ‘metabolite-like’. Also, on a truly prospective dataset of 457 compounds, 95.84% correct classification was achieved. Overall, we are confident that we contributed to the tasks of classifying metabolites, as well as to understanding metabolite chemical space better. This knowledge can now be used in the development of new drugs that need to resemble metabolites, and in our work particularly for assessing the metabolite-likeness of candidate molecules during metabolite identification in the metabolomics field.

The area of ‘Metabolomics’ is relatively young [1, 2] and describes the large-scale analysis of (often human and endogenous) metabolites. It comprises both the analytical approaches employed, such as mass spectroscopy (MS) as well as the analysis of the resulting data on a network- and phenotype level. Metabolomics is a particularly interesting research field as it allows the determination of biological phenotypes on a chemical basis, since endogenous metabolites are closer phenotype of an organism than for example gene expression [3]. As a consequence, new knowledge on biological processes can be obtained by investigating metabolites.

Various experimental techniques, most commonly MS and nuclear magnetic resonance (NMR), have been devised to detect and identify metabolites, with different approaches being necessary to cover different parts of the metabolite spectrum. In practice it is found that some metabolites with different lipophilicity can only be detected by one of the experimental techniques but not by others [4–9]. Different techniques might also be used depending on the type and quantity of sample to be analyzed, as well as the concentration and the molecular properties of

the metabolites. In general terms, NMR allows for a detailed characterization of the chemical structure of the (un)known compound, and it is the preferred technique for unambiguous identification of a chemical structure. On the downside, NMR requires abundant and pure sample, yielding low sensitivity. Conversely, MS offers high sensitivity and specificity, requiring less amounts of sample, but providing less information about the chemical structure, namely its elemental composition and some structural fragments.

However, despite its ability to describe a phenotype in many cases in a more relevant manner than other approaches, in metabolomics studies a major challenge exists, namely metabolite identification [10–12]. While many endogenous metabolites can be detected (and their spectrum determined), also elucidating their chemical structures is essential to properly interpret results, and to utilize the analytical data to finally answer biological questions [13]. However, the step from the analytical readout to the structural formula is often fraught with problems.

In the commonly employed MS-based profiling approaches (which are also used in our group), once metabolites are detected their elemental composition (or multiple elemental compositions) [14, 15] can be derived directly from MS data. Based on this elemental composition, matching chemical structures can be proposed following two approaches. In the first approach, molecular databases are queried for the presence of molecules with the same elemental composition (or similar spectral data), and hits are returned as candidate structures [13, 16]. However, the major shortcoming of this approach is that one can only find in databases what has been found before,

making the elucidation of novel metabolites impossible. In the second approach, which is meant to cover this shortcoming, the elemental composition and optionally other experimental data are provided to a 'structure elucidator', which will generate *in silico* all possible chemical structures which match the analytical constraints provided to the algorithm [17–19]. While one of the structures generated will be the metabolite of interest, depending on the elemental formula provided, the latter method in particular yields a large number of possible solutions. (For example, the elemental composition of phenylalanine, C₉H₁₁NO₂, yields 277,810,163 possible candidate structures.)

Due to the above reasons, molecular databases compiling structural information on endogenous metabolites are currently limited in size and they certainly do not cover metabolite space exhaustively. The number of possible metabolites is yet unknown [20]. While lipids alone are estimated to exist in the order of 20,000 different structures [21] plants are thought to contain around 200,000 metabolites [3]. Given these figures, the experimental data obtained until today is relatively scarce. A large database of metabolites such as the Human Metabolome Database (HMDB) [21] contains in its current version about 8,000 structures, which is only a fraction of the above numbers. Still, HMDB is the most comprehensive dataset to represent the Metabolite Space from a human point of view. Plant metabolomics makes use of different databases [12]. In addition metabolomics databases exist [22] that contain metabolites and the enzymatic reactions that connect them to pathways, such as in KEGG [23]; some databases contain metabolites grouped by organism such as in BioCyc [24] and other database relate metabolites with experimental information,

such as Metlin [25]. Still, given its number of data entries, the approach to match the MS or NMR spectrum to database spectra can only succeed in a fraction of cases.

Hence, solutions need to be ranked, based on the likelihood of a molecular structure to be a metabolite [26] – and, as we will outline in more detail below, this is one of the main aims of the current work of implementing a ‘metabolite-likeness’ model. In addition, our goal was to understand metabolites better from a chemical point of view, and this is what we will discuss in the remainder of this work, after setting our approach in context with the ‘prior art’ in the field of metabolite classification.

Focusing on metabolites of *E. coli*, Nobeli et al. [27] studied 745 metabolites of this organism by analyzing physiochemical descriptors, the diversity of scaffolds, and similarity-based compound clustering. It was observed that most of the *E. coli* metabolites are found between the 100 and 300 Da molecular weight region, that they contain up to 20 heavy atoms, and that they are mostly hydrophilic. In addition the low diversity of molecular scaffolds was observed. The clustering analysis performed revealed that it is difficult to use molecular similarity to group metabolites in ‘sub-classes’, since there is not a natural separation according to their two-dimensional structure similarity, concluding that the metabolite space of *E. coli* is homogeneous.

While Nobeli et al. focused on the metabolome of *E. Coli*, Gupta et al. [28] represented the chemical space of metabolites using the KEGG/LIGAND database, which includes metabolites from different species as well as xenobiotics. The

chemical space of non-metabolites was approximated by ZINC database [29], which contains small molecules that are commercially available. These molecules are often used as the search space, in virtual-screening research, or as background set in classification projects.

In this work it was concluded that hydroxyl groups, aromatic systems, and molecular weight are discriminating features between metabolite and non-metabolite chemical space. Furthermore, Self Organizing Maps (SOM), Random Forests (RF), and Classification Trees (CT) were employed to distinguish between the two classes of compounds, which were represented by 3D descriptors, topological descriptors, and global molecular descriptors, respectively. The best classification accuracy was 97%, achieved by the combination of RF and global molecular descriptors. (No external validation of such models is reported in their work, as opposed to our novel study, which includes a prospective validation set.)

While trying to discriminate metabolites from non-metabolites was the obvious starting point, it was then noted that also bioactive compounds, notably drugs, could be related to the metabolite/non-metabolite chemical spaces. All three of those sets were hence analyzed by Dobson et al. in a subsequent study [30]. Endogenous metabolites were selected from the HMDB, BioCyc, BiGG, and Edinburgh databases while drugs were compiled from DrugBank and KEGG DRUG. In addition, screening molecules from ZINC were the source for the background compound set. Molecules were represented using connectivity and path fingerprints, MDL Public Keys and E-state, and the similarity between them was determined by the Tanimoto coefficient.

In this work the authors concluded that *drugs are more similar to metabolites than to screening compounds*. Furthermore the distribution of molecular properties among the different families of compounds was studied and it was noticed that metabolites tend to have fewer heavy atoms than the other two groups of compounds. Another relevant physicochemical property identified was lipophilicity, which showed a bias in metabolites towards hydrophilicity, whereas drugs and screening compounds were more hydrophobic.

In the current study we are extending previous work by, compared to Gupta et al., focusing on a large set of human metabolites obtained from HMDB, instead of metabolites from multiple species, and an updated collection of background compounds from ZINC. We make use of different molecular descriptors such as ECFP_4 [31], FCFP_4, MDL Public Keys [32], and physicochemical properties, as well as classifiers like Support Vector Machines (SVM) [33], Random Forest (RF) [34] and Naïve Bayes (NB) [35] and evaluate their applicability to distinguishing metabolites from non-metabolites. In addition we include a prospective validation set to further assess model performance. Furthermore, Dobson et al. used molecular similarity to metabolites as an indicator of metabolite-likeness. In comparison, we assign our score based on the predictions given by different classification methods. The classifier presented here employs, at the time of publication, the most comprehensive collection of human metabolites and purchasable compounds. Furthermore we also make use of PCA and hierarchical clustering to understand which physicochemical properties as well as chemical functionalities are characteristic of metabolites, and discriminate them from non-metabolites. The

principal aim of this work is to establish a reliable metabolite classifier for candidate structures that need to be identified in metabolomics studies; however, apart from the classifier itself, also understanding metabolite space better was a second major aim of this work.

Methods

Datasets and Data Preprocessing

The Human Metabolome Database (HMDB) version 2.5 [21] served as source of the metabolite set. This database contains, in its original form, 7,886 human metabolites as determined by experimental analytical methods. The ZINC Database (ZINC) release 8 [29] was chosen to represent non-metabolite chemical space. From the different datasets provided by ZINC, we selected the subset “everything #10” (date 2010-06-17), since it includes 21.6 million compounds and it was the largest set at the time, and, hence, most representative of ‘all’ chemical space.

Molecules from the two datasets were standardized with PipelinePilot Student Edition 6.1 [36] using the 'washing' workflow suggested by Dobson et al. [30], which involved the selection of the largest fragment in the structure, the removal of salts and hydrogen atoms and the standardization of charges and stereochemistry. Because the ZINC database mainly contains molecules with a low molecular weight, a value of 1000 Daltons was set as the maximum molecular weight of any compound, metabolite or not, in this study. While this removes part of chemical space from the metabolite dataset, this step was necessary to avoid molecular weight to appear as a major discriminant between metabolites and non-metabolites

(which would not be relevant in the context of our future application of distinguishing metabolites from non-metabolites in cases of structures with an identical sum formula). Furthermore, when employing fingerprints for classification, the chemical distribution of features (as opposed to the molecular weight) will be used for classification, hence making the classification (in this feature space) size-independent. This filter removed 775 metabolites from the HMDB dataset. Furthermore, the constraint imposed on molecules to contain three or more atoms (in order to retain only small organic molecules in the dataset) removed 65 small molecules and ions from HMDB. Metabolites from HMDB that are considered drugs were also removed from the dataset, based on annotations as drugs in the fields “Taxonomy Family” and “Taxonomy Sub Class” provided by HMDB, removing 92 drugs from the dataset and reducing the metabolite dataset to 6,954 molecules. The number of molecules contained in ZINC was excessively large to perform clustering and classification, concerning the computational resources needed for such tasks, therefore selecting a subset was necessary. Such a subset was randomly selected from ZINC, which contained 194,350 molecules. All of these molecules passed the filtering based on molecular weight and the minimum number of atoms. The last dataset preprocessing step was the removal of metabolites (molecules contained in the HMDB database) from the ZINC dataset, where 8 molecules were removed from the non-metabolite set.

Training and Test Sets

Diversity selection [37, 38] was used in this work to prepare representative compound datasets for metabolites and non-metabolites with the intention of

reducing the bias that overrepresented families of molecules could have on the classification step. This initially appeared particularly crucial since lipids were hugely overrepresented in the HMDB database. After giving it more thought it was noted that this step certainly involves subjective elements since it, on the one hand, removes information about the distribution of data points in the original set. On the other hand, we assumed that there was a significant bias present in particular in the metabolite dataset not only due to 'natural' causes, but also due to the bias introduced by experimental techniques (such as MS and NMR), which are able to detect and identify compounds rather selectively. Hence, we came to the conclusion that close analogues should be removed carefully from the dataset. In this spirit, each dataset was independently clustered using the maximal dissimilarity partitioning algorithm implementation from the 'Cluster Molecules' component from PipelinePilot Student Edition 6.1 [36]. Molecules were represented by ECFP_4 fingerprints and the distance between each pair of molecules was calculated using the Tanimoto coefficient. The maximum dissimilarity of a cluster member to the cluster centre was 0.6, (that is, molecules from the same cluster possess a ECFP_4/Tanimoto similarity of at least 0.4). Finally cluster centers were selected as representatives of each cluster, which yielded 532 representatives for HMDB and more than 12,000 for ZINC. In order to have balanced training datasets for model building (where some algorithms are prone to majority class predictions), 532 random molecules were selected from ZINC. These two subsets of 532 molecules each were used for building the classification models. While these datasets are small, they were intended to remove much of the bias present in the original datasets. We also still made use of the additional compound information available

since from the remaining molecules not included in the training datasets the test set was built, where the remaining 6,422 metabolites as well as 6,422 randomly selected non-metabolites were joined to form an initial test set of 12,844 molecules. Hence, this very large test set was used to evaluate whether model generation with our training dataset assembled in the way just described would produce viable metabolite-likeness models.

Prospective Validation Sets

Predictive models are meant to be applied to novel, unseen molecules, and to estimate the performance on those new molecules the utilization of external validation sets is crucial. In order to determine prospective performance of our model, an external validation set was compiled, which includes 563 metabolites not yet part of HMDB (which were provided by the database curators). After filtering using the standardization protocol described above, the resulting prospective validation set contained 457 metabolites that were not included in any of the previous preprocessing steps (diversity selection, model building, and model evaluation). Furthermore, two other datasets of molecules were assembled for evaluation with the metabolite-likeness model, namely one of drugs, and one of bioactive compounds (as determined by experimental assays). To represent drugs DrugBank release 2.5 (date 23-11-2010) [39] was used, comprising 6,532 molecules. To represent bioactive molecules, ChEMBL [40] release 8 (date 09-12-2010) was employed. Both datasets were normalized using the protocol described above and from the 635,933 compounds in ChEMBL, 6,312 were randomly selected (the DrugBank dataset was used in full due to its smaller size). With these datasets we

evaluated if our metabolite-likeness model is able to detect the biogenic bias of drugs and bioactive compounds in general. With these three prospective validation sets (external validation set, drug set, bioactive compound set) we evaluated our best model, as derived in the parameter exploration, in two different ways. Firstly, the quality of the predictions for metabolites that were not involved at any stage of the model creation by employing an external validation set, was determined. Secondly, we tested the hypothesis that drugs (and, possibly to a lesser extent, bioactive molecules) are more similar to metabolites than to non-metabolites. This hypothesis could either be rejected or not from the distribution of metabolite-likeness scores as assigned by our model.

Molecular Descriptors

Molecular descriptors should be chosen with care depending for which problem they are going to be used [41, 42]. In this case different descriptor sets were used for classification as follows.

a) Atom Counts and Physicochemical Molecular Descriptors

Atom counts and physicochemical descriptors are rather simple, intuitive and easy to interpret by chemists. On the downside, they usually result in poorer classification results than more complex descriptors since no structural information is captured. In this study our descriptor set based on atom counts was called 'Atom Counts' and contained counts of the most common atom types in metabolites, namely H_Count, C_Count, N_Count, O_Count, F_Count, P_Count, S_Count, Cl_Count. 'Atom Counts' descriptors were computed using the component 'Element Count' from PipelinePilot

Student Edition 6.1 [36]. The physicochemical properties used were the Atom Counts descriptors mentioned above together with the following properties: the number of atoms (Num_Atoms in PipelinePilot), a calculated logP value (ALogP), a calculated logD value (LogD), the number of hydrogen donors (Num_H_Donors) and acceptors (Num_H_Acceptors), the number of rotatable bonds (Num_RotatableBonds), the number of rings (Num_Rings), the number of aromatic rings (Num_AromaticRings), a calculated value of solubility (Molecular_Solubility), a calculated value of the polar surface area (Molecular_PolarSurfaceArea), and a calculated value for the minimized energy (Minimized_Energy). All these properties, listed in detail in Table 1, were calculated with the components 'Element Count', 'Calculate Properties', 'ALogP', 'LogD', 'Surface Area and Volume', 'Molecular Energy' as implemented in PipelinePilot Student Edition 6.1 [36].

Descriptors	Properties
Atom Counts	H_Count, C_Count, N_Count, O_Count, F_Count, P_Count, S_Count, Cl_Count
PP_desc	Atom Counts, Molecular_Weight, Num_Atoms, ALogP, LogD, Num_H_Donors, Num_H_Acceptors, Num_RotatableBonds, Num_Rings, Num_AromaticRings, Molecular_Solubility, Molecular_PolarSurfaceArea, Minimized_Energy

Table 2 List of atom counts and physicochemical properties used to describe the molecules of this study. PP_desc include Atom Counts and the listed physicochemical properties.

b) Fingerprints

2D ECFP_X and FCFP_X are “Extended Connectivity” molecular fingerprints where features are descriptions of the neighborhood of the atoms up to a certain distance or radius X. In the ECFP fingerprint the atom identifier is based on the atom type, while in FCFP it is based on the functional class of the atom [31]. In this work, ECFP and FCFP fingerprints with radius 4 were calculated using the component 'Molecular Properties' in PipelinePilot Student Edition 6.1 [36] with the parameter 'Convert

Fingerprint To' set to 'Leave As-Is'. These fingerprints can produce thousands of features for a molecular library, including features that are present in very few molecules, which can easily lead to over fitting. Hence, we folded the fingerprints to a fixed length of 1024 bits, using PipelinePilot Student Edition 6.1 [36] component 'Convert Fingerprint', to an output format of 'Fixed length Array of Bits', 'Fixed Bit Length' of 1024, and 'Output Bit Order' of 'Pack Least-Significant First'.

MDL keys [32] were used as well for classification. MDL Public Keys are a key-based molecular representation defined by the presence or absence of 166 predefined keys, or molecular substructures. Since the size of this key set is only 166 bits, folding is not necessary.

Principal Component Analysis

Principal Component Analysis (PCA) is a mathematical transformation that projects the dataset onto a lower dimension defined by uncorrelated variables, the so-called 'principal components' [43]. Such components are ordered according to the percentage of variance in the dataset that they explain, which means that the first principal component explains the highest variance. We performed a PCA on the training set of metabolites and non-metabolites in order to understand better the nature of the chemistry contained in both classes. PCA was performed using the R library *FactoMineR* [44] and data was standardized to unit variance before analysis.

Hierarchical Clustering

Hierarchical Clustering groups objects together that are close in the particular representation chosen and assigns a hierarchy to the resulting clusters. This grouping can be agglomerative, where initially each object is a cluster by itself and where clusters are subsequently combined, or divisive, where the whole dataset is assigned to a single cluster initially which is then iteratively split into smaller clusters. Furthermore, two other factors determine the output of the clustering, the distance metric between objects and the method used to link two clusters, *i.e.* the method used to calculate the distance between clusters. We have used the agglomerative hierarchical clustering offered by *FactoMineR* [44] on the results of the PCA as described above in combination with an Euclidean distance metric and Ward's linkage method. Finally, the hierarchy of clusters is presented on a dendrogram that needs to be cut at some point to split the clusters. The criteria employed to cut the dendrogram was the default in *FactoMineR*, which splits the clusters at the point of maximal loss of intra-cluster inertia. The clustering results are used to evaluate if some natural grouping emerges from the data; in our case, whether metabolite space actually contains several distinct subspaces.

Classification Trees

Classification trees are machine-learning methods that use a univariate partition to split the dataset in subsets [45]. At each step the data is split using the predicting variable that optimizes a certain criteria. In our case, we make use of conditional inference trees (CIT) as implemented in the R package *party* [46]. Conditional inference trees perform a covariate selection that relies on permutation tests and

statistical significance. Applying CIT to a two-class classification problem can be seen as a binary tree where at each node the dataset is split into two subsets using the covariate that has the strongest association to the response variable. In the case that features are binary fingerprints, the presence or absence of a given feature determines the data split performed. Variables are selected if they maximize the 'purity' of the split, this is, that each subset contains mostly objects of one class. The result is a tree that depicts the best variables to split the data and provides information about relevant variables for each class of objects. In the course of the present study, classification trees were applied particularly to ECFP_4 fingerprints, in order to determine which features distinguish metabolite space from non-metabolite space.

Fragment Analysis

In this part of the work, we further analyzed the fragment composition of metabolite and 'purchasable chemistry' spaces as a means to better understand the composition of (and differences between) both compound spaces. From the point of view of a chemist, molecular fragments are easier to interpret and convey more meaning than a fingerprint or a sensitivity percentage. Therefore we used the component 'Generate Fragments' from PipelinePilot Student Edition 6.1 [36] to enumerate (in PipelinePilot terminology) rings, ring assemblies, bridge assemblies, chains, and Murcko assemblies (scaffolds that contain ring systems and ring systems connected by linkers, but no side chains) [47]. The top 20 most frequent fragments from our two datasets, human metabolites and purchasable compounds were collected and analyzed.

Machine Learning

Three machine-learning algorithms were used to generate the models of metabolite-likeness, namely Support Vector Machines (SVM) [33], Random Forests (RF) [34], and the Naïve Bayes Classifier (NB) [48]. We used the implementations of these algorithms in the statistical software package R [49]. For SVM, we employed the library *e1071* [50], which is an implementation of the standard C++ *libsvm* [51]. As for RF, we opted for the library *randomForest* [52], an R port of the original code of Breiman [34]. Again *e1071* was the library chosen for NB.

SVM is one of the most robust and widely used algorithms in machine learning and it belongs to the class of maximum margin classifiers [33, 53]. In a two-class problem, SVM tries to define a boundary that maximizes the separation between the two classes. Provided the classes are linearly separable, SVM builds a hyperplane with a maximal margin to neighboring objects of the two classes. When the linear separation is not feasible, a kernel function executes a nonlinear mapping of the data to a higher dimension where it can be linearly separated. SVM requires the tuning of two metaparameters, γ , which regulates the level of non-linear behavior of the kernel, and C , the cost of violating the constraints, in order to achieve an optimal performance. The kernel type was set to the default Gaussian Radial Basis Function (RBF). SVMs have been successfully used in molecular classification before, such as for classifying 'drug-likeness' [54, 55].

RF is an ensemble of classification trees [34] in which each tree classifies, or votes, the class of an object given a randomly chosen subset of the full variable set. Many

of such trees are grown (as determined by the variable n_{tree}) and majority voting is used to obtain one final classification result. RF requires the tuning of the metaparameter m_{try} , which determines the number of variables randomly sampled.

The last classification algorithm is the Naïve Bayes algorithm [48], which relies on the assumption that the variable values are conditionally independent of the class label. This strong assumption usually does not hold, but in practice this approach still allows building good models for multidimensional data, as was shown for bioactivity datasets before [56, 57]. Compared to SVM and RF, NB only requires one parameter to be tuned, the cut-off value for the class membership probability (equivalent to changing the choice of the 'prior'), which was however not explored in this work and it was set to its theoretical optimum (it was set to 50% in the case of balanced datasets, as proposed previously) [58]. According to this, a molecule with a predicted metabolite-likeness of 50% or higher is considered to be a metabolite, and with less than 50% metabolite-likeness, a non-metabolite.

Cross Validation and Model Generation

Concerning RF and SVM, k-fold cross validation [59–61] is a recommended method to tune metaparameters and avoid over fitting. We opted to apply a 5 fold cross validation, a previously recommended value for k [62, 63], to the 1,064 molecules in the training dataset. In the case of RF, for each cross validation split a range of values for m_{try} metaparameter were tested, while the number of trees in the forest, n_{tree} , was set to the default value of 500. The m_{try} giving the highest averaged Area Under the Curve (AUC) and smallest classification error was chosen as the optimal value for

building the model. Cross validation was performed in the same fashion for SVM (Table S1 shows the best values obtained for the metaparameters). Once the optimal metaparameters were selected, final RF (RF variable importance of PP_desc descriptors are listed in Table S2, and for MDL Public Keys in Table S3), SVM, and NB models were generated using the complete set of 1,064 molecules in the training dataset. This process of metaparameter determination and model building was performed for each pair of three different classifiers (RF, SVM, and NB) and five molecular representations (PP_desc, Atom Counts, ECFP_4, FCFP_4, and MDL Public Keys), resulting in a total of 15 different classification exercises.

Model Benchmarking

Once the training step was finished, we needed to evaluate what pair of classifier and representation gave the best results on the test set, consisting of an additional 6,422 metabolites as well as 6,422 non-metabolites that were not used at any stage during model training. To evaluate model performance we used sensitivity and specificity values derived from the confusion matrices, together with ROC curves and their associated AUC. After applying the models to the test set, the final step involved classification of the molecules contained to the prospective, external validation sets described above. The distribution of the metabolite-likeness scores for these datasets as well as the percentage of correctly classified compounds are discussed in the Results and Discussion section.

Results and Discussion

PCA and Hierarchical Clustering

PCA was performed to the training set and the loadings and scores plots for the first four dimensions are presented in Figure 1. For this PCA, we focus on physicochemical properties (PP_desc) for the sake of interpretability (PCA results for MDL Public Keys are presented in Figure S1 and Figure S2, and the percentage of variance explained in Table S4). Almost 71% of the variance is explained in the first four components. A slight separation between metabolites and non-metabolites can be observed in the score plots of PP_desc (Figure 1A and Figure 1C). The loadings plots for PP_desc (Figure 1B and Figure 1D) one can see which variables are correlated or inversely correlated with each class of compounds. For the first two dimensions (Figure 1B), the variables that contribute the most to the variance are Molecular Solubility, Molecular Weight, Molecular Polar Surface Area (PSA), and the number of carbon atoms per molecule (C_Count). Metabolites hence tend to have higher water solubility, lower molecular weight, and fewer carbon atoms than non-metabolites. These observations are in accordance to the work of Nobeli et al. [27] and Dobson et al. [30], who concluded that metabolites are hydrophilic and have less heavy atoms than non-metabolites. PSA tends to be bigger than the one of non-metabolites, suggesting that metabolites do not penetrate cell membranes as efficiently as the non-metabolites. Furthermore, the loadings plot for the third and fourth dimensions (Figure 1D), shows that the most contributing variables are Num Rings, Num Rotatable Bonds, N Count, S Count, and Minimized Energy. The number of rings, rotatable bonds, and minimized energy, for which metabolites obtain lower

values than non-metabolites, are indicators of molecular complexity, and, therefore, one can conclude that metabolites have simpler chemical structures than non-metabolites. Interestingly, metabolites also have fewer nitrogen and sulfur atoms than non-metabolites, as is the case for all atom types except for oxygen and phosphorus, which are more frequent for metabolites as opposed to non-metabolites.

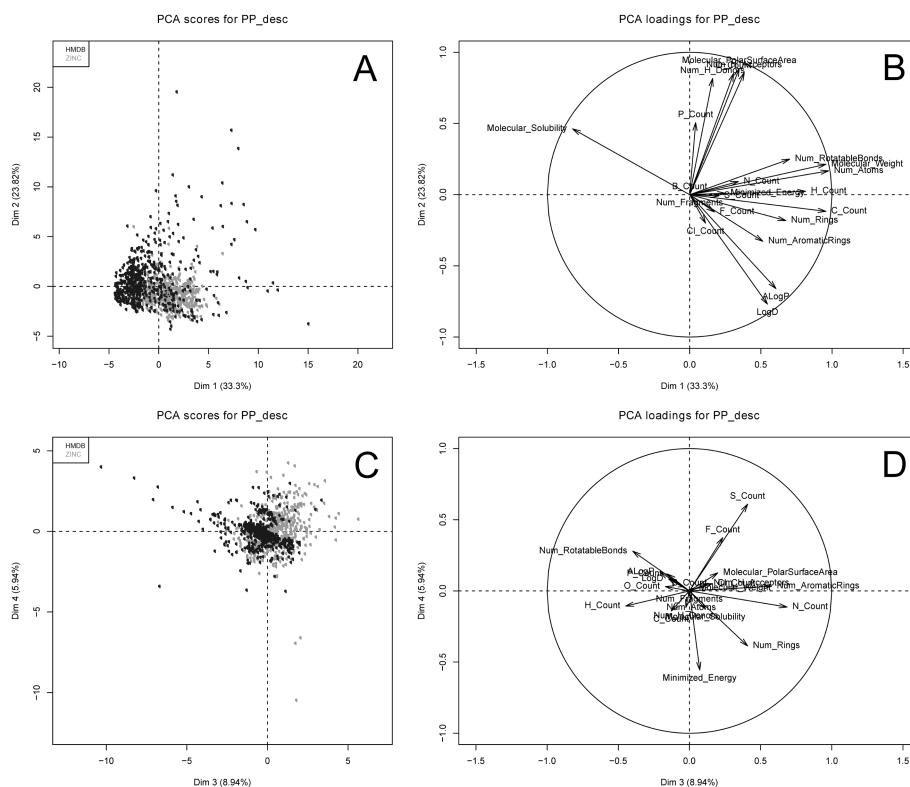


Figure 1 Principal Components Analysis of the PP_desc training set. PCA plots (A,C) and variable contributions (B,D) for the training datasets PP_desc.

The results of the PCA of PP_desc and MDL Public Keys were subject to hierarchical clustering. (Plots are presented in Figure S3). In both cases the optimal cluster split, according to the loss of intra-cluster inertia, returned 3 clusters. The distribution of metabolites and non-metabolites in each cluster is listed in Table 2. It can be seen that for PP_desc and MDL Public Keys 2 large clusters and a third small one are formed, each of them containing one dominant class of compounds. The first cluster

for PP_desc has a purity of 70.2% (370 metabolites and 157 non-metabolites), the second cluster has a purity of 89.65% (52 metabolites and 6 non-metabolites), and the third cluster has a purity of 77.03% (110 metabolites and 369 non-metabolites). Using MDL Public Keys, the first cluster has a purity of 78.81% (372 metabolites and 100 non-metabolites), the second cluster has a purity of 73.03% (134 metabolites and 363 non-metabolites), and the third cluster has a purity of 72.63% (26 metabolites and 69 non-metabolites). However, the purity of each cluster is not high and this, together with the lack of separation observed in the PCA, leads us to think that the separation of metabolites from non-metabolites requires the utilization of more sophisticated methods like random forests, or other nonlinear classifiers as explored in the following.

Cluster	Type	PP_desc	MDL Public Keys
1	HMDB	370	372
1	ZINC	157	100
2	HMDB	52	134
2	ZINC	6	363
3	HMDB	110	26
3	ZINC	369	69

Table 2 Cluster distribution of the molecules in the training datasets, using PP_desc and MDL Public Keys. The clustering performed was a hierarchical clustering and the dendrogram was cut at the point of maximal inertia loss.

Fingerprint Features and Fragment Analysis

A classification tree was built upon the training set, which was described using non-hashed ECFP_4 fingerprints (Figure 2). The results give a general idea of which chemical moieties are characteristic of each class of compounds. As expected, the most discriminating feature was the hydroxyl group, in agreement with the work by Gupta et al. [28], with a higher frequency among metabolites. On the other hand,

the presence of chemical moieties containing nitrogen, in particular secondary amines and secondary imines, is highly correlated with a class membership of the non-metabolites. Finally, in the case a molecule lacks hydroxyl functionalities (demonstrated to be metabolite-like moieties), but it also lacks five or three member rings, ether-like features, and primary amines, it will likely be a metabolite (which is the combination of features in the left-most branch of the tree in Figure 3).

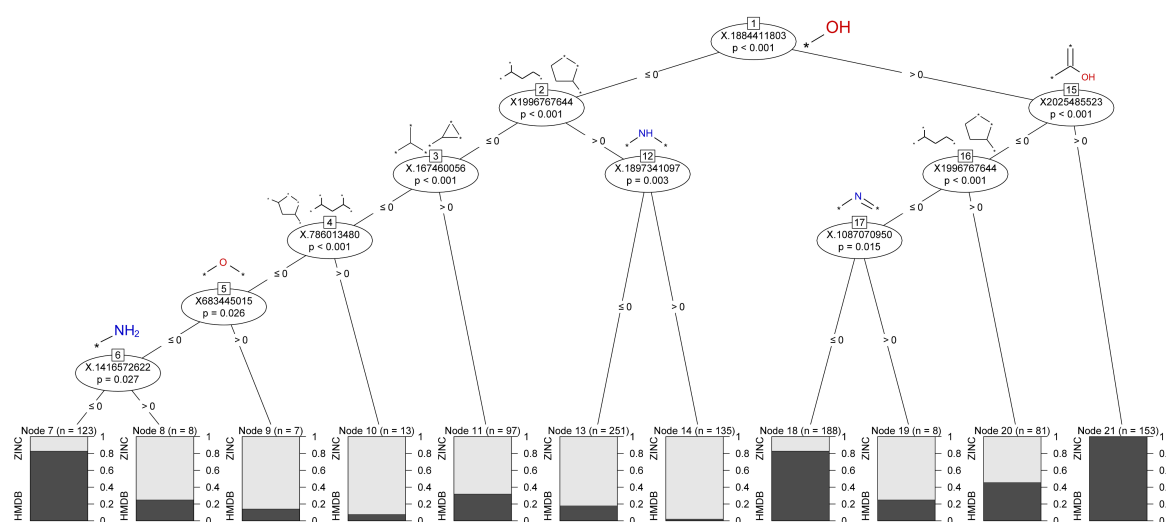


Figure 2 Conditional inference tree of the ECFP_4 features in the training set. Hydroxyls, carboxylic acids, and linear structures are associated with metabolites, whereas secondary amines and secondary imines are associated with non-metabolites.

When looking at the frequent fragments of metabolites (Figure 3) and non-metabolites (Figure 4), we corroborate this finding. Among metabolites, hydroxyls and carboxylic acids are frequent as well as rings containing oxygen atoms. In the case of non-metabolites, either rings or linear fragments containing nitrogen and sulfur abound, which is in accordance to the classification tree results, in accordance to the findings of Hert et al. [64]. Other frequent fragments of metabolites are the phosphate group, characteristic of some classes of metabolites like nucleotides and phospholipids, as well as the steroid and adenine scaffolds. This importance of class-specific fragments can make two metabolites from different classes very different,

and it hence poses a challenge when building models that aim to capture such diversity within a given class. One option is to build local models for each subclass of metabolites; but in this study we aimed at building a global model for metabolites, and as a result, we rely on complex classifiers to predict the metabolite-likeness of molecules. These classification models were built using the methods and data described in the methods section and they were applied to our test set.

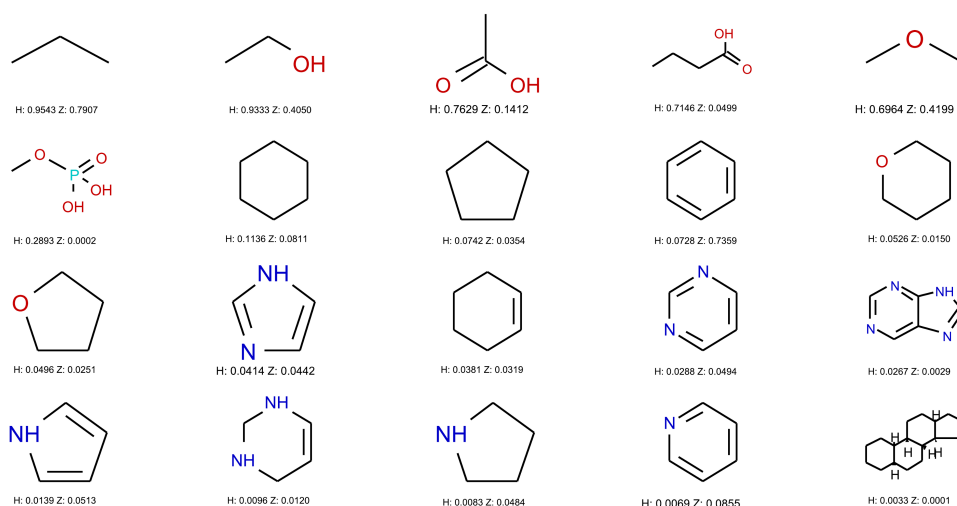


Figure 3 Top 20 most frequent fragments in HMDB. The 20 most frequent ring systems, chain assemblies, and Murcko assemblies in the metabolite data set (HMDB compounds). ‘H’ refers to the frequency of fragments in the HMDB dataset, ‘Z’ to the frequency of fragments in the ZINC dataset. Fragments with less than 4 heavy atoms were excluded. Oxygen containing rings, phosphate group, hydroxyl, carboxylic acid, and the steroid scaffold, among others, are common fragments in metabolites.

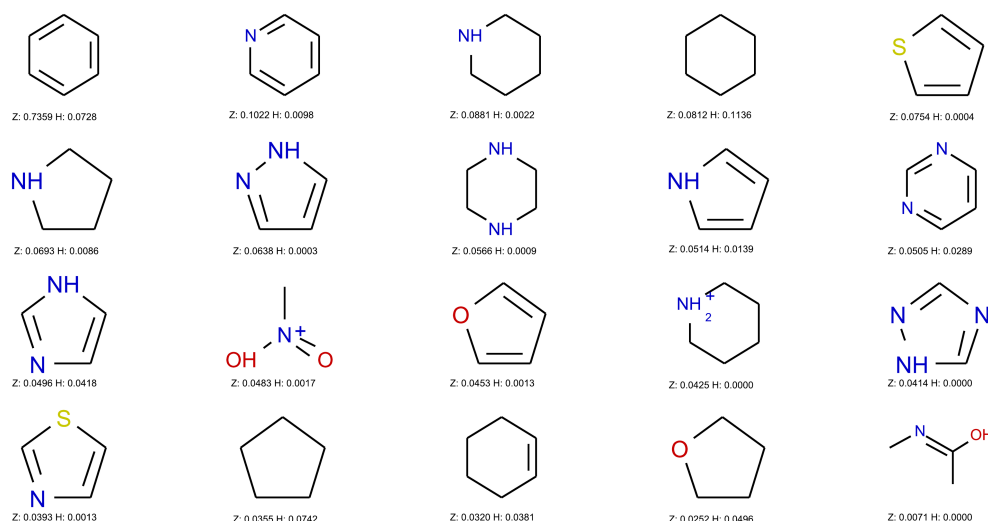


Figure 4 Top 20 most frequent fragments in ZINC. The 20 most frequent ring systems, chain assemblies, and Murcko assemblies among the ZINC compounds, here chosen as a non-metabolite-like set. ‘H’ refers to the

frequency of fragments in the HMDB dataset, 'Z' to the frequency of fragments in the ZINC dataset. Fragments with less than 4 heavy atoms were excluded. Nitrogen containing rings dominate the most frequent fragments.

Test Set

In this study we used 5 molecular representations and 3 classifiers. Our aim was to select which combination of molecular representation and classifier yielded the best classification results for metabolites. The classification results on the test set for each combination are presented in Table 3 and visualized graphically in Figure 5. MDL Public Keys and RF, reporting 99.84% sensitivity and 88.79% specificity, achieve best results. ECFP_4 is the best performing molecular representation when used with SVM, achieving 99.55% sensitivity, while PP_desc achieves the highest AUC of 98.66%. MDL Public Keys also outperformed the other representations for NB, with a sensitivity of 96.71%, specificity of 86.97%, and an AUC of 97.99%. Another representation that exhibits a solid performance across the whole study is ECFP_4 (which is in line with previous studies [65, 66]). This fingerprint has the best sensitivity for SVM, 99.55%, the second best AUC for RF, 99.07%, and the second best sensitivity, 97.15%, and AUC, 94.25% for NB. A conceptually related fingerprint, namely FCFP_4, shows surprisingly worse performance than MDL Public Keys and ECFP_4 fingerprints by having smaller AUC values for RF, SVM, and NB, 98.16%, 94.19%, and 80.80% respectively. Molecular descriptors, both PP_desc and Atom Counts, perform well: PP_desc reports better AUC for RF and SVM, 98.93% and 98.66% respectively, than FCFP_4, 98.13% and 94.19% respectively. Atom Counts descriptors also outperform FCFP_4 in SVM in terms of AUC, 98.02% the former and 94.19% the latter. On the other hand, PP_desc and Atom Counts underperformed when used with NB, where the AUC obtained was 61.57% and 58.95%, respectively.

	Random Forest			SVM			Naïve Bayes			Average		
	Sensitivity	Specificity	AUC	Sensitivity	Specificity	AUC	Sensitivity	Specificity	AUC	Sensitivity	Specificity	AUC
PP_desc	99.17%	88.60%	98.93%	96.82%	88.93%	98.66%	42.51%	86.56%	61.57%	79.50%	88.03%	86.39%
Atom Counts	97.91%	85.57%	97.33%	98.05%	84.10%	98.02%	36.66%	92.90%	58.95%	77.54%	87.52%	84.77%
ECFP4	99.80%	86.27%	99.07%	99.55%	83.43%	98.23%	97.15%	83.29%	94.25%	98.83%	84.33%	97.18%
FCFP4	99.55%	87.84%	98.16%	81.89%	86.53%	94.19%	99.75%	44.80%	80.80%	93.73%	73.06%	91.05%
MDL	99.84%	88.79%	99.13%	98.54%	86.48%	97.45%	96.71%	86.97%	97.99%	98.36%	87.41%	98.19%
Average	99.26%	87.41%	98.52%	94.97%	85.90%	97.31%	74.56%	78.90%	78.71%	89.59%	84.07%	91.52%

Table 3 Classification results of the test set. Results for the test set, including the percentage of correctly classified metabolites (Sensitivity), the percentage of correctly classified non-metabolites (Specificity) and the Area Under the Curve (AUC). It can be observed that the best combination of descriptor and classifier is MDL Public Keys and Random Forest and that the second best is ECFP_4 fingerprints and Random Forest. Interestingly, physicochemical descriptors (PP_desc) perform well both with Random Forest and Support Vector Machines classifiers. (A molecule is considered metabolite if its metabolite-likeness > 50%)

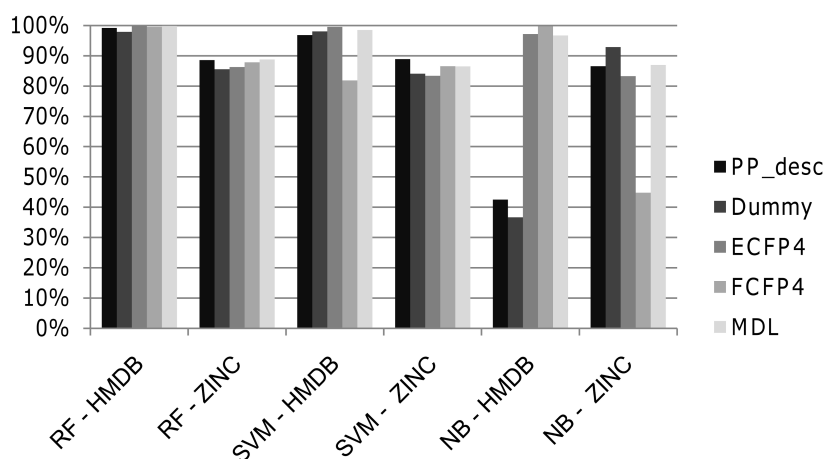


Figure 5 Classification accuracy on the test set. Percentage of correctly classified molecules of the test set for each combination of fingerprint and classifier. Sensitivity is in most cases larger than 90%, except for FCFP_4 and SVM, and Atom Counts and PP_desc and NB. Specificity is larger than 80% in most cases, except FCFP_4 and NB. It can be observed that metabolites are classified more accurately than non-metabolites when using RF and SVM.

By looking at the average AUC results for the different representations we conclude that MDL Public Keys (with 98.19%) and ECFP_4 (with 97.18%) are the best performing representations overall. If we observe the average results obtained by the classifiers, RF outperforms SVM and NB in each category with averages of 99.26% sensitivity, 87.41% specificity, and 98.52% AUC.

From the results presented in this work we see that with the optimal combination of molecular descriptors and classifier, MDL Public Keys and RF, 99.84% of the metabolites and 88.79% of the non-metabolites in the test set are classified correctly. These results are slightly better than those presented by Gupta et al. [28], who reported 97% correct predictions for KEGG metabolites using RF and global molecular descriptors, which are similar to the PP_desc descriptors used in the current work. While these 97% correct predictions were achieved on the dataset used to train the model, our 99.84% correctly classified metabolites were not employed in training the model. Interestingly, it is also observed in our predictions that metabolites have a smaller false positive rate than non-metabolites, which reinforces the idea that it is easier to determine *what makes a metabolite a metabolite*, than what makes a non-metabolite a non-metabolite. The ZINC molecules that have been classified as metabolites (some of them shown in Figure S4), form an interesting set for further research, since according to the models they exhibit metabolite-like features, which would give them an increased likelihood of being bioactive in experimental screening [64].

With respect to the classification algorithms, RF and SVM have demonstrated their status as the ‘state of the art’ in machine learning, as applied to this dataset. This good performance comes however at the expense of having to optimize metaparameters, which is more demanding for SVM, where finding the right gamma and cost results in changing the value ranges multiple times. From this experience, when facing a classification problem where objects are described by a large number of variables and only a modest computational power is available, RF is a good compromise.

As seen in previous research, ECFP_4 is a solid ‘all-round performer’ [65, 66], which obtains good results in combination with the different classification approaches. The most surprising feature is that with simpler molecular representations than ECFP_4, like MDL Public Keys or PP_desc molecular descriptors, one can achieve similar or slightly improved results from the above, as it has been observed before [67]. This finding confirms the idea that (at least known) ‘Metabolite Space’ is a well-defined subset of all ‘Chemical Space’, and that hence its diversity can be modeled with success using either 1D or 2D descriptors.

Apart from the discussion of general model performance we also investigated cases where our model failed, which may be either due to wrong data annotation or wrong predictions of the model. Figure 6 depicts false negative predictions, *i.e.* those metabolites with a metabolite-likeness value of 50% or lower, and which were therefore being considered as non-metabolites in combination with the MDL Public Keys and the RF classification method. Although these molecules would be

considered non-metabolites by our model, 9 out of 10 obtain a metabolite-likeness of 40% or more. It is interesting to note that the lowest scoring compound, debrisoquine with a score of 35.4%, is in fact a drug. Since it was not described as such by the HMDB taxonomy, our filtering step did not eliminate it. The same occurs for entacapone, which is a drug and has a predicted metabolite-likeness of 48.8%. Nevertheless, our classification method was able to assign to both drugs the lowest metabolite-likeness scores. Non-endogenous compounds are also present in this group of compounds, such as nicotine glucuronide, and 4b-Hydroxystanozolol, a metabolite of the synthetic anabolic steroid stanozolol. In the same fashion, we find in this set vanillylamine, with 49% of predicted metabolite-likeness, which is a metabolite of the natural product Vanillin and which structure resembles the endogenous metabolite 4-Methoxytyramine, which obtains a metabolite-likeness score of 48.8%. Unfortunately, some endogenous metabolites like Uroporphyrin II, 3-Methylhistamine, Melatonin, and Vitamin K1 2,3-epoxide, received a low score without an obvious reason, and they are hence false-negative predictions of our model

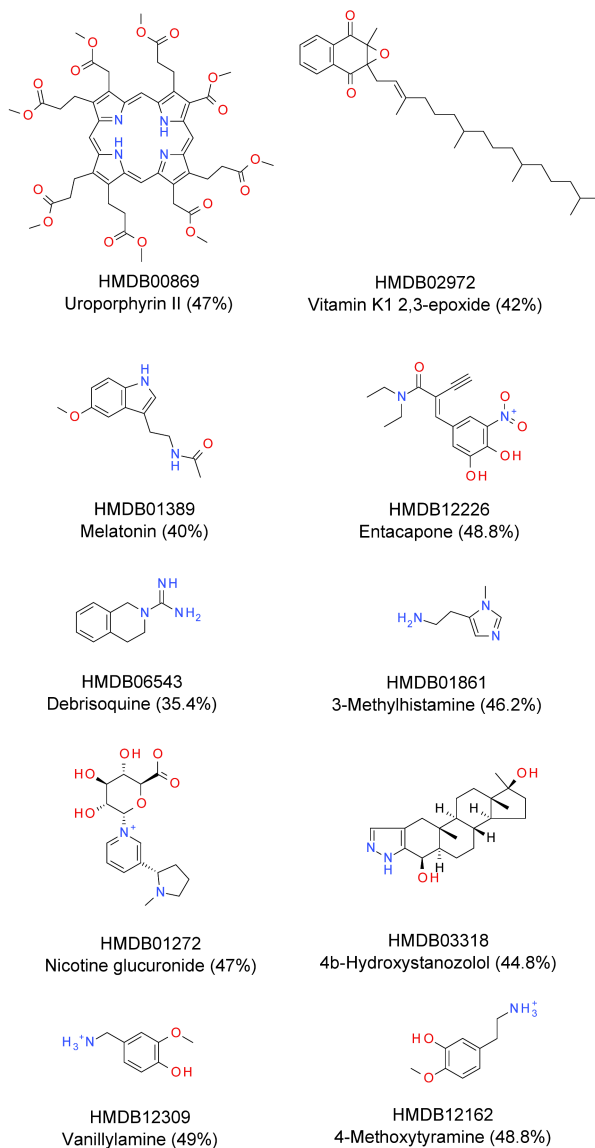


Figure 6 Metabolites in the test set predicted as non-metabolites. The 10 only false negative metabolites from the test set. These metabolites obtained a Metabolite-likeness score smaller than 50%, therefore being classified as non-metabolites, using the best model, MDL Public Keys and Random Forest. Debrisoquine obtains the lowest score; it is a drug that was not taxonomically described as such. 9 out of 10 compounds have 40% or more metabolite-likeness, which is very close to our cut-off used to predict metabolites.

Prospective Validation

Three prospective datasets containing metabolites, drugs, and small molecules, were next classified using our two best performing models, using RF and either MDL Public Keys or PP_desc. The results are displayed in Table 4 and indicate that 95.84% of the new metabolites (obtained after model training has been finished) are correctly

classified as metabolites, indicating the generalizability of our model to classify new data. As for the drugs (represented by DrugBank compounds), 54.37% are assigned a metabolite-likeness of 50% or higher, which is in accordance with our assumption that many drugs indeed resemble metabolites (as has been presented before [64]). For the third dataset, the screening compounds from ChEMBL, molecules predicted to be metabolites only represent 22.39% of the total dataset, hence a smaller percentage than for drugs.

	RF Prediction	
	Metabolites	Non-Metabolites
HMDB_unofficial	95.84%	4.15%
DrugBank	54.37%	45.62%
ChEMBL	22.39%	77.61%

Table 4 Percentage of molecules classified as metabolites or non-metabolites for three independent sets. 95.84% of independent metabolites are correctly classified. More than half of the drugs in DrugBank are considered metabolites. Only 22.39% of the screening compounds in ChEMBL are predicted as metabolites. (A molecule is considered metabolite if its metabolite-likeness > 50%)

In Figure 7 the distributions of metabolite-likeness for each dataset are visualized. We see that most of the new HMDB compounds (HMDB_unofficial) show high values of metabolite-likeness, while the ChEMBL molecules give values that are accumulating at the lower-scoring end of the distribution. The DrugBank molecules on the other hand are evenly distributed among all the metabolite-likeness ranges, with slight peaks at both the metabolite-like, as well as the non-metabolite-like end of the spectrum. This result is in accordance to the work of Ertl et al. [68], where a Natural Product-Likeness score was reported after studying natural products, drugs, and screening compounds. Natural products are molecules produced by living

organisms, and therefore they can be regarded as to some extent similar to the human metabolites we employed in our work. Ertl et al. concluded that drugs are more similar to natural products than screening compounds, a similar finding to what we have presented. This biogenic bias is also present in screening libraries, as presented by Hert et al. [64]; however, the wide spread of drugs along the spectrum of metabolite-likeness (in particular with slight peaks at either end of the scale) has not been previously reported.

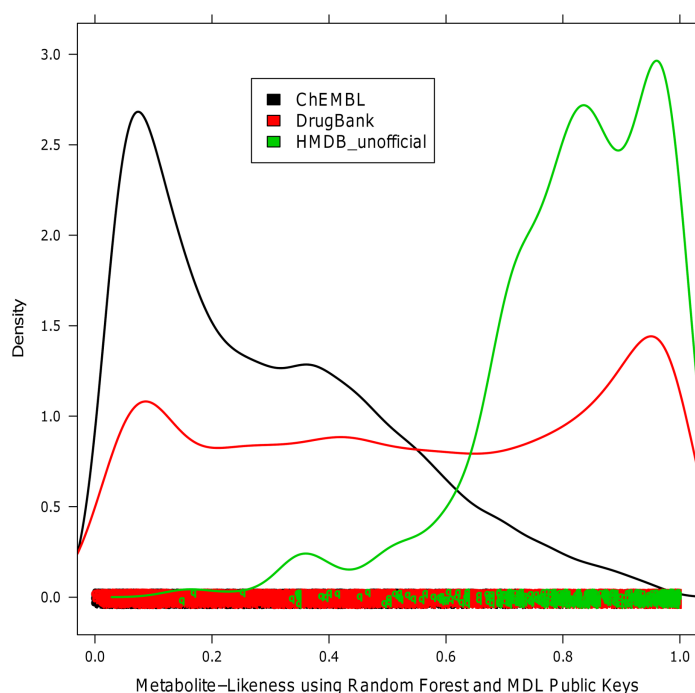


Figure 7 Metabolite-likeness distribution of the prospective validation sets. Distribution of predicted metabolite-likeness for the three classes of molecules in the prospective evaluation set using our best predicting model, RF and MDL Public Keys (namely metabolites from HMDB, drugs from DrugBank and bioactive compounds from ChEMBL). Most of the metabolites are predicted at a metabolite-likeness of 60% or higher. Most of non-metabolites from ChEMBL obtain low values. Drugs from DrugBank are spread across the whole range of values, with higher concentrations at both ends, which indicate a presence of synthetic drugs, for low values, and metabolite-like drugs at high values.

While numerical performance is one thing, the chemical interpretation of model predictions remains crucial. Hence, in order to explore further the results of the prospective validation, molecules of the three different classes (metabolites, drugs, bioactive compounds), which fall into different bins of metabolite-likeness scores,

are presented in Figure 8. The first noticeable feature is the absence of a metabolite with a predicted metabolite-likeness smaller than 10%, underlining the homogeneity of metabolites as a class (as opposed to non-metabolites). As a matter of fact, the metabolite HMDB13193 obtained the lowest metabolite-likeness, 17%, contains two chlorine atoms, which is not common in metabolites. Another interesting situation occurs with molecules that have a steroid scaffold, a common fragment in endogenous metabolites. Metabolite HMDB12524 and drug DB00180 (flunisolide) obtain metabolite-likeness values of 60.6% and 52%, respectively. Here flunisolide possesses a fluorine atom, which is not frequent in metabolites, and which might have hence reduced its metabolite-likeness score. Conversely, ChEMBL compound CHEMBL1163241 also has the steroid scaffold but obtains a score of just 35.2% on the metabolite-likeness scale, corresponding related to having two fluorine atoms and a secondary amine, features that the classification tree revealed to be common in non-metabolites. Finally, examples of compounds with high values of predicted metabolite-likeness are DB00131 (adenosine monophosphate), DB00125 (L-arginine), CHEMBL6422, and CHEMBL14568, which receive 84.2%, 99%, 82.8%, and 96.8% respectively. Adenosine monophosphate includes the phosphate group, frequently found in metabolites together with two hydroxyl groups. Metabolite-likeness features of L-Arginine, like linearity and a carboxylic group, outweigh the non-metabolite features like the nitrogen containing functional groups. Compound CHEMBL6422 possesses a carboxylic acid and hydroxyl functionalities, while and CHEMBL14568 is small, linear, and also exhibits a hydroxyl group, leading to a very high metabolite-likeness score.

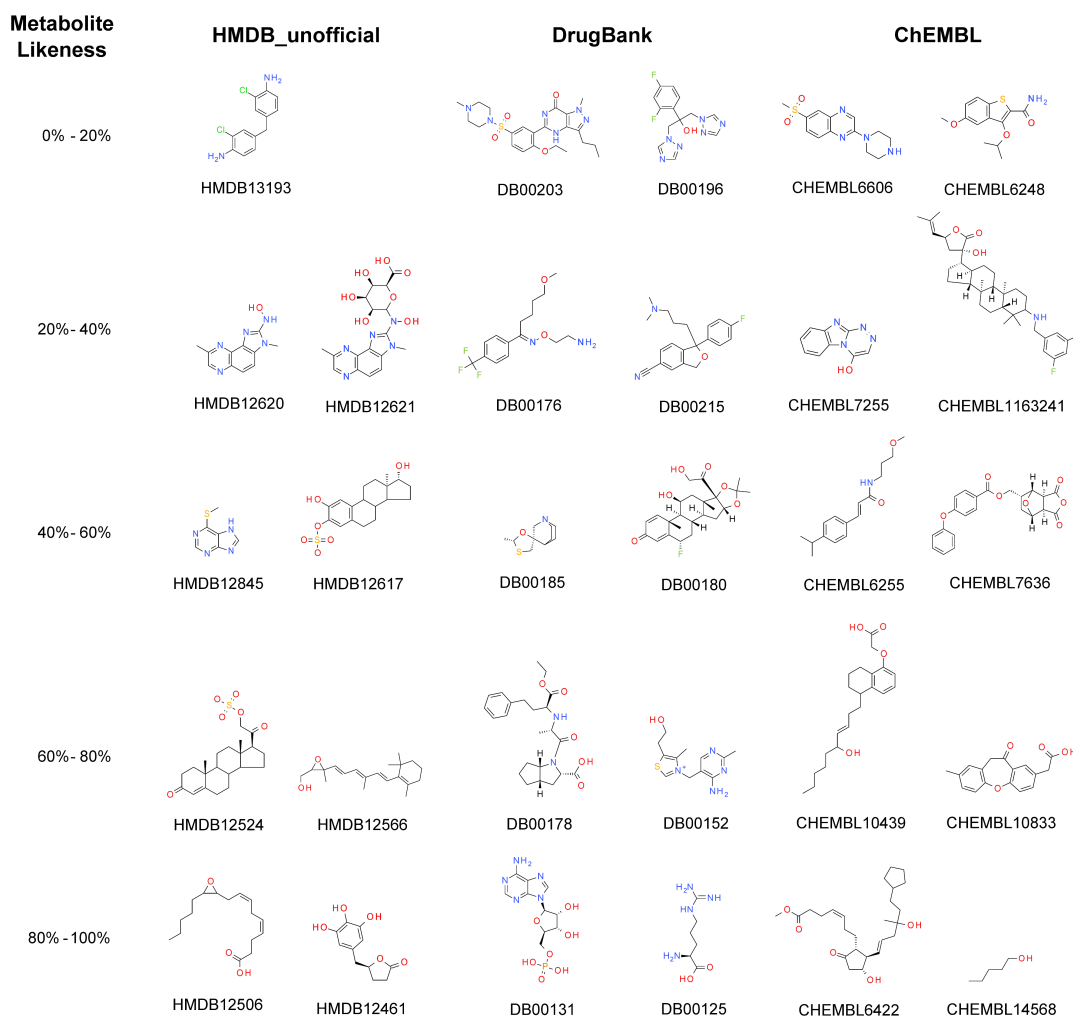


Figure 8 Molecules of the prospective validation sets with different predicted metabolite-likeness values. Compounds of the 3 classes present in the prospective evaluation set using our best predicting model, RF and MDL Public Keys, sorted according to their predicted metabolite-likeness. Non-metabolite compounds exhibit moieties characteristic of metabolites like carboxylic acids and phosphate groups, which make them obtain high values of metabolite-likeness.

The results obtained from the prospective validation demonstrate that our model is successful at identifying whether a molecule is a metabolite or not, which we expect to help studies that involve metabolite identification in the future. Furthermore, metabolite-likeness helps to detect non-metabolites that exhibit features characteristic of metabolites, which can be of interest for drug discovery. In our

future work, we will explore both of those avenues with results to be communicated shortly.

In this work we evaluated various machine-learning models with respect to their ability to discriminate metabolites from non-metabolites, and hence, to calculate the metabolite-likeness score of a given molecule. Our best model detects 99.84% of the metabolites from the test set and 95.84% of the metabolites from a prospective validation set, hence underlining the applicability of the classifier to the majority of novel metabolites. While we confirm that drugs are, on average, more metabolite-like than other compound classes, we noted a considerable spread of drugs across the metabolite-likeness spectrum, with two small (but distinct) peaks at either end of the spectrum, illustrating that both synthetic molecules and metabolite-like compounds may become successful drugs. As for the application side, metabolite-likeness is a tool to rank compounds that 'need' to resemble metabolites, which may be (as above) certain types of drugs, but also in particular candidate structures in metabolite identification. Given the performance of our model, we will now continue with our work to apply our model in precisely those areas. Accordingly, we expect to use this tool in metabolomics studies where no database match is found for the unknown compound and therefore, candidate structures are generated based on mass spectrometry data, e.g. elemental composition, using a structure generation tool. These output molecules would be then ranked according to their Metabolite-Likeness. Furthermore, we have also studied which functional groups, fragments, and physicochemical properties help describe the Metabolite Space. Our findings give a general idea of what metabolites look like, but also encourage us to

look closer at the different subclasses of metabolites and to explore the applicability of a local model approach if we want to expand our knowledge of metabolites.

References

1. German JB, Roberts MA, Fay L, Watkins SM: Metabolomics and the Nutritional Sciences Metabolomics and Individual Metabolic Assessment : The Next Great Challenge for Nutrition. *Journal of Nutrition* 2002;2486 –2487.
2. Nielsen J, Oliver S: The next wave in metabolome analysis. *Trends in biotechnology* 2005, 23:544–6.
3. Hall R, Beale M, Fiehn O, Hardy N, Sumner L, Bino R: Plant metabolomics: the missing link in functional genomics strategies. *Plant Cell* 2002, 14:1437–40.
4. Scalbert A, Brennan L, Fiehn O, Hankemeier T, Kristal BS, Van Ommen B, Pujos-Guillot E, Verheij E, Wishart D, Wopereis S: Mass-spectrometry-based metabolomics: limitations and recommendations for future progress with particular focus on nutrition research. *Metabolomics* 2009, 5:435–458.
5. Lu W, Bennett BD, Rabinowitz JD: Analytical strategies for LC-MS-based targeted metabolomics. *Journal of Chromatography B* 2008, 871:236–242.
6. Wishart D: Quantitative metabolomics using NMR. *TrAC Trends in Analytical Chemistry* 2008, 27:228–237.
7. Dettmer K, Aronov PA, Hammock BD: MASS SPECTROMETRY-BASED METABOLOMICS. *Mass Spectrometry Reviews* 2007:51– 78.
8. Dunn WB, Ellis DI: Metabolomics: Current analytical platforms and methodologies. *TrAC Trends in Analytical Chemistry* 2005, 24:285–294.
9. Lindon J, Nicholson J: Analytical technologies for metabonomics and metabolomics, and multi-omic information recovery. *TrAC Trends in Analytical Chemistry* 2008, 27:194–204.
10. Brown M, Wedge DC, Goodacre R, Kell DB, Baker PN, Kenny LC, Mamas M a, Neyses L, Dunn WB: Automated workflows for accurate mass-based putative metabolite identification in LC/MS-derived metabolomic datasets. *Bioinformatics* 2011, 27:1108–12.
11. Bowen BP, Northen TR: Dealing with the unknown: metabolomics and metabolite atlases. *Journal of the American Society for Mass Spectrometry* 2010, 21:1471–6.
12. Fiehn O, Kind T, Barupal DK: Data Processing, Metabolomic Databases and Pathway Analysis. In *Annual Plant Reviews Volume 43*. Wiley-Blackwell; 2011:367–406.
13. Kind T, Fiehn O: Advances in structure elucidation of small molecules using mass spectrometry. *Bioanalytical reviews* 2010, 2:23–60.
14. Kind T, Fiehn O: Metabolomic database annotations via query of elemental compositions: mass accuracy is insufficient even at less than 1 ppm. *BMC Bioinformatics* 2006, 7:234.
15. Kind T, Fiehn O: Seven Golden Rules for heuristic filtering of molecular formulas obtained by accurate mass spectrometry. *BMC Bioinformatics* 2007, 8:105.
16. Dunn WB, Broadhurst D, Begley P, Zelena E, Francis-McIntyre S, Anderson N, Brown M, Knowles JD, Halsall A, Haselden JN, Nicholls AW, Wilson ID, Kell DB, Goodacre R: Procedures for large-scale metabolic profiling of serum and plasma using gas chromatography and liquid chromatography coupled to mass spectrometry. *Nature Protocols* 2011, 6:1060–1083.
17. Molchanova MS, Shcherbukhin VV, Zefirov NS: Computer Generation of Molecular Structures by the SMOG Program. *Journal of Chemical Information and Modeling* 1996, 36:888–899.
18. Badertscher M, Korytko A, Schulz K-P, Madison M, Munk ME, Portmann P, Junghans M, Fontana P, Pretsch E: Assemble 2.0: a structure generator. *Chemometrics and Intelligent Laboratory Systems* 2000, 51:73–79.
19. Schymanski EL, Meinert C, Meringer M, Brack W: The use of MS classifiers and structure generation to assist in the identification of unknowns in effect-directed analysis. *Analytica Chimica Acta* 2008, 615:136–147.

20. Kind T, Scholz M, Fiehn O: How large is the metabolome? A critical analysis of data exchange practices in chemistry. *PLoS one* 2009, 4:e5440.
21. Wishart DS, Knox C, Guo AC, Eisner R, Young N, Gautam B, Hau DD, Psychogios N, Dong E, Bouatra S, Mandal R, Sinelnikov I, Xia J, Jia L, Cruz JA, Lim E, Sobsey CA, Shrivastava S, Huang P, Liu P, Fang L, Peng J, Fradette R, Cheng D, Tzur D, Clements M, Lewis A, Souza A De, Zuniga A, Dawe M, Xiong Y, Clive D, Greiner R, Nazzyrova A, Shaykhtudinov R, Li L, Vogel HJ, Forsythe I: HMDB: a knowledgebase for the human metabolome. *Nucleic Acids Research* 2009, 37:D603–610.
22. Go EP: Database resources in metabolomics: an overview. *Journal of neuroimmune pharmacology : the official journal of the Society on NeuroImmune Pharmacology* 2010, 5:18–30.
23. Kanehisa M, Goto S, Furumichi M, Tanabe M, Hirakawa M: KEGG for representation and analysis of molecular networks involving diseases and drugs. *Nucleic acids research* 2010, 38:D355–60.
24. Karp PD, Ouzounis C a, Moore-Kochlacs C, Goldovsky L, Kaipa P, Ahrén D, Tsoka S, Darzentas N, Kunin V, López-Bigas N: Expansion of the BioCyc collection of pathway/genome databases to 160 genomes. *Nucleic acids research* 2005, 33:6083–9.
25. Smith CA, Maille GO, Want EJ, Qin C, Trauger SA, Brandon TR, Custodio DE, Abagyan R, Siuzdak G: METLIN: A Metabolite Mass Spectral Database. *Therapeutic Drug Monitoring* 2005, 27:747–751.
26. Schymanski EL, Meringer M, Brack W: Automated Strategies To Identify Compounds on the Basis of GC/ESI-MS and Calculated Properties. *Analytical Chemistry* 2011, 83:903–912.
27. Nobeli I, Pongstingl H, Krissinel EB, Thornton JM: A Structure-based Anatomy of the E.coli Metabolome. *Journal of Molecular Biology* 2003, 334:697–719.
28. Gupta S, Aires-de-Sousa J: Comparing the chemical spaces of metabolites and available chemicals: models of metabolite-likeness. *Molecular Diversity* 2007, 11:23–36.
29. Irwin JJ, Shoichet BK: ZINC: a free database of commercially available compounds for virtual screening. *Journal of Chemical Information and Modeling* 2005, 45:177–82.
30. Dobson PD, Patel Y, Kell DB: “Metabolite-likeness” as a criterion in the design and selection of pharmaceutical drug libraries. *Drug discovery today* 2009, 14:31–40.
31. Rogers D, Hahn M: Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling* 2010, 50:742–54.
32. Durant JL, Leland B a, Henry DR, Nourse JG: Reoptimization of MDL keys for use in drug discovery. *Journal of chemical information and computer sciences* 2002, 42:1273–80.
33. Cortes C, Vapnik V: Support-vector networks. *Machine Learning* 1995, 20:273–297.
34. Breiman L: Random Forests. *Machine Learning* 2001, 45:5–32.
35. Klon AE, Glick M, Davies JW: Combination of a naive Bayes classifier with consensus scoring improves enrichment of high-throughput docking results. *Journal of medicinal chemistry* 2004, 47:4356–9.
36. Accelrys Pipeline Pilot, version 6.1.5; Accelrys Inc.: San Diego, CA, 2010. *Accelrys Pipeline Pilot, version 6.1.5; Accelrys Inc.: San Diego, CA* 2010.
37. Golbraikh A, Tropsha A: Predictive QSAR modeling based on diversity sampling of experimental datasets for the training and test set selection. *Journal of computer-aided molecular design* 2002, 16:357–69.
38. Schuffenhauer A, Brown N: Chemical diversity and biological activity. *Drug Discovery Today* 2006, 3:387–395.
39. Wishart DS, Knox C, Guo AC, Cheng D, Shrivastava S, Tzur D, Gautam B, Hassanali M: DrugBank: a knowledgebase for drugs, drug actions and drug targets. *Nucleic acids research* 2008, 36:D901–6.
40. Warr W: ChEMBL. An interview with John Overington, team leader, chemogenomics at the European Bioinformatics Institute Outstation of the European Molecular Biology Laboratory (EMBL-EBL). *Journal of Computer-Aided Molecular Design* 2009, 23:195–198.
41. Bender A: How similar are those molecules after all ? Use two descriptors and you will have three different answers. *Expert Opinion on Drug Discovery* 2010, 5:1141–1151.
42. Bender A, Glen RC: Molecular similarity: a key technique in molecular informatics. *Organic & biomolecular chemistry* 2004, 2:3204–18.
43. Wold S, Esbensen K, Geladi P: Principal component analysis. *Chemometrics and Intelligent Laboratory Systems* 1987, 2:37–52.
44. Josse J: FactoMineR : An R Package for Multivariate Analysis. *Journal Of Statistical Software* 2008, 25:1–18.

45. Breiman L, Friedman JH, Olshen RA, Stone CJ: *Classification and Regression Trees*. Wadsworth; 1984, p:368.
46. Hothorn T, Hornik K, Zeileis A: Unbiased Recursive Partitioning: A Conditional Inference Framework. *Journal of Computational and Graphical Statistics* 2006, 15:651–674.
47. Bemis GW, Murcko M a: The properties of known drugs. 1. Molecular frameworks. *Journal of medicinal chemistry* 1996, 39:2887–93.
48. Domingos P: On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. 1997, 130:103–130.
49. R Development Core Team: R: A Language and Environment for Statistical Computing. 2010.
50. Dimitriadou E, Hornik K, Leisch F, Meyer D, and Andreas Weingessel: e1071: Misc Functions of the Department of Statistics (e1071), TU Wien. 2010.
51. Chang C-C, Lin C-J: LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2011, 2:27:1–27:27.
52. Liaw A, Wiener M: Classification and Regression by randomForest. *R News* 2002, 2:18–22.
53. Noble WS: What is a support vector machine? *Nature biotechnology* 2006, 24:1565–7.
54. Li Q, Bender A, Pei J, Lai L: A large descriptor set and a probabilistic kernel-based classifier significantly improve druglikeness classification. *Journal of chemical information and modeling* 2007, 47:1776–86.
55. Byvatov E, Fechner U, Sadowski J, Schneider G: Comparison of Support Vector Machine and Artificial Neural Network Systems for Drug/Nondrug Classification. *Journal of Chemical Information and Computer Sciences* 2003, 43:1882–1889.
56. Bender A, Mussa HY, Glen RC, Reiling S: Molecular Similarity Searching Using Atom Environments, Information-Based Feature Selection, and a Naïve Bayesian Classifier. *Journal of Chemical Information and Computer Sciences* 2004:170–178.
57. Bender A, Mussa HY, Glen RC, Reiling S: Similarity searching of chemical databases using atom environment descriptors (MOLPRINT 2D): evaluation of performance. *Journal of chemical information and computer sciences* 2004, 44:1708–18.
58. Provost F: Machine Learning from Imbalanced Data Sets 101. *Invited paper for the AAAI'2000 Workshop on Imbalanced Data Sets*. 2000.
59. Fourches D, Muratov E, Tropsha A: Trust, but verify: on the importance of chemical structure curation in cheminformatics and QSAR modeling research. *Journal of Chemical Information and Modeling* 2010, 50:1189–204.
60. Hawkins DM: The problem of overfitting. *Journal of chemical information and computer sciences* 2004, 44:1–12.
61. Baumann K: Cross-validation as the objective function for variable-selection techniques. *Trends in Analytical Chemistry* 2003, 22:395–406.
62. Kohavi R: A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 1995:1137–1143.
63. Breiman L, Spector P: Submodel Selection and Evaluation in Regression: The X-Random Case. *International Statistical Review* 1992, 60:291–319.
64. Hert J, Irwin JJ, Laggner C, Keiser MJ, Shoichet BK: Quantifying biogenic bias in screening libraries. *Nature Chemical Biology* 2009, 5:479–83.
65. Hert J, Willett P, Wilton DJ, Acklin P, Azzaoui K, Jacoby E, Schuffenhauer A: Comparison of topological descriptors for similarity-based virtual screening using multiple bioactive reference structures. *Organic & biomolecular chemistry* 2004, 2:3256–66.
66. Bender A, Jenkins JL, Scheiber J, Sukuru SCK, Glick M, Davies JW: How similar are similarity searching methods? A principal component analysis of molecular descriptor space. *Journal of chemical information and modeling* 2009, 49:108–19.
67. Bender A, Glen RC: A discussion of measures of enrichment in virtual screening: comparing the information content of descriptors with increasing levels of sophistication. *Journal of Chemical Information and Modeling* 2005, 45:1369–75.
68. Ertl P, Roggo S, Schuffenhauer A: Natural product-likeness score and its application for prioritization of compound libraries. *Journal of Chemical Information and Modeling* 2008, 48:68–74.
69. Rojas-Chertó M, Kasper PT, Willighagen EL, Vreeken RJ, Hankemeier T, Reijmers TH: Elemental composition determination based on MS(n). *Bioinformatics* 2011, 27:2376–83.

70. Rojas-Chertó M, Peironcely JE, Kasper PT, Van der Hooft JJJ, De Vos RCH, Vreeken R, Hankemeier T, Reijmers T: Metabolite Identification Using Automated Comparison of High-Resolution Multistage Mass Spectral Trees. *Analytical Chemistry* 2012, 84:5524–5534.

71. Rojas-Chertó M, Van Vliet M, Peironcely JE, Van Doorn R, Kooyman M, Beek T Te, Van Driel M a, Hankemeier T, Reijmers T: MetiTree: a web application to organize and process high resolution multi-stage mass spectrometry metabolomics data. *Bioinformatics (Oxford, England)* 2012:2–4.

Supplementary materials

	mtry	Gamma	Cost
PP_desc	9	4.768372e-07	32768
Atom Counts	2	4.882812e-04	262144
ECFP4	10	3.051758e-05	16
FCFP4	30	1.907349e-06	8192
MDL	30	1.907349e-06	8192

Table S1 Optimal metaparameters for classifiers. mtry for Random Forest, Gamma and Cost for Support Vector Machines, obtained after performing Cross Validation on the training set.

Understanding and classifying metabolite space and metabolite-likeness

	HMDB	ZINC	MeanDecreaseAccuracy	MeanDecreaseGini
N_Count	1.3067688	1.2609013	1.0014388	46.898305
Molecular_Solubility	1.2788442	1.02582	0.9656276	96.271476
LogD	1.2191826	1.0680953	0.9642439	50.24542
Num_H_Donors	1.3046778	0.8545617	0.9567927	29.876567
Num_RotatableBonds	1.012057	1.17285022	0.8988967	26.85855
Molecular_Weight	1.1004754	0.99654993	0.8885767	31.778377
Minimized_Energy	1.1796855	0.69780848	0.8681832	32.409426
H_Count	1.2011563	0.24332713	0.8579638	24.73855
Molecular_PolarSurfaceArea	1.1207631	0.70494526	0.8549427	23.351115
ALogP	0.9248217	0.72155958	0.7985026	25.724204
Num_Atoms	0.7790244	0.76550378	0.7354873	20.597452
Num_H_Acceptors	0.9853438	0.35682066	0.7293065	10.555502
F_Count	1.0559811	-0.08105052	0.7200118	5.685859
Num_Rings	0.8475657	0.73452855	0.7184914	24.290726
C_Count	0.9386965	0.62885999	0.7131295	41.595612
Num_AromaticRings	0.7248383	0.81406126	0.6912011	22.524017
O_Count	0.8782411	0.45284146	0.6735855	10.741794
S_Count	0.7923043	0.02593264	0.4822221	4.077548
Cl_Count	0.7362878	-0.37414425	0.3897458	2.297109
P_Count	-0.133703	0.35386364	0.1807093	0.980051

Table S3 Importance given to the PP_desc descriptors by Random Forest. High values on Mean Decrease Accuracy and in Mean Decrease Gini indicate that this variable is important to discern between metabolites and non-metabolites. These importance values have been obtained from the Random Forest model built with the training set.

	HMDB	ZINC	MeanDecreaseAccuracy	MeanDecreaseGini
MDLPublicKeys.140	1.24138956	1.229133216	0.941920823	53.91841097
MDLPublicKeys.126	0.83892897	0.70907696	0.663644388	23.62313439
MDLPublicKeys.163	0.78447704	0.746221036	0.640603474	23.23559085
MDLPublicKeys.50	1.17048163	1.023474316	0.885259239	15.04553608
MDLPublicKeys.143	0.69289234	0.696664822	0.593501328	14.91132605
MDLPublicKeys.108	0.99957857	0.489609227	0.713694188	14.48014674
MDLPublicKeys.157	0.63767173	0.675822066	0.603715398	13.07839857
MDLPublicKeys.123	0.58418556	0.407174217	0.474627005	13.02194646
MDLPublicKeys.146	0.87558184	0.687488161	0.703815276	12.6453885
MDLPublicKeys.95	0.79652399	0.624375878	0.637617052	10.20590729
MDLPublicKeys.135	0.90220951	0.439277352	0.637256942	9.92262443
MDLPublicKeys.145	0.62671793	0.411886181	0.50576454	8.80818597
MDLPublicKeys.122	0.56884211	0.176402217	0.42077643	8.25408405
MDLPublicKeys.132	0.70757544	0.285691095	0.519554788	6.57286614
MDLPublicKeys.138	0.5532357	0.466305805	0.478651557	6.54776274
MDLPublicKeys.128	0.57303747	0.41742079	0.494394286	6.48147908
MDLPublicKeys.53	0.8537977	0.309679345	0.609916532	6.05340169
MDLPublicKeys.82	0.72725636	0.393078873	0.547102936	5.88552798
MDLPublicKeys.76	0.46225838	0.21721732	0.368584167	5.84646577
MDLPublicKeys.140	1.24138956	1.229133216	0.941920823	53.91841097

Table S4 Importance given to the MDL Public Keys by Random Forest. High values on Mean Decrease Accuracy and in Mean Decrease Gini indicate that this variable is important to discern between metabolites and non-metabolites. These importance values have been obtained from the Random Forest model built with the training set.

Component	Atom Counts	PP_desc	MDL Public Keys
1	25.44165	33.29614	11.54531
2	44.54251	57.11561	18.74551
3	58.67426	66.05534	23.8788
4	71.17018	71.99137	28.55269
5	82.20865	77.25728	32.71371
6	92.29973	82.01365	36.04002
7	98.71144	86.32577	38.95153
8	100	90.20601	41.51726

Table S5 Cumulative percentage of variance explained of the first 8 principal components. PCA was performed on the Atom Counts, PP_desc, and MDL Public Keys datasets.

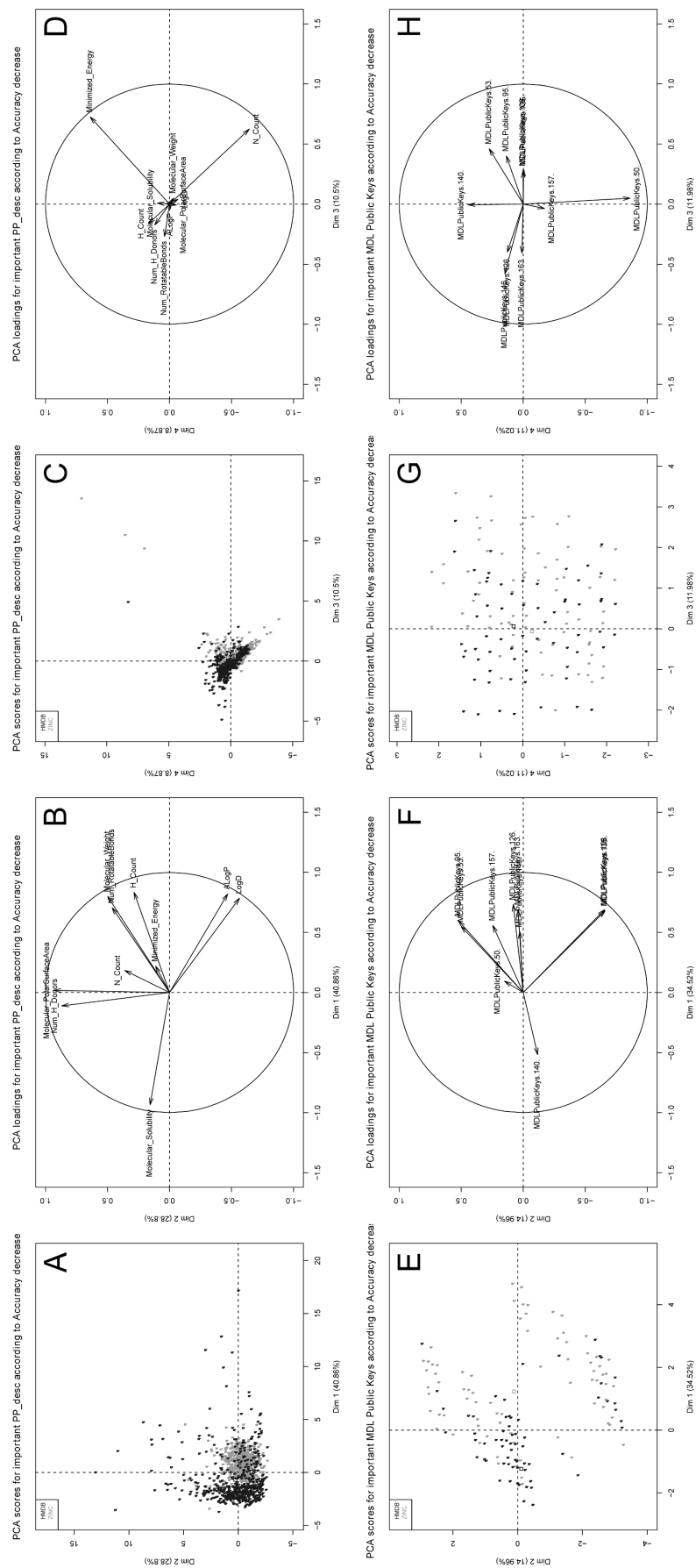


Figure S1 PCA of the PP_desc and MDL Public Keys that the RF model considers important. The importance criterion is the Mean Decrease Accuracy. The separation of both classes is slightly improved for PP_desc using these important variables if compared with the PCA score plot in Figure 1.

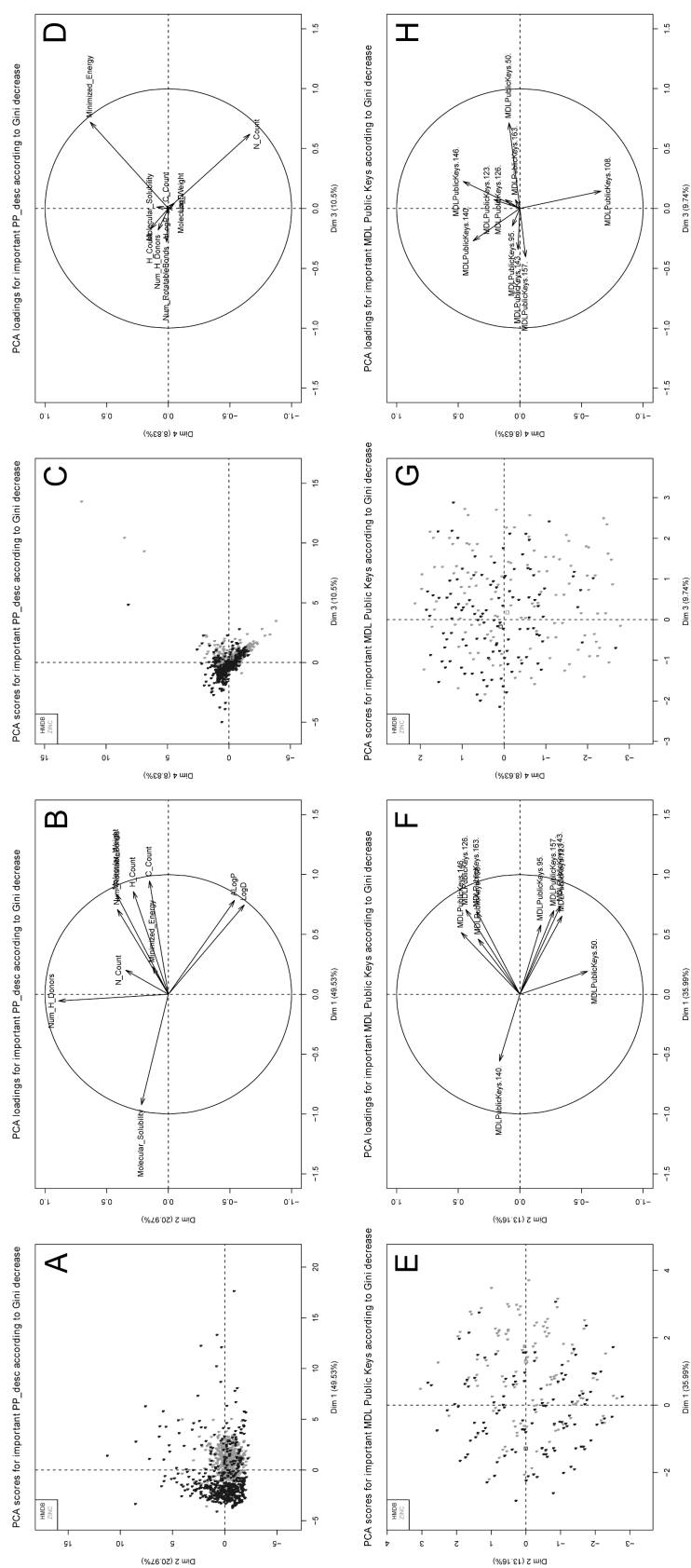


Figure S2 PCA of the PP_desc and MDL Public Keys that the RF model considers important. The importance criterion is the Mean Decrease Gini. The separation of both classes is slightly improved for PP_desc using these important variables if compared with the PCA score plot in Figure 1.

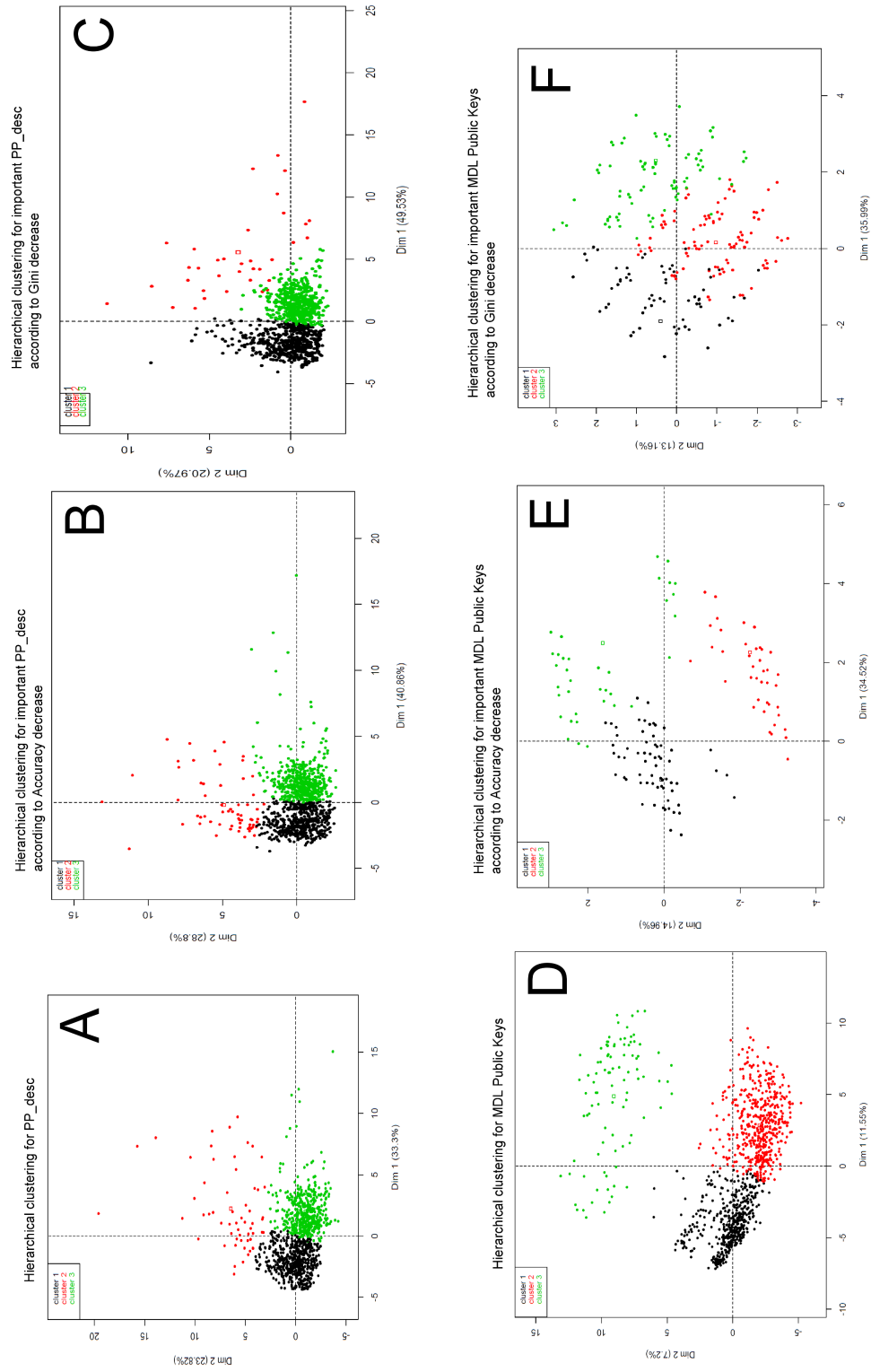


Figure S3 Hierarchical clustering of PP_desc and MDL Public Keys. Plots of the first two dimensions of the Hierarchical Clustering. For PP_desc: A, using all variables; B, using the important variables according to Accuracy decrease; C, using the important variables according to Gini decrease. For MDL Public Keys: A, using all variables; B, using the important variables according to Accuracy decrease; C, using the important variables according to Gini decrease. In all cases the optimal cut of the dendrogram, according to the maximum loss of inertia, returns 3 clusters.

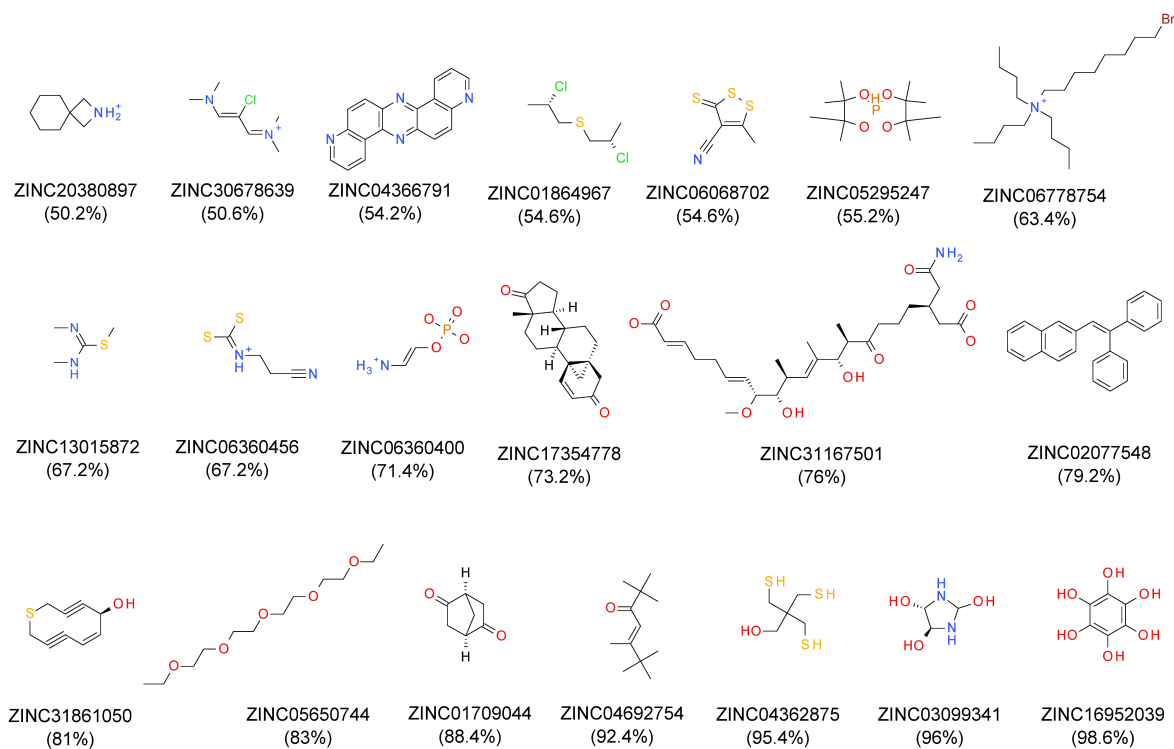


Figure S4 Non-metabolites predicted as metabolites. Some non-metabolites from the test set that obtained a Metabolite-likeness score greater than 50%, therefore being classified as metabolites, using the best model, MDL Public Keys and Random Forest. These are the 20 cluster centers selected from the clustering performed on all the false positives^{S4}.

4 *An automated pipeline for de novo metabolite identification using mass spectrometry-based metabolomics*

Published in *Analytical Chemistry* 2013, **7**:3576–3583.

An automated pipeline for de novo metabolite identification using mass spectrometry-based metabolomics

Metabolite identification is one of the biggest bottlenecks in metabolomics. Identifying human metabolites poses experimental, analytical and computational challenges. Here we present a pipeline of previously developed cheminformatic tools and demonstrate how it facilitates metabolite identification using solely LC-MSⁿ data. These tools process, annotate, and compare MSⁿ data, and propose candidate structures for unknown metabolites either by identity assignment of identical mass spectral trees or by de novo identification using substructures of similar trees. The working and performance of this metabolite identification pipeline is demonstrated by applying it to LC-MSⁿ data of urine samples. From human urine, 30 MSⁿ trees of unknown metabolites were acquired, processed and compared to a reference database containing MSⁿ data of known metabolites. From these 30 unknowns, we could assign a putative identity for 10 unknowns by finding identical fragmentation trees. For 11 unknowns no similar fragmentation trees were found in the reference database. Based on elemental composition only, a large number of candidate structures/identities were possible, so these unknowns remained unidentified. The other 9 unknowns were also not found in the database, but metabolites with similar fragmentation trees were retrieved. Computer assisted structure elucidation was performed for these 9 unknowns: for 4 of them we could perform de novo identification and propose a limited number

of candidate structures, and for the other 5 the structure generation process could not be constrained far enough to yield a small list of candidates. The novelty of this work is that it allows de novo identification of metabolites that are not present in a database by using MSⁿ data and computational tools. We expect this pipeline to be the basis for the computer-assisted identification of new metabolites in future metabolomics studies, and foresee that further additions will allow identifying even a larger fraction of the unknown metabolites.

Metabolomics is the study and characterization of metabolites, which are the small molecules (molecular weight below 1000 Daltons) of an organism, biofluid, tissue, or biocompartment. Metabolites are substrates or products of metabolic processes and therefore describe accurately the phenotype of an organism.[1] Metabolite identification is frequently cited as one of the major bottlenecks in metabolomics.[2–4] Knowing the identity of the metabolites that are relevant in studies is necessary for a proper biological interpretation of the results. This work focuses on Mass Spectrometry (MS) only rather than including Nuclear Magnetic Resonance (NMR) because the former is more sensitive than the latter.

While no agreement exists on how to perform metabolite identification, some guidelines do exist that define how to report identities of metabolites.[5] The highest reporting level is Level 1, where an identity is proposed and validated using two independent and orthogonal data sources relative to an authentic compound analyzed under identical experimental conditions, for instance accurate mass and Multi Stage Mass Spectrometry (MSⁿ) spectra or retention time and m/z or MSⁿ data.

Level 2 is used for putatively annotated compounds, where an identity is proposed based on MS/MS or MSⁿ spectral similarity of the unknown to the spectra of a known compound present in a database, but the identity is not validated with chemical reference standards. Level 3 includes putatively annotated compound classes, based on spectral similarity of the unknown to known compounds belonging to a certain chemical class. Level 4 includes unknown compounds that can be traced and quantified using spectral data in different experiments, but no structural information has been reported before. At the beginning of an identification project the unknown compounds can be divided into “*known unknowns*” and “*unknown unknowns*”. [6] A *known unknown* is a compound that has been previously described for a certain analytical platform, for instance by a certain mass and retention time window, but that has not yet been identified in the current study. An *unknown unknown* is a new compound that has not been previously described or identified.

MS experiments yield the m/z of the compound, from which the mass can be derived. For each mass, one or multiple elemental compositions are possible; and the more accurate the mass is determined the fewer candidate elemental compositions are obtained. The mass accuracy depends on the instrument employed and even for high accuracies, such as in the low part per million (ppm) or sub-ppm range, unique elemental compositions cannot always be obtained. [7] The number of possible elemental compositions can be reduced by incorporating information on the other molecules present in the sample and the possible biotransformations that could have occurred. [8] Additionally, a database of ionization products and frequent neutral losses when MS/MS data are available, can be used to annotate the

elemental compositions of metabolites.[9] In the case a unique elemental composition is available, multiple molecules can still be found with that composition. Additional information of the compound can be obtained by performing MS/MS experiments, where the compound is fragmented and the m/z of the resulting fragments can be measured. These spectra can then be matched with existing spectra databases for identity assignment or similarity search.[10]

As an alternative, MS^n data can be used to characterize a compound in more detail by fragmenting it, detecting its fragments, isolating them and fragmenting them multiple times.[11] The resulting information is a mass spectral tree of fragments connected hierarchically to the original parent ion,[11, 12] which contains more structural information of the unknown compound than regular MS and MS/MS data. MS^n data can be processed and enriched with open source tools like the Multistage Elemental Formula (MEF),[13] which creates a fragmentation tree where the parent ion and each fragment ion are annotated with their elemental composition, instead of the mass and a tree of neutral losses representing the fragmentation pattern of the compound. Actually, this tool can be used to exclude many possible elemental compositions for a given MS^n tree, so that often only one elemental composition for a spectral tree is obtained.

Different approaches have been recently presented to query and compare spectral data, most of them relying on concepts of fingerprint similarity. A fingerprint from the fragmentation tree of an unknown compound is an array of features like the elemental compositions of the fragments and the different branches, and it is used

to query a database of known compounds, for which a fragmentation tree fingerprint has been previously computed. The assumption for using fingerprint similarity is that similar fragmentation trees are produced by similar compounds.[14] Hypothetical fragmentation trees have been derived using a probabilistic model not from MS^n data, but from HPLC-MS/MS[15] or GC-TOF-MS,[16] and used to build a fingerprint comparison method[17] that could assign the class of unknown compounds and in some cases the identity. A different approach[18] involved building a spectral fingerprint directly from the MS/MS spectrum and relate it to a fingerprint containing structural information of the molecule. Recently, Rojas-Chertó et al[14] developed a similar approach using MS^n data to build fingerprints and use them to query experimental MS^n data, where the hierarchical relations between fragments in the fragmentation tree were measured experimentally instead of computationally simulated. These fingerprints were implemented in the web application MetiTree to process, handle, store and analyse MS^n spectra.[19]

In the best case, querying fragmentation trees using fingerprint similarity can return a perfect match if the unknown was present in the database, which would be a level 1 identification of a “*known unknown*” (if the unknown and the standard were measured in the same conditions). In a less favorable case, the unknown is not in the database and it is necessary to propose candidate structures via computer assisted structure elucidation (CASE) like our open source structure generator OMG.[20] In such situations, Rojas-Chertó et al[14] suggested to use the chemical structures of the similar trees in the fragmentation tree database to create the maximum common substructure (MCSS) under the assumption that the unknown metabolite,

which belongs to the same class, will possess the same moiety. This MCSS together with the elemental composition of the unknown could be the input for a structure generator that would produce all the possible molecules complying with these criteria. CASE has been used to identify pollutants and toxic compounds in environmental samples by generating candidates with a structure generator like MOLGEN and filter or rank them using specific criteria related to the problem at hand.[21] In a similar fashion, Schymanski et al[22] initially used gas chromatography coupled with electron-ionization mass spectrometry (GC/EI-MS) and possible filtering criteria were the prediction of spectra using tools like MetFrag,[23] retention index prediction and steric energy calculation, and in a posterior study[24] a consensus score combining these criteria was used to rank candidate molecules.

MS^n data and software tools to process and evaluate these data have been presented as the key factors of success for the identification of small molecules.[25] Many cheminformatics tools that contribute to the elucidation of compounds have been developed, but in the field of metabolite identification, they require the unknown metabolite to be present in a database like PubChem.[26–28] Furthermore, no combination of tools in a pipeline has been used for de novo metabolite identification as it was done for environmental pollutants, which used MS/MS data. Previous studies[29] used MS^n to identify plant metabolites, but required manual intervention and concluded that there is a need for pipelines of cheminformatics to improve metabolite identification. In the work presented here, we combine different tools in a pipeline that enables, for the first time, de novo identification of metabolites from MS^n data as well as identity assignment in an

automated fashion. In order to demonstrate the use of such an identification pipeline, we acquired 30 mass spectral trees of metabolites present in human urine and attempted to identify them with this pipeline.

Materials and methods

Mass spectral trees were acquired for the features measured in human urine samples. Details on analytical methods are provided in Supplementary Information. More than 450 metabolite features representing most probably metabolites were detected with deconvolution (using the software Dissect, Bruker Daltonics, Bremen, Germany) in urine. Mass spectral trees were acquired for the 30 most abundant peaks (Table S1) and processed with the metabolite identification pipeline presented. The identities of these 30 features and their trees were unknown upon selection. Our approach did not attempt to provide a comprehensive analytical coverage of urine metabolites. The aim of this study was to illustrate how the software pipeline can improve the identification of “*known unknowns*” and “*unknown unknowns*” in metabolomics.

Mass Spectral Tree Processing and MSⁿ Database

The first step in the pipeline (Figure 1) is to process and annotate the mass spectral trees into fragmentation trees. Mass spectral trees were processed using the MEF tool,[13] which resolves a unique elemental composition for each parent and fragment ion, as well as for the neutral losses. The result of using the MEF tool to process a mass spectral tree is a fragmentation tree and a neutral loss tree with elemental compositions assigned to the nodes of the tree. An in-house library of MSⁿ

data of reference metabolites was used as described by Rojas-Chertó et al.[14] This database contains fragmentation trees and neutral loss trees of 447 human metabolites and 118 plant polyphenolic metabolites. All MSⁿ spectra in the library were processed with MEF tool.

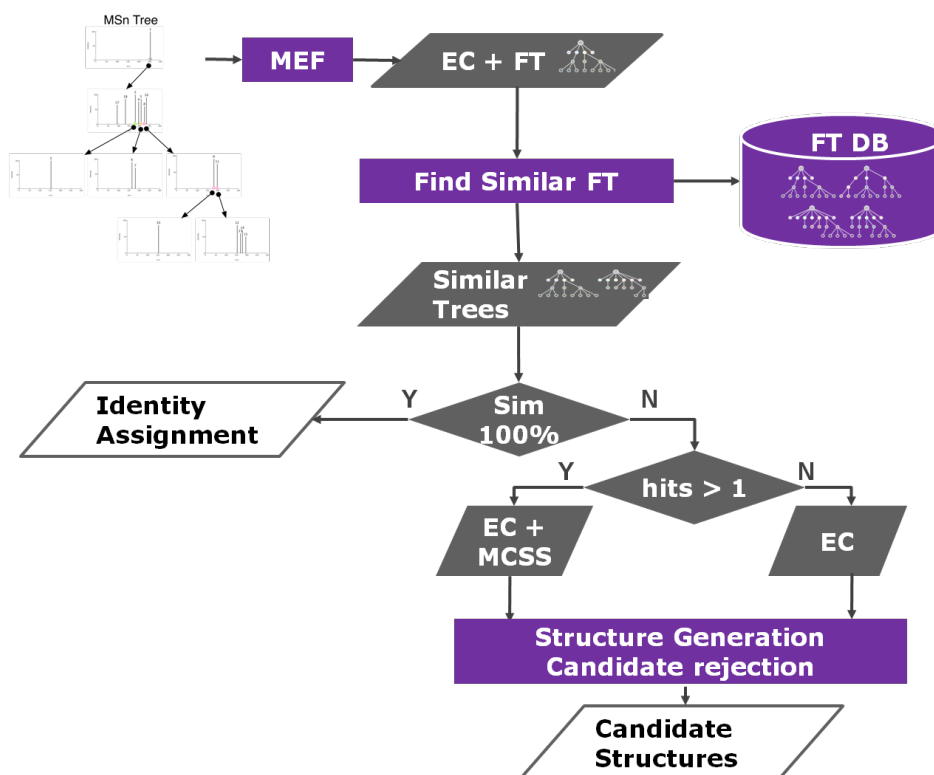


Figure 1 Metabolite identification pipeline. Abbreviations: MEF, Multistage Elemental Formula; EC, Elemental Composition; FT, Fragmentation Tree; MCSS, Maximum Common Substructure; Sim, Similarity.

Data Comparison and Fragmentation Tree Similarity Search

The 30 unknown metabolites were compared to the known metabolites stored in the MSⁿ database using the fragmentation tree fingerprint and similarity calculation presented by Rojas-Chertó et al.[14] A 10% similarity or more was considered to be relevant for identification purposes by educated guess.[14] In the case an unknown compound has 100% similarity with a metabolite in the database, we assign the identity to the “*known unknown*”, which in our case is level 1 identification. When

no metabolite is found with 100% similarity, we are facing the identification of an “*unknown unknown*”. In such case, multiple metabolites can be found with a certain degree of similarity, which is class assignment (level 3 identification) if these metabolites belong to the same class. Additionally, we used these similar compounds to calculate the maximum common substructure (MCSS) they shared and assumed it to be present in the structure of the unknown metabolite.

Candidate Structure Generation

We used the structure generator Open Molecule Generator (OMG)[20] to *in silico* generate all possible candidate structures for the unknowns (Figure 2), taking as an input the elemental composition of the unknown. OMG generates all the possible chemical structures containing exactly those atoms. This list of candidates, even for small elemental compositions, tends to contain millions or billions of possible molecules. Optionally, one can force the output molecules to contain one or multiple non-overlapping prescribed substructures, which reduces drastically the number of candidate structures generated. The bigger the substructure or the more substructures used, the fewer candidates are produced. In this work, we used the MCSS found in the similarity search to be present in the generated structures.

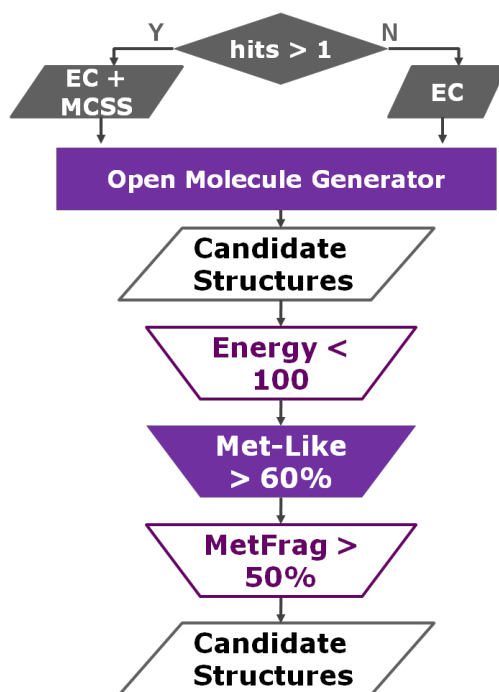


Figure 2 Structure generation and candidate rejection. Abbreviations: EC, Elemental Composition; MCSS, Maximum Common Substructure.

Candidate Structure Filtering

We used three filters (Figure 2) to remove unlikely candidate chemical structures: steric energy, metabolite-likeness, and fragmentation prediction. While OMG produces candidate structures that are valid according to the valence rule, many are unstable and therefore, unlikely to be found in a biological system. First, we used the component “Molecular Energy” from Pipeline Pilot Student Edition 6.1 [30] to calculate the internal energy of the generated structures and those with an energy value of 100 or above were removed. This threshold value was selected after observing that all the metabolites present in the Human Metabolome Database (HMDB)[31] have energy values below 100 when calculated with the same component. In order to use the energy score for further candidate ranking, we scaled energy values to unit range between 0% (for a candidate with energy value of 100) and 100% (for the candidate with the lowest energy value). Second, we used a

predictive model of Metabolite-Likeness[32] to remove candidate structures that are unlikely to structurally resemble human metabolites. We reported that almost all known human metabolites obtain a Metabolite-Likeness of 50% or more. Therefore, we set a conservative minimum threshold of 60% Metabolite-Likeness to consider structures for further identification. Third, we used the spectra prediction tool MetFrag[23] to remove candidates that cannot explain many of the peaks observed in the experimental spectra. MetFrag uses as an input a list of molecules and a list of the experimental spectral peaks, defined by the m/z and intensities. By cleavage of bonds, MetFrag fragments the molecules and computes for each one how many of the provided spectral peaks can be explained by the fragments. With this information, a score is built describing how well each candidate molecule can describe the experimental spectra. We used the settings of [M+H] mode, positive charge, 0.01 Mzabs and 10 Mzppm. We rejected candidate structures that did not obtain at least 50% MetFrag score. Lastly, we combined the three scores in a unique consensus score, as proposed by Schymanski et al[24] to rank the remaining structures in order and prioritize them for further manual identification by an expert.

Results

MS^n spectral trees of 30 unknown metabolites acquired in human urine were analyzed with the metabolite identification pipeline described in the Methods section. The fragmentation trees of the unknowns were used to query the MS^n database for identical or similar fragmentation trees. From the 30 unknown metabolites, 10 obtained a 100% fragmentation tree similarity match, 9 found one or more similar trees ($10\% < \text{similarity value} < 100\%$) and 11 did not obtain a single

hit in the database. At this stage, for these 11 unknowns we could only derive the elemental composition from the data. Using OMG to generate candidate structures for them would return billions of structures, therefore, these unknowns were not studied further and remained unidentified. This indicates that the MSⁿ database used in this study should be enriched with more and varied metabolites.

Identity Search

The database query returned a 100% similarity match for 10 fragmentation trees (Table S2). This is the highest possible similarity score and implies that both the fragmentation tree and neutral loss tree are identical for the unknown metabolite and the standard compound in the database. These 10 identified metabolites are creatinine, acetaminophen, phenylalanine, 7-methylxanthine, uric acid, hippuric acid, paraxanthine, o-tyrosine, l-acetylcarnitine and tryptophan.

Both the authentic standards present in the database and the unknown metabolites from urine were acquired using high resolution MS and MSⁿ in the same lab using the same equipment (as described in Methods). Hence, we are confident to have achieved a full Level 1 identification as proposed in the MSI[5] for these 10 unknown compounds. The authentic standards were acquired by direct infusion, in order to obtain deep and wide mass spectral trees, containing as much structural information as possible. Therefore, they miss an associated retention time. Ideally, these standards should be measured in the same HPLC system as the unknowns in order to have an extra analytical technique to support the full identification. It is interesting to mention that despite being characteristic of the chemical structure, a mass

spectral tree could theoretically not be unique for a given molecule, i.e. two isomeric structures with the same elemental composition but different structure could produce the same mass spectral tree. Hence, the need of complementary analytical methods, like NMR, to validate the identification of metabolites.

Similarity Search

For 9 of our 20 remaining unknown metabolites, we found in the database metabolites with similar fragmentation trees. Three metabolites only found one similar metabolite in the database. In such cases, we were neither able to propose the class of the unknown nor to extract a maximum common substructure, since we would need at least two similar metabolites of the same class. The only possible course of action according to our pipeline was to generate candidate structures using OMG for the elemental composition. Additionally, candidate molecules that are structurally dissimilar to the metabolite in the database could be removed using a chemical similarity filter, but this was out of the scope of the current work, because the resulting list of candidates would be too large. Unknown 16 returned 21 similar metabolites, which produced a very small MCSS (C-C). Such a MCSS, when used in OMG would not constrain the generation process and return billions of candidate structures. Therefore we did not proceed with the identification of this unknown.

Identification of unknowns

Five unknowns returned two or three similar metabolites in the database. All the similar metabolites are found in urine according to HMDB. We calculated a MCSS

An automated pipeline for de novo metabolite identification using mass spectrometry-based metabolomics

from these metabolites, generated structures using OMG and filtered the candidates using the three filtering criteria. For unknown 28, similarity search returned two similar metabolites, with 25% to 3-methoxytyramine and 11% to sinapic acid using fragmentation tree similarity (Table 1).

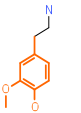
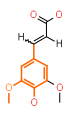
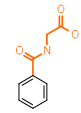
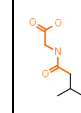
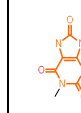
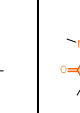
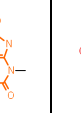
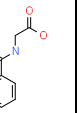
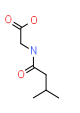
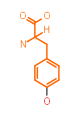
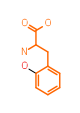
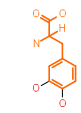











Unknown	28		9		17		15		27			
Candidate Structures	6.8M		150M		Billions		Billions		Billions			
EC	Hits	C ₉ H ₁₀ O ₂	2	C ₇ H ₇ NO ₄	2	C ₆ H ₆ N ₄ O ₃	2	C ₉ H ₉ NO ₄	2	C ₉ H ₁₃ NO ₄ P ₂		3
Similar Structures												
Similarity	25%	11%	19%	11%	18%	10%	24%	12%	32%	30%	13%	
MCSS												
Candidate Structures MCSS	82		65,445		4		8		281			
Candidate structures filtering	8		2,312		4		5		182 (40)			

Table 1 De novo metabolite identification of “unknown unknown” metabolites that are not present in the MSⁿ database, but have a degree of fragmentation tree similarity with one or more fragmentation trees of known metabolites in the database. Candidate structures are generated with Open Molecule Generator using the EC and MCSS and filtered using energy, Metabolite-Likeness and MetFrag.

The MCSS used as prescribed substructure in the candidate generation process with OMG returned only 82 molecules, instead of 6.8 million molecules if only the elemental composition was used. This list of candidate structures was reduced by using energy, metabolite-likeness and MetFrag filters. This resulted in 8 candidate structures, which are presented in Table 2 sorted by a consensus score (CS).









Candidate Structure	Energy	Met likeness	MetFrag	Consensus Score
	HMDB / InChIKey			
1 	-1.35 (100%)	81.0%	99.5%	93.5%
	None / XNKMCBYYBMXTPO-UHFFFAOYSA-N			
2 	-1.24 (99.9%)	80.8%	94.4%	91.7%
	HMDB13744 / YOMSJEATGXXYPX-UHFFFAOYSA-N			
3 	12.42 (86.41%)	68.6%	93.4%	82.8%
	None / QTPWGUHKASDDHO-UHFFFAOYSA-N			
4 	57.76 (41.86%)	75.4%	93.6%	70.2%
	None / HUYRKDFVJJCZQOM-UHFFFAOYSA-N			
5 	64.07 (35.45%)	72.2%	90.2%	66.0%
	None / WNOKBMFZDXCSRN-UHFFFAOYSA-N			
6 	61.54 (37.95%)	61.8%	93.3%	64.4%
	None / YOYNEOCOQAQSSV-UHFFFAOYSA-N			
7 	83.35 (16.43%)	66.6%	93.6%	58.9%
	None / QVXRGADGDBVPIE-UHFFFAOYSA-N			
8 	94.26 (5.66%)	77.2%	93.4%	58.9%
	None / SYCBYIPPZGRLQD-UHFFFAOYSA-N			

Table 2 Candidate structures for unknown 28

For unknown 9, the database query returned two similar metabolites, with fragmentation tree similarity of 19% to hippuric acid and 11% to isovalerylglycine (Table 1). OMG generates more than 150 million molecules for the elemental composition of this unknown and using the MCSS derived from the similar metabolites, this list is reduced to 65,445 compounds and filtered further to 2,312 candidate structures, of which 1,279 obtained a CS of 90% or higher. This made a

selection of smaller list of candidate structures not feasible. For unknown 17 two metabolites with similar fragmentation trees, 18% similarity to 1,3-dimethyluric acid and 10% to 1,3,7-trimethyluric, were found in the database (Table 1). OMG generated a much smaller list of candidate structures, only 4, using the MCSS as constraint (Table 3).





Candidate Structure	Energy	Met likeness	MetFrag	Consensus Score
	HMDB / InChIKey			
1 	18.91 (100%)	98.4%	100%	99.4%
	HMDB01970 / ODCYDGXXCHTFIR-UHFFFAOYSA-N			
2 	21.09 (97.31%)	97.6%	100%	98.3%
	HMDB01973 / XJEJWDFDVPDMAS-UHFFFAOYSA-N			
3 	21.49 (96.82%)	96.2%	100%	97.7%
	HMDB11107 / YHNNPKUFPWLTOP-UHFFFAOYSA-N			
4 	18.98 (99.91%)	98.4%	89.7%	96.0%
	HMDB03099 / QFDRTQONISXGJA-UHFFFAOYSA-N			

Table 3 Candidate structures for unknown 17

Similarity search returned 2 metabolites similar to unknown 15, with fragmentation tree similarity of 24% to hippuric acid and 12% to isovalerylglycine. At this step we observed three things: i) the elemental composition of the unknown was the elemental composition of to hippuric acid with an extra oxygen atom; ii) the fragmentation tree similarity of 24% was due to the neutral loss tree, which were identical for the unknown and the compound in the database, indicating that both compounds had a similar structure and fragmentation pattern; iii) the fragmentation tree measured for the unknown was almost identical to the one in the database,

except for an additional oxygen atom in each of the fragment ions. This indicated that the chemical structure of the unknown was the structure of hippuric acid, which we used as MCSS (Table 1), with an additional oxygen atom. OMG generated 8 candidate molecules using the MCSS as constraint (Table 4), which result of adding one oxygen atom in all possible ways to hippuric acid. Further filtering using our three criteria removed candidates 6, 7 and 8, which despite having favorable values of energy score and metabolite-likeness, were not able to explain any of the experimental fragments and therefore MetFrag assigned them a 0% score. A close examination of the *in silico* fragments proposed by MetFrag for the experimental fragment revealed that all of them contained a phenol group, a feature that is not present in the three rejected candidates. Therefore, we propose that unknown 15 has the same structure as a compound with an oxygen atom attached to the benzene ring. The position of the oxygen in the phenol group remains unknown. Additionally, NMR measurements of standards could be used to elucidate the position of the oxygen in the molecule and confirm the identity of this unknown.

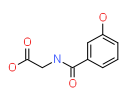
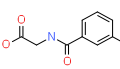
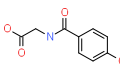
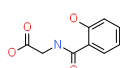
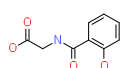
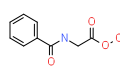
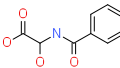
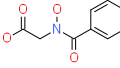
Candidate Structure	Energy	Met likeness	MetFrag	Consensus Score
	HMDB / InChIKey			
	-1.51 (100%)	96.8%	100%	98.9%
	HMDB06116 / XDOWFNMYJRHEW-UHFFFAOYSA-N			
	-1.51 (100%)	96.8%	100%	98.9%
	HMDB06116 / XDOWFNMYJRHEW-UHFFFAOYSA-N			
	-1.49 (99.99%)	96.8%	100%	98.9%
	HMDB13678 / ZMHLUFWWWPBTIU-UHFFFAOYSA-N			
	-1.39 (99.88%)	94.6%	100%	98.1%
	HMDB00840 / ONJSZLXSECQROL-UHFFFAOYSA-N			
	-1.39 (99.88%)	94.6%	100%	98.1%
	HMDB00840 / ONJSZLXSECQROL-UHFFFAOYSA-N			
	-1.11 (99.61%)	96.2%	0%	65.2%
	None / NVVKRZSRWCCEAU-UHFFFAOYSA-N			
	-0.53 (99.03%)	98.2%	0%	65.7%
	HMDB02404 / GCWCVCCEIQXUQU-MRVPVSSYSA-N			
	1.04 (97.49%)	87%	0%	61.5%
	None / FMYVYJPEMYKYRE-UHFFFAOYSA-N			

Table 4 Candidate structures for unknown 15

Similarity search for unknown 27 returned three metabolites with fragmentation tree similarity of 32% to l-tyrosine, 30% to o-tyrosine, and 13% to dl-dopa (Table 1). OMG generated a list of 281 candidate structures using the MCSS, which was reduced to 182 after filtering. We observed that two of the similar metabolites in the database had a phenol (benzene ring with an attached oxygen atom) and the third

one a catechol, (benzene ring with two attached oxygen atoms). Hence, we assumed that our unknown also had at least one oxygen atom attached to the benzene ring. We selected from the 182 candidates those that contained a phenol, which resulted in a final list of 40 candidate structures. A P-P bond was present in all the candidates, which despite being a rarity among known metabolites did not penalize the scores obtained by the molecules. This P-P moiety would immediately raise an alarm flag for any metabolite identification expert. It could be caused by poor experimental acquisition of the mass spectral tree or by an incorrect assignment of the elemental composition by MEF. Inspection by an expert determined that for a m/z of 262.03809 MEF should produce an elemental composition like $C_9H_{13}NO_6P$, which belongs to phosphotyrosine, instead of $C_9H_{14}NO_4P_2$. Therefore, we confirmed that the analytical conditions were identical for this unknown as for the other compounds and that all the elemental compositions generated and forced MEF to use the elemental composition $C_9H_{13}NO_6P$ for the parent ion, but it failed to annotate the elemental compositions of the fragments. In other words, this elemental composition could not explain the fragment ions measured experimentally. As a result, we considered the experimental data and the elemental composition $C_9H_{14}NO_4P_2$ to be valid and all 40 candidates to be possible. Ideally, authentic standards of them should be measured and compared with the spectral data of the unknown.

Tentative validation of MCSS assignment

We assessed whether the use of the MCSS and the filtering can lead to a wrong identification or to miss the good molecule in the list of candidate structures. We

applied the structure generation and filtering strategy to the 10 identified metabolites. For only four of these metabolites similar trees were found in the database and a MCSS could be generated (Table S3). We observed in each of these four cases that the MCSS found is a substructure of the metabolite, with which OMG generated among others the good structure. The filtered list of candidate structures always contained the good molecule, which was ranked high according to the consensus score in three of the four cases. In previous work[14], it was observed that with a tree similarity below 20% the MCSS obtained was not very informative. In these four examples, the MCSS used were informative enough when obtained from metabolites with at least 12% tree similarity. When including metabolites with tree similarity between 12% and 10% the MCSS was a carboxylic acid for unknowns 22, 12, and 18. For unknown 28 it was a benzene ring. These MCSS belong to the metabolite, but OMG would return millions of candidates, therefore we did not use them for further confirmation. From this we conclude that the use of the pipeline can provide good candidate structures. Further validation should be performed to understand whether there are cases for which the pipeline could lead to incorrect results.

Discussion

The results presented demonstrate how this metabolite identification pipeline can be used to identify metabolites using MS^n data from human urine samples. This workflow could be adapted to work with MS/MS data, although data processing and similarity search of spectra should be then modified. Here we only used MS^n data and applied it to those features for which a similar fragmentation tree was present in

the MSⁿ database. Such MSⁿ database can be used locally, MetiTree,[19] or online, Massbank[33] and MzCloud. The number of metabolites that can be identified in this way depends on how comprehensive the database of MSⁿ is. Furthermore, we showed for the first time how metabolites not present in a database could be identified.

Having substructure information is crucial to identify unknown metabolites. In our case, we observed that using a large MCSS (or alternative multiple prescribed substructures) reduced significantly the number of candidate structures, therefore future work should focus on developing more reliable ways of generating more or larger MCSS. In the case of unknown 9, the MCSS found was linear, which allowed for the formation of many rings and therefore, a list of more than 2,000 candidate structures. For the same unknown and for unknown 28 we observed that the filtering using energy, metabolite-likeness and MetFrag yielded a 10-fold reduction in the number of candidates, proving the value of incorporating these criteria. In those cases where the MCSS described most of the structure of the unknown, OMG produced a short list of candidates and this was not significantly reduced with the filters, since most of the structures were acceptable. Additionally, more filters could be added in the future depending on the data available, like retention time prediction[34–36]. Fragmentation prediction by MetFrag proved to be useful at rejecting candidates, like for unknown 15, that did not have an oxygen atom attached to a ring but to a chain.

The use of mass spectral trees was crucial to assign identities and to derive structural information of the unknown metabolite from similar metabolites. We observed that very similar metabolites could have low fragmentation tree similarity, because their fragmentation trees were different. Fortunately, the structural resemblance was captured in the neutral loss trees, which in some cases were identical between the unknown and a similar metabolite, despite having different fragmentation trees. This shows the importance of including neutral loss information in the fragmentation tree fingerprint approach and encourages future research on how to better combine fragmentation tree and neutral loss tree information for similarity search.

Conclusion

In this work we have presented a pipeline that enables metabolite identification using MS^n data and that can be used in metabolomics studies involving experimental data. Starting from the experimental MS^n data of unknown metabolites, this pipeline processes, annotates, and compares MS^n data, and assigns the identity or provides a few putative identities for de novo identification of unknown metabolites.

By means of fragmentation tree similarity, this pipeline can assign the identity to an unknown metabolite, provided its MS^n spectra have been previously measured and stored in a database. In the case this metabolite is not in the database, this pipeline is capable of doing de novo metabolite identification by extracting common moieties in similar compounds and using structure generation to propose candidate structures. De novo identification is in itself the biggest contribution of this work to

the field of metabolomics as the pipeline does not require the unknown metabolite to be present in any database to propose a handful of possible structures.

While the unknown is not required to be in a database to be identified, the number of the candidate structures returned will be fewer, provided substructures of the unknown can be discovered. Ideally, these substructures could be found by matching subtrees of the unknown with a database of annotated MS^n trees, i.e. where a structure has been assigned to the fragment ions. Unfortunately, these annotated databases are not yet available for MS^n data, and therefore we searched for similar metabolites to the unknown in the MS^n database and generated the MCSS. On the one hand, it appears necessary to enrich MS^n databases with experimental data of more and varied metabolites to increase the chances of finding similar metabolites. On the other hand, finding too many compounds with similar fragmentation trees can produce a small MCSS if the chemical structures are different, which will not constrain enough the generation of candidate molecules. Therefore, it is interesting to study better ways to find similar compounds, like an initial clustering of the known metabolites and a posterior MCSS calculation within each cluster could benefit de novo identification. Additionally, the similarity threshold of fragmentation trees could be modified in order to obtain less similar compounds and as consequence a larger MCSS, provided we have a rich and comprehensive database.

To the best of our knowledge this is the first implementation of a metabolite identification pipeline that enables identity assignment and de novo metabolite identification and that makes use solely of LC- MS^n data, and we foresee that further

additions such the ones proposed above will allow to identify even a larger fraction of the unknown metabolites.

References

1. Fiehn O: Metabolomics--the link between genotypes and phenotypes. *Plant molecular biology* 2002, 48:155–71.
2. Dunn WB, Erban A, Weber RJM, Creek DJ, Brown M, Breitling R, Hankemeier T, Goodacre R, Neumann S, Kopka J, Viant MR: Mass appeal: metabolite identification in mass spectrometry-focused untargeted metabolomics. *Metabolomics* 2012.
3. Johnson CH, Gonzalez FJ: Challenges and opportunities of metabolomics. *Journal of Cellular Physiology* 2012, 227:2975–81.
4. Scalbert A, Brennan L, Fiehn O, Hankemeier T, Kristal BS, Van Ommen B, Pujos-Guillot E, Verheij E, Wishart D, Wopereis S: Mass-spectrometry-based metabolomics: limitations and recommendations for future progress with particular focus on nutrition research. *Metabolomics* 2009, 5:435–458.
5. Sumner LW, Amberg A, Barrett D, Beale MH, Beger R, Daykin C a., Fan TW-M, Fiehn O, Goodacre R, Griffin JL, Hankemeier T, Hardy N, Harnly J, Higashi R, Kopka J, Lane AN, Lindon JC, Marriott P, Nicholls AW, Reily MD, Thaden JJ, Viant MR: Proposed minimum reporting standards for chemical analysis. *Metabolomics* 2007, 3:211–221.
6. Wishart DS: Computational strategies for metabolite identification in metabolomics. *Bioanalysis* 2009, 1:1579–1596.
7. Kind T, Fiehn O: Metabolomic database annotations via query of elemental compositions: mass accuracy is insufficient even at less than 1 ppm. *BMC Bioinformatics* 2006, 7:234.
8. Rogers S, Scheltema R a, Girolami M, Breitling R: Probabilistic assignment of formulas to mass peaks in metabolomics experiments. *Bioinformatics* 2009, 25:512–8.
9. Draper J, Enot DP, Parker D, Beckmann M, Snowdon S, Lin W, Zubair H: Metabolite signal identification in accurate mass metabolomics data with MZedDB, an interactive m/z annotation tool utilising predicted ionisation behaviour “rules”. *BMC bioinformatics* 2009, 10:227.
10. Roux A, Xu Y, Heilier J-F, Olivier M-F, Ezan E, Tabet J-C, Junot C: Annotation of the Human Adult Urinary Metabolome and Metabolite Identification Using Ultra High Performance Liquid Chromatography Coupled to a Linear Quadrupole Ion Trap-Orbitrap Mass Spectrometer. *Analytical Chemistry* 2012, 84:6429–6437.
11. Kasper PT, Rojas-Chertó M, Mistrik R, Reijmers T, Hankemeier T, Vreeken RJ: Fragmentation trees for the structural characterisation of metabolites. *Rapid Communications in Mass Spectrometry* 2012, 26:2275–86.
12. Sheldon MT, Mistrik R, Croley TR: Determination of ion structures in structurally related compounds using precursor ion fingerprinting. *Journal of the American Society for Mass Spectrometry* 2009, 20:370–6.
13. Rojas-Chertó M, Kasper PT, Willighagen EL, Vreeken RJ, Hankemeier T, Reijmers TH: Elemental composition determination based on MS(n). *Bioinformatics* 2011, 27:2376–83.
14. Rojas-Chertó M, Peironcely JE, Kasper PT, Van der Hooft JJJ, De Vos RCH, Vreeken R, Hankemeier T, Reijmers T: Metabolite Identification Using Automated Comparison of High-Resolution Multistage Mass Spectral Trees. *Analytical Chemistry* 2012, 84:5524–5534.
15. Rasche F, Svatos A, Maddula RK, Böttcher C, Böcker S: Computing fragmentation trees from tandem mass spectrometry data. *Analytical Chemistry* 2011, 83:1243–51.
16. Hufsky F, Rempt M, Rasche F, Pohnert G, Böcker S: De novo analysis of electron impact mass spectra using fragmentation trees. *Analytica Chimica Acta* 2012, 739:67–76.
17. Rasche F, Scheubert K, Hufsky F, Zichner T, Kai M, Svatos A, Böcker S: Identifying the unknowns by aligning fragmentation trees. *Analytical Chemistry* 2012, 84:3417–3426.
18. Heinonen M, Shen H, Zamboni N, Rousu J: Metabolite identification and molecular fingerprint prediction via machine learning. *Bioinformatics* 2012, 28:2333–2341.

19. Rojas-Chertó M, Van Vliet M, Peironcely JE, Van Doorn R, Kooyman M, Beek T Te, Van Driel M a, Hankemeier T, Reijmers T: MetiTree: a web application to organize and process high resolution multi-stage mass spectrometry metabolomics data. *Bioinformatics* 2012, 28:2707–2709.
20. Peironcely JE, Rojas-chertó M, Fichera D, Reijmers T, Coulier L, Faulon J-L, Hankemeier T: OMG : Open Molecule Generator. *Journal of Cheminformatics* 2012, 4.
21. Schymanski EL, Bataineh M, Goss K-U, Brack W: Integrated analytical and computer tools for structure elucidation in effect-directed analysis. *Trends in Analytical Chemistry* 2009, 28:550–561.
22. Schymanski EL, Meringer M, Brack W: Automated Strategies To Identify Compounds on the Basis of GC/EI-MS and Calculated Properties. *Analytical Chemistry* 2011:903–912.
23. Wolf S, Schmidt S, Müller-Hannemann M, Neumann S: In silico fragmentation for computer assisted identification of metabolite mass spectra. *BMC Bioinformatics* 2010, 11:148.
24. Schymanski EL, Gallampois CMJ, Krauss M, Meringer M, Neumann S, Schulze T, Wolf S, Brack W: Consensus Structure Elucidation Combining GC/EI-MS, Structure Generation and Calculated Properties. *Analytical Chemistry* 2012, 84:3287–3295.
25. Kind T, Fiehn O: Advances in structure elucidation of small molecules using mass spectrometry. *Bioanalytical reviews* 2010, 2:23–60.
26. Zhou B, Wang J, Resson HW: MetaboSearch: Tool for Mass-Based Metabolite Identification Using Multiple Databases. *PLoS one* 2012, 7:e40096.
27. Brown M, Dunn WB, Dobson P, Patel Y, Winder CL, Francis-McIntyre S, Begley P, Carroll K, Broadhurst D, Tseng a, Swainston N, Spasic I, Goodacre R, Kell DB: Mass spectrometry tools and metabolite-specific databases for molecular identification in metabolomics. *The Analyst* 2009, 134:1322–32.
28. Menikarachchi LC, Cawley S, Hill DW, Hall LM, Lai S, Wilder J, Grant DF: MolFind: A Software Package Enabling HPLC/MS-Based Identification of Unknown Chemical Structures. *Analytical Chemistry* 2012, 84:9388–94.
29. Hooft JJJ, Vervoort J, Bino RJ, Vos RCH: Spectral trees as a robust annotation tool in LC–MS based metabolomics. *Metabolomics* 2011, 8:691–703.
30. Accelrys Pipeline Pilot, version 6.1.5; Accelrys Inc.: San Diego, CA, 2010. Accelrys Pipeline Pilot, version 6.1.5; Accelrys Inc.: San Diego, CA 2010.
31. Wishart DS, Knox C, Guo AC, Eisner R, Young N, Gautam B, Hau DD, Psychogios N, Dong E, Bouatra S, Mandal R, Sinelnikov I, Xia J, Jia L, Cruz JA, Lim E, Sobsey CA, Shrivastava S, Huang P, Liu P, Fang L, Peng J, Fradette R, Cheng D, Tzur D, Clements M, Lewis A, Souza A De, Zuniga A, Dawe M, Xiong Y, Clive D, Greiner R, Nazyrova A, Shaykhtudinov R, Li L, Vogel HJ, Forsythe I: HMDB: a knowledgebase for the human metabolome. *Nucleic Acids Research* 2009, 37:D603–610.
32. Peironcely JE, Reijmers T, Coulier L, Bender A, Hankemeier T: Understanding and Classifying Metabolite Space and Metabolite-Likeness. *PLoS ONE* 2011, 6:e28966.
33. Horai H, Arita M, Kanaya S, Nihei Y, Ikeda T, Suwa K, Ojima Y, Tanaka K, Tanaka S, Aoshima K, Oda Y, Kakazu Y, Kusano M, Tohge T, Matsuda F, Sawada Y, Hirai MY, Nakanishi H, Ikeda K, Akimoto N, Maoka T, Takahashi H, Ara T, Sakurai N, Suzuki H, Shibata D, Neumann S, Iida T, Tanaka K, Funatsu K, Matsuura F, Soga T, Taguchi R, Saito K, Nishioka T: MassBank: a public repository for sharing mass spectral data for life sciences. *Journal of Mass Spectrometry* 2010, 45:703–714.
34. Boswell PG, Schellenberg JR, Carr PW, Cohen JD, Hegeman AD: A study on retention “projection” as a supplementary means for compound identification by liquid chromatography-mass spectrometry capable of predicting retention with different gradients, flow rates, and instruments. *Journal of chromatography. A* 2011, 1218:6732–41.
35. Hall LM, Hall LH, Kertesz TM, Hill DW, Sharp TR, Oblak EZ, Dong YW, Wishart DS, Chen M-H, Grant DF: Development of Ecom50 and retention index models for nontargeted metabolomics: identification of 1,3-dicyclohexylurea in human serum by HPLC/mass spectrometry. *Journal of Chemical Information and Modeling* 2012, 52:1222–37.
36. Creek DJ, Jankevics A, Breitling R, Watson DG, Barrett MP, Burgess KE V: Toward Global Metabolomics Analysis with Hydrophilic Interaction Liquid Chromatography-Mass Spectrometry: Improved Metabolite Identification by Retention Time Prediction. *Analytical Chemistry* 2011:8703–8710.

Supplementary materials

Chemicals

All reagents and chemicals were of HPLC grade purity or higher and purchased from Sigma-Aldrich.

Human urine samples

Urine samples were collected from healthy volunteers (3 males and 2 females) in the morning. The samples were individually diluted with water in a ratio of 1:1 (v/v) to a final volume of 2 mL. The samples were subsequently centrifuged at 16,100 rpm for 10 min at 10 °C. The supernatant was collected.

HPLC-MS

LC separation was carried out on an Agilent 1200 LC system. Samples were separated in an Atlantis C18 T3 column (Waters, 100 x 2.1 mm, 3 µm) using a mobile phase linear gradient from 98% water/2% acetonitrile + 0.1% formic acid to 98% acetonitrile/2% water + 0.1% formic acid. The injection volume was 5 µL and the flow 250 µL/min.

MS detection was carried out on a Finnigan LTQ-Orbitrap XL instrument (Thermo Electron Corp.). Electrospray ionization was carried out in the positive ionization mode. Mass spectra were acquired in the centroid mode in the range m/z 60-1000 at a resolution of 60,000.

The LTQ-Orbitrap was adapted with a chip-based nano-electrospray ionization source/fractionation robot (NanoMate Triversa, Advion BioSciences). The eluent flow was split by the NanoMate, at 249.075 $\mu\text{L}/\text{min}$ to the fraction collector and 925 nL/min to the nano-electrospray source. LC-fractions were collected every 5 s (i.e., 21 μL) into a 384 wells plate (Twin tec, Eppendorf), cooled at 10⁰C.

Mass spectral tree acquisition

The chip-based nano-electrospray ionization source (Triversa NanoMate, Advion Biosciences) was also used for automated direct sample infusion of the collected fractions into the LTQ-Orbitrap. MSⁿ data of the collected fractions were recorded using a data-dependent scanning function with the criteria to select the highest peak and from this the five most intense ions detected for MS² and the three most intense ions for the rest of the MSⁿ levels. For signal averaging, the mass spectrometer was set with five microscans. The Orbitrap was operated at 30,000 resolution, a normalized collision energy of 35% and an isolation window of 1 Th.

Unknown	m/z	Elemental Composition [M+H]	Depth of tree	# fragments
1	227.0775	C8H11N4O4	MS4	16
2	243.13405	C11H19N2O4	MS5	33
3	271.07492	C11H15N2O4S	MS5	36
4	313.08582	C13H17N2O5S	MS5	132
5	447.10675	C21H23N2O5S2	MS5	28
6	146.08102	C6H12NO3	MS3	5
7	152.07057	C8H10NO2	MS2	2
8	167.05627	C6H7N4O2	MS2	2
9	170.04449	C7H8NO4	MS4	8
10	181.06073	C8H9N2O3	MS4	11
11	265.11835	C13H17N2O4q	MS5	40
12	180.06531	C9H10NO3	MS4	7
13	204.12299	C9H18NO4	MS3	8
14	197.06703	C7H9N4O3	MS4	9
15	196.06024	C9H10NO4	MS4	7
16	185.09195	C8H13N2O3	MS4	17

An automated pipeline for de novo metabolite identification using mass spectrometry-based metabolomics

17	183.05112	C6H7N4O3	MS3	6
18	182.08107	C9H12NO3	MS5	19
19	169.03558	C5H5N4O3	MS4	10
20	205.09718	C11H13N2O2	MS5	10
21	181.07181	C7H9N4O2	MS3	4
22	166.08606	C9H12NO2	MS4	9
23	268.1	C10H14N5O4	MS2	2
24	114.06562	C4H8N3O	MS2	1
25	144.10164	C7H14NO2	MS2	2
26	195.06522	C4H12N4O3P	MS4	12
27	262.03809	C9H14NO4P2	MS5	34
28	151.07513	C9H11O2	MS3	8
29	271.16553	C13H23N2O4	MS5	38
30	310.20163	C17H28NO4	MS4	40

Table S1 30 MSⁿ trees of unknown metabolites acquired from human urine, with retention time, m/z, elemental composition, MS level achieved and number of fragments in the fragmentation tree.

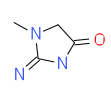
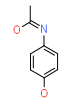
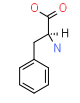
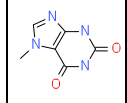
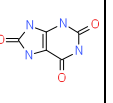
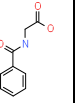
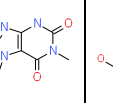
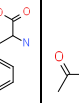
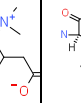
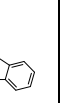
Unknown	24	7	22	8	19	12	21	18	13	20
m/z	114.06	152.07	166.09	167.05	169.03	180.06	181.07	182.08	204.12	205.10
Elemental Composition [M+H]	C ₄ H ₈ N ₃ O	C ₈ H ₁₀ N ₂ O ₂	C ₉ H ₁₂ N ₂ O ₂	C ₆ H ₇ N ₄ O ₂	C ₅ H ₅ N ₄ O ₃	C ₉ H ₁₀ N ₂ O ₃	C ₇ H ₉ N ₄ O ₂	C ₉ H ₁₂ N ₂ O ₃	C ₉ H ₁₈ N ₂ O ₄	C ₁₁ H ₁₃ N ₂ O ₂
MS ⁿ DB match	Creatinine	Acetaminophen	Phenylalanine	7-Methylxanthine	Uric acid	Hippuric acid	Paraxanthine	O-Tyrosine	L-Acetylarnitine	Tryptophan
Structure										

Table S2 10 Identified metabolites that are found in the MSⁿ database with 100% fragmentation tree similarity. Chemical structures belong to the metabolite found in the database.




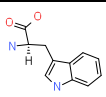
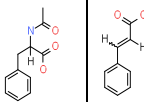
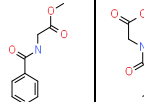
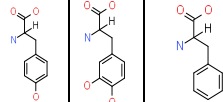
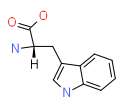
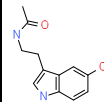




Unknown		22		12		18		20			
Structure											
EC	Hits	C ₉ H ₁₀ O ₂	11	C ₉ H ₉ NO ₃	2	C ₉ H ₁₁ NO ₃	8	C ₁₁ H ₁₂ N ₂ O ₂		4	
Similar Structures											
Similarity		27%	15%	24%	12%	86%	19%	12%	44%	15%	13%
MCSS											
Candidate Structures MCSS		92		475,242		9		28,925			
Candidate structures filtering		43		419		8		952			
Rank		4th		1st		7th		217th			

Table S3 Validation of the MCSS assignment on 4 identified metabolites. The metabolite contains the MCSS found and the correct structure is in the filtered list of candidate structures.

5 *De novo* *identification of* *metabolites with* *open molecule* *generator for* *metabolomics*

In preparation

De novo identification of metabolites with open molecule generator for metabolomics

Metabolite identification is a major bottleneck in metabolomics. Proposing chemical structures for unknown metabolites is essential to give interpretation to scientific results and it requires new software tools. We have implemented PMG, an open source parallel structure generator, especially designed for metabolomics. It is an extended version of OMG, the earlier released structure generator. PMG produces molecules faster than OMG, it accommodates multiple prescribed substructures and it removes unstable structures using a bad list of rings and substructures that are not found in human metabolites. These substructures can be obtained from different sources like MSⁿ, NMR, and manual annotation or library search. PMG has been tested using elemental compositions and substructures of known human metabolites as well as unknown metabolites found in human urine. The new PMG algorithm represents a 100-fold increase in speed versus OMG in most cases, while the use of bad rings and bad substructures yields a 10-fold reduction of candidate structures in the best cases. In all of the test cases the good chemical structure is returned in the list of candidate structures. We expect PMG in its current form to significantly contribute to the de-novo identification of metabolites and due to its open source nature, to be the core of future metabolomics identification software. In addition, PMG can be also used for

the de-novo identification of other molecules than metabolites or applied in other applications requiring a structure generator.

One of the major bottlenecks in metabolomics is metabolite identification, the precise elucidation of the chemical structure of a metabolite.[1–3] When an unknown metabolite is not present in reference molecular databases, a de novo identification of the metabolite is necessary, and one needs other tools and algorithms than library searches. Mass spectrometry (MS) is a common analytical tool used in metabolomics. It returns the mass over charge ratio (m/z) of an ion (or more ions) after ionization of a molecule, from which elemental compositions (ECs) can be derived. In addition multi stage mass spectrometry (MS^n) has been used in metabolite identification [4, 5] since it offers substructure information of the unknown molecule by fragmenting ions subsequently into smaller ions. With such a strategy, the information obtained for an unknown molecule is its EC and sometimes one or more substructures. In order to generate candidate chemical structures of the unknown metabolite the use of a structure generator (ideally open source)[6] like the recently introduced Open Molecule Generator (OMG)[7] is obviously a very attractive option.

A structure generator receives as input the elemental composition of the unknown and optional constraints, like wanted and unwanted substructures, to limit the search space. Ideally, a structure generator produces all the possible molecules without duplicates. Structure generation is a combinatorial problem that can lead to a computational explosion in order to get complete results. In practice, it is more

desirable to get a short list of realistic candidate structures,[4] which in any case contains the structure of the unknown, at the cost of longer computational time, than a long list of (mostly) unrealistic structures.[8] A short list of candidates is easier to check manually by an expert and to filter computationally. Ultimately, the structure generator should return one or a handful of candidate structures, allowing the experimental validation of the proposed identity of the unknown molecule. On the downside, including more constraints in the generation process requires more computation time. Some constraints can be evaluated for non-finished molecules during the generation process and other constraints can be evaluated only for complete molecules. In either case, these checks add to the number of computations performed per molecule. The use of faster computers or faster algorithms can circumvent the impact of this increase in computation time.

In our earlier work with OMG [7] we observed that executing in parallel the structure generator calculations was possible but would not be the ultimate solution for the poor speed performance of OMG compared to the commercial structure generator MOLGEN[9]. The original OMG code generated molecules using the canonical augmentation path approach, which required the calculation of the canonical representative of each graph (two isomorphic graphs or molecules have the same canonical representative). It is known in graph theory that obtaining the canonical representative of a graph is more complex than checking if the graph is the canonical representative.[10, 11] Therefore, removing the use of the graph canonizer and using a computationally cheaper canonicity test was a logical way to improve the performance of our algorithm.

In computer assisted structure elucidation, constraints like prescribed substructures or required physicochemical properties are often used to reduce the number of generated molecules at the expense of increasing the computation time. Such constraints can be used to reject unwanted molecules while they are generated or after all the molecules have been produced, the former being less computationally demanding than the latter. In a previous metabolomics study,[4] substructure information was indispensable to turn the identification of unknown metabolites into a tractable problem. In addition, posterior filtering of unstable molecules based on high values of force field internal energy and on low metabolite-likeness[12] prediction yielded a short list of candidate structures.

This work extends upon previous studies in which we introduced OMG,[7] and upon lessons learned after attempting to identify human metabolites using an automated pipeline of software tools and MSⁿ data.[4] We observed the need for a faster algorithm, which should as well accommodate more constraints derived from metabolomics analytical data to generate a shorter list of candidate structures. Here we present Parallel Molecular Generator (PMG), an improved OMG, a generic open source multi-core structure generator for metabolomics. It takes as an input the elemental composition of the unknown, and optionally several prescribed substructures, a list of bad rings and lists of good and bad substructures. PMG accommodates a faster canonization algorithm and allows the use of multiple CPUs. Its open source nature allows users to implement other constraints that are adequate for their requirements. We assessed the speed improvement and the effect of (i) using multiple cores generating candidate structures for elemental

compositions, (ii) constraining with bad rings and bad substructures, and (iii) with and without prescribed substructures. In order to demonstrate how the input of elemental compositions and substructures are used by PMG, two different test sets were used. The first set contained elemental compositions from known metabolites present in the Human Metabolome Database (HMDB)[13] and the substructures were drawn using Marvin.[14] The second set contained known and unknown metabolites measured in human urine, for which the elemental composition was obtained using the Multi-stage Elemental Formula (MEF) tool[15] and the substructures were derived using MSⁿ data, fragmentation tree similarity[16] and the maximum common substructure (MCSS) approach. We present the effect in time and number of generated molecules when using PMG in multiple processors with and without constraints. We demonstrate how PMG can contribute to transform the identification challenge of an unknown metabolite into a solvable problem.

Materials and methods

Structure Generation Via Orderly Generation

There are many approaches to generate a complete set of non-duplicate molecules, like the homomorphism principle used by MOLGEN, the orderly generation [17, 18] and the canonical augmentation path proposed by McKay[19]. Brinkman[20], Faulon[21] and Meringer[22] provided a simple description of these methods.

Structure generation with OMG[7] can be regarded as a search tree with multiple levels where the root contains the atoms of the elemental composition without bonds and the leaves or end points of the tree represent complete molecules. It

follows an algorithm first proposed by Faulon[21], where at each level a bond is added to the intermediate molecules. The structure generator should return all possible valid molecules without duplicates. In order to improve the speed of the algorithm used in OMG two strategies are devised: i) usage of faster computational techniques to check for and remove duplicates and ii) in some cases, make use of (a priori) constraints that can be applied early in the search tree to prune branches, since the less branches in the tree the less duplicate check and removal is needed.

The choice of algorithm to generate molecules determines the completeness of the results and the speed of obtaining them. In the current approach followed for PMG, we use orderly generation and consider a special ordering on graphs, such that for every two graphs (with the same number of vertices, bonds and degree), we can determine the bigger and the smaller one. Among isomorphic graphs, we consider the smallest (minimal) one as the canonical one. Each node (atom) is given a number, and therefore each edge (bond) can be represented as a triplet $\langle x,y,d \rangle$ where x and y ($x < y$) are the numbers associated to the connected atoms and d shows the degree of the bond. An edge $\langle x,y,d \rangle$ is smaller than the edge $\langle x', y', d' \rangle$ if :

$x < x'$; or,

$x = x'$ and $y < y'$; or,

$x = x'$ and $y = y'$ and $d > d'$.

This is, if the labeling of the atoms is smaller, or for a similar labeling, the degree of the edge is smaller. In orderly generation, we start from the smallest possible graph, i.e., a graph with vertices but no edges (which can be seen as a triplet $\langle 0,0,0 \rangle$), and continue adding edges following this principle: we can only add an edge to a graph

which is bigger than all the edges already in the graph. Furthermore, we only keep growing a graph if it fulfills a criterion, like being semi-canonical or minimal. Graph theory will then guarantee us that in the end we will generate all possible graphs and exactly one instance of them.

Generating the edges in ascending order already removes some intermediate possible structures and therefore increases speed. However, the test for minimality is computationally expensive. Nevertheless, testing for minimality is still less computationally expensive than generating the equivalent minimal graph. Generating the minimal graph, which is equivalent to canonizing the graph, was the approach used in OMG. We expect that substituting the canonicity test for the minimality test will increase the speed of the algorithm. An alternative to the minimality test to speed up the computation is to use a simpler though not complete test, called semi-canonicity, for intermediate structures and test minimality only in complete structures.[23] By using this test instead of the full minimality check, we can very quickly reject most of the non-minimal structures. However, we may end up with some duplicate structures. Therefore, we still need to do the full minimality check on the final structures to keep only the canonical structures. The ideal solution is to combine the two approaches above. More precisely, we can first apply the very quick semi-canonicity check to reject most of the non-minimal structures, and then apply immediately the full minimality check on the remaining intermediate structures. Alternatively, intermediate tests are performed once blocks in the adjacency matrix that stores the graph are filled, for further details we refer to the

work of Meringer.[22, 23] This way, we will end up performing the costly operation of minimality test less.

PMG implements three structure generation modes: (default mode) semi-canonicity and minimality tests combined, (mode 0) semi-canonicity test for intermediate molecules and minimality test for finished molecules, and (mode 1) minimality test for intermediate and finished molecules. All three modes generate the same molecules, but they require a different amount of time depending if the input is only the EC or also prescribed substructures. On the one hand, semi-canonicity is faster to check than minimality, but it removes less intermediate molecules, which results in more branches in the search tree. On the other hand, minimality, yet more computationally demanding, constrains more and produces fewer branches in the search tree. The effect on computation time of performing a faster test more times (semi-canonicity) versus a slower test less times (minimality) needs to be assessed. Additionally we assess the impact of using or not using good and bad substructure and bad rings have on the overall performance of the different modes available in PMG.

Multi-core execution

The algorithm used for OMG allowed theoretically parallel execution, but this was not implemented as such. In the current PMG we have implemented the new algorithm in a thread-safe way to allow its execution in parallel. We tested how well the three execution modes and the different multi-core setups (increasing number of cores) performed. In order to show how PMG can be applied in a metabolomics

context, we used the elemental compositions and prescribed substructures used in[7] and compared the results, both in time and structures generated, by PMG with those obtained by OMG.

Priori Constraints

PMG, as OMG did, accepts prescribed non-overlapping substructures. These are substructures that should be present in the finished molecules and they fix how a part of the whole structure is connected, reducing drastically the number of generated molecules. In our previous work we obtained prescribed substructures for unknown metabolites by finding metabolites with similar MS^n spectra to the unknown, and calculating their maximum common substructure (MCSS). The MCSS is the biggest part of the chemical structures that they share. The assumption was that our unknown would have a chemical structure that is similar to the structures of metabolites having similar spectra. We demonstrated that removing molecules with a high force field energy value significantly reduced the number of candidate structures. Additionally, we observed that certain ring moieties contribute significantly to an increase of the energy value of a molecule. Therefore, we have implemented an efficient way to test if certain unrealistic molecular rings and ring systems (Figure 1) are present in our intermediate molecules. The bad rings implemented in PMG are: a ring of any size with a triple bond; an atom with two double bonds, which are in a ring; two three member rings fused together; a three member ring with a double bond; a four member ring with two double bonds. When a double or triple bond is created, or a ring is closed we test if whether unwanted rings are present in the molecule. These bad rings match all types of heavy atoms,

i.e. they are not specific for carbon atoms. Rejecting intermediate molecules with bad rings reduces the number of finished candidate structures and speeds up the execution of PMG.

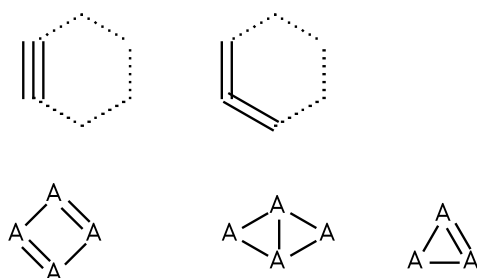


Figure 1 Bad rings used as prior constraint in PMG.

Posterior Constraints

Not all constraints can be applied from the beginning to intermediate structures to reduce the search space. Therefore, these constraints need to be applied on finished molecules before they are considered as acceptable candidates. As mentioned above, prescribed substructures should not overlap. In metabolomics studies it is possible to have multiple substructures that we want our candidate structures to contain, but we cannot know if they overlap or not. In such a situation, the biggest (more constraining) substructure is provided as prescribed substructure (a priori constrain) and the smaller substructures (which we refer to as good list) tested on finished molecules (posterior constrain). PMG makes use of the Small Molecule Subgraph Detector (SMSD) library[24] to query if the substructures in the good list are present in the finished molecules. Finished molecules will only be accepted provided they contain all the substructures present in the good list.

When multiple substructures are available and it is not possible to know if they overlap, the biggest substructure should be provided as prescribed and the smaller substructures as good list. These substructures can be obtained from analytical data, for instance, by matching MS/MS spectra with a database of annotated spectra[25], by manual[26] or automated[27] interpretation of MSⁿ data, by calculating the MCSS of metabolites with similar MSⁿ spectra,[16] or by interpreting NMR spectra.[28] If an analytical method has been used that includes derivatization of metabolites, the attached moiety can be provided as well in the good list. Alternatively the good list can be built using biological knowledge, for instance, if the unknown belongs to a metabolic pathway where all its metabolites share common substructures.

Apart from rejecting molecules that contain bad rings and do not contain substructures from the good list, we implemented a bad list rejection option in PMG using SMSD. Here, the user can provide other substructures different than the bad rings that he does not want in his finished molecules. Finished molecules will be rejected if they contain one of the unwanted substructures in the bad list. We offer an example of a bad list containing benzene and cyclopentane fused with 3 and 4-membered rings (Figure 2). These ring systems are energetically unfavorable and unstable, thus not likely present in metabolites. To support this claim, we queried these ring systems in HMDB and we did not find human metabolites that contained them. Alternatively, different unwanted substructures can be drawn with a chemical editor (like Marvin Sketch or ISIS/ ChemDraw), stored as an SD file and provided as bad list. Again, biological knowledge can be used to derive a bad list. For example, a certain metabolic pathway is studied and all the metabolites involved, including the

unknown, do not contain rings. To prevent PMG of generating ring containing molecules a bad list could be assembled with different rings.

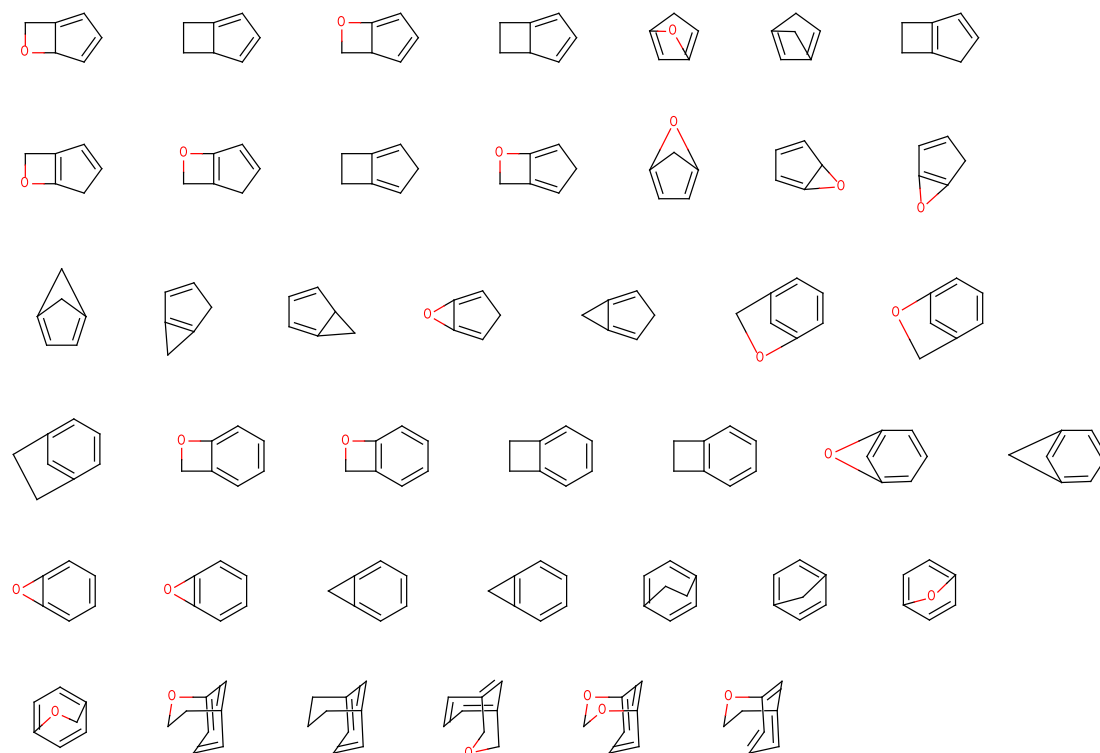


Figure 2 Bad list of substructures used as posterior constraint in PMG.

We evaluated the effect and the suitability for metabolite identification of the improvements in PMG. Firstly, to test the speed improvement of PMG, we used the same elemental compositions (Tables 1 and 2) and fragments (Table 3) employed in the original OMG publication.[7] These elemental compositions belong to human metabolites found in HMDB. The fragments, manually sketched, were present in the metabolites and likely to be found in experimental MS^n data. We assessed the speed gain of the new algorithm by comparing the total time per elemental composition of the single core execution of the three modes in PMG with OMG, when the input is only an elemental composition or an elemental composition and prescribed substructures. Secondly, we also assessed the effect of using multiple cores by

comparing the number of molecules generated per second when using one, two, four, seven and ten cores. Lastly, we used elemental compositions and substructures of real unknown metabolites found in human urine to test the effect of using the filters of prescribed substructures, bad rings, good and bad lists. MS^n data of known and unknown metabolites present in human urine were acquired, processed with MEF and compared to a spectral database of known metabolites. From those metabolites in the database found to be similar to the unknowns, a MCSS was calculated and proposed as a prescribed substructure. Additionally, the fragmentation trees of these unknowns underwent manually annotation, which in some cases provided additional substructures, which were used as good list for the evaluation of PMG.

Results and discussion

The speed of the three algorithms implemented in PMG is compared for some typical metabolites, which we detected earlier in urine[4] with the original results of OMG. PMG was executed in the same computer and under the same conditions used to obtain the results for OMG in our previous work. All three algorithms in PMG outperform OMG when using only elemental formulas containing carbon, oxygen and hydrogen atoms (Table 1) and other additional elements like nitrogen, phosphorus or sulfur (Table 2). For elemental compositions with a small number of elements like $C_3H_4O_3$ and $C_2H_5NO_2$, OMG and PMG need approximately the same amount of computation time. When the elemental composition has more elements and the structure generators should produce thousands or millions of molecules, PMG can be 40 times faster than OMG using only one core. From the three algorithms, semi-

canonicity & minimality combined appeared to be the fastest, followed by minimality and finally by semi-canonicity. The test for minimality is more time expensive than the test for semi-canonicity but at the same time it is more constraining. With these results we conclude that it is helpful to use the combined semi-canonicity & minimality mode when the only input is the elemental composition.





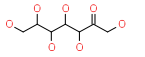
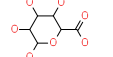
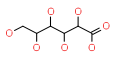
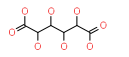
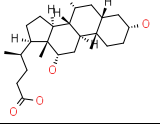

Structure	Name HMDB ID Elemental Composition	# Candidate Structures	OMG	PMG - SC-Min	PMG - SC	PMG - Min
			Time (s)	Time (s)	Time (s)	Time (s)
	Pyruvic acid HMDB00243 C ₃ H ₄ O ₃	152	0.51	0.34 (1 core)	0.33 (1 core)	0.33 (1 core)
	Malic acid HMDB00156 C ₄ H ₆ O ₅	8,070	27.07	1.85 (1 core)	1.95 (1 core)	1.89 (1 core)
	D-Xylose HMDB00098 C ₅ H ₁₀ O ₅	18,092	125	4.75 (1 core)	6.73 (1 core)	5.19 (1 core)
	D-Fructose HMDB00660 C ₆ H ₁₂ O ₆	267,258	5,035	121 (1 core)	305 (1 core)	145 (1 core)
	Sedoheptulose HMDB03219 C ₇ H ₁₄ O ₇	4,106,823	186,248	427 (10 cores)	2259 (10 cores)	567 (10 cores)
	Pectin HMDB03402 C ₆ H ₁₀ O ₇	3,183,337	46,320	129 (10 cores)	453 (10 cores)	155 (10 cores)
	Galactonic acid HMDB00565 C ₆ H ₁₂ O ₇	767,569	22,475	55.1 (10 cores)	166 (10 cores)	70.22 (10 cores)
	Galactaric acid HMDB00639 C ₆ H ₁₀ O ₈	8,568,129	186,730	484 (10 cores)	1,704 (10 cores)	599 (10 cores)
	Cholic acid HMDB00619 C ₂₄ H ₄₀ O ₅	* More than 2,147,483,646	* not available	* not available	* not available	* not available
	Phenyllactic acid HMDB00779 C ₉ H ₁₀ O ₃	48,496,265	** not available	877 (10 cores)	4,653 (10 cores)	901 (10 cores)

Table 1 Number of candidate substructures produced by PMG using only an elemental composition (only C,H,O) and the time in seconds to generate them using 1 or 10 cores. Abbreviations: SC-Min, Semi-canonicity & minimality combined; SC, semi-canonicity; Min, minimality. SC-Min outperforms the other two methods and is the preferred method when only using elemental compositions as input.



















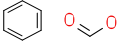
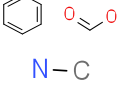
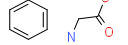
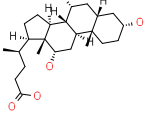
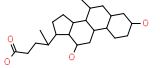
Structure	Name HMDB ID Elemental Composition	# Candidate Structures	OMG	PMG - SC-Min	PMG - SC	PMG - Min
			Time (s)	Time (s)	Time (s)	Time (s)
	Glycine HMDB00123 C ₂ H ₅ NO ₂	97	0.45	0.34	0.33	0.33
	Acetyl-glycine HMDB00532 C ₄ H ₇ NO ₃	26,530	126	4.76	6.13	4.93
	Phenylalanine HMDB00159 C ₉ H ₁₁ NO ₂	516,741,797	* not available	17,254.79 (10 cores)	* not available	* not available
	Glutamic acid HMDB00148 C ₅ H ₉ NO ₄	685,392	12,348	22.17 (10 cores)	47.40 (10 cores)	22.87 (10 cores)
	Phosphoenolpyruvic acid HMDB00263 C ₃ H ₅ O ₆ P	83,977	761	29.48	34.87	30.01
	Creatinine HMDB00562 C ₄ H ₇ N ₃ O	303,601	3,921	128	201	127
	Guanidinoacetic acid HMDB00128 C ₃ H ₇ N ₃ O ₂	124,808	1,962	68.13	82.04	67.59
	Cytosine HMDB00630 C ₄ H ₅ N ₃ O	491,299	3,952	135	198	134
	Uric acid HMDB00289 C ₅ H ₄ N ₄ O ₃	* More than 464,899,034	* not available	* not available	* not available	* not available
	Histamine HMDB00870 C ₅ H ₉ N ₃	134,278	26.56	74.14	180	74.33
	D-Cysteine HMDB03417 C ₃ H ₇ NO ₂ S	15,978	131	6.38	7.23	6.36
	p-Cresol sulfate HMDB11635 C ₇ H ₈ O ₄ S	* More than 82,000,000	* not available	* not available	* not available	* not available

Table 2 Number of candidate substructures produced by PMG using only an elemental composition and the time in seconds to generate them using 1 or 10 cores. Abbreviations: SC-Min, Semi-canonicity & minimality combined; SC, semi-canonicity; Min, minimality. SC-Min and Min perform equally well. SC-Min is the preferred method when only using elemental compositions as input, since it performs better than Min when the elemental compositions contain only C,H and O.

The speed improvement becomes more significant when we provide prescribed substructures to the structure generator (Table 3). In this case, PMG using the semi-canonicity & minimality mode achieves in some cases a 100-fold increase in speed compared to OMG. We observe this for both small metabolites with prescribed substructures (phenylalanine) and large metabolites with substructures (cholic acid).

De novo identification of metabolites with open molecule generator for metabolomics

In some other cases, like malic acid, PMG only achieved a 4-fold improvement. We also see that the semi-canonicity & minimality mode performs similarly to the semi-canonicity mode and both outperform the minimality mode. In all cases the same number of candidate structures were obtained with the three PMG modes. Since semi-canonicity & minimality is the best performing algorithm with only elemental compositions and also when substructures are provided, we conclude that semi-canonicity & minimality is the best generic algorithm for structure generation and should be used as default. We also observe that, in some cases, PMG produces more molecules than OMG when using prescribed substructures. This had to do with the way the original OMG algorithm handled atom elements of multiple valences (N valence 3 and 5; S valence 2,4, and 6; P valence 3 and 5) when using substructures. PMG improved this and generates all possible and valid valences when using prescribed substructures.

Structure	Name HMDB ID Elemental Composition	Prescribed Substructures	OMG		PMG - SC-Min		PMG - SC	PMG - Min
			# Candidate Structures	Time (s)	# Candidate Structures	Time (s)	Time (s)	Time (s)
	Glycine HMDB00123 C ₂ H ₅ NO ₂		6	0.54	6	0.26	0.22	0.22
	D-Cysteine HMDB03417 C ₃ H ₇ NO ₂ S		210	3.18	210	0.97	0.91	0.98
	Phenylalanine HMDB00159 C ₉ H ₁₁ NO ₂		107,155	19,386	119,955	356	356	2,002
			595	271	595	3.47	3.47	5.56
			289	172	289	2.48	2.38	2.39
			26	25.15	26	0.97	0.94	0.92
	Cholic acid HMDB00619 C ₂₄ H ₄₀ O ₅		334	120	334	0.77	0.78	0.88

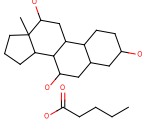






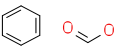
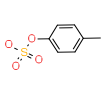


			2,505	119	2,50	1.47	1.38	1.38
	Malic acid HMDB00156 C ₄ H ₆ O ₅		1,436	4.69	1,436	1.16	1.12	4.90
	Uric acid HMDB00289 C ₅ H ₄ N ₄ O ₃		6,069,863	155,828	7,357,453	9,593	* not available	* not available
	Phenyllactic acid HMDB00779 C ₉ H ₁₀ O ₃		26,164	163	29,580	24.97	24.89	570
			525	3.97	525	1.16	1.16	1.46
	p-Cresol sulfate HMDB11635 C ₇ H ₈ O ₄ S		13,177	63.05	13,177	12.78	57.33	10.34
			17,232	1,204	19,132	64.85	110	989

Table 3 Number of candidate substructures produced by PMG using an elemental composition and prescribed substructures and the time in seconds to generate them using 1 core. Abbreviations: SC-Min, Semi-canonicity & minimality combined; SC, semi-canonicity; Min, minimality. SC-Min performs similar to SC and outperforms Min, therefore SC-Min is the preferred method when only using elemental compositions as input.

The impact of running PMG on multiple cores for the structure generation of molecules where only the elemental composition is provided, is presented in Figure 3, and for those with given elemental compositions and prescribed substructures in Figure 4. The results are presented in molecules per second (in logarithmic scale) using the semi-canonicity & minimality combined method. We see for both cases about a 10-fold improvement from OMG to PMG in the number of molecules generated per second for a single core execution (Figures 3 & 4). When only elemental compositions are used as input, the speed of generation of molecules with PMG increase rather linear with the number of cores. When using also prescribed substructures as input, we observe a comparable improvement in the speed of generation of molecules with the increasing the number of cores only for half of the examples, and especially for those that need longer computation times.

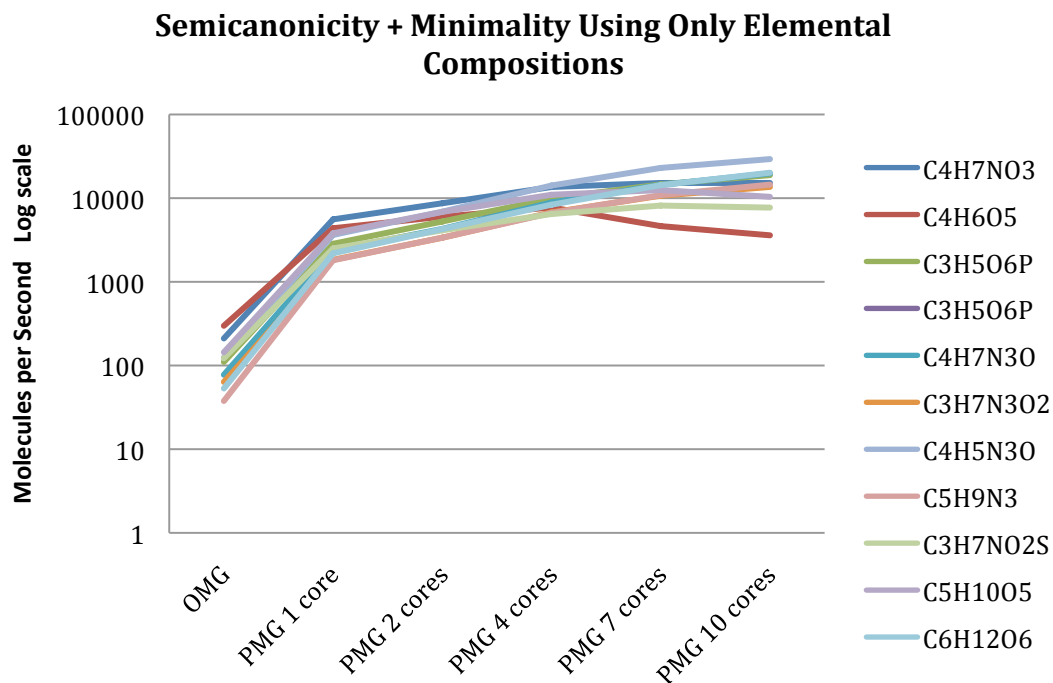


Figure 3 Number of molecules generated (logarithmic scale) only with elemental compositions using OMG and PMG in single core, and PMG in multiple cores. PMG in single core is 10 times faster than OMG. Multicore execution of PMG achieves near linear speed up.

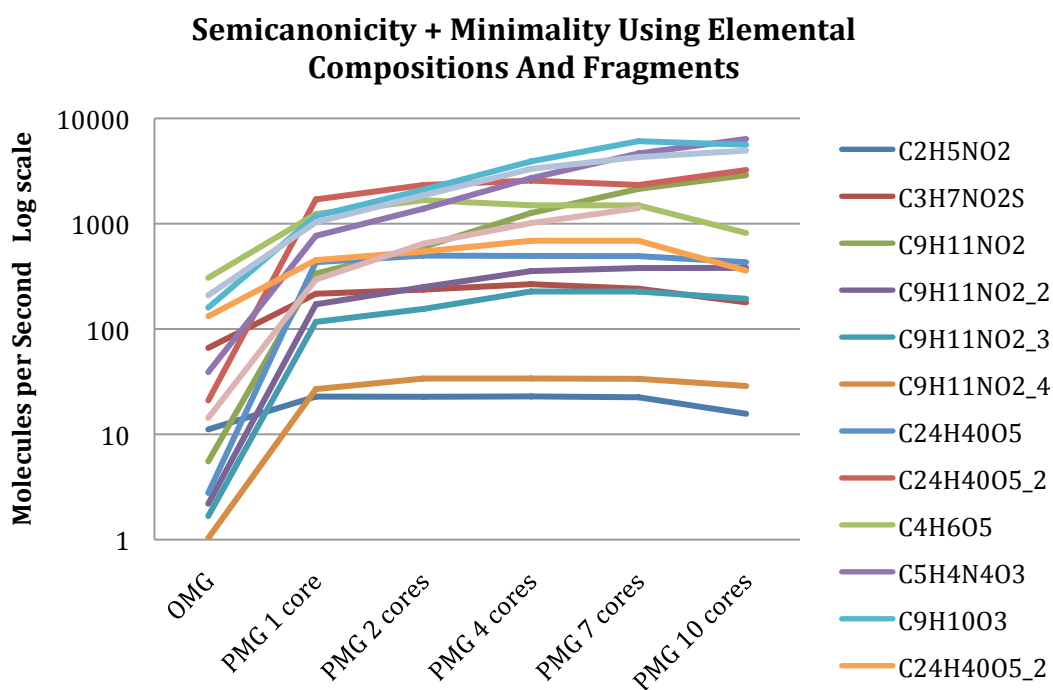


Figure 4 Number of molecules generated (logarithmic scale) with elemental compositions and prescribed substructures using OMG and PMG in single core, and PMG in multiple cores. In most of the cases, PMG in single core achieves more than 50-fold increase in speed compared to OMG. Multicore execution of PMG achieves from no speed improvement to linear speed up.

The results of using bad rings as a prior constraint and a bad list as posterior constraint are presented in Table 4: the number of molecules generated by PMG unconstrained are compared with PMG using the above constraints, where both executions used the semi-canonicity & minimality combined method and 10 cores. We observed that in some cases ($C_2H_5NO_2$, $C_7H_{14}O_7$, $C_6H_{10}O_7$) PMG only rejected a few molecules, or in other cases no rejection of structures at all was achieved ($C_5H_{10}O_5$, $C_6H_{12}O_6$). The reason for this is that the number of hydrogen atoms in these formulas impedes rings to be formed, therefore no bad rings or bad substructures can be found. For formulas that allow ring formation, PMG using the constraints removes in some cases 82% ($C_4H_5N_3O$), 62% ($C_4H_7N_3O$) or 61% ($C_5H_9N_3$) of the candidate structures. In these cases, constraining the generation of molecules in PMG with bad rings and bad list of substructures can provide a significant reduction in the number of generated molecules.

Formula	# Candidate Structures	Calcualtion time (s)	# Candidate Structures	Calculation time (s)
	Without constraints		With bad rings and bad list constraints	
$C_2H_5NO_2$	97	0.35	95	0.69
$C_4H_7NO_3$	26,530	1.75	21,329	6.70
$C_9H_{11}NO_2$	516,741,797	17,254.80	235,017,993	77,723
$C_5H_9NO_4$	685,392	22.17	582,745	168
$C_3H_4O_3$	152	0.35	129	0.80
$C_4H_6O_5$	8,070	2.24	7,464	4.00
$C_3H_5O_6P$	83,977	4.45	74,422	23.63
$C_9H_{10}O_3$	48,496,265	877	28,468,157	8,065
$C_4H_7N_3O$	303,601	15.36	115,475	25.32
$C_3H_7N_3O_2$	124,808	9.16	78,753	22.43
$C_4H_5N_3O$	491,299	16.77	88,400	23.51
$C_5H_9N_3$	134,278	9.25	52,574	12.41
$C_3H_7NO_2S$	15,978	2.07	12,054	4.62
$C_5H_{10}O_5$	18,092	1.74	18,092	7.01
$C_6H_{12}O_6$	267,258	13.27	267,258	74,571
$C_7H_{14}O_7$	4,106,823	427	4,106,823	1,201
$C_6H_{10}O_7$	3,183,337	129	3,057,256	836

Table 4 Number of candidate structures generated by PMG with and without bad rings and bad list constraints and the time in seconds to generate them using 1 core.

We tested further the use of prior and posterior constraints in addition to the elemental compositions and prescribed substructures (Table 5) of unknown metabolites found in human urine in our earlier work.[4] Some prescribed substructures were obtained using the fragmentation tree similarity[16] and MCSS methods. Other prescribed substructures and the good list substructures were annotated by an expert after manual interpretation of the MSⁿ trees. For unknown 28, PMG produces 86 candidate structures using a prescribed substructure and only 12 using the same prescribed substructure and the bad rings constraint. This reduction is comparable to the one obtained by OMG and a posterior filtering based on energy, metabolite-likeness[12] and fragmentation prediction.[29] In other cases, like unknown 9, using prescribed substructure and good list results in PMG generating one unique candidate structure. For unknown 27, prescribed substructures and good list produced 45 structures, which was reduced to 3 candidates using the bad rings constraints. As we can see, only in a few cases the use of bad rings and bad list filtering reduces further the number of candidates. But in those cases, the effect is similar to removing molecules with high force field energy and low metabolite-likeness. It also has to be taken into account that the prescribed substructure fixes a big part of the chemical structure, making it difficult to produce bad rings with the remaining atoms. The combination in PMG of prescribed substructures, good list, bad list and bad rings achieves a significant reduction in the number of generated molecules for real unknown metabolites.





















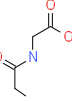
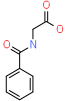
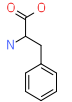

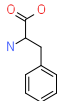
Unknown	Elemental Composition	Prescribed Substructure	Good list	# Candidate Structures	
				without	with
				bad rings and bad list constraints	
7	C ₈ H ₉ NO ₂			32	32
7	C ₈ H ₉ NO ₂			1	1
7	C ₈ H ₉ NO ₂			270	270
7	C ₈ H ₉ NO ₂			42	42
12	C ₉ H ₉ NO ₃			121,363	50,003
18	C ₉ H ₁₁ NO ₃			4,251	4,245
20	C ₁₁ H ₁₂ N ₂ O ₂			1,891	1,748
22	C ₉ H ₁₁ NO ₂			165	165
28	C ₉ H ₁₀ O ₂			86	12
9	C ₇ H ₇ NO ₄			80,489	53,194
9	C ₇ H ₇ NO ₄			1	1
9	C ₇ H ₇ NO ₄			17,680	14,800
15	C ₉ H ₉ NO ₄			5	5
15	C ₉ H ₉ NO ₄			5	5
27	C ₉ H ₁₃ NO ₄ P ₂			45	3
27	C ₉ H ₁₃ NO ₄ P ₂			281	264

Table 5 Number of candidate structures generated by PMG (SC-Min; 10 cores) applied to the identification of unknown metabolites in urine using prescribed substructures, good list, and with/without bad rings and bad list constraints.

In previous work we used OMG to identify unknown metabolites in human urine.[4]

After the generation step we used an internal energy filter and a metabolite-likeness filter to reduce further the list of candidates. The use of these filters after PMG is obviously possible, but was not in the scope of this work. However, we observed

that using the bad rings and bad substructures constraints was rather equivalent to remove energetically unfavorable molecules.

Conclusion

We presented PMG, an open source multi-core structure generator developed for de-novo metabolite identification. We implemented a new algorithm based on semi-canonicity and minimality tests, with a multi-core architecture. This implementation generates candidate structures up to 100 times faster than our previous structure generator OMG. Next to the elemental composition, PMG allows as input prescribed substructures and bad and good lists of substructures. The use of good substructures limits the search space while the bad rings removes energetically unfavorable molecules. This results in a short list of candidate structures, without missing the correct structure. We have used substructures obtained from MSⁿ spectra, derived by manual annotation and using cheminformatic tools, to illustrate the use of PMG. Alternatively, these substructures could have originated from NMR or from database search.

We are convinced that PMG will improve de-novo metabolite identification by providing a short yet correct list of candidate structures at a high-speed. Obviously, PMG can be also applied to the identification of also other molecules than metabolites only or applied in other applications requiring a structure generator.

References

1. Dunn WB, Erban A, Weber RJM, Creek DJ, Brown M, Breitling R, Hankemeier T, Goodacre R, Neumann S, Kopka J, Viant MR: Mass appeal: metabolite identification in mass spectrometry-focused untargeted metabolomics. *Metabolomics* 2012.
2. Johnson CH, Gonzalez FJ: Challenges and opportunities of metabolomics. *Journal of Cellular Physiology* 2012, 227:2975–81.
3. Scalbert A, Brennan L, Fiehn O, Hankemeier T, Kristal BS, Van Ommen B, Pujos-Guillot E, Verheij E, Wishart D, Wopereis S: Mass-spectrometry-based metabolomics: limitations and recommendations for future progress with particular focus on nutrition research. *Metabolomics* 2009, 5:435–458.
4. Peironcely JE, Rojas-Cherto M, Tas A, Vreeken R, Reijmers T, Coulier L, Hankemeier T: An Automated Pipeline For De Novo Metabolite Identification Using Mass Spectrometry-Based Metabolomics. *Analytical Chemistry* 2013, 7:3576–3583.
5. Van der Hooft JJJ, De Vos RCH, Mihaleva V, Bino RJ, Ridder L, De Roo N, Jacobs DM, Van Duynhoven JPM, Vervoort J: Structural elucidation and quantification of phenolic conjugates present in human urine after tea intake. *Analytical chemistry* 2012, 84:7263–71.
6. Kind T, Fiehn O: Advances in structure elucidation of small molecules using mass spectrometry. *Bioanalytical reviews* 2010, 2:23–60.
7. Peironcely JE, Rojas-Cherto M, Fichera D, Reijmers T, Coulier L, Faulon J-L, Hankemeier T: OMG: open molecule generator. *Journal of Cheminformatics* 2012, 4:21.
8. Laue R, Gr T, Meringer M, Kerber A: Constrained Generation of Molecular Graphs. In volume 69 of DIMACS series in discrete mathematics and theoretical computer science: Graphs and discovery. American Mathematics Society; 2005, 69:319–332.
9. Kerber A, Laue R, Grüner T, Meringer M: MOLGEN 4.0. *Match* 1998, 37:205 – 208.
10. Faulon J-L: Isomorphism, Automorphism Partitioning, and Canonical Labeling Can Be Solved in Polynomial-Time for Molecular Graphs. *Journal of Chemical Information and Modeling* 1998, 38:432–444.
11. Kaski P, Östergård PJ: Classification of Designs. In *Classification Algorithms for Codes and Designs* SE - 6. Springer Berlin Heidelberg; 2006, 15:175–218.
12. Peironcely JE, Reijmers T, Coulier L, Bender A, Hankemeier T: Understanding and Classifying Metabolite Space and Metabolite-Likeness. *PLoS ONE* 2011, 6:e28966.
13. Wishart DS, Knox C, Guo AC, Eisner R, Young N, Gautam B, Hau DD, Psychogios N, Dong E, Bouatra S, Mandal R, Sinelnikov I, Xia J, Jia L, Cruz JA, Lim E, Sobsey CA, Shrivastava S, Huang P, Liu P, Fang L, Peng J, Fradette R, Cheng D, Tzur D, Clements M, Lewis A, Souza A De, Zuniga A, Dawe M, Xiong Y, Clive D, Greiner R, Nazzyrova A, Shaykhtudinov R, Li L, Vogel HJ, Forsythe I: HMDB: a knowledgebase for the human metabolome. *Nucleic Acids Research* 2009, 37:D603–610.
14. Marvin 5.11.0 ChemAxon (<http://www.chemaxon.com>): Marvin 5.11.0. 2012.
15. Rojas-Chertó M, Kasper PT, Willighagen EL, Vreeken RJ, Hankemeier T, Reijmers TH: Elemental composition determination based on MS(n). *Bioinformatics* 2011, 27:2376–83.
16. Rojas-Chertó M, Peironcely JE, Kasper PT, Van der Hooft JJJ, De Vos RCH, Vreeken R, Hankemeier T, Reijmers T: Metabolite Identification Using Automated Comparison of High-Resolution Multistage Mass Spectral Trees. *Analytical Chemistry* 2012, 84:5524–5534.
17. Colbourn C, Read R: Orderly algorithms for graph generation. *International Journal of Computer Mathematics* 1979, 7:167–172.
18. Faradzev IA: Constructive Enumeration of Combinatorial Objects. in *Problèmes combinatoires et théorie des graphes*, University of Paris, Orsay 1978:131–135.
19. McKay B: Isomorph-Free Exhaustive Generation. *Journal of Algorithms* 1998, 26:306–324.
20. Brinkmann G: Isomorphism rejection in structure generation programs. *Discrete Mathematical Chemistry* 2000, 51:25 – 38.
21. Faulon J-L, Visco DP, Roe D: Enumerating Molecules. In *Reviews in Computational Chemistry*. John Wiley & Sons, Inc.; 2005:209–286.
22. Meringer M: Structure Enumeration and Sampling. In *Handbook of Chemoinformatics Algorithms*. 2009:235–271.
23. Meringer M: Fast Generation of Regular Graphs and Construction of Cages. *Journal of Graph Theory* 1999:137–146.

24. Rahman SA, Bashton M, Holliday GL, Schrader R, Thornton JM: Small Molecule Subgraph Detector (SMSD) toolkit. *Journal of Cheminformatics* 2009, 1:12.
25. Kertesz TM, Hill DW, Albaugh DR, Hall LH, Hall LM, Grant DF: Database searching for structural identification of metabolites in complex biofluids for mass spectrometry-based metabolomics. *Bioanalysis* 2009, 1:1627–43.
26. Hooft JJJ, Vervoort J, Bino RJ, Vos RCH: Spectral trees as a robust annotation tool in LC–MS based metabolomics. *Metabolomics* 2011, 8:691–703.
27. Ridder L, Van der Hooft JJJ, Verhoeven S, De Vos RCH, Van Schaik R, Vervoort J: Substructure-based annotation of high-resolution multistage MS(n) spectral trees. *Rapid communications in mass spectrometry : RCM* 2012, 26:2461–71.
28. Elyashberg M, Blinov K, Molodtsov S, Smurnyy Y, Williams AJ, Churanova T: Computer-assisted methods for molecular structure elucidation: realizing a spectroscopist's dream. *Journal of Cheminformatics* 2009, 1:3.
29. Wolf S, Schmidt S, Müller-Hannemann M, Neumann S: In silico fragmentation for computer assisted identification of metabolite mass spectra. *BMC Bioinformatics* 2010, 11:148.

6 *Conclusions and perspectives*

Conclusions and perspectives

Metabolite identification is still one of the biggest bottlenecks in metabolomics. The de novo structure elucidation of metabolites is very time consuming and poses several challenges. However, without the identity of a peak detected with mass spectrometry-based metabolomics methods its biological role cannot be interpreted. In addition, effective integration with other 'omics' data such as genomics and proteomics requires the identity of metabolites. Therefore there is a huge need for more efficient strategies to identify metabolites. Multi-stage mass spectrometry (MS^n) is a promising type of mass spectrometry from which information on the fragmentation pattern of the metabolite can be obtained.

In this thesis new algorithms and methods that enable the de novo identification of metabolites have been developed. The aim was to find methods to propose candidate structures for unknown metabolites using MS^n data as starting point. Ideally this list of candidate identities should be as short as possible by using additional constraints so that an expert can easily inspect it and select some structures for further validation. The algorithms and methods, which have been developed in this thesis and a parallel project, have been integrated to into a semi-automated pipeline to identify new metabolites starting from multi-stage mass spectrometry. The focus in this thesis was on human metabolites. The discovery of new metabolites will improve our capability to understand disease via its metabolic fingerprint, to develop personalized treatments and to discover new drugs. In

addition, the cheminformatics methods presented in this thesis increase our understanding on the properties of human metabolites.

In Chapter 2 a structure generator, the Open Molecule Generator (OMG), was developed. This tool allowed generating candidate structures for unknown metabolites with a certain elemental composition and known substructure(s). While research in computer assisted structure elucidation (CASE) dates back to the 1960s, recent developments have been scarce. The most notable example was MOLGEN, an efficient, but at the same time “black box” commercial structure generator. Therefore the in Chapter 2 developed structure generator was customizable and open source: this allowed to implement methods and algorithms that were relevant to the envisioned identification pipeline as described in Chapter 5. Being open source, OMG permits other scientists to improve it further and to adapt it to future needs.

The efficiency of the generation of structure candidates for a given elemental composition (EC) and substructure(s) was demonstrated for a number of metabolites. The results showed that OMG produced all possible chemical structures for a given input (EC and/or substructure(s)). These results only contained chemically valid structures according to the valence rule, whereas MOLGEN could produce some unwanted atom types: for instance, for P valence 5, MOLGEN would generate P atoms with a triple and a double bond. OMG allowed also to use several substructures as constrain, whereas MOLGEN allows only one substructure as constrain. The use of prescribed substructures constrained significantly the number

of candidate structures obtained. They reduced for example a list of hundreds of millions of molecules (unconstrained generation) to a few thousand structures (constrained generation). While this is a significant reduction, reducing the list further using additional constraints appeared essential to turn the identification of unknowns into a tractable problem. In conclusion, a structure generator was developed which was superior to MOLGEN in several aspects. However, OMG was slow when generating many molecules. The reason was that the algorithm used in OMG was based on the canonical augmentation path. This algorithm produces all possible molecules without duplicates, but it requires the use of a canonizer for duplicate removal. This canonizer calculates the canonical representative for each intermediate (unfinished) molecule. This conversion from a molecule to its canonical representative is computationally time expensive. We concluded that a better algorithm with a better duplicate removal approach was required to speed up calculations and make the tool more attractive for practical use. Such an algorithm was developed in Chapter 5.

In Chapter 3 we studied the nature of human metabolites. The aim was two-fold: (i) to learn what characteristics differentiate metabolites from other small (< 1000 Da) non-metabolite molecules and (ii) to predict the metabolite-likeness of a chemical structure, i.e. how likely it is to be a human metabolite. In the classification of molecules, as in many machine-learning problems, one has to deal with an interpretability trade off. While easy to understand molecule descriptions (such as physicochemical properties or scaffolds) combined with easy to interpret predictive models mostly provide poor predictive results, complex descriptions (structural

fingerprints) together with difficult to interpret models provide often better predictions. In other words, if we want to accurately predict metabolite-likeness we will probably not be able to easily understand in chemical terms what these predictions are based on.

We observed that metabolites are a heterogeneous family of compounds and compared to non-metabolites, metabolites have simpler structures, less ring systems, more hydrophilic groups, less nitrogen and sulfur atoms, and more oxygen and phosphor atoms. These easy to interpret features were not complex enough to be used in a model that would discern between metabolite and non-metabolite molecules. Therefore, we developed and validated a metabolite-likeness predicting model, which used the molecular descriptor MDL Public Keys and the Random Forest classification algorithm to assign a metabolite score to a molecule. This model achieved the highest classification accuracy at the expense of low interpretability. The model was effectively used in Chapter 4 to reject non-metabolite candidate structures for unknown metabolites, demonstrating that metabolite-likeness prediction is one of the tools that can be routinely used in metabolite identification studies.

In Chapter 4 the tools developed in Chapters 2 and 3 of this thesis and developed in a parallel project within the Netherlands Metabolomics Centre [69–71] were integrated into a pipeline to identify metabolites. This pipeline was composed of four modules. The first module annotated MSⁿ data to obtain a fragmentation tree comprising fragment ions and neutral losses of known elemental composition using the

Multistage Elemental Formula (MEF) tool [69]. The second module compared this fragmentation tree of an unknown with the fragmentation trees of known metabolites in a home-build database. If a fragmentation tree with 100% similarity was found, the identity of the unknown was provisionally assigned. If more than one fragmentation tree was found with less than 100% similarity, a substructure of that unknown metabolite was calculated via the maximum common substructure (MCSS) from the metabolites being similar to the fragmentation tree. The third module was the OMG structure generator, using as input the elemental composition and, if found, a substructure of that metabolites. The fourth module used three filters to reduce the list of candidates generated by OMG: (i) a Metabolite-likeness filter, which kept only molecules resembling human metabolites, (ii) a internal energy filter, which kept only energetically stable molecules and (iii); the MetFrag filter, which predicted the mass spectral fragmentation of molecules and kept only those candidate molecules that could explain half of the fragments observed experimentally.

This metabolite identification pipeline was tested for the identification of 30 different MSⁿ spectra obtained from unknown metabolites in a human urine sample. For 10 of the 30 unknowns a perfect match of the acquired fragmentation tree and a fragmentation tree of a known metabolite in the database was found, and therefore, these metabolites were provisionally identified. For 9 unknown metabolites two or more similar metabolites were found in the MSⁿ database, which allowed the calculation of a maximum common substructure as input to constrain the number of structures obtained by the OMG structure generator. For 3 out of these 9 unknowns,

OMG and the different filters provided a short list of 8 or less candidate structures. For 6 out of these 9 unknowns, the list of candidates was excessively large, i.e. larger than 40 candidates. Lastly, 11 unknowns remained unidentified because no similar metabolite was found in the database and therefore OMG using only the elemental composition provided more than one million of candidate structures.

In summary, the developed identification pipeline proved to be useful at identifying unknown metabolites using only MS^n data. If a metabolite is already in a MS^n database, the metabolites can be well provisionally identified based only on the MS^n spectrum; obviously, in this manner only metabolites already known can be identified. In order to identify truly new metabolites, i.e. de novo identification, one needs to produce a short list of candidate structures for a given fragmentation tree, and that is only possible if sufficient substructure information is available and powerful filters are used. This actually depends heavily on the availability of comprehensive MS^n database of known metabolites from which similar metabolites can be found and a maximum common substructure can be derived. It would be very beneficial if annotated subtrees in the database with their corresponding substructures would be available. This would allow finding multiple annotated subtrees with their substructures for a MS^n tree of an unknown metabolites in such a database. This would provide multiple prescribed substructures as input for the structure generator. Such multiple substructures as input are only possible using manual interpretation of a MS^n tree. The three filters used (Metabolite-Likeness, internal energy and fragmentation prediction) proved to be useful at reducing the number of candidates to an amount that could be inspected by an expert for further

validation. Obviously this identification pipeline for known and fully unknown molecules (i.e. not reported in any database) does not only apply to human metabolites, but can be applied also to plant metabolites or other types of molecules.

In Chapter 5, we implemented and tested the Parallel Molecule Generator (PMG). This structure generator addresses some of the lessons learned in Chapter 2 (OMG) and Chapter 4 (metabolite identification pipeline). Firstly in both chapters we observed that using multiple substructures as constraints could make many identification problems feasible using the in Chapter 4 developed identification pipeline. Secondly, OMG should be improved by using a faster algorithm, by reducing the need for a full canonizer, i.e. use a less computationally demanding method to remove duplicate structures, and by parallel execution of the algorithm. Lastly, in Chapter 4 we have learned that filters like removing energetically unfavorable structures are important and they should be already incorporated in the structure generating process. The rationale for implementing such filters in PMG is that providing a short list of candidates produced slowly is more desirable than a long list of candidates produced quickly, since a shorter list of candidates brings us closer to the identification of the unknown metabolite. We achieved a reduction in the number of possible candidate structures for a given elemental composition by including several constraints, i.e. using prescribed substructures, good and bad lists, exclusion of energetically unfavorable bad rings. In addition, we reduced the impact on the computational time of including these constraints in the method by implementing two new algorithms. These run in parallel, therefore they produce

results faster, they can accommodate more constraints and their results are as complete as those the original OMG algorithm.

The increase in speed using PMG compared to the OMG was evaluated for the same elemental compositions and associated substructures as we used for the validation of the OMG in Chapter 2. In terms of generating molecules using a single core computing, PMG provided all possible structures about 40-fold faster compared to OMG with elemental compositions as only input. The speed increases was even up to 100-fold for PMG compared to OMG when also prescribed substructures were provided as input. The time to generate molecules could be further reduced by executing PMG in multiple cores, which OMG does not allow, represents an almost linear speed up increase as more cores are added.

The efficiency in reducing the number of structures obtained with the additional constrains was tested for the unknown metabolites found in human urine of Chapter 4. An expert annotated manually 30 MSⁿ spectral trees, of unknown metabolites found in human urine, which provided substructures for the good list. For the unknowns for which the elemental composition allowed structures with rings, the use of all additional constraints as introduced in PMG removed up to 82% of the candidate structures. In conclusion, PMG is a further improvement in the efficient generation of candidate structures for a given elemental composition, substructures and using additional constraints. We expect that the open source nature of PMG allows further improvements by other researchers, especially when more knowledge over the type of molecules to be identified is known.

Metabolomics is a growing field that still suffers from some limitations specific for metabolites, and some limitations that are also observed for other 'omics' areas such as proteomics. So far no generic procedure is available that allows the identification and quantification of all metabolites present in a sample compared to sequencing of a full genome, which is possible for currently just a few thousands of euros. One challenge is that databases containing metabolites and experimental data such as MS, MSⁿ, and NMR spectra of metabolites have been for many decades kept in-house of companies or research groups, and the available databases are containing only a fraction of the metabolites being expected. Fortunately, more international consortia are being established to tackle the challenges in metabolomics.

The research described in this thesis has shown that the success of de novo metabolite identification relies on the synergy between analytical chemistry methods (i.e. LC-MSⁿ) and cheminformatics tools. It can be expected that the analytical instrumentation and methods will further develop and faster methods will require less amount of sample and will detect more metabolites, for which masses and fragment ions will be detected with mass spectrometry with more accuracy and better reproducibility. The key factor for the success of MS as a standard technique for metabolite identification is its ability to produce substructure information for many analytes. Important will be also to obtain MSⁿ spectra on-the-flight, i.e. without the need of fractionation prior to direct infusion into the MS.

In this thesis it has been presented that knowing the elemental composition and certain substructures of an unknown metabolite allows to limit the number of possible structures of that unknown. In this thesis the concept of the maximum common substructure (MCSS) from similar MS^n spectra was used to derive a substructure for the unknown metabolites. However, better alternative to determine substructures in an unknown metabolite should be developed as there are cases that not the correct substructure was obtained, or the substructure was too small. A better alternative could be to relate shared branches or subtrees among metabolites with shared substructures, i.e. the building block principle. Or, even better, an MS^n database of known metabolites should have annotated the branches and subtrees of the metabolites with their corresponding substructures. In such an approach, matching a subtree of the unknown metabolite with and annotated subtrees in the database will provide a higher confidence that the unknown contains that substructure(s). And as shown in this thesis the more substructures are used as constraints for the structure generator the fewer candidate structures are obtained. Having an MS^n database with tens of thousands of annotated metabolites would allow the identification of metabolites in a semi high-throughput fashion.

Generally high benefits can be expected from the computerization of human expertise, as was for example demonstrated for chess playing supercomputers and artificial intelligence software. However, for metabolite identification in the current situation also for the in this thesis developed pipeline the input of a human expert is still required. Human expertise is necessary at different steps of the metabolite identification process. Humans are required to evaluate the correctness of the

analytical data acquired from biological samples. It does not matter how good a software pipeline is, if the initial input is of bad quality, the output will be of bad quality. Software helps us at performing repetitive tasks more efficiently. Where software underperforms humans is at detecting anomalies in a pipeline. An expert can use tools like common sense and intuition to detect that results are overly optimistic, on the one hand, or to focus on those candidates that have more chances of succeeding. However, it can be expected that with the further development of the identification pipeline the required input from a human expert will become less and less over time.

Samenvatting

Samenvatting

Metabolomics is één van de 'omics' disciplines die gebruikt wordt om biologische systemen beter te begrijpen. Het is uitermate geschikt om variatie op fenotypisch niveau te begrijpen die niet volledig op genetische basis verklaard kan worden. In metabolomics worden metabolieten bestudeerd. Dit zijn kleine moleculen die betrokken zijn bij allerlei metabolische processen. Terwijl genen gedurende de levensduur van een organisme niet significant veranderen, veranderen metabole processen, en de daarbij betrokken metabolieten, continu. Vandaar dat metabolieten ideale markers zijn voor de staat waarin een organisme zich bevindt.

Een klassiek voorbeeld van het gebruik van metabolieten binnen biomedisch onderzoek is het vinden van metabolieten die verschillend zijn tussen een groep van zieke en gezonde patiënten. De significante aanwezigheid (of afwezigheid) van een bepaald metaboliet (of een groep van metabolieten) in de zieke groep kan gebruikt worden als indicator, ook wel biomarker genoemd, voor die ziekte. Voordat metabolieten gebruikt kunnen worden om de biologie beter te begrijpen, moet eerst de chemie van metabolieten begrepen worden. Met andere woorden, wat zijn de chemische structuren van de metabolieten die indicatoren zijn voor een bepaalde ziekte? Het beantwoorden van deze vraag is het doel van Metaboliet Identificatie, één van de grootste knelpunten binnen metabolomics, en het doel van dit proefschrift.

In dit proefschrift heb ik gewerkt aan het maken van een aaneenschakeling van software tools die het mogelijk maken om chemisch structuren te identificeren van nieuwe metabolieten gevonden in humane monsters waarbij alleen LC-MSn data wordt gebruikt.

Identificatie van nieuwe metabolieten wordt een uitdaging als de structuur van de metabolieten niet terug te vinden is in een database of wanneer deze niet vergeleken kunnen worden met de structuren van chemische standaard verbindingen. Vandaar dat de structuur voorspeld moet worden. In Hoofdstuk 2 beschrijf ik de ontwikkeling van de Open Molecule Generator (OMG). OMG is een open-source structuur generator die voor een onbekend metaboliet, gegeven de elementaire compositie, een volledige lijst met mogelijke chemische structuren produceert. De gebruikte kanonische augmentatie aanpak (origineel binnen de graaftheorie ontwikkeld om grafen te genereren) is daarvoor zodanig aangepast dat alle mogelijke molecuulstructuren zonder duplicaten worden gegenereert. Tevens accepteert de generator als additionele invoer substructuren die de uiteindelijke structuren van de onbekende metabolieten moeten bevatten. OMG genereert miljoenen kandidaat structuren gegeven de elementaire atoomsamenstelling van gangbare humane metabolieten, zelfs wanneer substructuren als additionele invoer worden gebruikt om het aantal mogelijkheden te beperken. Additionele stappen waren nodig om deze lijst met mogelijke structuren te verkleinen zodat uiteindelijk alleen maar een beperkt aantal mogelijke structuren overblijft.

In Hoofdstuk 3 implementeerde ik een model dat voorspelt in hoeverre een molecuul met een bepaalde chemische structuur lijkt op een metaboliet, de Metabolite-likeness. Met andere woorden, het model bepaalt hoe waarschijnlijk een molecuul een humaan metaboliet is. Twee mogelijke redenen zijn aan te geven betreffende de toepassing van deze tool. Ten eerste, om beter de intrinsieke eigenschappen van humane metabolieten te begrijpen en ten tweede om de Metabolite-likeness te gebruiken als een filter om kandidaat structuren geproduceerd door OMG, die niet op humane metabolieten lijken, te verwijderen. Ik heb daarvoor humane metabolieten uit de Human Metabolome Database (HMDB) vergeleken met niet-metaboliet moleculen uit de ZINC database. Ik heb verschillende klassificatie modellen en chemische descriptorren getest en gezien dat uiteindelijk de combinatie van de Random Forest klassificatie methode met MDL Public Keys descriptorren het beste Metabolite-likeness voorspelden.

In Hoofdstuk 4 creëerde ik een metaboliet identificatie pipeline door OMG en Metabolite Likeness te koppelen aan twee, ook op dezelfde afdeling ontwikkelde, tools, de Multistage Elemental Formula (MEF) tool en een algoritme die MSn data met elkaar vergelijkt samen met een energie en een fragmentatie voorspeller filter. Het idee was om een pipeline van tools te hebben die als invoer MSn data van onbekende metabolieten gebruikte om ze vervolgens te identificeren of een kleine lijst met kandidaat structuren te genereren. Deze pipeline werd getest aan de hand van een humaan urine monster waarvoor MSn spectra van 30 onbekende metabolieten was opgenomen. Van deze 30 onbekenden werden er 10 geïdentificeerd doordat hun spectra samenvielen met spectra in een database met

bekende metabolieten. Voor 3 onbekenden werd een relatief korte lijst met minder dan 8 kandidaat structuren verkregen. Voor 6 andere onbekenden werd een lange lijst met kandidaten verkregen. Ten slotte kon voor de overige 11 onbekenden geen lijst met kandidaten gegenereerd worden. Deze resultaten ondersteunden het idee dat metaboliet identificatie mogelijk is gebruikmakende van alleen MSn data in combinatie met slimme cheminformatica tools, maar dat er nog steeds ruimte voor het aanbrengen van verbeteringen is.

Deze verbeteringen werden ingevoerd in Hoofdstuk 5 in de vorm van de Parallel Molecule Generator (PMG). In hoofdstuk 4 leerden we dat het verkrijgen van een korte lijst met kandidaten essentieel is om nieuwe metabolieten te identificeren. Dit is de reden waarom wij in PMG meer chemische randvoorwaarden toevoegden, zoals de verwijdering van moleculen die instabiele ringstructuren en ongewenste substructuren bevatten, om de lijst met kandidaat structuren verder te verkleinen. Omdat de berekening van deze additionele randvoorwaarden extra computertijd kost, werd PMG zodanig geïmplementeerd dat het parallel kon draaien in een multi-core omgeving. Deze verbetering resulteerde in een significante afname van de tijd nodig om de kandidaatlijst te genereren. PMG was gemiddeld 100 maal sneller dan OMG voor onbekende metabolieten waarvoor een lijst met substructuren voorhanden was.

Kortom, dit proefschrift laat zien dat het succes van de-novo metaboliet identificatie sterk bepaald wordt door een goede synergie tussen de analytische chemische methoden en de cheminformatica tools. Tevens werd gepresenteerd dat als de

elementaire compositie en bepaalde substructuren van een onbekende metabooliet bekend zijn, het mogelijk is om de hoeveelheid mogelijke structuren voor de onbekende metabooliet te beperken.

Acknowledgements

Acknowledgements

There are many persons have helped me to finish this thesis, and it is a nice challenge to properly thank all of them. This journey started with Eelke, Andreas, Michael and Prof. Ijzerman. They offered me the most exciting research project a master student could dream of and ignited my love for research and motivated me to go for a PhD.

During this PhD I learned a lot from my (abundant) PhD supervisors. They turned me into a better scientist. I appreciate very much that they stayed patient when facing my temperament. With their optimism and their continuous questions they helped me a lot. They opened my eyes that a PhD student should create something new and that that is going to be tough.

I would like to thank the students I was lucky to supervise, Remco, Bart and Anja. You brought new insights in my research and helped me to be more accountable. I only wish I could have offered you longer projects. My special thanks to the colleagues that contributed with their lab work: Piotr, Richard, Marco and Justin. Without you acquiring MSn trees I would have had nothing to test my tools with. Michael thanks for enlightening me computationally and doing it with a smile and for the updates in the app world. I would also like to thank Loes and Anneke for supporting me in all a PhD student needs. Special thanks to the ABS/NMC crew for all support and discussions during coffee breaks. My gratitude goes all the way to

Evry, France, where Pablo and Davide welcomed me every time I was visiting them. Davide, thanks for teaching me how to cook proper risotto and for offering a temporary home when necessary.

If there is a person that deserves credit for a lot of this thesis is Miquel. He helped me scientifically and personally during four years. Moltes gracies mestre, ho hem aconseguit!! Thanks to my dearest friends Alvaro, Laura and Juan Carlos, no hay dia que no os eche de menos, gambiteros. My gratitude goes also to Jordi and Lisa for keeping my social life alive ... i pels sopars vegetarians per llepar-se'n els dits. Special thanks go to my daily support groups where I can vent my frustrations, La Tertulia and Club M, who needs a psychologist with such great friends?

Long time ago, after my first BSc year I was doubting whether a university study was the right thing for me. Now I am allowed to defend my PhD thesis. This transformation has only been possible with the encouragement and good values my parents provided me: gràcies papa, gracias mama, os quiero.

And above everybody, there is Lin. What a ride these years have been. I would not have achieved this without you. You offered me your patience, sweetness and understanding day in and day out. Esta tesis es para ti, Gatita.

Curriculum vitae

Curriculum vitae

Julio Eduardo Peironcely Miguel was born on 9 December 1982 in Barcelona, Catalonia, Spain. He completed his BSc in Computer Science in 2006 at Universitat Autònoma de Barcelona. In September 2006 he joined the MSc Bioinformatics offered jointly by Leiden and TU Delft universities. During his studies, he developed an interest in research using and developing software for life sciences. He started with the data integration of different microarray experiments, under the supervision of Dr. F. Verbeek. He continued between January and September 2008, with a research project between the Medicinal Chemistry group (supervisors Dr. A. Bender and Dr. E. vd Horst) and the Leiden Institute For Advanced Computer Science (LIACS) (supervisor Dr. M. Emerich). In this project he devised a new way to measure the similarity of GPCR receptors according to their sequence similarity and their ligand similarity. From October 2008 till December 2012 he performed the work presented in this thesis as PhD student in Analytical Biosciences at Leiden University (supervisor Prof. Dr. T. Hankemeier and Dr. T. Reijmers), TNO Quality of Life (supervisors Dr. L. Coulier and Dr. I. Bobbeldijk-Pastorova) and the Institute Of Systems and Synthetic Biology at Evry University (supervisor Prof. Dr. J-L. Faulon). During his PhD, Julio was appointed visiting scientist at the Institute Of Applied Mathematics (IPAM) at the University of California Los Angeles (UCLA). Currently Julio works as a data scientist in industry where he continues to use his computational research skills and ingenuity to solve new types of challenges.

Publications

Publications

1. **Peironcely JE**, Reijmers T, Coulier L, Bender A, Hankemeier T: Understanding and Classifying Metabolite Space and Metabolite-Likeness. PLoS ONE 2011, 6:e28966.
2. **Peironcely JE**, Rojas-Cherto M, Fichera D, Reijmers T, Coulier L, Faulon J-L, Hankemeier T: OMG: open molecule generator. Journal of Cheminformatics 2012, 4:21.
3. **Peironcely JE**, Rojas-Cherto M, Tas A, Vreeken R, Reijmers T, Coulier L, Hankemeier T: An Automated Pipeline For De Novo Metabolite Identification Using Mass Spectrometry-Based Metabolomics. Analytical Chemistry 2013, 7:3576–3583.
4. **Peironcely JE***, Jaghoori MM*, Tas A, Reijmers T, Coulier L, Faulon J-L, Hankemeier T: De Novo Identification of Metabolites With Open Molecule Generator For Metabolomics. In preparation

Not part of this thesis

5. Doddareddy MR, Westen GJP Van, Horst E Van Der, **Peironcely JE**, Ijzerman AP, Emmerich M, Jenkins JL, Bender A: Chemogenomics : Looking at Biology through the Lens of Chemistry. Analysis 2009.
6. Van der Horst E*, **Peironcely JE***, Ijzerman AP, Beukers MW, Lane JR, Van Vlijmen HWT, Emmerich MTM, Okuno Y, Bender A: A novel chemogenomics analysis of G protein-coupled receptors (GPCRs) and their ligands: a potential strategy for receptor de-orphanization. BMC bioinformatics 2010, 11:316.

7. Horst E Van Der, **Peironcely JE**, Westen GJP Van, Hoven OO Van Den, Galloway WRJD, Spring DR, Wegner JK, Vlijmen HWT Van, Ijzerman AP, Overington JP, Bender A: Chemogenomics Approaches for Receptor Deorphanization and Extensions of the Chemogenomics Concept to Phenotypic Space. *Curr. Top. Med. Chem.* 2011, 44.
8. Rojas-Chertó M, **Peironcely JE**, Kasper PT, Van der Hooft JJJ, De Vos RCH, Vreeken R, Hankemeier T, Reijmers T: Metabolite Identification Using Automated Comparison of High-Resolution Multistage Mass Spectral Trees. *Analytical Chemistry* 2012, 84:5524–5534.
9. Rojas-Chertó M, Van Vliet M, **Peironcely JE**, Van Doorn R, Kooyman M, Beek T Te, Van Driel M a, Hankemeier T, Reijmers T: MetiTree: a web application to organize and process high resolution multi-stage mass spectrometry metabolomics data. *Bioinformatics* 2012, 28:2707–2709.
10. Kirchmair J, Howlett A, **Peironcely JE**, Murrell DS, Williamson MJ, Adams SE, Hankemeier T, Van Buren L, Duchateau G, Klaffke W, Glen RC: How Do Metabolites Differ from Their Parent Molecules and How Are They Excreted? *Journal of Chemical Information and Modeling* 2013, 53:354–367.
11. Jaghoori MM, Jongmans S-STQ, De Boer F, **Peironcely JE**, Faulon J-L, Reijmers T, Hankemeier T: PMG: Multi-core Metabolite Identification. *Electronic Notes in Theoretical Computer Science* 2013, 299:53–60.

*Authors contributed equally