



Università degli Studi di Padova

DIPARTIMENTO DI SCIENZE DEL FARMACO
Scuola di Dottorato di Ricerca in Scienze Molecolari
Indirizzo Scienze Farmaceutiche
XXV Ciclo

TESI DI DOTTORATO DI RICERCA

**Chemoinformatics Approaches
For New Drugs
Discovery**

Supervisore:

Ch.mo Prof. Stefano Moro

Direttore della scuola/Coordinatore d' indirizzo:

Ch.mo Prof. Antonino Polimeno

Ch.mo Prof. Alessandro Dolmella

Dottorando:

Marco Fanton

31 Gennaio 2013

Dedicated to ...

Pamela, my strength

my family, permanent reference

Stefano, teacher, guide and friend

...small details make the greatness of works...

Riassunto

Il termine chemoinformatica si riferisce all'uso di metodi informatici per risolvere problemi chimici ed ha come oggetto strutture molecolari e loro rappresentazioni, proprietà e dati collegati; passaggio cruciale è la traduzione di sistemi atomici interconnessi in rappresentazioni e modelli in silico, garantendo il completo e corretto trasferimento dell'informazione chimica. Negli ultimi 20 anni i database chimici sono evoluti da semplici archivi molecolari a strumenti di ricerca per l'identificazione di nuovi candidati farmaci, grazie allo sviluppo di tecnologie di high-throughput che permettono una continua e costante espansione delle librerie chimiche come testimoniato da database pubblici quali PubChem [<http://pubchem.ncbi.nlm.nih.gov/>], ZINC [<http://zinc.docking.org/>], ChemSpider [<http://www.chemspider.com/>]. Requisiti fondamentali per qualsiasi libreria chimica sono l'unicità e disambiguità molecolare, la correttezza chimica (relativa ad atomi, legami, ortografia chimica), la standardizzazione dei formati di archiviazione e registrazione molecolare. Lo scopo di questo lavoro è lo sviluppo di strumenti e masse dati chemoinformatici applicabili al processo di identificazione di nuovi farmaci.

La prima fase del progetto si è focalizzata sull'analisi dello spazio chimico commerciale in termini di ridondanza molecolare e correttezza dei modelli in-silico, allo scopo di identificare un descrittore molecolare univoco e non ambiguo utilizzabile nella indicizzazione di librerie molecolari; questo ha permesso di unificare una libreria di 42 milioni di composti commercialmente disponibili e di implementare *MMsDusty*, un'applicativo web per l'unificazione di librerie chemoinformatiche.

Uno dei prodotti principali del progetto è *MMsINC*[®], una piattaforma chemoinformatica basata su una libreria iniziale di 4 milioni di modelli molecolari di elevata qualità e priva di ridondanza, espansa poi a circa 460 milioni di strutture. La piattaforma permette di effettuare analisi chemoinformatiche tramite funzioni appositamente sviluppate (ricerca per similarità, sottostruttura, descrittori molecolari) oltre ad essere interfacciata col PDB (*Protein Data Bank*) [<http://www.rcsb.org/pdb/home/home.do>] e correlata ai farmaci attualmente in commercio.

La seconda piattaforma sviluppata è *pepMMsMIMIC*, un protocollo di analisi ed identificazione di peptidomimetici basato su screening di librerie chimiche multiconformerone tramite FP (*fingerprints*) farmacoforici, allo scopo di identificare piccole molecole organiche in grado di mimare geometricamente e chimicamente peptidi o proteine endogeni.

Infine è stato sviluppato un protocollo di analisi conformazionale esaustiva di librerie chimiche, fondamentale per la predizione di modelli molecolari tridimensionali di alta qualità, richiesti nelle applicazioni chemoinformatiche; ottimizzando l'esplorazione torsionale all'interno degli intervalli degli angoli diedri più frequenti rilevati nelle strutture organiche risolte ai raggi X del CSD (*Cambridge Structural Database*) su 89 milioni di grafi molecolari, sono stati generati 2.6×10^7 conformeri di alta qualità.

Nel complesso la piattaforma ed i protocolli sviluppati permettono di effettuare analisi chemoinformatiche su librerie molecolari di grosse dimensioni, garantendo elevata qualità, correttezza ed unicità del dato chimico e della sua rappresentazione in silico tramite modelli tridimensionali.

Abstract

Cheminformatics uses computational methods and technologies to solve chemical problems. It works on molecular structures, their representations, properties and related data. The first and most important phase in this field is the translation of interconnected atomic systems into in-silico models, ensuring complete and correct chemical information transfer. In the last 20 years the chemical databases evolved from the state of molecular repositories to research tools for new drugs identification, while the modern high-throughput technologies allow for continuous chemical libraries size increase as highlighted by publicly available repository like PubChem [<http://pubchem.ncbi.nlm.nih.gov/>], ZINC [<http://zinc.docking.org/>], ChemSpider [<http://www.chemspider.com/>]. Chemical libraries fundamental requirements are molecular uniqueness, absence of ambiguity, chemical correctness (related to atoms, bonds, chemical orthography), standardized storage and registration formats.

The aim of this work is the development of cheminformatics tools and data for drug discovery process. The first part of the research project was focused on accessible commercial chemical space analysis; looking for molecular redundancy and in-silico models correctness in order to identify a unique and univocal molecular descriptor for chemical libraries indexing. This allows for the 0%-redundancy achievement on a 42 millions compounds library. The protocol was implemented as *MMsDusty*, a web based tool for molecular databases cleaning. The major protocol developed is *MMsINC*[®], a cheminformatics platform based on a starting number of 4 millions non-redundant high-quality annotated and biomedically relevant chemical structures; the library is now being expanded up to 460 millions compounds. *MMsINC*[®] is able to perform various types of queries, like substructure or similarity search and descriptors filtering. *MMsINC*[®] is interfaced with PDB (*Protein Data Bank*) [<http://www.rcsb.org/pdb/home/home.do>] and related to approved drugs.

The second developed protocol is called *pepMMsMIMIC*, a peptidomimetic screening tool based on multiconformational chemical libraries; the screening process uses pharmacophoric fingerprints similarity to identify small molecules able to geometrically and chemically mimic endogenous peptides or proteins.

The last part of this project lead to the implementation of an optimized and exhaustive conformational space analysis protocol for small molecules libraries; this is crucial for high quality 3D molecular models prediction as requested in cheminformatics applications. The torsional exploration was optimized in the range of most frequent dihedral angles seen in X-ray solved small molecules structures of CSD (*Cambridge Structural Database*); by applying this on a 89 millions structures library was generated a library of 2.6×10^7 high quality conformers.

Tools, protocols and platforms developed in this work allow for cheminformatics analysis and screening on large size chemical libraries achieving high quality, correct and unique chemical data and in-silico models.

Contents

	Page
I Introduction	1
Chapter 1 Chemistry First of All	3
Chapter 2 Chemoinformatics Aim	5
Chapter 3 Chemoinformatics Learning Process	7
3.1 Objects Representation	8
3.2 Data Collecting	9
3.3 Learning Process	9
Chapter 4 Chemoinformatics History	11
4.1 Introduction	11
4.2 Structural Databases	11
4.3 Quantitative Structure-Activity Relationships	12
4.4 Molecular Modeling	12
4.5 Structure Elucidation	12
4.6 In-silico Chemical Reactions	12
II MATERIALS AND METHODS	13
Chapter 5 Chemical Objects Representation	15
5.1 Introduction	15
5.2 Chemical Nomenclature and Notation	16
5.2.1 Systematic Nomenclature	17
5.2.2 Line Notations	17
5.2.2.1 Wiswesser Line Notation	18
5.2.2.2 ROSDAL	19
5.2.2.3 SMILES	20
5.2.2.4 Sybyl Line Notation	22
5.3 Molecular Constitution Coding	25
5.3.1 Molecular Graphs	25
5.3.2 Matrix Representation	27
5.3.2.1 Adjacency Matrix	27
5.3.2.2 Distance-based Matrix	28
5.3.2.3 Atom Connectivity Matrix	29
5.3.2.4 Incidence Matrix	29

5.3.2.5	Bond Matrix	30
5.3.2.6	Bond-Electron Matrix	30
5.3.3	Connection Tables	33
5.3.4	Input and Output of Chemical Structures	35
5.3.5	Structure Exchange Formats	35
5.3.6	Molfiles and SDfiles	36
5.3.6.1	Molfiles	36
5.3.6.2	SDfiles	37
5.3.7	Unambiguous and Unique Representations	38
5.3.7.1	Structure Isomers and Isomorphism	38
5.3.7.2	Canonicalization process	40
5.3.7.3	The Morgan Algorithm	41
5.3.7.3.1	Morgan Algorithm: an example	41
5.3.7.3.1.1	Step 1: Classification of atoms by considering their neighborhood (relaxation process)	41
5.3.7.3.1.2	Step 2: Assigning unique, invari- ant atoms numbering	43
5.3.8	Special Notations	44
5.3.8.1	Fragment Coding	44
5.3.8.2	Fingreprints	45
5.3.8.2.1	Hashed-Fingerprints	45
5.3.8.3	Hash Codes	46
5.3.9	Stereochemistry representation	46
5.3.9.1	Detection and Specification of Chirality	48
5.3.9.1.1	Ordered Lists	48
5.3.9.1.2	Rotational Lists	49
5.3.9.1.3	Permutation Descriptors	49
5.3.9.2	Stereochemistry in Molfile and SMILES	50
5.3.9.2.1	Stereochemistry in Molfiles	50
5.3.9.2.2	Stereochemistry in SMILES	52

III Results and Discussion 53

Chapter 6	Redundancy Management in Chemoinformatics Libraries	55
6.1	Introduction	55
6.2	InChI	56
6.2.1	InChI Generation	56
6.2.1.1	Normalization	57
6.2.1.1.1	Additional Normalization Rules	58
6.2.1.1.1.1	Step 1. Disconnect salts	58
6.2.1.1.1.2	Step 2. Disconnect metals	58
6.2.1.1.1.3	Step 3. Eliminate radicals	58
6.2.1.1.1.4	Step 4. Variable protonation pro- cessing (charges and mobile H)	59
6.2.1.1.1.5	Step 5. Charges and mobile H Processing	59
6.2.1.2	Canonicalization	61
6.2.1.3	Serialization	62

6.2.2	InChI Layers Type	62
6.3	SMILES and InChI used implementations	68
6.3.1	SMILES Implementation	68
6.3.2	InChI Implementation	68
6.4	Datasets	69
6.5	InChI and SMILES Strings Reproducibility Assessment	69
6.6	Redundancy Identification Efficiency	71
6.6.1	Single Catalogue Redundancy Analysis	71
6.6.2	Pan-catalogues Redundancy Analysis	77
6.6.3	SMILES Errors Identification	78
6.7	MMsDusty Pipeline	90
6.8	Conclusion	90
Chapter 7	MMsINC [®] : a large-scale chemoinformatics platform	91
7.1	Introduction	91
7.2	Database Creation	92
7.2.1	First Redundancy Washing	93
7.2.2	Step 2: Tautomers Generation	93
7.2.3	Step 3: Ionic States Generation	93
7.2.4	Step 4: Conformer Selection	93
7.2.5	Step 5: Second Redundancy Washing	93
7.2.6	Step 6: Unstable Tautomer Elimination	94
7.2.7	Step 7: Molecular Descriptors calculation	94
7.3	Database Validation: Subsets Analysis	95
7.4	Informatic Structure	96
7.5	Database Querying	96
7.5.1	Identical Structure Search	96
7.5.2	Substructure Search	96
7.5.3	Molecular Scissoring Search	97
7.5.4	Similarity Search	97
7.5.5	PDB-Similarity Search	97
7.5.6	Descriptors Filtering	98
7.6	Results Displaying	98
7.7	Implementation	102
7.8	Conclusion	102
Chapter 8	pepMMsMIMIC: a peptidomimetics screening platform	103
8.1	Introduction	103
8.2	The pepMMsMIMIC Protocol	103
8.2.1	pepMMsMIMIC workflow	106
8.2.1.1	Conformers Generation: MultiConf-MMsINC [®]	106
8.2.1.2	Pharmacophoric Fingerprint Generation	106
8.2.1.2.1	Fingerprint Coding	108
8.2.1.2.1.1	First Criterion	108
8.2.1.2.1.2	Second Criterion	109
8.2.1.3	USR-based Molecular Shape Recognition	111
8.2.1.4	Scoring Metrics	113
8.2.1.5	Querying pepMMsMIMIC	113
8.2.1.6	Results Displaying	116
8.2.1.7	Implementation	116

8.2.1.8	Preliminary Validation	116
8.2.1.9	Conclusion	117
Chapter 9	Exhaustive Conformational Analysis	119
9.1	Introduction	119
9.2	Conformers Population Generation	120
9.2.1	Exhaustive Systematic Search	121
9.2.2	Model-based search	123
9.2.3	Stochastic search	123
9.3	Localized and Exhaustive Conformational Space Analysis	124
9.3.1	Molecular Ring Systems	124
9.3.2	Acyclic Flexible Chains	125
9.3.3	Conformers Generation and Geometric optimization . . .	125
9.3.3.1	Selection of Initial Torsional Angle	126
9.3.3.2	Local Minimum Search	127
9.3.3.3	Classification of Conformers Population	127
9.3.4	Protocol Validation	128
9.4	Conclusion	128

List of Figures

	Page
2.1 Bioinformatics and Chemoinformatics cooperation	6
3.1 From Data to Knowledge	8
5.1 Structural information hierarchy	16
5.2 Phenylalanine different line notations	18
5.3 WLN coding rules examples	19
5.4 SMILES coding rules	21
5.5 Generation of SMILES	22
5.6 SLN coding rules	23
5.7 Graph-theory representations	25
5.8 Phenylalanine weighted and labeled graph	25
5.9 Graph theory basics	26
5.10 Graph theory basics(continue)	27
5.11 Adjacency Matrix	28
5.12 Redundant Adjacency Matrix	28
5.13 Distance Matrix	29
5.14 Incidence Matrix	30
5.15 Bond Matrix	30
5.16 Bond-Electron Matrix	31
5.17 Valence Electrons number from bond-electron matrix	32
5.18 Connection Table	33
5.19 Redundant Connection Table	33
5.20 Non-Redundant Connection Table	34
5.21 Standard Exchange File Formats	35
5.22 L-Alanine Molfile	37
5.23 SDfile	38
5.24 Functional Isomerism	39
5.25 Structural Isomorphism problem	39
5.26 Canonicalization process	40
5.27 Morgan Algorithm	41
5.28 Extended Connectivity	42
5.29 Equivalent Classes Determination	42
5.30 Morgan Algorithm-based Canonicalization	43
5.31 Molecular Fragments Coding	44
5.32 Fingerprint Bitstring Constitution	45
5.33 Hashed-Fingerprint Constitution	46
5.34 Stereochemistry Representation	47

5.35	Ordered Lists	49
5.36	Permutation Descriptors Determination	49
5.37	Operations on Permutation Descriptors	50
5.38	Molfile Stereochemical Flags	51
5.39	Parity Values Determination	52
5.40	SMILES Stereochemistry Definition	52
6.1	Normalization Process	57
6.2	Radical Elimination Process	58
6.3	Aromatic Bonds Conversion	59
6.4	H-Transfer Tautomerism Possibilities	60
6.5	Guanine Possible Tautomers	60
6.6	Guanine Normalization and Canonical Numbering	60
6.7	Mobile Positive Charge Detection	61
6.8	Canonicalization Process	62
6.9	InChI Layer Generation Flowchart	63
6.10	InChI Layers	64
6.11	S-Glutamic Acid Non-Standard InChI	65
6.12	Python CTs Reshuffling Code	70
6.13	InChI vs SMILES Redundancy Identification Rate	72
6.14	Redundancy Identification Statistics-1	75
6.15	Redundancy Identification Statistics-2	76
6.16	SMILES Errors in Redundancy Detection	80
6.17	InChI/BabelCanSmiles/CACTVSusmiles strings comparison-1	81
6.18	InChI/BabelCanSmiles/CACTVSusmiles strings comparison-2	82
6.19	SMILES String failure-1	84
6.20	SMILES String failure-2	85
6.21	SMILES String failure-3	86
6.22	SMILES String failure-4	87
6.23	SMILES String failure-5	88
6.24	SMILES String failure-6	89
7.1	Public Molecular Databases Comparison	92
7.2	MMsINC [®] Molecular Descriptors	94
7.3	MMsINC [®] Substructure Search	99
7.4	MMsINC [®] Similarity Search	100
7.5	MMsINC [®] Search Report	101
8.1	pepMMsMIMIC Architecture	105
8.2	The pepMMsMIMIC WorkFlow	106
8.3	Key Residues Coding	107
8.4	AminoAcids Side Chains Features	108
8.5	Centroids-pair Distance Classification	110
8.6	USR Coding	111
8.7	USR-based Shape Similarity Score	112
8.8	pepMMsMIMIC Web Page	114
8.9	pepMMsMIMIC User Interface	115
8.10	pepMMsMIMIC Validation	117
9.1	Potential Energy Profile	121

9.2	Conformational Space Representation	122
9.3	Piperonylbutoxide Conformational Space Exploration	123
9.4	Atom Clashes	125
9.5	Experimental Distribution of Torsional Angle Values	126
9.6	Energetic Symbolic Function	127
9.7	Energetic Minimization of Torsion Angle	127
9.8	Torsion Angle Library for 2-L-Benzylsuccinate	128
9.9	Superimposition of X-ray solved structure and predicted conformers	128

List of Tables

	Page
5.1 Molecular nomenclature systems comparison	17
5.2 Line notation systems comparison	24
5.3 Matrix representation comparison	32
5.4 Connection Tables pro and con.	34
6.1 CACTVS Hash Strings	68
6.2 InChI vs SMILES: Redundancy Identification Efficiency	78
7.1 MMsINC [®] Chemical Subsets Representation	95

Nomenclature

CDK	Chemistry Development Kit
CPU	Central Processing Unit
CSD	Cambridge Structural Database
CT	Connection Table
EC	Extended Connectivity
FP	Fingerprint
GPU	Graphic Processing Unit
PDB	Protein Data Bank
ROSDAL	Representation of Organic Structures Description Arranged Linearly
SLN	Sybyl Line Notation
SMILES	Simplified Molecular Input Line Entry System
USR	Ultrafast Shape Recognition
WLN	Wiswesser Line Notation

Part I
Introduction

Chapter 1

Chemistry First of All

Chemoinformatics is simply a new name for an old discipline. It is based on the application of informatics methods and technologies to solve chemical problems. Then, even if using computational methods, chemical rules and knowledge come before any algorithm, software or CPU *Central processing Unit*. The amount of information that has to be processed is often quite large, so chemoinformatics deals with complex and extended chemical spaces instead of single molecules as in other *Molecular Modeling* applications.

"*Chemical space is the space spanned by all possible (i.e. energetically stable) stoichiometric combinations of electrons, atomic nuclei and topologies (isomers) in molecules*". It has been estimated that the number of chemical entities made by 10 atom types (C, H, N, O, P, S, F, Cl, Br, I) and with molecular weight less than 500 Da, ranges from 10×10^{40} to 10×10^{120} structures. Actually more than 40 millions different compounds are known and this immense amount of information can be processed only by electronic means, using computational methods: here chemoinformatics comes in.

The primary requirement for any chemoinformatics application is the high-quality of data, in order to ensure complete and accurate chemical properties and chemical description transfer to in-silico models; poor quality data lead to bad chemoinformatics protocol performances, negatively affecting the drug discovery process.

"Many scientists TRUST chemistry and biology databases that are so often reused, reanalyzed and integrated...The authors of such articles do not appear to analyze for problems caused by poor DATA QUALITY or hypotheses that are incorrect due to poor underlying data." (Cit. Antony Williams, ChemSpider).

Regardless the continuous increase in computational methodologies capabilities, databases size, CPU, GPU and data transfer rate performances, the weak point of chemoinformatics applications is the poor chemical quality of used data; this is primarily due to a lack of consciousness of this problem and consequently to a lack of suitable tools to fix it. In this research project the identification and improvement of poor chemical quality data comes before any chemoinformatics protocol implementation, application or study.

Chapter 2

Chemoinformatics Aim

It was clear since some decades ago that the large amount of chemical information accumulated by chemists can be made accessible to the scientific community only in electronic form by storing it in databases. This new discipline based on storage, manipulation and processing of chemical information was missing a proper name. In most cases it was called *Chemical Information* or *Computer Chemistry* but the term *Chemoinformatics* appeared only recently. Here are some of the first citings:

"The use of information technology and management has become a critical part of the drug discovery process. Chemoinformatics is the mixing of those information resources to transform data into information and information into knowledge for the intended purpose of making better decisions faster in the area of drug lead identification and organization". (Cit. K. Brown, Annual Reports in Medicinal Chemistry 1998, 33, 375±384)
"Chemoinformatics: A new name for an old problem." (Cit. M. Hamm, R. Green, Current Opinion in Chemical Biology 1999, 3, 379±383)

"Chem(o)informatics is a generic term that encompasses the design, creation, organization, management, retrieval, analysis, dissemination, visualization, and use of chemical information". (Cit. G. Paris, August 1999 Meeting of the American Chemical Society).

For our purpose the best definition of Chemoinformatics is: *The application of informatics methods to solve chemical problems*. There is no clear distinction between bioinformatics and chemoinformatics even if by tradition the first one deals with small molecules while the second one deals with protein and genes. Proteins structure and function, ligands binding to their receptors, substrates conversion to products by enzymes, these are all areas where chemo and bioinformatics work together to improve chemical knowledge, especially in drug design process. Genomics methods allow for protein targets identification for new drug candidates development while chemoinformatics methods allow for new lead structures identification and optimization into drugs.

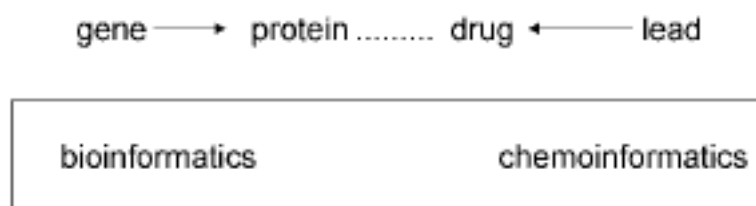


Figure 2.1: Bioinformatics and Chemoinformatics cooperation.[§]

This work is focused on small and medium-sized molecules.
However the protocols here presented works on macromolecules too.

Chapter 3

Chemoinformatics Learning Process

Contents

3.1	Objects Representation	8
3.2	Data Collecting	9
3.3	Learning Process	9

How informatics can help drug discovery and chemistry in general? Chemistry works on making compounds with desired properties, so the first task is to predict which structure has the desired properties. To make predictions it is necessary to pass through a process of learning: deductive or inductive. Deductive learning requires a theory that allows to make hypothesis and to calculate the property of interest.

In chemistry this theory is called quantum mechanics; chemical properties are related to 3D structure by the Schrödinger equation. Great progress in theory development and in hardware and software technology advances, allows the calculation of many interesting chemical properties of reasonable size with high accuracy. However in some areas there are limitations due to lack of development of theory or to excessive computational time required.

In the inductive learning process, starting from observations, inferences are made to predict new observations. Observations need a scheme that allows for ordering them and recognizing common or different features; based on observations a model is built to make predictions by analogy. Inductive learning is the oldest way of acquiring chemical knowledge: chemists do experiments, make properties measurement, run reactions to build models that allows for predictions.

An example is the bromination of monosubstituted benzene derivatives. Substituents with atoms carrying free electron pairs bonded directly to the benzene ring (OH, NH₂, etc) lead to o- and p-substituted derivatives while double bonds substituents (NO₂, CHO, etc.) provide m- substituted derivatives.

In order to improve chemical knowledge, many experiments has been performed producing an enormous data amount; here comes deductive learning to derive

knowledge from data.

Data, information, and knowledge are generally defined as:

- *Data*: any observation provides data as result of a physical measurement, a yes/no answer to whether a reaction occurs or not, or the determination of a biological activity.
- *Information*: if data are put into context with other data, we obtain information. Biological activities measurement make more sense if are known the molecular structures of the studied compounds.
- *Knowledge*: knowledge needs some level of abstraction. Pieces of information are ordered in a model; rules are derived from observations; predictions can be made by analogy.

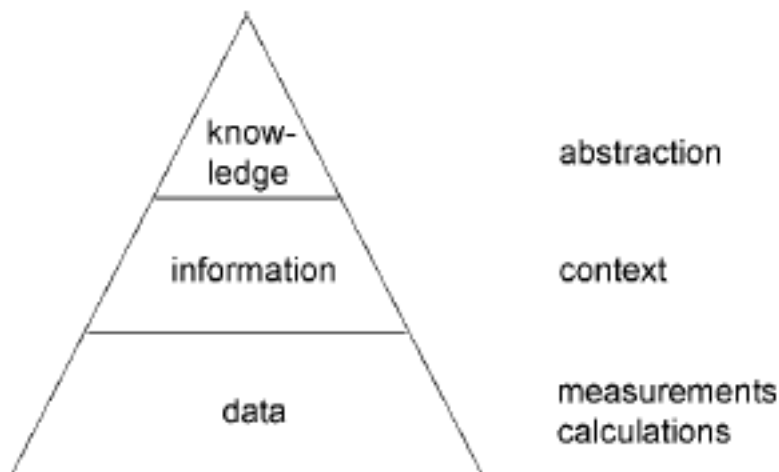


Figure 3.1: From Data to Knowledge through information.[§]

In Chemoinformatics the abstraction is performed to gain knowledge about compounds properties. Physical, chemical or biological data are associated with each other or with compounds structures; then inductive learning methods allow for obtaining a model to make predictions. The knowledge acquisition requires three major tasks: *Objects representation*, *Data collecting*, *Learning process*.

3.1 Objects Representation

First of all chemical compounds have to be represented and this is principally do by their molecular structure in various forms of sophistication or abstraction. The large amount of known compounds is manageable only by databases storing. This is explained in *MATERIAL AND METHODS* section.

3.2 Data Collecting

Chemistry is focused on production of compounds with a variety of physical, chemical or biological properties. Properties measurements is required to order them in a quantitative manner for further optimization. The enormous amounts of available data need databases for storage and management. Chemical Databases description is provided in *MATERIAL AND METHODS* section.

3.3 Learning Process

Inductive and deductive learning ways have already been mentioned in Chapter 3. Some chemical compounds properties can be calculated explicitly by quantum mechanical methods but molecular mechanics methods can often achieve quite high accuracy in properties calculation too. Learning process applications are not the scope of this work but they have been mentioned to give a complete scenario of Chemoinformatics field.

Chapter 4

Chemoinformatics History

Contents

4.1	Introduction	11
4.2	Structural Databases	11
4.3	Quantitative Structure-Activity Relationships	12
4.4	Molecular Modeling	12
4.5	Structure Elucidation	12
4.6	In-silico Chemical Reactions	12

4.1 Introduction

Chemoinformatics evolved in the past four decades thanks to chemists that developed computer methods to manage the chemical information and structures. The major development and improvement start since 1980 due to increment in calculator performances.

4.2 Structural Databases

Chemical compounds and data storage or searching are probably the earliest expression of chemoinformatics. The work done at National Bureau of Standards, Washington DC, 1957, showed for first time that chemical structures can be searched by user-defined substructures through atom-by-atom searching. In 1960 the National Science Foundation funded the Chemical Abstracts Service to develop storage and searching methods in databases. In the same years Swiss and German chemical companies such as BASF, Hoechst and Thomae developed storing methods for their in-house chemical libraries. ICI (UK) built a database of several hundred thousand structures based on *WLN (Wiswesser Line Notation)*[1, 2, 3]. Works made at the National Institutes of Health introduced fragment-based screening to enhance the speed of substructure searching.

4.3 Quantitative Structure-Activity Relationships

Hammett and Taft in the 1950s worked on separation and quantification of steric and electronic influences on chemical reactivity. Starting from this in 1964 Hansch began to quantify steric, electrostatic and hydrophobic effects and their influences on chemical properties and biological activity of drugs too. Free-Wilson analysis was introduced to relate biological activity to presence/absence of specific substructures in a molecule.

4.4 Molecular Modeling

In the late 1960s Marshall, at Washington University St. Louis, MO, USA, developed methods for protein structures visualization on graphic screens.

4.5 Structure Elucidation

In 1964 at Stanford started the DENDRAL project, a prototypical application of artificial intelligence techniques to chemical problems. Chemical structure generators were developed and mass spectra information was used to prune chemical graphs in order to derive the chemical structure associated with a certain mass spectrum.

4.6 In-silico Chemical Reactions

In 1967, a Sheffield group presented a work on indexing chemical reactions for database building, while in 1969, a Harvard group developed a computer-assisted synthesis design.

Part II

**MATERIALS AND
METHODS**

Chapter 5

Chemical Objects Representation

Contents

5.1	Introduction	15
5.2	Chemical Nomenclature and Notation	16
5.2.1	Systematic Nomenclature	17
5.2.2	Line Notations	17
5.3	Molecular Constitution Coding	25
5.3.1	Molecular Graphs	25
5.3.2	Matrix Representation	27
5.3.3	Connection Tables	33
5.3.4	Input and Output of Chemical Structures	35
5.3.5	Structure Exchange Formats	35
5.3.6	Molfiles and SDfiles	36
5.3.7	Unambiguous and Unique Representations	38
5.3.8	Special Notations	44
5.3.9	Stereochemistry representation	46

5.1 Introduction

Chemistry relies heavily on experimental observations and data. The enormous increase in the number of compounds and related data led in the past decades to inefficient data-handling; the only way to fix this was by electronic means using chemoinformatics. While in other scientific disciplines are only used text and numbers for data storage and transfer, chemistry need to use molecules. At the beginnings compounds have been characterized by giving them names and quickly names were substituted by symbols to compact long names. Not last the improvement in molecular structures determination led to the necessity of assigning graphs to compounds. The 2D representation is the universal way to describe molecules both in chemistry or chemoinformatics. In this geometric model, each atom is identified by its atomic symbol and bonds by lines. Such

a kind of model is incomplete and simplified because it only describes molecular topology and not topography (3D structure); tridimensional representation of molecules requires additional information about spatial positioning of atoms, dihedral angles and distances between atoms. Finally if chemical properties (e.g. electrostatic potential) have to be mapped onto a 3D molecular representation surface, more complex information are needed. This hierarchical representation is illustrated in figure 5.1.

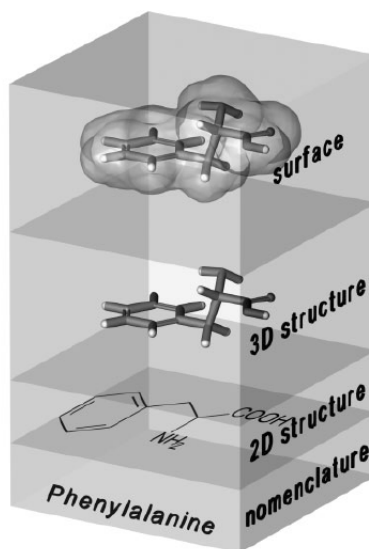


Figure 5.1: Hierarchical molecular representation with different level of structural information.[§]

The first step in chemoinformatics is to translate molecules into bits, so into a computer language, the so called machine code. Computers can only handle 0 and 1 bit values, so coding process is requested in order to transfer data. Molecules have to be represented by machine-readable code; different approaches allow for this: nomenclature, chemical notation, mathematical notation. The major advantage of chemical nomenclature or notation systems is the capability to translate molecular structures between different codes. Since different codes may contain different information, unambiguous and unique coding is not always ensured. In an unambiguous coding an exact chemical structure corresponds to one and only one notation or nomenclature. In many coding languages the same structure could correspond to more than one notation, arising the problem of strings collision, which negatively affects chemical libraries and databases quality. In fact a fundamental chemoinformatics requirement is the uniqueness: coding must result in only one *unique* structure or nomenclature (notation) depending on transformation direction.

5.2 Chemical Nomenclature and Notation

Nomenclature is a technical language, a set of terms and rules for a specific field of knowledge. In 1870, D.I. Mendeleev and L. Meyer, compiled elements

in the periodic table. Nowadays Chemical elements are defined by their own symbols plus additional information as charge or neutrons number.

5.2.1 Systematic Nomenclature

The systematic IUPAC nomenclature characterizes compounds by a unique name (which can be quite long and complicated) in order to systematically describe structure fragments; this notation is used for molecular databases indexing as in the Chemical Abstracts Service. IUPAC nomenclature does not allow for cheminformatics direct extraction of additional information such as bond orders or molecular weight. Organic compounds nomenclature is based on two principal features: the longest continuous aliphatic chain of carbon atoms and branching/rings presence. Functional groups specification allow for chemical family definition. Complex structures requires complex rules to define a unique name which length increases with chemical complexity. For example the structure with trivial name *phenylalanine* corresponds to IUPAC definition: *2-amino-3-phenylpropanoic acid*. Trivial name and systematic nomenclature are both alphanumerical strings but useless for computer applications; in fact a single chemical structure could be described by more than one valid string leading to unambiguous but not unique structure/name correlation.

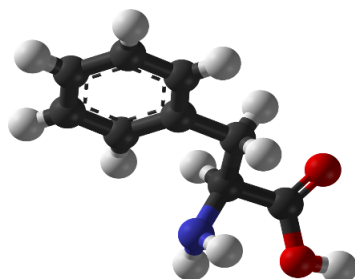
Advantages	Disadvantages
<i>Trivial names</i>	
short, concise, easy to memorize	many available
widespread	no clear systematics
unambiguous	no evidence of stereochemistry
<i>IUPAC nomenclature</i>	
standardized systematic classification	extensive nomenclature rules
include stereochemistry	alternative names are allowed
widespread	complicated names
unambiguous	
allow reconstruction	

Table 5.1: Molecular nomenclature systems comparison

5.2.2 Line Notations

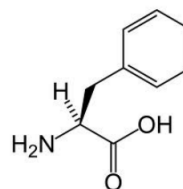
Line notations allow for molecules representation by alphanumeric strings. IUPAC nomenclature is a kind of line notation but it doesn't provide structural information directly from the name. Line notation coding idea come first the computer era but compactness of such coding made it suitable for computer handling. Development of line notation coding start between 1960 and 1970. In the next sections the four most popular line notations are explained: *Wiswesser(WLN)*, *ROSDAL*, *SMILES*, *Sybyl (SLN)*. WLN is the oldest one while SMILES is quite an important representation. *InChI (International Chemical Identifier)* [<http://www.iupac.org/inchi/>][4] is the most recently devel-

oped coding by IUPAC; as it has been widely used in this work, it will be discussed later in chapter 6.



L-Phenylalanine

Systematic name:	L-Phenylalanine
IUPAC name:	(2S)-2-amino-3-phenylpropanoic acid
Empirical formula:	C ₉ H ₁₁ NO ₂
Condensed formula:	C ₆ H ₅ CH ₂ CH(NH ₂)CO ₂ H
WLN(Wiswesser):	VQY1ZR
ROSDAL:	1O-2=3O,2-4-5N,4-6-7=-12-7
SMILES:	NC(Cc1ccccc1)C(O)=O C1=CC=C(C=C1)CC(C(=O)O)N C1=CC=C(C=C1)C[C@@H](C(=O)O)N



InChI: InChI=1S/C9H11NO2/c10-8(9(11)12)6-7-4-2-1-3-5-7/h1-5,8H,6,10H2,(H,11,12)/t8-/m0/s1

Figure 5.2: Phenylalanine different line notations.

5.2.2.1 Wiswesser Line Notation

Wiswesser Line Notation was developed by William J. Wiswesser and introduced in 1946. WLN uses the standard elements symbols while functional groups, ring systems, positions of ring substituents, and positions of condensed rings are described by letters or symbols combination. There are 40 used symbols in WLN from this 3 sets:

- capital letters: A-Z for elements, atom groups, branches, and ring positions;
- numbers: 0-9 for alkyl chains length or rings number;
- special characters: " ", "/", "-", and " " (blank) for rings/substitution positions.

WLN is very compact but unambiguity is achieved only by a complex set of rules. Much work has been done to develop softwares for WLN conversion into a connection table and vice versa, but it was never completely solved. As symbols sequential arrangement is not unique only one unambiguous sequence per molecule is allowed in WLN.

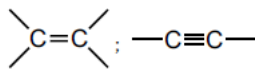
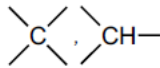
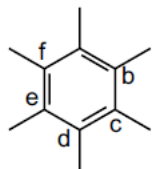

Class	Structural unit	WLN coding
Hydrogen	H	H
Alkanes	C_nH_{2n+2}	n (e.g., CH_3CH_2 ; CH_2CH_2)
Alkenes; alkynes		U; UU
Branched chains		X, Y
Aromatic rings		R
Substituted derivatives		R B, C, D, E, F
(Hetero)cyclic hydrocarbons		L.n.J; T.n.J L: beginning of a carbocyclic ring; T: beginning of a heterocyclic ring; n: number of atoms of the ring system; J: termination of the ring system
Alkyl halides	-X (X = F, Cl, Br, I)	F, G, E, I
Alcohols; ethers	-OH; -O-	Q; O
Ketones; aldehydes	-CO-; -CO-H	V; VH
Carboxylic acids; esters	-COOH; -CO-O-	VQ; VO

Figure 5.3: WLN coding rules examples.[§]

WLN allowed for indexing of Chemical Structure Index (CSI) at the Institute for Scientific Information (ISI) and of Crossbow System of Imperial Chemical Industries (ICI). With the introduction of connection tables in 1965 and molecular editors in the 1970s, WLN lost its importance.

5.2.2.2 ROSDAL

ROSDAL (*Representation of Organic Structures Description Arranged Linearly*) was developed by S. Welford, J. Barnard, and M.F. Lynch in 1985 for the Beilstein Institute in order to transmit structural information between users and the Beilstein DIALOG system during database data retrieval by queries. ROSDAL ASCII string syntax simply codes a chemical structure using alphanumeric symbols[5]. Atoms are arbitrarily assigned unique numbers, except for hydrogens. Carbon atoms are identified only by digits while other atom types are identified by their atomic symbol too. Bond symbols are inserted between atom numbers; branches are enclosed in commas. ROSDAL coding is unambiguous but not unique. ROSDAL notation is build by these steps:

1. Structure diagram drawing and atoms random unique numbering
2. Atomic symbols writing after atom index
3. Carbon atoms not indexed by numbers

4. Bond types describing as: single, double, triple, any ("-", "=", "#", "?")
5. Commas separated branches and substituents indexing

5.2.2.3 SMILES

SMILES (*Simplified Molecular Input Line Entry System*) was introduced in 1986 by David Weininger at the US Environmental Research Laboratory, USEPA, Duluth, MN[6, 7], [<http://www.daylight.com>]. This coding is highly compressed and still widespread used as a universal molecular nomenclature for structures representation and exchange of structural information. Over WLN and ROSDAL, SMILES is based on a smaller and simpler set of coding rules:

1. Atomic symbols-based atoms representation
2. Automatic and implicit hydrogen atoms saturation of free valences (H not represented)
3. Neighboring atoms sequential representation
4. Double and triple bonds coding by "=" and "#" respectively
5. Branches representation by parentheses
6. Rings description digits on to the two connecting ring atoms

SMILES has been improved since 1988 and the present set of coding rules has been implemented by DAYLIGHT [<http://www.daylight.com/dayhtml/smiles/index.html>]. Enhancements of SMILES are XSMILES, SMARTS, SMIRKS, STRAPS, CHUCKLES, CHORTLES, CHARTS, USMILES). SMIRKS language allow for chemical reaction coding; SMARTS is used for molecular patterns (substructures) definition suitable for substructural searching in databases. USMILES (UniqueSMILES) is a special SMILES coding developed by Daylight; it is independent of the internal atomic numbering and results always in the same canonical, unambiguous, and unique description of the compound, granted by Morgan's algorithm, as explained later. Due to the compactness of textual coding, graphical input is not required and computational performances are higher than with previous described notations.

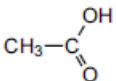
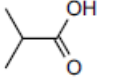


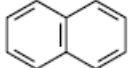
SMILES code	Chemical structure	Compound name
<i>Atoms:</i> Atoms are represented by their atomic symbols. Ambiguous two-letter symbols (e.g., Nb is not NB) have to be written in square brackets. Otherwise, no further letters are used. Free valences are saturated with hydrogen atoms.		
C	CH ₄	methane
[Fe+ 2] or [Fe+ +]	Fe ²⁺	iron (II) cation
<i>Bonds:</i> Single, double, triple, and aromatic (or conjugated) bonds are indicated by the symbols " . ", " = ", " # " and " : ", respectively; single and aromatic bonds should be omitted.		
C=C	H ₂ C=CH ₂	ethene
O=CO	HCOOH	formic acid
<i>Disconnected structures in the molecule:</i> Individual parts of the compound are separated by a period. The period indicates that there is no connection between atoms or parts of a molecule. The arrangement of the parts is arbitrary.		
[Na+].[OH-]	NaOH	sodium hydroxide
<i>Branches:</i> Branches are indicated within parentheses.		
CC(=O)O		acetic acid
CC(C)C(=O)O		isobutyric acid
<i>Cyclic structures:</i> Rings are described by breaking the ring between two atoms and then labeling the two atoms with the same number.		
C1CCCCC1		cyclohexane
<i>Aromaticity:</i> Aromatic structures are indicated by writing all the atoms involved in lower-case letters.		
o1cccl		furan
c12c(ccc1)ccc2 same as c1cc2ccccc2cc1		naphthalene

Figure 5.4: SMILES coding rules.[§]

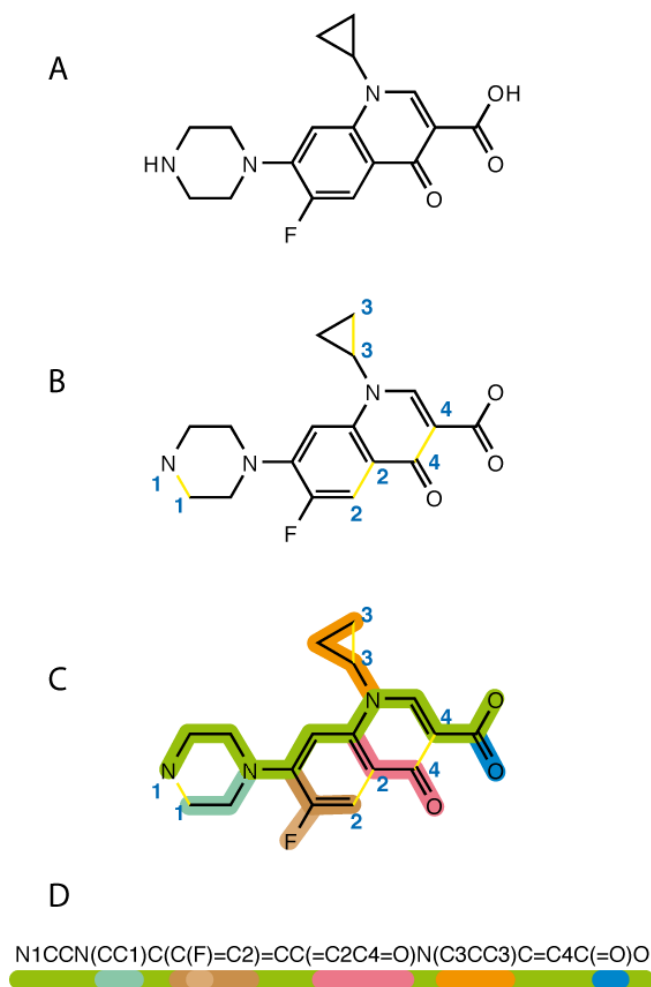


Figure 5.5: Generation of SMILES: break cycles, then write as branches off a main backbone.

5.2.2.4 Sybyl Line Notation

SLN (*Sybyl Line Notation*) was developed by Tripos Inc[8]. It is a modification of SMILES with two principal differences: first explicit hydrogen atoms are required and second it allows for representation of fragments, substructure and combinatorial libraries. These features make it suitable for database storage. SLN uses six basic rules four of which are similar to the SMILES notation. SLN in addition can deal with macro-atoms which are specification of groups of atoms such as aminoacids.

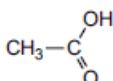


SLN	Chemical structure	Compound name
<i>Atoms:</i> Atoms are represented by their atomic symbols. The first letter is upper-case, and in two-letter symbols the second letter is lower-case. Hydrogen atoms must be specified.		
CH4	CH ₄	methane
NH2	-NH ₂	amine
<i>Bonds:</i> Single bonds are omitted; double, triple, and aromatic bonds are indicated by the symbols "=", "#", and ":", respectively. In contrast to SMILES, aromaticity is not an atomic property, but a property of bonds. A period indicates the start of a new part of the structure.		
HC(-O)OH	HCOOH	formic acid
Na.OH	NaOH	sodium hydroxide
<i>Branches:</i> Branches are indicated by parentheses.		
CH3C(-O)OH		acetic acid
<i>Cyclic structures:</i> Ring closures are described by a bond to a previously defined atom which is specified by a unique ID number. The ID is a positive integer placed in square brackets behind the atom. An "@" indicates a ring closure.		
C[15]H2CH2CH2CH2CH2@15		cyclohexane
O[6]:CH:CH:CH:CH:@6		furan

Figure 5.6: SLN coding rules.[§]

Actually the most used notation is SMILES even if it is affected by some limitations in structures coding as explained hereinafter. In table 5.2 are summarized advantages and disadvantages for described coding systems.

Advantages	Disadvantages
Wiswesser Line Notation	
concise linear code unambiguous	large number of complex rules coding prone to errors
includes stereochemistry unique if rules are followed	only those substructures contained in the coding can be retrieved in a substructure search no support for coding reactions
ROSDAL	
simple code, easy to learn fast data exchange format includes stereochemistry unambiguous	no support for coding reactions not unique
SMILES	
simplest linear code easy to learn fast data exchange format supports Markush, stereochemistry and reaction coding unambiguous	not unique (except Unique SMILES) some problems with aromaticity perception
Sybyl Line Notation	
simple code, easy to learn Markush and macro atom definitions includes stereochemistry fast data exchange format unambiguous	not unique aromaticity has to be normalized no valence rules no support for coding reactions

Table 5.2: Line notation systems comparison

5.3 Molecular Constitution Coding

Chemical structures could be drawn on paper or by graphic softwares on a computer. Even if this pictures contains much chemical information for chemists, they are not directly processable by calculators: it is requested their conversion into another representation format. Nomenclature and line notations formats were discussed in sections 5.2.1 and 5.2.2.

5.3.1 Molecular Graphs

A structure diagram and a topological graph are analogue representations so graph theory[9, 10, 11] is applicable to molecular constitution coding. Mathematically speaking chemical structure diagrams are graphs consisting of nodes (vertices), which represent atoms, and edges, which are the bonds. Graphs are often simplified by representing carbons only as vertices where bonds meet. These are topological graphs because only linkages between atoms are shown while topographical (3D) information is not provided. A chemical structure diagram has no bonds-direction and nodes are labeled by atom symbols. In graph theory no geometric information is considered and two nodes (atoms) can have several edges (bonds) between them (multiple bonds).

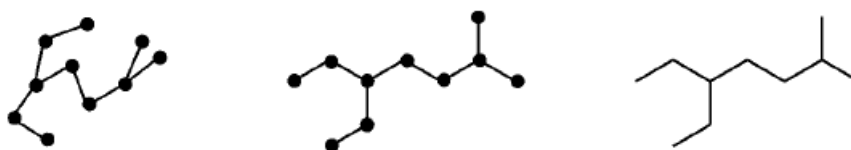


Figure 5.7: Different graph representations of the same diagram: only connections are considered, not length or angles.[§]

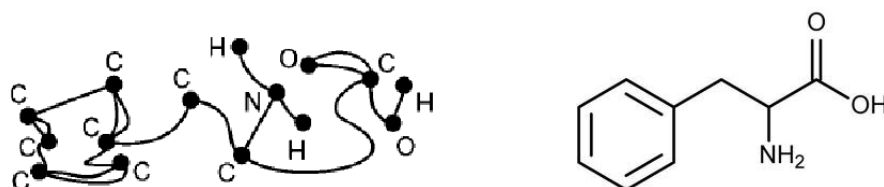


Figure 5.8: Phenylalanine weighted and labeled graph: different atoms and bonds types (left side).[§]

In order to mathematically process chemical models or representation, they have to be transferred to graph theory. Basics of graph theory are described in figure 5.9.

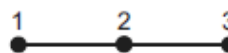
Nodes (dots) are *adjacent* when they are connected by the same edge.



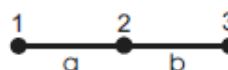
If the nodes of a graph are marked (e.g., with digits), the graph is termed *labeled*. In the example, node 1 is adjacent to node 2 but not to node 3.



The *degree* (or valency) of a node is determined by the number of distinct edges that end in a given node. (e.g., nodes 1 and 3 have the degree 1, and node 2 has the degree 2).



An edge is *incident* to the two joining nodes (e.g. *a* is incident to 1 and 2).



A graph is *connected* if at least one edge is between all the nodes. Thus, from any given node in a connected graph, all the other nodes can be reached.



Conversely, a *disconnected graph* (null graph) contains isolated nodes without edges (in chemistry, these may be mixtures of compounds or collections of substructures).



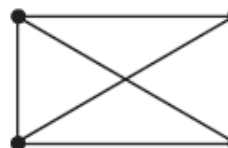
A *digraph* (or *directed graph*) has directed edges between two nodes (e.g., a weighted orientation).



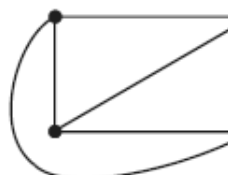
Structure diagrams are *undirected graphs*.



A graph is *complete* if all the nodes are connected (adjacent) to all the other nodes.



A graph is *planar* if it can be drawn on a plane without edges crossing, with intersections only at the edges (independently of how it is drawn). For example, cubane can be drawn as a planar graph.



Euler path: A connected graph can be traversed in one path (which ends at the node where it began) if all nodes have an even degree (see the Königsberg bridge problem, Section 2.4.1).

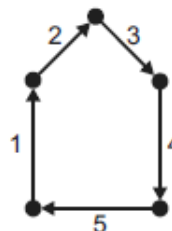
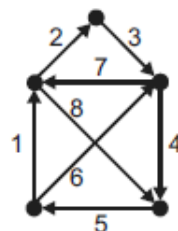
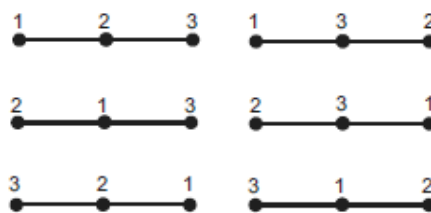


Figure 5.9: Graph theory basics.[§]

Euler circuit (the house of Santa Claus): a graph can be drawn in one path if the degree of all the nodes is a multiple of 2 and two nodes have an odd degree (for starting and ending the path). The drawing has to start at one of the nodes with an odd degree.



Isomorphism: If a labeled graph has n defined nodes, it can be represented by $n!$ labeled graphs. In the example with $n = 3$, the six graphs are isomorphic.



see Fig. 2-41 in Section 2.5.2.2

Figure 5.10: Graph theory basics(continue).[§]

5.3.2 Matrix Representation

A molecular graph can also be represented as a matrix: calculation of paths and cycles are based on matrix operations. The matrix of a structure with n atoms consists of an array of $n \times n$ entries. Molecular representations by matrix could be done. A molecule with its different atoms and bond types can be represented in matrix form in different ways depending on what kind of entries are chosen for the atoms and bonds. Molecules could be represented as a variety of matrices depending on what kind of descriptors are used for atoms and bonds: adjacency, distance, incidence, bond, and bond-electron matrices. Hydrogen atoms are sometimes not represented but calculable using valence rules of atoms. In redundant matrices each atom is described twice, in one column and in one row; in non-redundant matrices each element is represented only once (the top right or bottom left triangle of the matrix).

5.3.2.1 Adjacency Matrix

The adjacency matrix of a n atoms molecule is a square- $(n \times n)$ - matrix with entries listing all atoms connectivities. A row-column intersection has a value of 1 if the corresponding atoms are connected otherwise its value is 0. Thus an adjacency matrix is a Boolean matrix with bits 0 or 1.

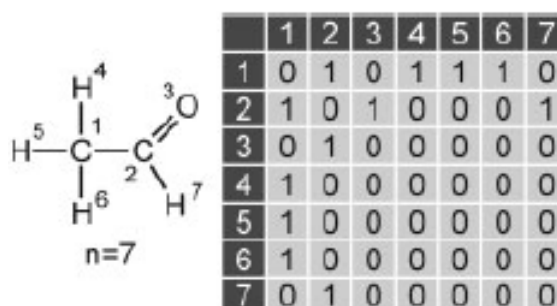


Figure 5.11: Adjacency Matrix(7x7) of ethanal.[§]

Diagonal elements are always zero and the matrix is symmetric around the diagonal elements (undirected, unlabeled graph). Since it is a redundant matrix, it can be reduced to half of its entries (figure 5.12). The requested storage space depends only on the number of nodes (atoms) and not on bonds number. In an adjacency matrix all information are contained in the much smaller non-redundant matrix; this kind of matrix is unsuitable for molecule constitution reconstructing because bonds orders information are not provided.

	1	2	3	4	5	6	7
1	0	1	0	1	1	1	0
2	1	0	1	0	0	0	1
3	0	1	0	0	0	0	0
4	1	0	0	0	0	0	0
5	1	0	0	0	0	0	0
6	1	0	0	0	0	0	0
7	0	1	0	0	0	0	0

	1	2	3	4	5	6	7
1		1		1	1	1	
2	1		1				1
3		1					
4	1						
5	1						
6	1						
7		1					

	1	2	3	4	5	6	7
1		1		1	1	1	
2			1				1
3							

	1	2	3
1		1	
2			1
3			

Figure 5.12: Redundant Adjacency Matrix simplification for ethanal (from left to right) by: omitting zero values, reducing it to the top right triangle, omitting the hydrogen atoms.[§]

5.3.2.2 Distance-based Matrix

Distance matrices describe the shortest distance between atoms in molecules, expressed as geometric distances (in Å) or as topological distances (in number of bonds).

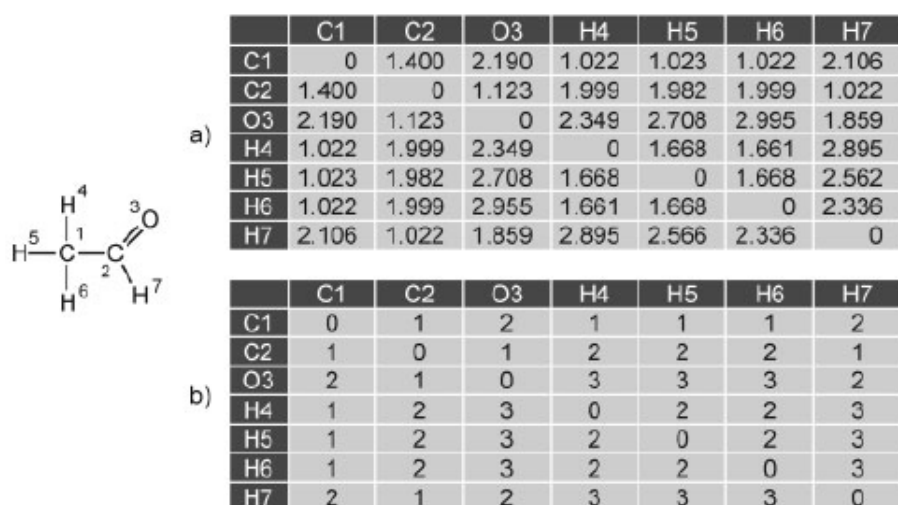


Figure 5.13: Ethanal distance matrices: a) geometric distances in Å and b) topological distances (shortes-path number of bonds between atoms).[§]

5.3.2.3 Atom Connectivity Matrix

Adjacency and distance matrices provide molecular connectivity information but no atom type or bond order description. Atom Connectivity Matrix, introduced by Spialter[12], provides more atoms/bonds information but it was abandoned.

5.3.2.4 Incidence Matrix

It is an $n \times m$ matrix where nodes (atoms) define the columns (n) and edges (bonds) correspond to rows (m). An entry is assigned value 1 if the corresponding edge ends in this particular node.

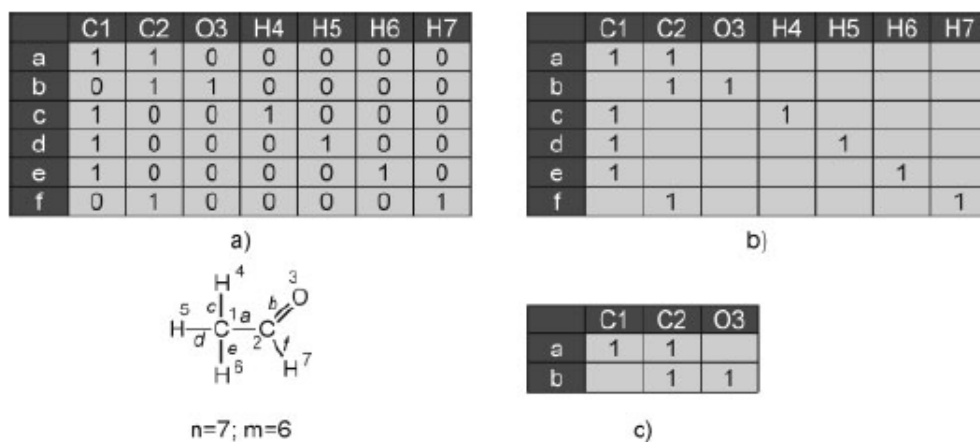


Figure 5.14: Ethanal redundant incidence matrices a) compression by: b) omitting the zero values, c) omitting the hydrogen atoms. In the non-square matrix, atoms are listed in columns and bonds in rows.[§]

5.3.2.5 Bond Matrix

A bond matrix is similar to adjacency matrix with additional information about bonds order of the connected atoms. Entries value could be: 0 (non bonds between considered atoms), 1 (single bond), 2 (double bond) or 3 (triple bond). This matrix is redundant.

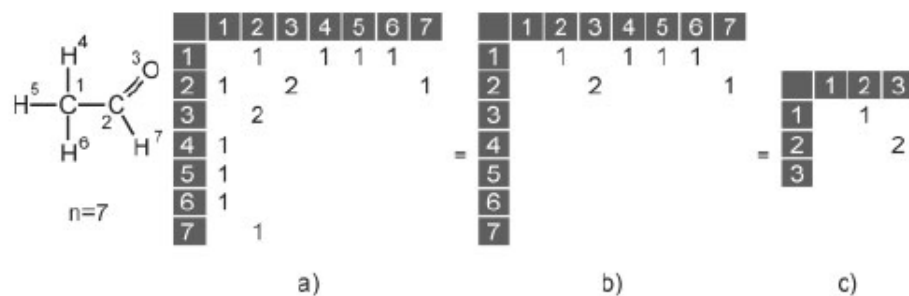


Figure 5.15: a) Redundant ethanal bond matrix with the zero values omitted. b) Compressed by reduction to the top right triangle. c) Omitting the hydrogen atoms.[§]

5.3.2.6 Bond-Electron Matrix

BE-matrix was introduced in the Dugundji-Ugi model[13]. It is both an extension of the bond matrix and a modification of atoms connectivity matrix. In addition BE matrix provides to the entries of bond values in the off-diagonal elements, the number of free valence electrons on the corresponding atom in the

diagonal elements (e.g., $O_3 = 4$ in Figure 5.16). In this kind of matrix are listed all atoms valence electrons: ones involved in bonds and free ones.

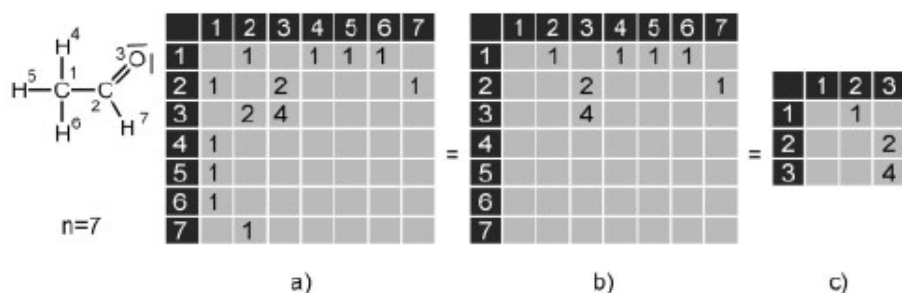


Figure 5.16: a) Redundant ethanal bond-electron matrix with the zero values omitted. b) Compressed by reduction to the top right triangle. c) Omitting the hydrogen atoms.[§]

A BE-matrix has interesting mathematical properties that reflect chemical information:

- The sum s_i of all entries of a row b_{ji} or column b_{ij} , is equal to the number of valence electrons of atom i . Eq. (1).

$$s_i = \sum_j b_{ij} = \sum_j b_{ji} \quad (1)$$

This is called row/column sum (Figure 5.17: carbon atom 2 has $1 + 2 + 1 = 4$ valence electrons.)

- The sum over all entries of the BE-matrix (S) correspond to the valence electrons total number. Eq. (2).

$$S = \sum_i \sum_j b_{ij} \quad (2)$$

In the example, ethanal has 36 valence electrons.

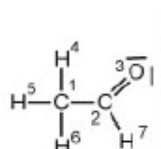
- If valence electrons number calculated does not agree with the standard number of valence electrons in an atom, this atom carries a charge; thus the diagonal element b_{ii} has more or fewer valence electrons than the nominal value b_{ii}^0 of the respective atom i . The charge is obtained by subtracting the sum of the row values from the nominal value. Eq. (3).

$$\Delta b = b_{ii}^0 - b_{ii} \quad (3)$$

- The cross sum \hat{s}_i , that is the sum over all the entries in a row and a column of atom i ($= 2s_i$ according to Eq. (1)) with the diagonal element b_{ii} of atom i counted only once, indicates the total number of valence electrons in atom i . (Eq. (4)).

$$\hat{s}_i = 2s_i - b_{ii} \quad (4)$$

In Figure 5.17, the oxygen atom 3 has $2 + 4$ (row) + $2 + 4$ (column) - 4 (diagonal element) = 8 electrons: the oxygen atom obeys the octet rule.



	1	2	3	4	5	6	7	row sum	Element
1		1		1	1	1		4	C
2	1		2				1	4	C
3		2	4					6	O
4	1							1	H
5	1							1	H
6	1							1	H
7		1						1	H
column sum	4	4	6	1	1	1	1	36	
cross sum	8	8	8	2	2	2	2		

$n=7$

Figure 5.17: The BE-matrix of ethanal allows to determine atoms valence electrons number (sum of each row).[§]

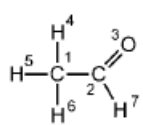
The BE-matrix is also useful for chemical reactions representation.

Advantages	Disadvantages
General	
the molecular graph is completely coded (each atom and bond is represented)	the number of entries in the matrix grows with the square of the number of atoms
matrix algebra can be used	no stereochemistry included
Adjacency matrix	
describes connections of atoms	no bond types and bond orders
contains only 0 and 1 (bits)	no number of free electrons
Distance matrix	
describes geometric distances	no bond types or bond orders
	no number of free electrons
	cannot be represented by bits
Incidence matrix	
describes connections and bonds	no bond types and bond orders
contains only 0 and 1(bits)	no number of electrons
Bond matrix	
describes connections and bond orders	no number of free electrons
	cannot be represented by bits
Bond-Electron matrix	
describes connections, bond orders, valence	cannot be represented by bits

Table 5.3: Matrix representation comparison

5.3.3 Connection Tables

In a matrix representation the number of entries increases with the square of molecule atoms number. What is needed is a molecular representation where entries number increases as a linear function of atoms number. This can be achieved by listing atoms and bonds in a table: rows and columns indices identify an entry distinguishing each atom and each bond in a molecule and giving the connections between atoms. This kind of representation is called connection table (*CT*). *CTs* were introduced in the early 1980s and are to date the most complete and performant chemical structure representation way in computer systems. In a *CT* each one atom of a molecule is labelled arbitrarily and then arranged in an atoms list (Figure 5.18). Bond information are stored in a second table with indices of the connected atoms, and with bonds order stored as an integer code (1 = single bond, 2 = double bond, etc.) in the third column. Atoms and the bonds lists, are linked through the atom indices.

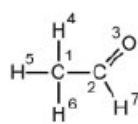


Atom list	
1	C
2	C
3	O
4	H
5	H
6	H
7	H

Bond list		
1 st atom	2 nd atom	bond order
1	2	1
2	3	2
2	7	1
1	4	1
1	5	1
1	6	1

Figure 5.18: Connection table for ethanal with arbitrarily labeled atoms.[§]

Alternative redundant *CT* is shown in Figure 5.19; the first two columns give the index of an atom and the corresponding element symbol. The bonds list is integrated into a tabular form. An atom can be bonded to several other atoms: atom with index 1 is connected to the atoms 2, 4, 5, and 6. These information can also be stored on a single line: a row contains one focused atom followed by the indices of all bonded atoms. Bond orders directly follows atom indices of these connected atoms. Atom 1 (carbon) is connected to carbon atom 2 and to hydrogen atoms 4, 5, and 6 by single bonds.



atom index	element	1 st index of atom	bond order	2 nd index of atom	bond order	3 rd index of atom	bond order	4 th index of atom	bond order
1	C	2	1	4	1	5	1	6	1
2	C	1	1	3	2	7	1		
3	O	2	2						
4	H	1	1						
5	H	1	1						
6	H	1	1						
7	H	2	1						

Figure 5.19: Ethanal Redundant Connection Table.[§]

Since each bond connects two atoms, each atom is defined twice. Starting from this, a non-redundant and compressed *CT* is obtained by listing bonds

only once and by omitting hydrogen atoms (Figure 5.20); this allow for storage space saving keeping intact molecular structure information. A CT is extendable by adding lists of free electrons, charges, atom parity, etc. If atoms indexing is changed, CT changes, so it is unambiguous but not unique, which can be achieved by canonicalization (see below). There are various CTs formats, distinguished in internal and external: usually internal connection tables are redundant (ensuring maximum flexibility and data processing speed), while external CTs are non-redundant for disk space saving.

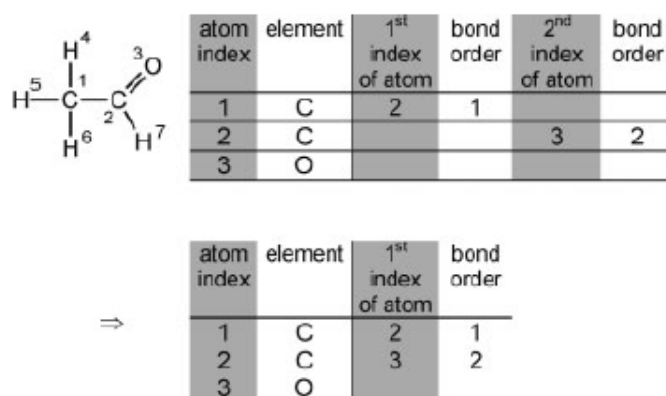


Figure 5.20: Ethanal Non-Redundant Connection Table. Only non-H atoms are described; lowest indices bonds are counted once (see Figure 5.19).[§]

Advantages	Disadvantages
the graph is completely coded	in most compact codes, hydrogen atoms are omitted and can be derived only indirectly
the number of entries grows linearly with the number of atoms	needs more than bits as entries
atom types, connections, and bond orders are described separately	
extensions allow addition of information on free electrons, stereochemistry, etc.	
widely used representation of chemical structure information	

Table 5.4: Connection Tables pro and con.

5.3.4 Input and Output of Chemical Structures

Computers language is based on bits packed into words or bytes, without understanding what atoms or bonds are. Human beings do not deal with bits very well: chemists uses 2d or 3D models to describe molecules. The problem is transfer these models to computers and make computers understand them. Graphical editor allow chemists for drawing molecular structures that could be converted into one of the structural representations described before or directly into a machine readable language. The reverse process is the output of molecular structures. Available molecular representations are:

- nomenclature: IUPAC names, trivial names, registration identifiers;
- line notations: Wiswesser line notation (WLN), ROSDAL, Sybyl line notation, SMILES, etc.
- connection tables;

Molecular representation formats could also be interconverted each other.

5.3.5 Structure Exchange Formats

<i>File format</i>	<i>Suffix</i>	<i>Comments</i>	<i>Support</i>
MDL Molfile	*.mol	Molfile; the most widely used connection table format	www.mdli.com
SDfile	*.sdf	Structure-Data file; extension of the MDL Molfile containing one or more compounds	www.mdli.com
RDfile	*.rdf	Reaction-Data file; extension of the MDL Molfile containing one or more sets of reactions	www.mdli.com
SMILES	*.smi	SMILES; the most widely used linear code and file format	www.daylight.com
PDB file	*.pdb	Protein Data Bank file; format for 3D structure information on proteins and polynucleotides	www.rcsb.org
CIF	*.cif	Crystallographic Information File format; for 3D structure information on organic molecules	www.iucr.org/iucr4op/cif/
JCAMP	*.jdx, *.dx, *.cs	Joint Committee on Atomic and Molecular Physical Data; structure and spectroscopic format	www.jcamp.org/
CML	*.cml	Chemical Markup Language; extension of XML with specialization in chemistry	www.xml-cml.org

Figure 5.21: Standard Exchange File Formats.[§]

Data processing asks for interaction and cooperation of different software systems and databases; the exchange of chemical structure information plays a basic role since the internal file format of one software system has to be understood by another. Exchange process is handled through an external ASCII file. Many different file formats have been developed since the 1970s, but the MDL Molfile format developed at Molecular Design Limited (now MDL Information Systems, Inc.) became a de facto standard file format[14]. Extensions of MDL Molfile format, led to SDfile, RGfile, Rxnfile, or RDfile, with special additional information each one. Major standard exchange file formats are summarized in figure 5.21.

5.3.6 Molfiles and SDfiles

Only two file formats have widely been accepted by the cheminformatics community as standard formats for chemical information exchange: the Molfile and SDfile[15] formats first described by Dalby et al. from Molecular Design Limited (MDL). While Molfile describes a single molecular structure which can contain disjointed fragments, an SDfile (SD stands for structure-data) contains structure and data (properties) for any number of molecules, which makes it especially convenient for handling large sets of molecules - for example for data transfer between databases or from databases to data analysis tools. Both Molfile and SDfile are based on Connection Tables.

5.3.6.1 Molfiles

Figures 5.22 present the L-Alanine molecule and corresponding Molfile. A Molfile consists of a header block specific to Molfiles (lines 1-3) and a connection table (*Ctab* or *CT*) (lines 4-18), which is common to all MDL's CTfile formats. The header block is so formatted:

- line 1: molecule name (trivial name, ID, alphanumeric code, etc.)
- line 2: user's name, software name, date and time of file creation
- line 3: comments
- line 4-18: connection table (*Ctab*) core, description of given compound constitution.

CT first line (*counts line*), specifies atoms and bonds number, chirality information (1 or 0 in the chiral flag entry). The last entry specifies *Ctab* format version (In figure 5.22 V2000). Atoms enumerated in counts line are described in atoms block; each one is represented by a single row, which contains Cartesian coordinates, atomic symbol and charge. Coordinates could define a 2D or 3D molecular model, as declared in the second file line. 3D CT has non-zero value in atoms block third column (z-coordinates). Bonds are described in the bonds block in which each line specifies which two atoms are bonded, bond type and the stereo configuration. Atoms indices reflect their sequential order in the atom block. The last part, properties block, can contain miscellaneous properties. The last line is "M END" which indicates the end of file and thus the end of molecule.

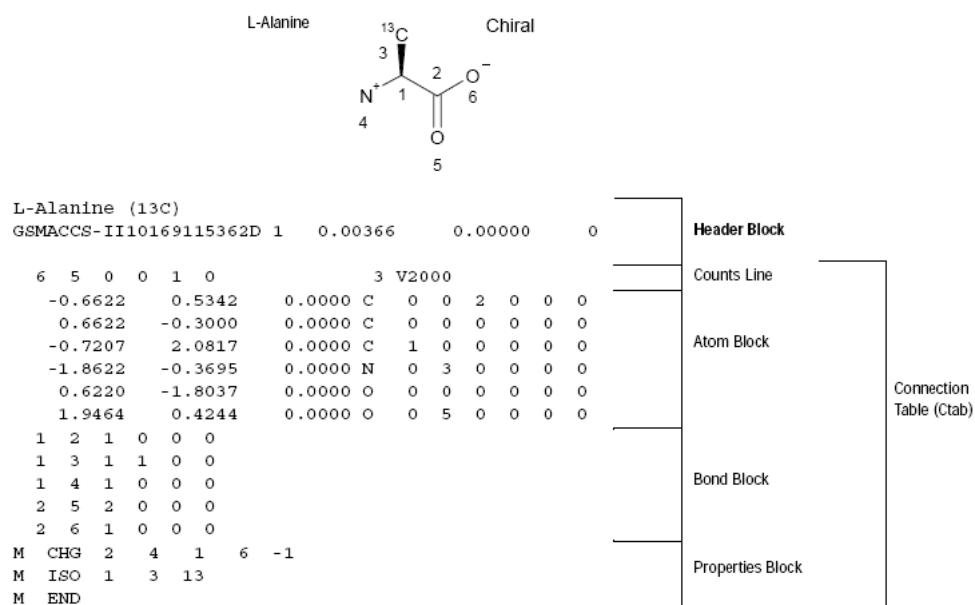


Figure 5.22: L-Alanine Molfile.[§]

5.3.6.2 SDfiles

As mentioned before an SDfile contains structural information and associated data for one or more compounds; thus it is particularly useful for data exchange between databases and computational software. In an SDfile, each molecule is represented by its Molfile (CT) with additional data describing its non-structural properties in text fields (molecular weight, molecular descriptors, biological activity, etc.). Each one molecule in the SDfile is terminated by a "\$\$\$\$" delimiter while it is started by a data header line (a molecular name like an ID, a code or a trivial name). Figure 5.23 shows structure and SDfile of sulfuric diamide (sulfamide) molecule, extracted from PUBCHEM [<http://pubchem.ncbi.nlm.nih.gov/>].

```

82267
-OEChem-01061302493D

9 8 0 0 0 0 0 0999 V2000
0.0001 -0.1566 0.0042 S 0 0 0 0 0 0 0 0 0 0 0 0
0.0007 -0.9136 -1.2324 O 0 0 0 0 0 0 0 0 0 0 0 0
0.0001 -0.8269 1.2900 O 0 0 0 0 0 0 0 0 0 0 0 0
1.3107 0.9492 -0.0306 N 0 0 0 0 0 0 0 0 0 0 0 0
-1.3116 0.9479 -0.0312 N 0 0 0 0 0 0 0 0 0 0 0 0
2.2234 0.5484 -0.2857 H 0 0 0 0 0 0 0 0 0 0 0 0
1.4024 1.5552 0.7960 H 0 0 0 0 0 0 0 0 0 0 0 0
-2.2239 0.5463 -0.2868 H 0 0 0 0 0 0 0 0 0 0 0 0
-1.4043 1.5538 0.7954 H 0 0 0 0 0 0 0 0 0 0 0 0
1 2 2 0 0 0 0
1 3 2 0 0 0 0
1 4 1 0 0 0 0
1 5 1 0 0 0 0
4 6 1 0 0 0 0
4 7 1 0 0 0 0
5 8 1 0 0 0 0
5 9 1 0 0 0 0
M END
> <PUBCHEM_COMPOUND_CID>
82267

> <PUBCHEM_CONFORMER_ID>
0001415B00000001

> <PUBCHEM_MMFF94_ENERGY>
6.474

$$$$

```

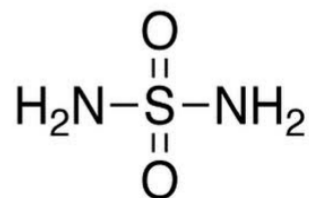


Figure 5.23: Sulfamide Sample SDfile.[§]

5.3.7 Unambiguous and Unique Representations

In cheminformatics unique representation of chemical structures is essential. Database and chemical libraries handling with registry, storage, and retrieval systems requires a one-to-one correspondence of a unique and invariant notation with the respective chemical structure. Canonical coding process is needed.

5.3.7.1 Structure Isomers and Isomorphism

Atoms can be arranged in many different bonding situations and often different structural formulas could match same empirical formula (figure 5.24).

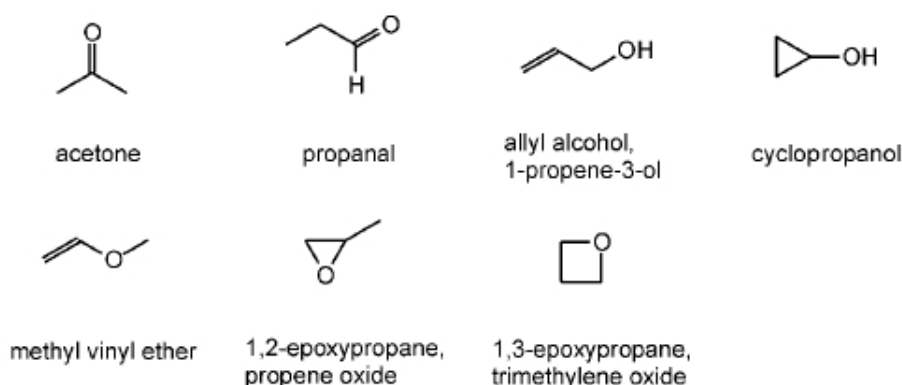


Figure 5.24: Structure diagrams expressing C_3H_6O empirical formula.[§]

In isomorphic structures the atoms (nodes) of structures (graphs) correspond one-to-one, preserving the adjacency of the nodes; thus, the topology of considered molecules is identical. In figure 5.25 this is described with phenylalanine; the three substituents H, COOH, and NH₂ can be positioned arbitrarily on the terminal carbon atom.

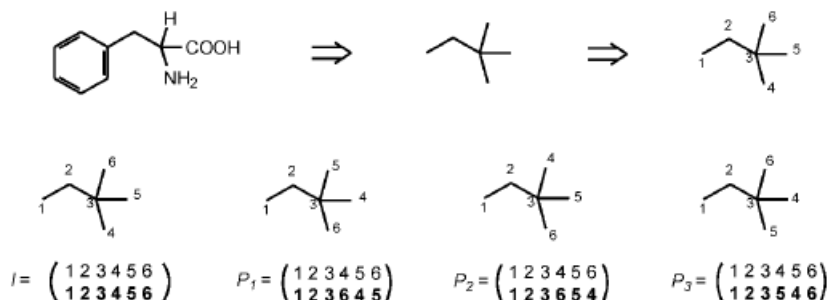


Figure 5.25: Structural Isomorphism problem. Phenylalanine is simplified to a core without the substituents and arbitrarily numbered atoms (top of figure). Substituents position can be permuted (without changing the constitution, second line) using a permutation group (bottom of figure): first mapping line is the original atoms numbering while second one describes the changed atoms numbering (I=original, P=permuted).[§]

The determination of isomorphic structures requires a mathematical operation called permutation: two or more structures are isomorphic if they are interconvertible by permutation (Eq. (5)); P_3 and P_2 are identical if a P_x operation is applied. Considering atom 4 of P_3 (Figure 5.25, third line): In P_3 atom 4 takes the place of atom 5 of 5 in P_2 . To replace atom 4 in P_2 at position 5, both have to be interchanged, by writing the number 4 at the position of 5 in P_x . Applying this to all s substituents, the result is a new permutation P_x identical to P_1 .

$$P_x P_2 = P_3 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 6 & 4 & 5 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 6 & 5 & 4 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \end{pmatrix} \quad (5)$$

In databases it is necessary to compare existing entries with new ones in order to check for redundancy; registration and retrieval of compounds in databases or chemical libraries are based on isomorphism detection algorithms that compare structure diagrams searching for identical molecular graphs.

5.3.7.2 Canonicalization process

A connection table is ambiguous and not unique: the same molecule may be represented by a variety of different connection tables with different atoms numbering (see Figure 5.19). A structure with n atoms can be numbered in $n!$ corresponding to up to $n!$ different CTs (in case of symmetric molecule, some CTs are identical). A three atoms molecule, e.g. hypochlorous acid ($ClOH$), has $3! = 6$ different atomic numberings and so six different CTs.

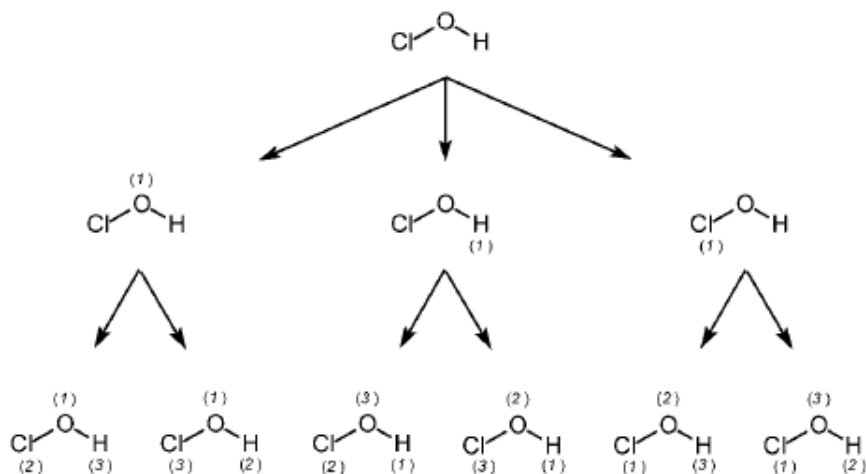


Figure 5.26: Six different atoms numbering for $HClO$.[§]

Canonicalization is necessary to define one and only one standard and reproducible atoms numbering: this is called canonicalization process and lead the molecule to be represented by only one CT or bond matrix. The most used and performant technique is based on the Morgan Algorithm[16] and its improvements.

5.3.7.3 The Morgan Algorithm

The scope of Morgan Algorithm is to obtain invariant-labeled atoms. The classification is done by considering neighbors number (*connectivity*) for each one atom in an iterative manner (*extended connectivity, EC*). Finally Morgan Algorithm produces an unambiguous and unique atoms numbering. Coding process is based on two major aspects:

- Unique Coding: A n atoms molecule has $n!$ different atoms labeling (a 12 atoms structure has about 0.5 billion possible connection tables); the Morgan Algorithm reduces them to only one.
- Consideration of stereochemistry: The parity - R/S or cis/trans - of a stereocenter can be obtained by considering Morgan atoms numbers similarly to CIP rules.

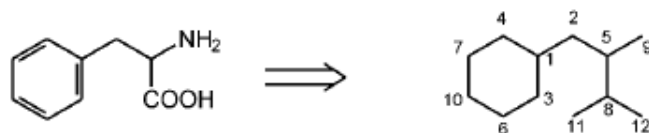


Figure 5.27: Morgan Algorithm: canonical numbering for phenylalanine.[§]

It has to be noticed that in some cases the algorithm fails, because of in some structures the numbering show oscillatory behavior. Moreover, even if equivalent atoms has the same EC value, they are not necessarily always equivalent. This are the major causes of failure in redundancy washing on chemical libraries: a low performances canonicalization process leads to wrong identification of duplicate compounds. In chapter 6 are explained methods developed during this research project and able to address this chemoinformatics crucial issue.

5.3.7.3.1 Morgan Algorithm: an example In addition to provide unique and invariant atoms numbering, the Morgan Algorithm can identify constitutionally equivalent atoms. This is done by a two steps process:

- Relaxation process: calculates the EC
- Assignment of atoms numbering

5.3.7.3.1.1 Step 1: Classification of atoms by considering their neighborhood (relaxation process) Considering organic structures constituted by C, N, O, H, and halogens, atoms are classified into four classes depending on the number of non-hydrogen attached atoms. Class number values ranges from 1 to 4 (primary-quatarnary C atom), corresponding to node/atom degree. Hydrogen atoms number can be obtained by application of valence rules. The first iteration process extracts information about each node degree; then Morgan takes the neighboring atoms into account by summing class values of all connected atoms. This led to a new class value for each one atom: EC value that represents the neighborhood of the adjacent atoms (Figure 5.28).

an EC=9.

Globally the sequence is:

1. First sphere atoms EC values results from NAn (*Neighboring Atoms number*), Eq.(6)

$$EC(1) = nNA(1) \quad (6)$$
2. Number of equivalent classes (c) for the first sphere is determined. (c) is equivalent to different EC values number.
3. In the higher sphere (s) the EC values are given by the sum of EC values of directly connected neighboring atoms of the former sphere, Eq. (7):

$$EC(i) = nNA(i) \quad (7)$$
4. At each sphere the number of equivalence classes (c) is determined.
5. Iteration is continued until the number of equivalent classes is equal to or lower than the previous iteration one.
6. The iteration with the highest number of equivalent classes enters the next step.

5.3.7.3.1.2 Step 2: Assigning unique, invariant atoms numbering

The highest equivalence classes number iteration is taken as the starting point for the canonicalization: the highest EC value atom is labeled with the sequence number 1. This is the most deeply embedded atom in the structure, from which Morgan process start the numbering. From the initial atom all the first-neighbor atoms are assigned according to the magnitude of their extended connectivity value. The neighbor atoms of the second(or current + 1) atom are assigned in an equivalent manner. This is done for all the atoms; arbitrary decisions are made when numbering the equivalent atoms.

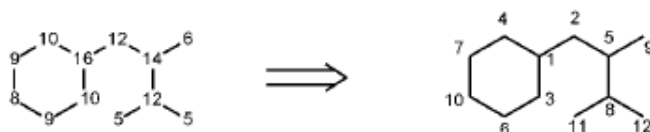


Figure 5.30: Morgan Algorithm-based Canonicalization: starts at the highest EC value atom (in the example: 16), which gets the number 1. All other atoms are numbered according to their EC values.[§]

Globally the sequence is:

1. Highest EC value atom obtains sequence number 1 (current atom).
2. All the connected neighboring atoms are enumerated 2, 3, 4, etc., according to their decreasing EC values. In case of atoms which have the same EC value, they are numbered serially following specific rules: atom types (C before N) or bond types (single before double), charges, etc.

3. The next highest numbered atom compared with the current atom becomes the current one. All unnumbered atoms connected to the current one, are numbered serially according to their decreasing EC values. Atoms with equivalent EC values are numbered following the specific rules.
4. This process is continued until all the atoms are canonically enumerated.

There are anyway some problems in the Morgan algorithm in finding the terminating condition of step 1 (oscillatory behaviour in number of equivalent classes [17] or special atoms with isospectral points [18]).

5.3.8 Special Notations

In addition to the molecular representations described above, there are other specific notations for specific applications.

5.3.8.1 Fragment Coding

Fragments allows for indexing of particular features in chemical structures [19, 20]. These are small assemblies of atoms (functional groups, ring systems, etc.), that has to be specified in advance. Performances of fragment coding systems depend on fragments definition rules.

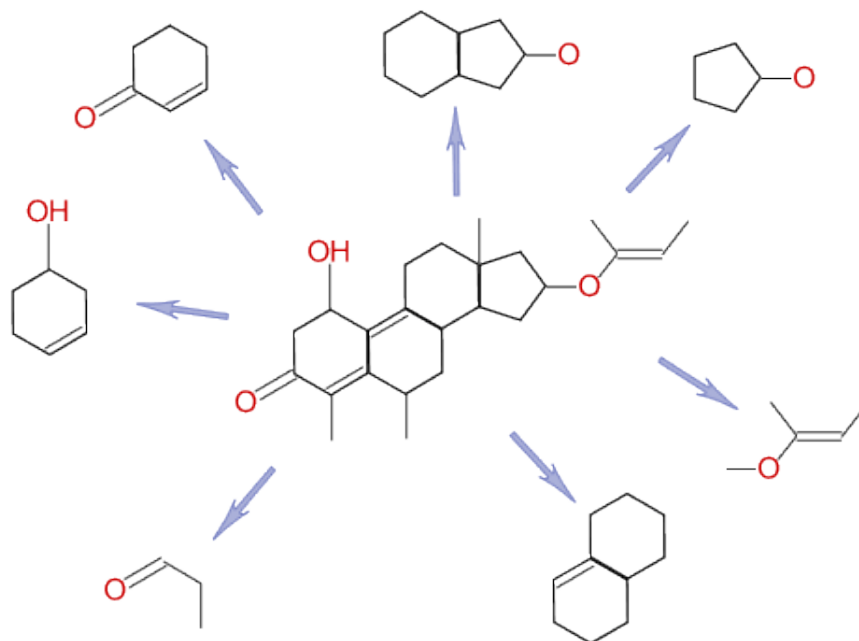


Figure 5.31: Molecular Fragments Coding.[§]

Fragment codes are ambiguous: different structures could have the same fragment code, because it does not describe molecular topology (atoms connections). Fragments characterize molecules classes and this is important in

patents chemical structures description. The principal application of fragments coding is the substructure search on molecular databases in order to identify all molecules characterized by a specific set of chemical groups.

5.3.8.2 Fingerprints

A *fingerprint (FP)* identifies a specific molecule as a human fingerprint identifies a person. Structural keys describes specific molecular features indicating the presence or not of particular atoms assemblies. The fragments are coded in binary keys so the fingerprint results in a sequences of 0 and 1 (bit strings): 0 indicates the absence of that specific chemical group while 1 the presence (at least one time in molecule). Fingerprints have lengths of 150-2500 bits depending on used coding rules and molecular complexity. It is necessary to define in advance a fragments library; if it has a number of fragments equal to the number of bit in the bitstring, bits could be correlated 1:1 to the fragments. On the other hand if a structure contains only a few defined fragments, only a few bits are set. This molecular representation is ambiguous but allows very efficient similarity search.

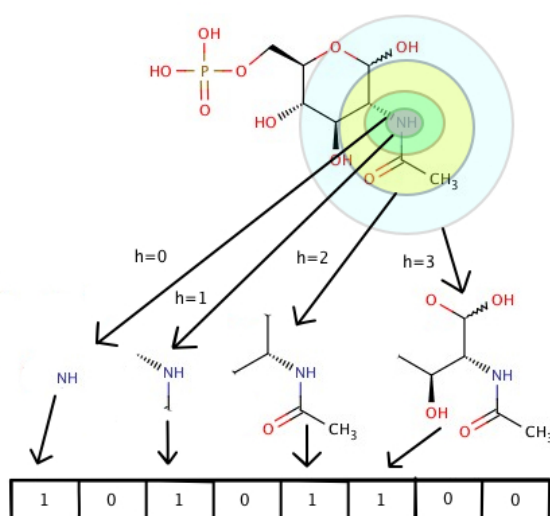


Figure 5.32: Fingerprint Bitstring Constitution: only existing features get the "1" bit.[§]

5.3.8.2.1 Hashed-Fingerprints In the *Hashing* technique, molecular bonds are traversed obtaining information about substructures and molecular relationships[21]; fragments received are assigned "1" bit in the string but the FP may include collision entries. *Hashed-FP* do not need for pre-defined fragments library, getting a deeper and more complete molecular description. However, no direct bitstring-chemical features correlation exists because of fragments library has not been defined.

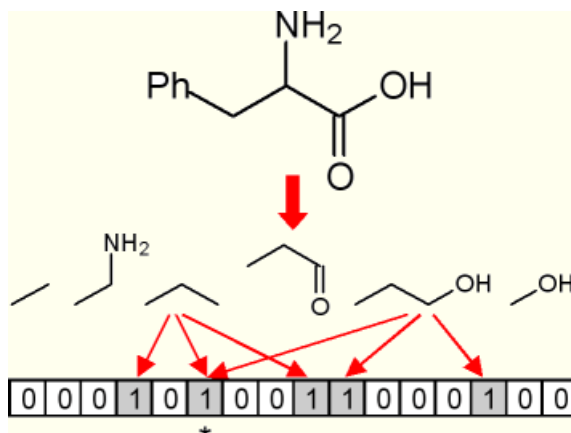


Figure 5.33: Hashed-Fingerprint Constitution: asterisk indicates a bitstring collision.[§]

5.3.8.3 Hash Codes

Hash coding is an informatics method[22, 23]. Coding strings coming from CTs may be quite large and thus not useful for structures storage addressing. The hashing procedure splits the input into multiple small pieces: this allows for high data transfer rate, for example during databases quering.

Hashing (*key transformation*) creates storage addresses from alphanumeric keys; data are separated into ID-labeled fragments which are not directly accessible but need transforming into a *hash-code*(*fixed characters-number code*). Produced code is highly compressed and depends only on input information(e.g. CT, thus molecular topology). Hash codes does not contain structural data, it is only used as an addressing-key to stored data entries. Has code has a pre-defined bit length. A 32 bits hash code could have 2^{32} (4 294 976 296) possible values, while a 64 bits one, 2^{64} values. Due to the fixed length, different data entries could share the same hash code (*address collision*); higher input data number correspond to higher probability in address collision. To avoid collisions it is possible to code about 10000 entries with a 32 bit has code and about 100 million with a 64 bit one[24].Hashing reduces multidimensional data to only one dimension, so information gets lost and complete data reconstruction from code is impossible; thus it is not usable for direct data access, but on the other hand, dued to intensive data reduction, it is an high-performances method for databases managing and indexing.

5.3.9 Stereochemistry representation

Stereoisomers are compounds that shares the same topology but not the same geometry (topography). Stereochemistry has a crucial role in target (protein)-ligand interactions; the commercial chemical space analysis performed in this research project has highlighted that almost all line notation coding system, except for InChI, difficultly deal with stereochemistry definition of chemical libraries compounds, because of missing rules on stereocenters definition or because of lacks in enantiomers discrimination as explained in Chapter 7. The

stereochemistry is expressed in chemical representations as wedged bonds (substituent in front of viewer's plane) and hashed bonds (substituent point away from reference plane). It is crucial to completely and unambiguously transfer the stereochemistry information from 3D to 2D representation.

ok	warning	undefined	Ok	ok	ok	ok
undefined	undef	undefined	Ok	warning	ok	warning
undefined	warn: bonds inside 180° sector (examples)					ok

Figure 5.34: Stereochemistry Representation for tetra-substituted (top) and tri-substituted (bottom) stereocenters: only unambiguous definition are allowed. [H]

	ok	undefined	ok	undefined	Undefined	undefined
input						
interpreted as						
	ok	ok	ok	ok	Undefined	undefined
input						
interpreted as						

In cheminformatics application, stereogenic units are identified/generated by substituents permutation. Beyond chiral centers (atoms), other stereogenic units are planes and axis of chirality; detection of chirality is possible for chiral atoms, but not always in case of different stereogenic units.

5.3.9.1 Detection and Specification of Chirality

In 1950s Cahn, Ingold, and Prelog proposed *CIP rules*[25, 26, 27, 28] which became the official way to describe stereoisomerism. Rules refers to atoms directly connected to stereogenic unit and are summarized as follows:

1. Decreasing atomic number ligands ranking
2. If the order of two substituents cannot be determined, it is decided by atomic numbers comparison of next atoms in the connected skeleton
3. In case of atoms with same atomic number but different mass number, the higher mass number atom is assigned the highest priority
4. Double and triple bonds account for two or three single bonds

To decide the R or S configuration, the molecule is oriented so that the lowest priority group points directly away from observator; the other three groups, are traversed from higher to lower priority. Clockwise direction correspond to R configuration while anticlockwise describes S configuration. The older implementations of CIP rules for computer-based chirality detection are: LHASA[29], CHIRON[30], and CACTVS[31] software packages, while more recently several commercial molecular editors and visualizers (CambridgeSoft's ChemOffice, ACD's I-Lab, Accelrys' WebLab, and MDL's AutoNom) implemented CIP rules. The most used and recent chemoinformatics methods to detect/describe stereoisomerism are:

- Ordered Lists
- Rotational List
- Descriptor permutation

5.3.9.1.1 Ordered Lists Starting from a tetrahedral atom the, 4 substituent can be arranged according to their priority (1-4) in $4!$ (24) ways; these arrangements belongs to two simmetry classes. A class can be interconverted into the other by single permutation of two ligands, while two permutations led to an isomer of the same class. Substituents priorities are assigned by considering atomic number as in CIP rules.

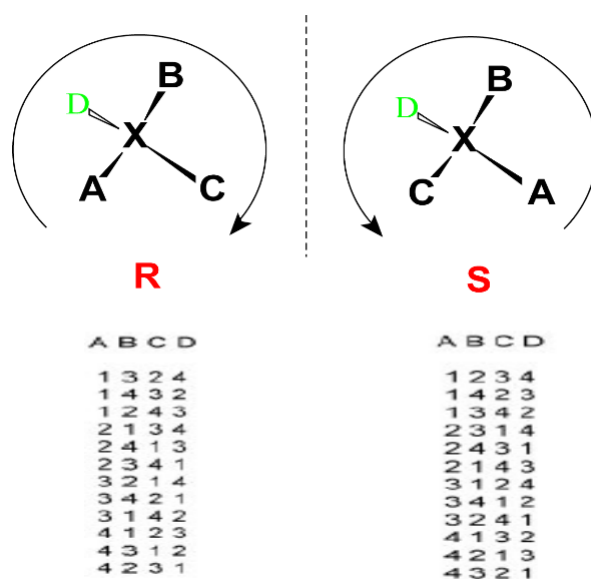


Figure 5.35: Ordered List of 24 priority arrangements around a tetrahedral stereocenter. The two classes of permutation are separated: *R* configuration on left and *S* configuration on right.[§]

Another technique, *PLR*[32], assigns priorities by using Morgan Algorithm unique-numbering(see paragraph 5.3.7.3). CIP and PLR priorities are not correspondant so stereoisomers in PLR are marked as Y and X. The same approach could be used for stereochemistry determination at double bonds.

5.3.9.1.2 Rotational Lists Rotational lists[33] was first introduced in the *Standard Molecular Data (SMD)* format. Geometrical arrangements around a stereocenter are listed(e.g., square, tetrahedron, ...) and atoms are also numbers-labeled; in this way a stereoisomer is defined by its rotational list in the *stereo block* of a SMD file.

5.3.9.1.3 Permutation Descriptors Stereodescriptors works in pairs (*R/S*, *Cis/Trans*) only if a molecule contains atoms with a maximum coordination number of four; computers can deal with this situation representing the pairwise descriptors by "0" and "1" bits. Mathematically +1 or -1 permutation descriptors are used. With coordination numbers higher than four permutation group sign (+/-)is no sufficient. Stereochemistry description by permutation groups require molecule stereocenter splitting obtaining a skeleton and its ligands; each one is then numbered independently.

§] §] §]

Figure 5.36: Permutation Descriptors Determination. Both skeleton and its ligands are numbered independently(skeleton indices in italics, ligands indices in bold).

Skeleton numbering can be arbitrary but fixed; ligands numbering is based on CIP rules. A reference stereoisomer corresponds to the situation with ligand 1 on skeleton site 1, ligand 2 on skeleton site 2, etc.; this reference stereoisomer is assigned a +1 descriptor. Comparison of each one stereoisomer with the reference one, allows for permutation descriptors determination. In Figure 5.37 this is shown for a reflection operation on the reference isomer, obtaining an inversion of the stereocenter; then a transposition of two ligand indices brings the structure to correspond with reference. The stereoisomer obtained by a reflection operation has $(-1)1 = (-1)$ descriptor.

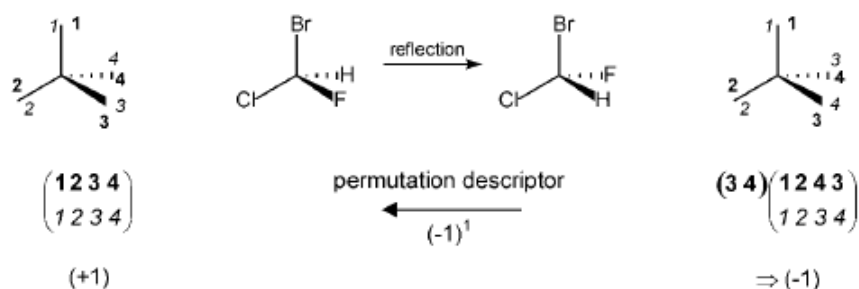


Figure 5.37: Operation on Permutation Descriptors: reflection on reference stereoisomer produce a stereocenter inversion.[§]

5.3.9.2 Stereochemistry in Molfile and SMILES

Stereocenters definition by stereo descriptors is incorporated in the two most used structure representations, Molfile and SMILES.

5.3.9.2.1 Stereochemistry in Molfiles Connection table of a Molfile only stores information about molecular topology; stereoinformation are encoded as stereodescriptors which are stored in specific fields of Molfile (Figure 5.38). Molfile stereodescriptor is based on PLR ordered list. Sometimes stereochemistry is unknown so the descriptor has more than 2 possible values (R and S); this could be bypassed using the *parity value*, which is calculated by comparison of parity value and ordered list. Morgan Algorithm atoms-indices are permuted until they are in ascending order: if this is achieved by an odd number of permutations, the parity value is 1, while it is 2 for an even permutations number. In addition it could be 0 for undefined stereochemistry or 3 for unknown stereochemistry.

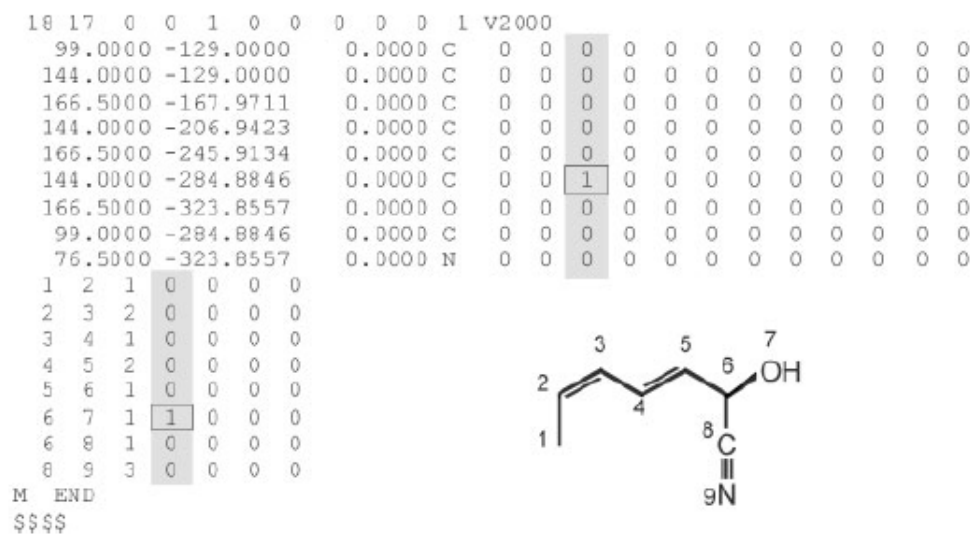


Figure 5.38: Molfile Stereochemical Flags. Parity values are marked in the gray columns.[§]

In the example in Figure 5.38, atom 6 (row 6, column 7 of the atom block) has a parity value of 1; in fact to obtain an ascending Morgan numbers order are needed an odd number of permutations. The process is so summarized:

1. Atoms numbering canonicalization by Morgan Algorithm
2. Permutation reiteration to bring Morgan indices in ascending order.

In Figure 5.39 example only one permutation is required (3 to 4), corresponding to a parity value of 1 and thus to the R-isomer. Stereobonds are defined in the bond list (4th column of CT):

- 0 correspond to a single non-stereo bond
- 1 for up (a wedged bond)
- 4 for either up or down
- 6 for down (a hashed bond).

Cis/Trans or *E/Z* double bonds configuration is obtained by the x,y,z coordinates of the atom block only if the value is 0; value 3 indicates that the double bond is either cis or trans. In the bond block of Figure 5.38 the stereocenter is set to 1 (up) at atom 6 (row 6, column 4 in the bond block), while the double bonds configuration is determined by the x,y coordinates of the atom block.

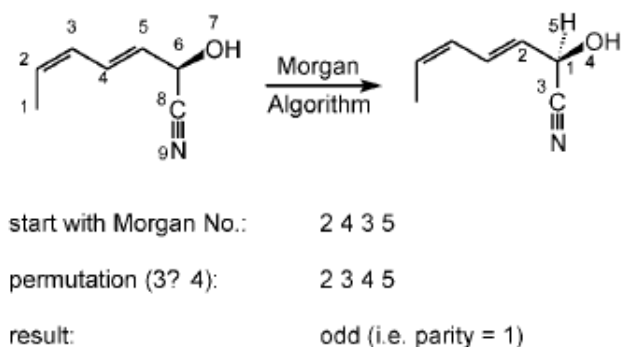


Figure 5.39: Parity Values Determination. After structure canonicalization (top), parity value is determined by the number of permutations needed to bring into ascending order the Morgan indices (bottom).[§]

5.3.9.2.2 Stereochemistry in SMILES In the SMILES notation, clockwise or anti-clockwise atoms ordering, is coded by @ or @@ respectively. Reading the SMILES code from the left, the three atoms after the identifiers @ or @@ describe the stereochemistry of the stereocenter. These three atoms representation depends only on writing order, and not on atoms priorities. Double bonds stereoisomerism is coded in SMILES by \ or / which specify the connected atoms relative direction (parallel or opposite) at double bond.

C(=C(/CCC[C@](C(O)=O)(C)C)Br)/(C)C

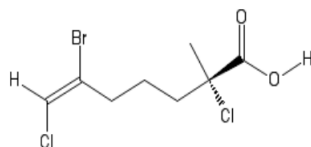


Figure 5.40: SMILES Stereochemistry Representation. @ defines tetrahedral carbon configuration while \ defines double bond *Cis/Trans* configuration.[§]

Part III

Results and Discussion

Chapter 6

Redundancy Management in Chemoinformatics Libraries

Contents

6.1	Introduction	55
6.2	InChI	56
6.2.1	InChI Generation	56
6.2.2	InChI Layers Type	62
6.3	SMILES and InChI used implementations	68
6.3.1	SMILES Implementation	68
6.3.2	InChI Implementation	68
6.4	Datasets	69
6.5	InChI and SMILES Strings Reproducibility Assessment	69
6.6	Redundancy Identification Efficiency	71
6.6.1	Single Catalogue Redundancy Analysis	71
6.6.2	Pan-catalogues Redundancy Analysis	77
6.6.3	SMILES Errors Identification	78
6.7	MMsDusty Pipeline	90
6.8	Conclusion	90

6.1 Introduction

In the broadest sense a molecular database is a chemical data storage system usable for queries based on different metrics. Fundamental requirements for such a system are first of all the complete redundancy absence, and then the chemical correctness(atom types, bond types, bonds angles and distances) of stored structures; this second requirement is achieved only by ensuring completeness and accuracy of chemical information transfer process during in-silico models and representations building. The scope of this first phase of research project, was to compare efficiency in molecular redundancy identification for two chemical line notation systems: InChI and SMILES. Redundancy heavily affects

chemoinformatics protocols performances and results quality. Using redundant chemical libraries in virtual screening projects, could lead to an over-estimation of geometric positioning process quality; in the same manner a similarity search query leads to an over-estimation of query/library similarity rate. This work highlights how InChI and SMILES are not equally efficient in redundancy identification, depending on libraries size and composition.

6.2 InChI

InChI (*International Chemical Identifier*)[4, 34] was developed in order to provide a characters string for uniquely representing chemical compounds; this requires transformation of an input connection-table into an output alphanumeric string. In order to be a digital molecular signature two properties are requested:

1. Different compounds (different CTs) must have different identifiers
2. A single compound must have a single identifier

In order to achieve this, InChI needs to include all of chemical features that characterize a specific compounds, while drawing conventions are suppressed. InChI has a hierarchical and layered structure which reflects different molecular details levels; each level describes a distinct class of chemical information with layers ordered to provide sequential structural refinement. Layers are:

1. Basic connectivity
2. Overall charge
3. Mobile/fixed H-atoms(expresses tautomerism)
4. Isotopic composition
5. Stereochemistry

InChI layers are each one appended in a strictly defined order: each layer has only one preceding (*parent*) layer except for the first layer; repeating layers are suppressed while each layer depends on prior layers. Thus the same layers of two different compounds are not comparable; each layer do not depends on successive layers. If two InChI strings are identical up to a layer, it means that the structural characteristics of the two compounds are identical up to that point.

6.2.1 InChI Generation

The original idea of InChI is the normalization step which allows for conventions removal maintaining compound description. The main layer (atoms and bonds) is mandatory while all other layers are provided only if the corresponding input information is described; in this way chemical description is adjustable at different detail levels. InChI is generated from input structure in three steps:

1. Normalization (removal of not needed information and information splitting into layers)

2. Canonicalization (drawing-independent atoms labelling)
3. Serialization (labels conversion into InChI).

6.2.1.1 Normalization

For the main layer construction are only needed identities of each atom and its covalently-bonded atom/atoms. Information about pi-bonds, charge, isotopic composition, tautomerism and stereochemistry are ignored. This *normalization* process avoids the complexities commonly encountered in structure representation. Nitro groups are an example of this; representations of zwitterions and special valencies are well known cheminformatics coding problems, avoided by normalization step. This kind of representation describes only the single-bond network of a molecule, while excess or deficit of electrons (charges) is represented in a separate layer. In figure 6.1 are summarized input, normalized and canonicalized structures; large numbers designate classes of equivalent atoms while small numbers (canonical numbers) are used in the serialization step.

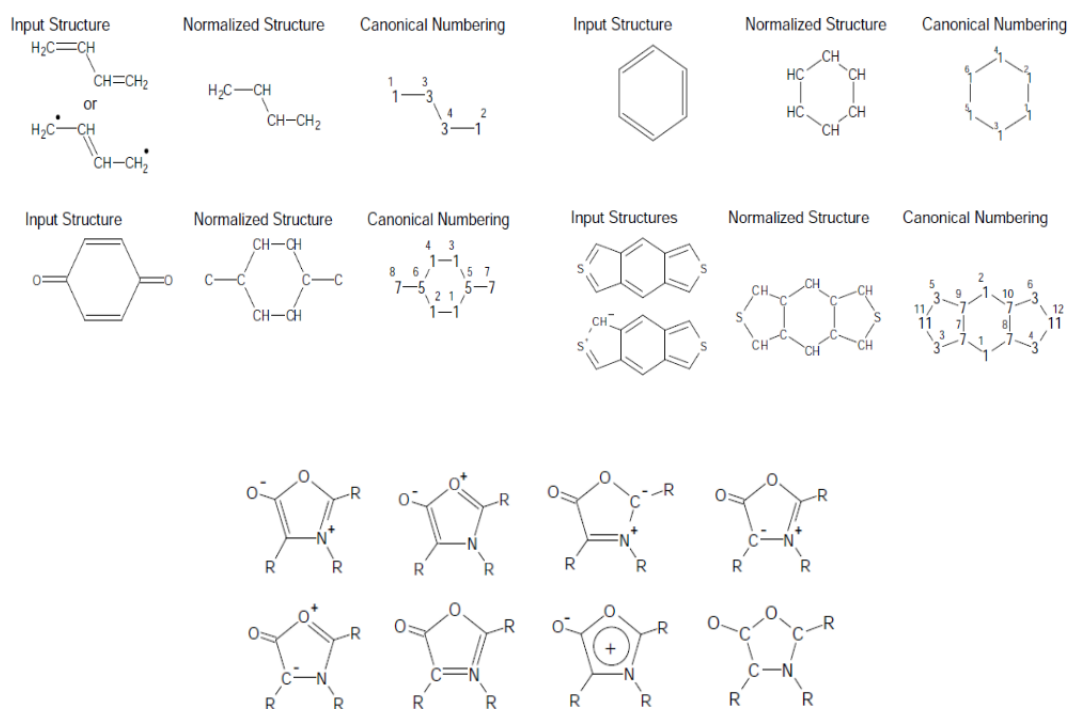


Figure 6.1: Normalization Process. Each one molecule is represented as starting input, normalized and canonicalized structure (top); different tautomeric representations with the last being the normalized one (bottom).^[#]

Hydrogen atoms are required for normalization step and thus unambiguous H atoms description is needed in order to obtain a reliable InChI.

6.2.1.1.1 Additional Normalization Rules InChI applies 5 additional normalization rules to an input structure. 1-3 eliminates structure drawing conventions to avoid interferences with later steps; step 4 adjusts variable protonation and step 5 performs tautomerism discovery. The additional normalization steps are:

1. Disconnect salts
2. Disconnect metals
3. Eliminate radicals
4. Variable protonation processing
5. Charges and mobile H processing

6.2.1.1.1.1 Step 1. Disconnect salts Salts could be represented either as connected or disconnected; InChI always treats salts as disconnect fragments. Salts are defined in this way:

M-X or Y-M-X

where M is the metal atom and HX, HY are acids. In connected salts, metals has only single bonds and no H-atoms connected.

6.2.1.1.1.2 Step 2. Disconnect metals Metal atoms are disconnected in the main layer and their charges if possible are moved to the metal atom. Step 2 is applied to F, Cl, Br, I, At, O, S, Se, Te, N, P, As and B atom types.

6.2.1.1.1.3 Step 3. Eliminate radicals Beyond radical, this step performs conversion of aromatic bonds to alternating single and double bonds (via radical elimination).

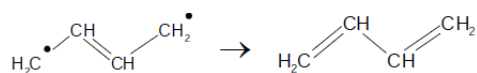


Figure 6.2: Radical Elimination Process.[#]

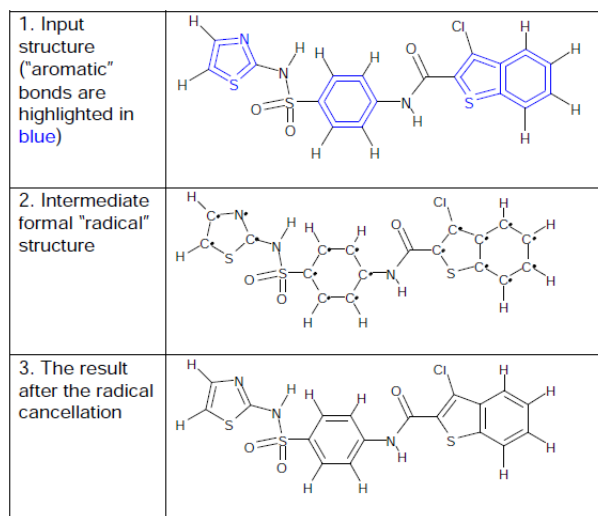


Figure 6.3: Aromatic Bonds Conversion. Aromatic systems are converted into alternating single-double bond systems.^[#]

6.2.1.1.1.4 Step 4. Variable protonation processing (charges and mobile H) This step allows for representation of substances with variable or unknown protonation states; the existence of charges (+1/-1) on non-metal atoms are requested to start this step. This step is so composed:

1. Protons Removal from Charged Heteroatoms; these protons are stored in a proton (charge) layer.
2. Protons Removal from Neutral Heteroatoms; until total charge is positive, proton removal procedure is performed, requesting in some cases a *Hard Proton Removal* (this is the case of some protonated atoms that are concealed by alternating bond conventions).
3. Protons Addition to Reduce Negative Charge; if a negative global charge is present due to previous steps, protons are added to minimize charge.

6.2.1.1.1.5 Step 5. Charges and mobile H Processing Atoms with hydrogen exchange possibilities are grouped as *mobile H group*.

1. *Simple Tautomerism Detection*

Main layer is the same for any arrangement of mobile hydrogen atoms: this is achieved by mobile H removal and labeling of H-donor and H-receptor atoms. These H-atoms are identified using H-transfer tautomerism rules, listed in figure 6.4 and explained in figures 6.5, 6.6. using guanine.

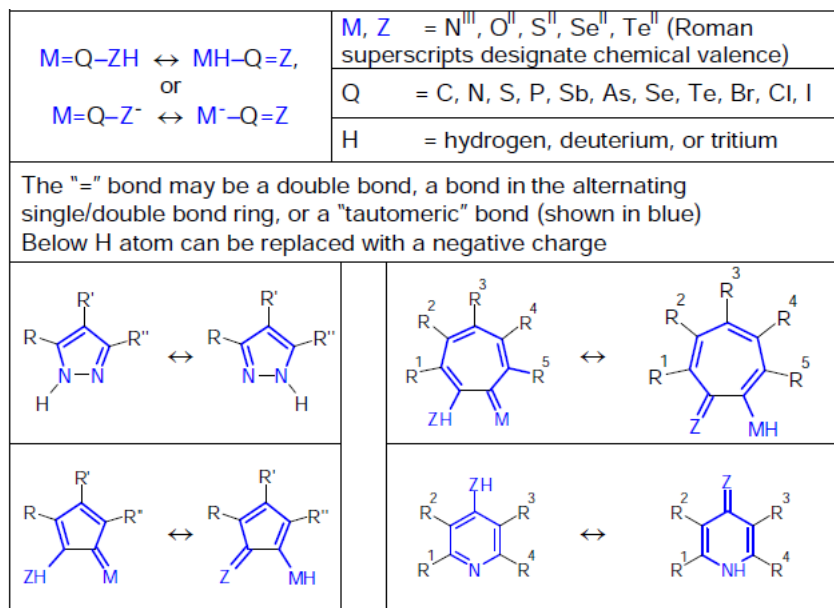


Figure 6.4: H-Transfer Tautomerism Possibilities. These rules allow to identify H-donor and H-acceptor atoms.[#]

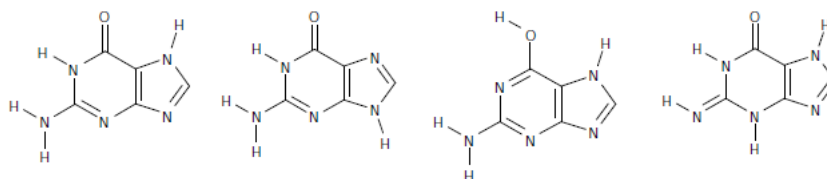


Figure 6.5: Guanine Possible Tautomers(not all).[#]

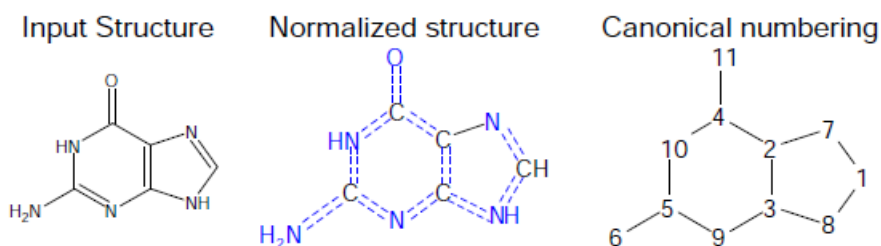


Figure 6.6: Guanine Normalization and Canonical Numbering. Donor/Acceptors of H-bond and changeable bonds are highlighted in blue.[#]

Guanine InChI, considering FixedH layer, is:

InChI=1/C5H5N5O/c6-5-9-3-2(4(11)10-5)7-1-8-3/h1H,(H4,6,7,8,9,10,11)/f/h8,10H,6H2

while the annotated version is:

```
InChI=
{version}1
/{formula}C5H5N5O
/c{connections}6-5-9-3-2(4(11)10-5)7-1-8-3
/h{H_atoms}1H,(H4,6,7,8,9,10,11)
/f{fixed_H:formula}
/h{fixed_H:H_fixed}8,10H,6H2
```

where:

```
/h{H_atoms}1H,(H4,6,7,8,9,10,11): 1 H on atom number1 , 4 H atoms
are shared by atoms 6,7,8,9,10, and 11
/h{fixed_H:H_fixed}8,10H,6H2: 2 H on atom 6, 1H on atom 8, 1 H on
atom 10
/f{fixed_H:formula} empty because fixed H layer and Main layer share
the same chemical formula.
```

This example illustrates important features of InChI:

- Ignoring fixed H layer (/f) leads to equivalent InChIs for different guanine tautomeric forms
- Including fixed H layer allows for specific guanine tautomers definition

This behaviour of fixed-H InChI is widely used in the redundancy elimination process, as explained in chapter 7.

2. Moveable Positive Charges Detection

N-atoms positive charges are treated as moveable through bonds between these atoms; the same for phosphorus atoms. Atoms able to exchange positive charges are grouped into a mobile charge group.

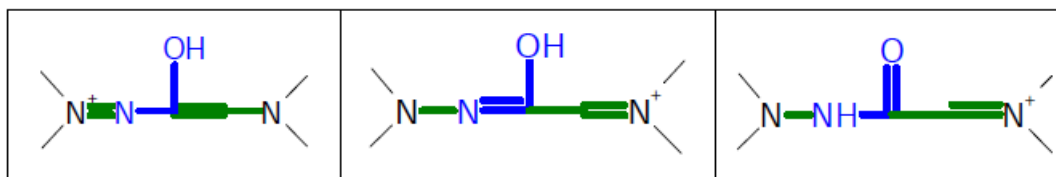


Figure 6.7: Mobile Positive Charge Detection. Positive charge moving creates a tautomeric pattern (blue). Bonds changed by charge movement are highlighted in green. The three structures share the same identifier.^[†]

6.2.1.2 Canonicalization

Canonicalization generates drawing-independent atom-labels. In absence of stereochemistry, the canonicalization process uses a modified algorithm for dealing with layers^[35]. The stereochemical canonicalization is instead based on

mapping of non-stereochemical canonical numbering, using EC, in order to find the smallest internal representation of the stereo layer, keeping previous layers unchanged.

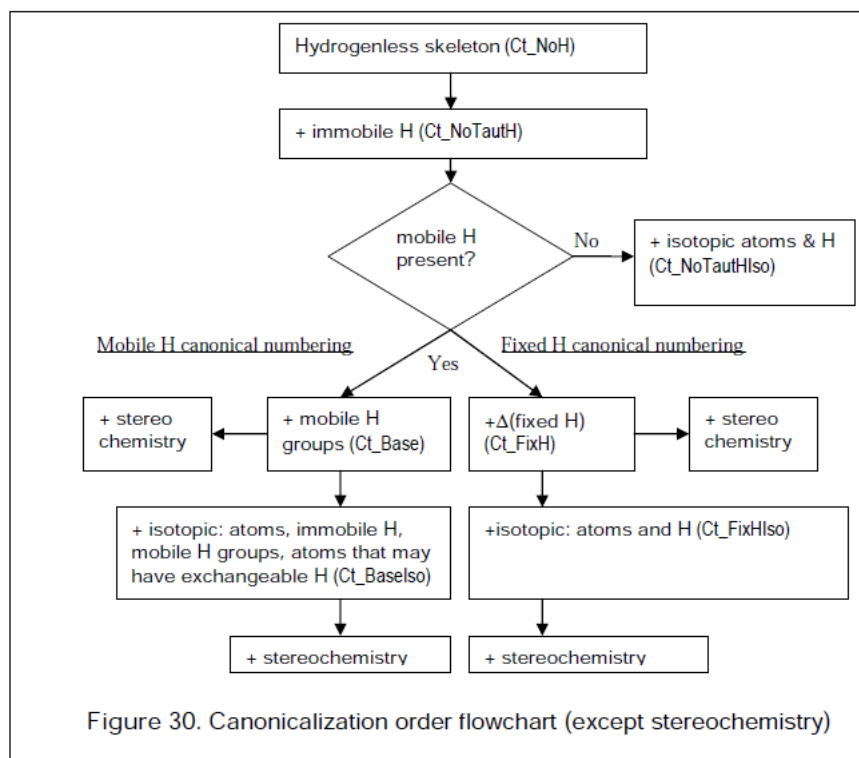


Figure 6.8: Canonicalization Process(stereochemistry is excluded).[#]

6.2.1.3 Serialization

Serialization is the last InChI generation phase; all labels generated by canonicalization are gathered and converted into InChI string.

6.2.2 InChI Layers Type

InChI may be composed of up to five distinct varieties of layers, for five different class of structural information. Chemical formula and connections layers are generated starting from only connectivity information.

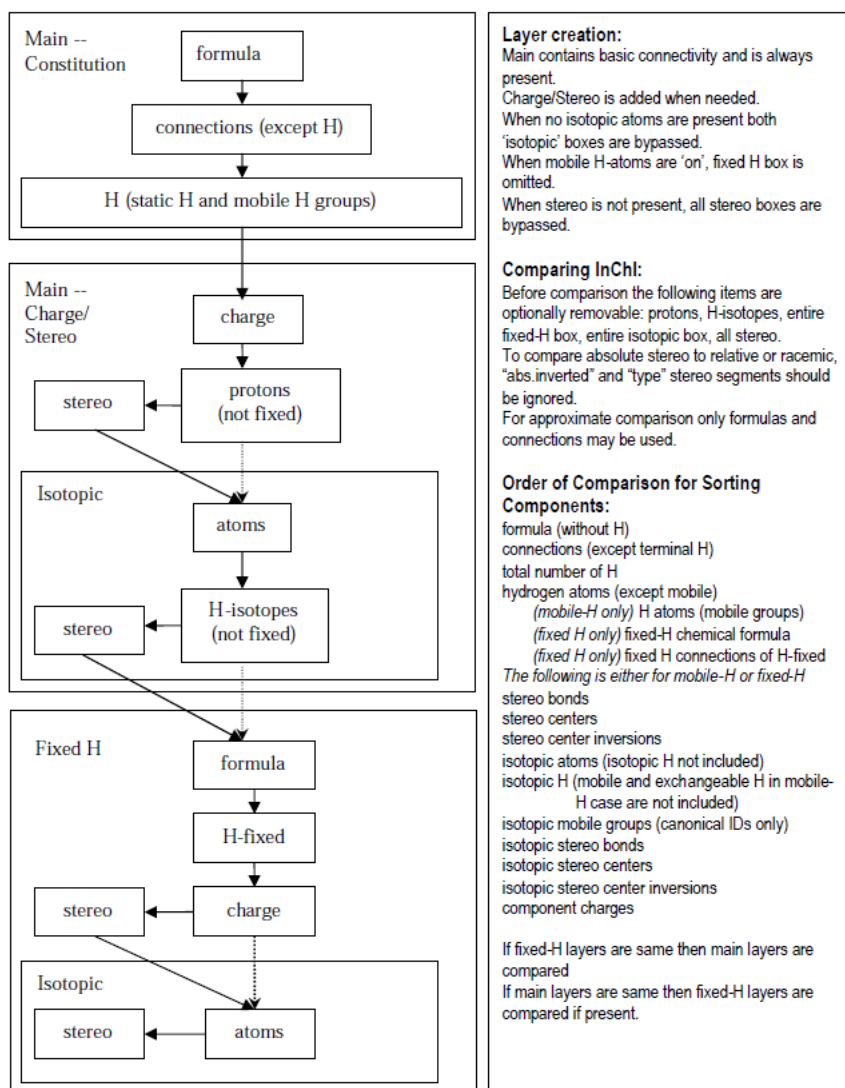


Figure 6.9: InChI Layer Generation Flowchart.[#]

```

{InChI version}
1. Main Layer (M):
/{formula}
/c{connections}
/h{H_atoms}
2. Charge Layer
/q{charge}
/p{protons}
3. Stereo Layer
/b{stereo:dbond}
/t{stereo:sp3}
/m{stereo:sp3:inverted}
/s{stereo:type (1=abs, 2=rel, 3=rac)}
4. Isotopic Layer (MI):
/i{isotopic:atoms}*
/h{isotopic:exchangeable H}
/b{isotopic:stereo:dbond}
/t{isotopic:stereo:sp3}
/m{isotopic:stereo:sp3:inverted}
/s{isotopic:stereo:type (1=abs, 2=rel, 3=rac)}
5. Fixed H Layer (F):
/f{fixed_H:formula}*
/h{fixed_H:H_fixed}
/q{fixed_H:charge}
/b{fixed_H:stereo:dbond}
/t{fixed_H:stereo:sp3}
/m{fixed_H:stereo:sp3:inverted}
/s{fixed_H:stereo:type (1=abs, 2=rel, 3=rac)}
(6.) Fixed/Isotopic Combination (FI)
/i{fixed_H:isotopic:atoms}*
/b{fixed_H:isotopic:stereo:dbond}
/t{fixed_H:isotopic:stereo:sp3}
/m{fixed_H:isotopic:stereo:sp3:inverted}
/s{fixed_H:isotopic:stereo:type (1=abs, 2=rel, 3=rac)}
/o{transposition}

```

Figure 6.10: InChI Layers.[#]

Layers are illustrated below with an example from *"User's Guide: IUPAC International Chemical Identifier (InChI) Program"*[34]. The example molecule is the isotopically substituted (S)-Glutamic acid.

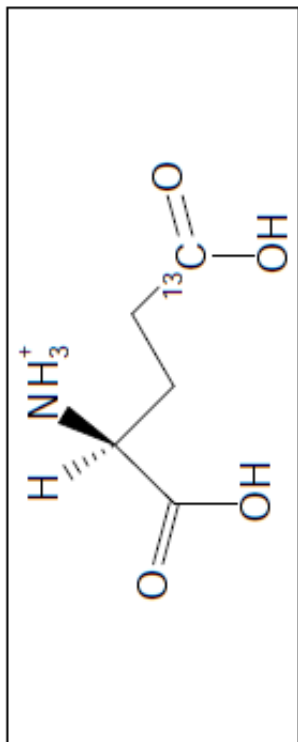


Figure 6.11: S-Glutamic Acid Non-Standard InChI.[#]

The complete InChI for isotopically substituted S-Glutamic Acid is:

InChI=1/C5H9NO4/c6-3(5(9)10)1-2-4(7)8/h3H,1-2,6H2,(H,7,8)(H,9,10)/p+1/t3-/m0/s1/i4+1/fC5H10NO4/h6-7,9H/q+1

Referring to Figure 6.4 layers are so composed:

1. Main Layer:
 - (a) *Chemical Formula* (the conventional Hill-sorted elemental formula):
/C5H9NO4
 - (b) *Connections* (lists of bonds divided into three sublayers):
 - i. all bonds except ones to non-bridging H-atoms: **/c6-3(5(9)10)1-2-4(7)8**
 - ii. bonds of all immobile H-atoms: **/h1-2H2,3H,6H2,**
 - iii. location of mobile H-atoms: **(H,7,8)(H,9,10)**
2. Charge Layer: represents net charge; it is independent of other layers and when omitted indicates that the charge is not specified. Charge layer appears in two sublayers:
 - (a) Component Charge: single components charges
 - (b) Protons: number of H^+ removed from/added to the compound in order to represent it regardless to its degree of protonation (to make same components with variable protonation, like aminoacids, identical): **/p+1**
3. Stereochemical Layer: it is composed of two sublayers where the first one is independent of the second, but not vice-versa.
 - (a) Double Bond sp^2 (Z/E) Stereo. This stereo configuration is represented in 2-dimensional drawings.
 - (b) Tetrahedral sp^3 Stereo and Allenes. It is represented using wedge/hatch (out/in) bonds. Relative sp^3 stereochemistry is represented before absolute stereochemistry: **/t3/m0/s1**
4. Isotopic Layer: it describes isotopically labeled atoms but mobile isotopic hydrogen atoms are listed separately: **/i4+1**
5. Fixed-H Layer: detects mobile H atoms and define their bounded atoms. It accounts for any needed changes in earlier layers:
 - (a) **/fC5H10NO4**
 - (b) **/h6-7H,9H**
 - (c) **/q+1**
 - (d) **/tm**
 - (e) **/m0**
 - (f) **/s1**
 - (g) **/im**

Referring to isotopically substituted S-Glutamic Acid:

InChI=1/C5H9NO4/c6-3(5(9)10)1-2-4(7)8/h3H,1-2,6H2,(H,7,8)(H,9,10)/p+1/t3-/m0/s1/i4+1/fC5H10NO4/h6-7,9H/q+1

the annotated InChI string is so composed:

```
InChI=
{version}1
/{formula}C5H9NO4
/c{connections}6-3(5(9)10)1-2-4(7)8
/h{H_atoms}3H,1-2,6H2,(H,7,8)(H,9,10)
/p{protons}+1
/t{stereo:sp3}3-
/m{stereo:sp3:inverted}0
/s{stereo:type (1=abs, 2=rel, 3=rac)}1
/i{isotopic:atoms}4+1
/f{fixed_H:formula}C5H10NO4
/h{fixed_H:H_fixed}6-7,9H
/q{fixed_H:charge}+1
```

Generally InChI layers organization look like this:

Main Layer

```
/{formula}
/c{connections}
/h{H_atoms}
```

Charge layer

```
/q{charge}
/p{protons}
```

Stereo layer

```
/b{stereo:dbond}
/t{stereo:sp3}
/m{stereo:sp3:inverted}
/s{stereo:type (1=abs, 2=rel, 3=rac)}
```

Isotopic Layer

```
/i{isotopic:atoms}
/h{isotopic:exchangeable_H}
/b{isotopic:stereo:dbond}
/t{isotopic:stereo:sp3}
/m{isotopic:stereo:sp3:inverted}
/s{isotopic:stereo:type (1=abs, 2=rel, 3=rac)}
```

Fixed H layer

```
/f{fixed_H:formula}
/h{fixed_H:H_fixed}
/q{fixed_H:charge}
/b{fixed_H:stereo:dbond}
/t{fixed_H:stereo:sp3}
/m{fixed_H:stereo:sp3:inverted}
/s{fixed_H:stereo:type (1=abs, 2=rel, 3=rac)}
```

Fixed H layer (isotopic part)

```
/i{fixed_H:isotopic:atoms}
/b{fixed_H:isotopic:stereo:dbond}
/t{fixed_H:isotopic:stereo:sp3}
/m{fixed_H:isotopic:stereo:sp3:inverted}
```

```
/s{fixed_H:isotopic:stereo:type (1=abs, 2=rel, 3=rac)}
/o{transposition}
```

Some particular features of InChI are:

- Main layer is always generated while other layers could appear or not, depending on presence/absence of associated information.
- Layer contents depend on prior layers.
- Charge layer accounts for overall charge, thus it is independent of others.
- Protons layer refers to the entire structure.

6.3 SMILES and InChI used implementations

6.3.1 SMILES Implementation

SMILES (see paragraph 5.2.2.3) translates extended connection tables into alphanumeric strings. Since the canonicalization algorithm is public, the commercial toolkit [<http://www.daylight.com>] was not used; free and open source SMILES implementations are widely used. During this work, were generated and compared SMILES strings of 65 commercially available chemical catalogues; input structure format was SDF (see 5.3.6.2), for a total amount of about 42 million compounds. Used SMILES implementation are:

- *OpenBabel Canonical SMILES* [<http://openbabel.org/>]
- *CACTVS uSMILES* and *CACTVS HashStrings* *E_HASHSY* and *E_HASHY* [<http://www.xemistry.com/>]

Since they stores almost all structural information details of InChI, the two CACTVS hash strings were selected for comparison study. Particularly *E_HASHSY* string can distinguish between different tautomers, stereoisomer, isotopes and ionic states; it exhibits a very high performance rate in redundancy identification and so, it represents a good comparison test especially for InChIs.

Name	Taut-sensible	Stereo-sensible	Charge-sensible
E_HASHY	✗	✗	✓
E_HASHSY	✗	✓	✓

Table 6.1: CACTVS Hash Strings

6.3.2 InChI Implementation

In this work was used the official *IUPAC InChI-Generator version 1.03* [<http://www.inchi-trust.org/downloads/>] to convert CTs into string notations. Chemically speaking two tautomers of the same molecule are different entities, but in chemical catalogues usually only one tautomer is represented.

Considering that in real bio-pharmaceutical space tautomers could behave differently and thus they are distinct entities, in the same way in cheminformatics studies is required to treat each one tautomer of the same compound as a single compound; for example *N π -Histamine* and *N τ -Histamine* exhibit different binding affinities toward *H₂* receptor. Since standard InChI does not account for tautomerism detection (see 6.2.1.1.1.5, Step 5), a custom InChI string was generated and used to detect redundant molecular structures while distinguishing different tautomers of the same molecule; not doing so leads to lacks in redundancy identification and elimination. All InChIs generated in this work are *fixedH-InChIs* (tautomer-sensible); this allows for correct identification of duplicate molecules, considering tautomers as different entities. Since *fixedH-InChIs* vary along tautomers, it allows for 100% in-silico chemical libraries redundancy elimination.

6.4 Datasets

The whole study was performed using two datasets:

- *MMsINC*[®] ver. 1.1: a 4 millions non-redundant cheminformatics database [36]
- a dataset of 42 millions of chemical entries from 65 different vendors commercial catalogues

Such a large chemical space ensures for the highest chemical heterogeneity the highest level of statistical significance of data analysis.

6.5 InChI and SMILES Strings Reproducibility Assessment

In order to compare the redundancy identification capabilities for both SMILES and InChI notations, the stability and reproducibility assessment of both strings is mandatory. The ideal scenario is that InChI/SMILES do not change for the same molecule, even if recovered from different connection tables. This issue was investigated with an in-house developed *Python* script[37]; the tool iterates CTs reshuffling for each molecule, changing only the atoms numbering and not bonds type and order, ensuring no-changes in molecular topology; for each one reshuffling cycle, InChI and SMILES are re-generated and stored. Atoms numbering reshuffling is performed randomly *n* times and the bond connections in the new CTable bond block are rebuilt respecting the original connectivity matrix. The canonicalization algorithms of both InChI and SMILES generators, ideally must invariantly label the CT atoms, obtaining a unique sequential atom numbering regardless of drawing or numbering order. Once the iteration process ends, *n* InChIs and *n* SMILES strings are compared. Totally were performed 100 x 3.96 (396) millions random permutations for the first dataset (*MMsINC*[®] v.1.1) and 10 x 42 (420) millions in the second dataset. The reshuffling steps did not produce InChI/SMILES strings modification (fluctuation) for the same molecule, thus assessing that the canonicalization step works properly and independently of atoms numbering or drawing order.

```

1 import sys, os
2 import random
3 def reorder(dizionario):
4     n_keys = dizionario.keys()
5     # print n_keys
6     random.shuffle( n_keys )
7     # print n_keys
8     return n_keys
9 atoms = {}
10 bonds = {}
11 r = 1
12 f = open(sys.argv[1], "r")
13 a = 0
14 for l in f:
15     if r > 4 and len(l.strip()) > 25:
16         a = r - 4
17         # print l.rstrip()
18         atoms[a] = l.rstrip()
19         r += 1
20 f.close()
21 f = open(sys.argv[1], "r")
22 r = 1
23 for l in f:
24     if r > 4 and len(l.strip()) < 25 and "M" not in l[0:2]:
25         # print l.rstrip()
26         a1 = l[3:].strip()
27         a2 = l[3:6].strip()
28         bonds[ " ".join([a1, a2]) ] = l.rstrip()
29         r += 1
30 f.close()
31 new_atoms = reorder( atoms )
32 #header
33 f = open(sys.argv[1], "r")
34 r = 1
35 for l in f:
36     if r <= 4:
37         # print l.rstrip()
38         r += 1
39 for atom in new_atoms:
40     # print atoms[atom]
41     idx = {}
42 for atom in atoms:
43     old_index = str(atoms.keys().index(atom) + 1)
44     new_index = str(new_atoms.index(atom) + 1)
45     idx[ old_index ] = new_index
46 #print bonds.keys()
47 for bond in bonds:
48     a1 = bond.split()[0].strip()
49     a2 = bond.split()[1].strip()
50     new_a1 = idx[ a1 ]
51     new_a2 = idx[ a2 ]
52     prev_string = bonds[bond]
53
54     at1 = prev_string[:3].strip()
55     at2 = prev_string[3:6].strip()
56     print " *(3 - len(new_a1)) + new_a1 + " *(3 - len(new_a2)) + new_a2 + prev_string[6:]
57 print "M END"
58

```

Figure 6.12: Python CTs Reshuffling Code

6.6 Redundancy Identification Efficency

Once assessed InChI/SMILES stability and reproducibility, the redundancy identification rate was investigated, in order to verify if the two strings could be used as unique and unambiguous molecular descriptors. Strings comparison was performed using in-house developed BASH [<http://www.gnu.org/software/bash/>] and Python code, according to this scheme:

1. Alphanumeric strings sorting: in the sequentially ordered resulting list, redundant strings must stand one behind the other;
2. Unique strings identification (corresponding to unique molecular entries)
3. Molecular topography rebuilding starting from CTs of unique entries

The above mentioned workflow was applied independently on both InChI/SMILES strings lists, using two distinct pipelines:

- Single vendor catalogues, processed one by one in an isolated manner
- All 65 vendors catalogues, processed as an all-vendors merged catalogue (all against all)

Crossed comparisons between InChI and SMILES strings for each one molecule, allowed for discovery of redundancy identification errors.

6.6.1 Single Catalogue Redundancy Analysis

The identified redundancy ratio is very similar using both InChI and SMILES strings as summarized in Figure 6.13. In almost all cases, the absolute difference between the % of redundancy identified by InChI and SMILES is less than 0.1%; differences higher than 1% were detected only for 4 vendors with a maximum value of 8.3%.

This high comparable InChIs/SMILES behaviour is probably due in minor part to the fact that vendors perform an internal *fast and rough* redundancy cleaning or filtering of their catalogues, so duplicates can be partially removed depending on size and constitution of the dataset. A 1000 entries file is easily cleanable to obtain a unique molecular set using InChI, SMILES or any other kind of molecular descriptor, because of the limited chemical space considered; on the other hand a 10 millions entries dataset needs a more robust redundancy pruning pipeline.

The principal reason for this comparable strings performances is the limited vendors catalogues size; the represented chemical space is quite small and strings collision probability is smaller than the one occurs in a large size datasets. In fact large libraries are characterized by a high scaffolds diversity, bringing to difficult in distinguishing very similar but not identical chemical compounds.

Looking at absolute redundancy for each vendor's catalogue it's clear the lack of uniqueness because of absolute redundancies detected by InChIs are around 50% or more for 10 vendors out of 65 and because of almost all minor size vendors (< 1 million molecules) are characterized by a significant redundancy too; the average InChIs-detected redundancy over all vendors is around 14%, with only 4 vendors having 0% redundancy in their catalogues. Non-redundant datasets are very small (30-2000 compounds) and thus are characterized by a limited structural complexity. Moreover, only 10 vendors out of 65 are characterized by an absolute redundancy < 1%. *CACTVS HASHSY* string has the closest InChIs performance in redundancy identification: it detects for all 65 catalogues, a difference in % *identified redundancy* less than 1%, and for 21 catalogues out of 65 a 0% difference, with respect to InChIs. *HASHSY* string is the one with the best performance among all non-InChI strings but hash strings (see 5.3.8.3) cause the complete loss of any chemical/structural information: the input structure file is translated into an alphanumeric string, used as a storage address key for data entries, so no chemical structure reconstruction is possible. Input structure chemical information restoring is possible, with different success rates, only starting from SMILES and InChI strings. *CACTVS HASHY* string is not accurate as *HASHSY* because the first one is not stereosensitive; 30 catalogues out of 65 have a *HASHY/InChI* absolute redundancy difference higher than 1% and 7 higher than 10% with a maximum of about 39%. In frequent cases the SMILES- or HASH-driven duplicates identification pipelines recognize a different number of duplicates compare to InChI. There are several explanation for that: in some cases it is the translation-program assigns the same string to different structures (e.g stereoisomers) and, consequently, duplicates are over-estimated. In other cases, the translation-code assigns different strings to the same structure (usually due to errors in the CT codification or some bugs in canonicalization algorithms) and consequently, duplicates are underestimated. Considering this results, it is not surprising that find a commercial database with 0% redundancy is very difficult. Three common weaknesses occur during databases construction/maintain processes:

- database owner did not perform any redundancy cleaning process before releasing its catalogue
- database owner performed a redundancy elimination process using a non-InChi-driven approach
- database owner did not perform redundancy elimination process for any updated version of the catalogue
- database owner did not perform at all any redundancy elimination process.

Concluding, the developed cleaning pipeline can guarantee the identification and removal of duplicates independently from libraries size and represented chemical space. Concerning single vendors' catalogues thus, it's required to perform an InChI-idriven redundancy analysis prior to any chemoinformatics study. It's fundamental that the used redundancy identification pipeline/descriptor, behaves in a linear manner ensuring stable duplicates discovery capabilities, on both limited or extended chemical space libraries. In order to verify the performances stability, the above described InChI-driven redundancy elimination process was applied to a larger chemical space obtained by joining all vendors'

catalogues into a global one. Three common weaknesses occur during databases construction/maintain processes:

- database owner did not perform any redundancy cleaning process before releasing its catalogue
- database owner performed a redundancy elimination process using a non-InChi-driven approach
- database owner did not perform redundancy elimination process for any updated version of the catalogue
- database owner did not perform at all any redundancy elimination process

It is fundamental to assess that the used redundancy identification pipeline or descriptor(s) has a linear performance maintaining stable duplicates discovery capabilities, on both limited or extended chemical space libraries. In order to verify the performances stability, the above described InChI-driven redundancy elimination process, was applied to an extended chemical space obtained by joining all vendors' catalogues in one. In figure 6.14 and 6.15 is reported the redundancy distribution for each one vendor, computed according to InChI, SMILES and Hash strings.

Vendor	Input structures number	Computed unique InChI	% Redundancy detected by InChI	Computed unique OpenBabel CANSMILES	Computed unique CACTVS HASHSY	Computed unique CACTVS HASHSY	% Redundancy detected by CANSMILES	Computed unique CACTVS uSMILES	% Redundancy detected by uSMILES	Ir - CS %	Ir - uSR %	% Redundancy by HASHSY	Ir - HASHSY %	%Redundancy by HASHSY	Ir - HASHSY %
PUBCHEM	23660297	20243932	14.44	20242329	20233588	17280972	14.45	20302390	14.19	0.01	0.25	14.4829006661	0.04	26.96	12.52
ZINC8	22742630	11088923	51.24	11078671	11052128	7930419	51.29	11138666	51.13	0.05	0.11	51.4034744442	0.16	65.13	13.89
ENAMINE	19830161	18761859	5.39	18760869	18759083	18694627	5.39	18784884	5.27	0.00	0.12	5.4012570061	0.01	5.73	0.34
AKOS	55806899	4075585	26.97	4075813	4067501	3979176	26.97	4207000	24.62	0.00	2.35	27.1148470828	0.14	28.70	1.73
EMOLECULES	4303078	4282564	0.48	4282428	4280663	4281685	0.48	4283487	0.46	0.00	0.02	0.5209061978	0.04	1.43	0.95
DOORSY	4188559	3613409	13.73	3614197	3612680	3581308	13.71	3618511	13.61	0.02	0.12	13.74888573039	0.02	14.50	0.77
SCIENTIFIC EXCHANGE	1206181	1185754	1.68	1185831	1185650	1185417	1.67	1185750	1.68	0.01	0.00	1.688863682	0.01	1.71	0.03
LIFE CHEMICALS	1082962	539230	50.21	539719	538935	537576	50.16	539612	50.17	0.05	0.04	50.2350059663	0.03	50.36	0.15
SPECS	1050905	229907	78.12	229505	226850	222301	78.16	227763	78.33	0.04	0.20	78.4043276885	0.28	78.85	0.72
CHEM BRIDGE	968324	664266	32.79	664214	662930	658221	32.79	664343	32.78	0.01	0.01	32.9238185049	0.14	33.40	0.61
VITAMIN	858816	848613	0.84	848625	848598	848423	0.84	848649	0.84	0.00	0.00	0.8434055919	0.00	0.86	0.02
OTAVA	847205	740338	12.59	740645	740390	739794	12.58	740619	12.58	0.01	0.01	12.6079284235	0.02	12.68	0.09
CHEMBL NTD	659251	330778	49.83	330634	329729	325840	49.85	331349	49.74	0.02	0.09	49.9843003651	0.16	50.57	0.75
CHEMBL	635926	597957	5.97	597326	597163	566715	6.07	599169	5.78	0.10	0.19	6.095520548	0.12	10.88	4.91
CHEBI8	607556	593723	2.28	593402	593391	563114	2.33	593985	2.23	0.05	0.04	2.3314723252	0.05	7.31	5.04
ALINDA	521249	266419	48.89	265394	263676	262377	49.08	265163	49.13	0.20	0.24	49.4145792126	0.53	49.66	0.78
PRINCETON	492149	482116	2.04	482148	482099	481661	2.03	483071	1.84	0.01	0.19	2.0420644967	0.00	2.13	0.09
IBS	439784	425992	3.14	426030	425989	424543	3.13	426064	3.12	0.01	0.02	3.1367671402	0.00	3.47	0.33
ASINEX	429738	393229	8.50	393168	393102	390485	8.51	396965	7.63	0.01	0.87	8.5251944208	0.03	9.13	0.64
ZELINSKY INSTITUTE	379786	367840	3.15	367840	367817	367586	3.15	368351	3.01	0.00	0.13	3.1515116408	0.01	3.21	0.07
TIMTEC	275867	212401	23.01	212536	212215	211033	22.96	213747	22.52	0.05	0.49	23.0734375623	0.07	23.50	0.50
PHARMEKS	274136	254272	7.25	254299	254237	250578	7.24	254340	7.22	0.01	0.02	7.2588058482	0.01	8.59	1.35
CHEMICALBLOCK	253807	146643	42.22	147213	146523	125140	42.00	156882	38.19	0.22	4.03	42.2699137534	0.05	50.69	8.47
NATURAL_CMPDS	185930	150867	19.13	149596	149495	118149	19.54	165045	11.23	0.41	7.88	19.5960845479	0.47	36.46	17.33
SYNTHON	110060	95552	49.53	95595	95127	94794	49.50	95317	49.75	0.04	0.21	49.9209666668	0.39	50.22	0.69
MAYBRIDGE	109403	65291	40.32	65803	65056	61185	39.85	66719	39.02	0.47	1.31	40.5354514958	0.21	44.07	3.75
ARVI	90044	79820	11.35	79820	79787	79416	11.31	80335	10.78	0.05	0.57	11.3910976856	0.04	11.80	0.45
ZINC NATURAL PRODUCTS	89425	89065	0.40	89078	89018	60681	0.39	89207	0.24	0.01	0.16	0.4551299972	0.05	32.14	31.74
KEYORGANICS	58710	52800	6.89	52801	52798	52723	6.89	52960	6.61	0.00	0.28	6.8982542761	0.00	7.03	0.14
MATRIX	45323	44309	2.24	44310	44309	43816	2.24	44392	2.05	0.00	0.18	2.2372746729	0.00	3.33	1.09
APOLLO SCIENTIFIC	42062	39420	6.28	39427	39419	39042	6.26	39485	6.13	0.02	0.15	6.2835813799	0.00	7.18	0.90
LIPID_MAPS	40613	22738	44.01	22934	22543	19446	43.53	22596	44.36	0.48	0.35	44.4931425888	0.48	52.12	8.11

Figure 6.14: Redundancy Identification Statistics over 65 vendors' catalogues. Redundancy rates are computed by: I(InChI), uS(cactvs unique smiles), CS(open babel Canonical smiles), HASHISY, HASHSY. (r=redundancy).

Vendor	Input structures number	Computed unique InChIs	% Redundancy detected by InChI	Computed unique OpenBabel CANSMILES	Computed unique HSHSY	Computed unique CACTVS HASHSY	% Redundancy detected by CANSMILES	Computed unique CACTVS uSMILES	% Redundancy detected by uSMILES	Ir - CSr %	Ir - uSr %	% Redundancy by HASHSY	Ir - HASHSYr %	%Redundancy by HASHSY	Ir - HASHSYr %	
BINDING DB	34444	34002	1.28	34000	33066	34048	1.29	34048	1.15	0.01	0.13	1.2832423644	0.00	0.00	2.72	
FLUOROCHEM	28607	28023	2.04	28027	27681	28080	2.03	28080	1.84	0.01	0.20	2.0449540322	0.00	0.00	3.24	
ANALITYCON	28356	26909	5.10	26897	26897	26938	5.15	26938	5.00	0.04	0.10	5.1452955283	0.04	0.04	7.61	
ARONIS	23742	23718	0.10	23718	23718	23718	0.10	23718	0.09	0.00	0.01	0.1010866818	0.00	0.00	0.15	
TOSLAB	23542	21143	10.19	21138	21069	21141	10.19	21141	10.20	0.00	0.01	10.2115368278	0.02	0.02	10.50	
MDPI	23530	21919	6.85	21918	21905	21974	6.85	21974	6.61	0.00	0.23	6.9060773481	0.06	0.06	8.27	
ASISCHEM	18514	18501	0.07	18501	18501	18501	0.07	18501	0.07	0.00	0.00	0.070211133	0.00	0.00	0.00	
PEAKDALE	15555	15369	1.20	15368	15357	15399	1.20	15399	1.00	0.01	0.19	1.2021857923	0.01	0.01	1.27	
OAKWOOD	13397	13240	1.17	13241	13215	13251	1.16	13251	1.09	0.01	0.08	1.1719041576	0.00	0.00	0.93	
DRUGBANK3	13383	6773	49.39	6711	6699	6253	49.85	6759	49.50	0.01	0.10	49.8439587536	0.55	0.55	53.28	
HUMAN_MATABOLOME_DB	7896	7835	0.65	7834	7827	7840	0.66	7840	0.58	0.01	0.06	0.6593963987	0.01	0.01	4.55	
COMBIBLOCK	7199	7114	1.18	7114	7114	7129	1.18	7129	0.97	0.00	0.21	1.1807195444	0.00	0.00	1.28	
BIOBLOCKS	5926	2625	55.70	2625	2598	2625	55.70	2625	55.70	0.00	0.00	55.703676704	0.00	0.00	56.83	
ACBLOCKS	5795	5789	0.10	5789	5789	5794	0.10	5794	0.02	0.00	0.09	0.1035375324	0.00	0.00	27.11	
CHEMIK	3977	3767	5.28	3767	3755	3779	5.28	3779	4.98	0.00	0.30	5.280362082	0.00	0.00	5.58	
SINOVA	3932	3886	1.17	3886	3843	3892	1.17	3892	1.02	0.00	0.15	1.1698890977	0.00	0.00	2.26	
LABOTEST	3165	3088	2.43	3090	3053	3091	2.37	3091	2.34	0.06	0.09	2.4960505529	0.06	0.06	3.54	
MAGELLAN BIOSCIENCE	2325	1936	16.73	1925	1921	1933	17.20	1933	16.86	0.47	0.13	17.376344086	0.65	0.65	23.74	
TRACTUSCHEM	2164	2041	5.68	2042	2022	2067	5.64	2067	4.48	0.05	1.20	5.6839186691	0.00	0.00	6.56	
MICROSOURCE	2160	2008	7.04	2001	1961	2003	7.36	2003	7.27	0.32	0.23	7.3611111111	0.32	0.32	9.21	
INFARMATIK	2113	1381	34.64	1380	1375	1381	34.69	1381	34.64	0.05	0.00	34.6900141978	0.05	0.05	34.93	
EXCLUSIVE_CHEMISTRY	2056	2056	0.00	2056	2056	2056	0.00	2056	0.00	0.00	0.00	0	0.00	0.00	0.00	
SEQUOIA RESEARCH	2041	1979	3.04	1978	1856	1978	3.09	1978	3.09	0.05	0.05	3.1357177854	0.10	0.10	9.06	
CHEMIVATE	1108	1108	0.00	1108	1108	1108	0.00	1108	0.00	0.00	0.00	0	0.00	0.00	6.03	
ADESIS	960	953	0.73	953	904	953	0.73	953	0.73	0.00	0.00	0.7291666667	0.00	0.00	0.00	
AFICHEMFARM	948	920	2.95	920	911	926	2.95	926	2.32	0.00	0.63	2.9535964979	0.00	0.00	3.90	
CDD	921	430	53.31	428	420	428	53.53	428	53.53	0.22	0.22	53.5287730727	0.22	0.22	54.40	
SMPDB	847	842	0.59	842	804	843	0.59	843	0.47	0.00	0.12	0.5903187721	0.00	0.00	5.08	
CMLD BOSTON UNIV	791	782	1.14	782	625	783	1.14	783	1.01	0.00	0.13	1.1378002528	0.00	0.00	4.49	
MICROCOMICHEM	435	435	0.00	435	435	435	0.00	435	0.00	0.00	0.00	0	0.00	0.00	19.85	
SINOFLUORO	279	267	4.30	267	265	267	4.30	267	4.30	0.00	0.00	4.3010752688	0.00	0.00	5.02	
OMEGACHEM	123	122	0.81	122	74	122	0.81	122	0.81	0.00	0.00	0.8130081301	0.00	0.00	39.02	
ITHEMBA_INTERMEDIATE	30	30	0.00	30	30	30	0.00	30	0.00	0.00	0.00	0	0.00	0.00	0.00	
TOTAL	280251	260991	14.90	260988	71133443	64653459	261275	261275	7.10	7.10	7.241475053	13.46	13.46	13.46	13.46	
AVERAGE																

Figure 6.15: Redundancy Identification Statistics (continue from fig. 6.14).

6.6.2 Pan-catalogues Redundancy Analysis

This analysis was performed applying the same pipeline described above with an input dataset obtained by joining all 65 vendors' catalogues into a single SD file, containing more than 92 millions compounds. The global redundancy was investigated through three different but sequential approaches:

- Dataset 1: 92355744 (92.35 Millions) compounds derived by all 65 commercial catalogues merging. Single catalogues did not undergo any redundancy elimination process. On Dataset 1 has been generated InChIs, uSMILES and CANSMILES for each molecular structure, and the above described redundancy cleaning pipeline was applied.
- Dataset 2: 71483311 (71.48 Millions) compounds derived by merging all 65 starting catalogues, previously cleaned one by one using InChI. The redundancy elimination was not applied in a cross manner (all against all) but in an isolated manner for each catalogue. As for 1, on Dataset 2 has been generated InChIs, uSMILES and CANSMILES for each molecular structure, performing the same redundancy elimination pipeline as above.
- Dataset 3: 42191880 (42.19 Millions) non-redundant compounds, corresponding to the InChi-based cleaned version of Dataset 2; thus dataset 3 correspond to a cross-cleaned version (all catalogues against all) of Dataset 2. Dataset 3 is redundancy free, so there are 42191880 unique structures in it. Also in this case has been generated InChIs, uSMILES and CANSMILES and performed the redundancy cleaning pipeline one more time.

Results are collected in Table 6.2. The InChi performance is excellent, stable and independent of the input structural dataset composition, size and previously performed cleaning processes. Considering all sets 1-3, InChI-driven duplicates identification performance is highly efficient and reaches the 100% duplicates discovery in every case. Both uSMILES and CANSMILES have lower duplicate identification performance compared to InChIs in all three cases, even if CANSMILES reaches a better result respect uSMILES. In particular, in set 1 and 2, the number of unique compounds reported by SMILES-driven analysis is higher compared to the one reported using the corresponding InChI. Curiously, starting from dataset 3, both uSMILES-driven and CANSMILES-driven cleaning processes reduce the overall number of unique structures compared to the InChI-driven process: this means that SMILES are not able to distinguish almost similar (but not identical) compounds. In other words both kind of SMILES aren't able to perform like InChI even if the input dataset is redundancy-free. In addition, the strings generation time is about 3.3 hours for 42 million compounds set in case of InChIs and about 72 hours for the same set in case of SMILES, on a Linux OS, Intel Core i7 cpu machine.

DATASET 1 (42073344 really unique molecules)			
#input mols	#Unique InChIs	#Unique uSMILES	#Unique CANSMILES
92355744	42073344	46889084	42198186
DATASET 2 (42073344 really unique molecules)			
#input mols	#Unique InChIs	#Unique uSMILES	#Unique CANSMILES
71206303	42073344	46763085	42182482
DATASET 2 (42073344 really unique molecules)			
#input mols	#Unique InChIs	#Unique uSMILES	#Unique CANSMILES
42073344	42073344	42061272	42102661

Table 6.2: InChI vs SMILES: Redundancy Identification Efficiency

6.6.3 SMILES Errors Identification

A deeper analysis inside SMILES failures, highlighted the cause for lacks in real duplicate molecules identification. All molecules processed by InChI and SMILES generator, were sorted two times: the first one on the basis of their SMILES strings, and the second one using their InChIs strings. Sorted lists entries have been classified according to this scheme:

- Duplicate according to SMILES definition
- Duplicate according to InChI definition
- Unique according to SMILES definition
- Unique according to InChI definition

This classification allows for identification of True and False unique/duplicates entries, so there are 4 possible molecular redundancy classes:

- True duplicate
- False duplicate
- True unique
- False unique

In more details, since each one molecule is defined by a SMILES string and an InChI string, was developed a parsing process of the two strings sorted lists. First InChI and SMILES strings are grouped according to their redundancy: all identical strings constitute a group, which reflects structural information of the same identical chemical entity. The parser performs a pairwise comparison of entries number, group by group, on both InChI and SMILES grouped lists. In an ideal case, all InChIs of the same group representing the same chemical entity need to be identical; as well as, all SMILES of the same group representing the same chemical entity need to be identical. If a molecule is 3 times represented

in the dataset, the corresponding InChI-grouped and SMILES-grouped sub-lists must contain 3 string notations each one; more than 3 means that different entities has been assigned the same identifier, while less than 3 means that the same structure has been assigned different strings. All chemical structures corresponding to groups in which the number of entries for an InChI strings group differs from the one for SMILES strings groups were extracted and analyzed: strings collision happend for structures that are the same based on their InChI strings but not on the basis of smiles strings. Looking inside these structures the InChI superior performance was assessed one more time.

Concerning SMILES, 3 kind of errors were identified:

1. False duplicate identification: really different compounds, identified as duplicate
2. False unique identification: really duplicate compounds, identified as unique
3. Wrong identification of different features: really different compounds with wrong *different-features* identification.

Type 1 errors are principally related to molecules with stereogenic N atoms (aziridines, bicyclic compounds with bridgehead N atoms, oximes, imines, di-AZO groups and cis/trans double bonds containing N) and stereogenic P atoms; in all this cases SMILES is not able to distinguish different stereoisomers.

Type 1 errors are less frequent, but highlights how SMILES coding is not unambiguous because of its dependence on CT atoms numbering.

Type 3 errors, is caused by a wrong identification of difference-features between two molecules; thus really different structures are identified as so, but the difference-feature is not correctly recognized. Also in this case are involved molecules with stereogenic C atoms and R=NR1 groups.

Fig. 6.13 summarizes some examples of false duplicate/unique compounds detected by SMILES. In some cases, such as for *B* or *D*, the errors are probably due to lacks of some chemical rules in SMILES implementation, such as the *NH2-axial/equatorial* recognition in the case of *B* or an undetectable phosphorus stereochemistry in *D*. In other cases, such as for *E*, errors seem to derive from the canonicalization algorithm or in other early steps of the string generation.

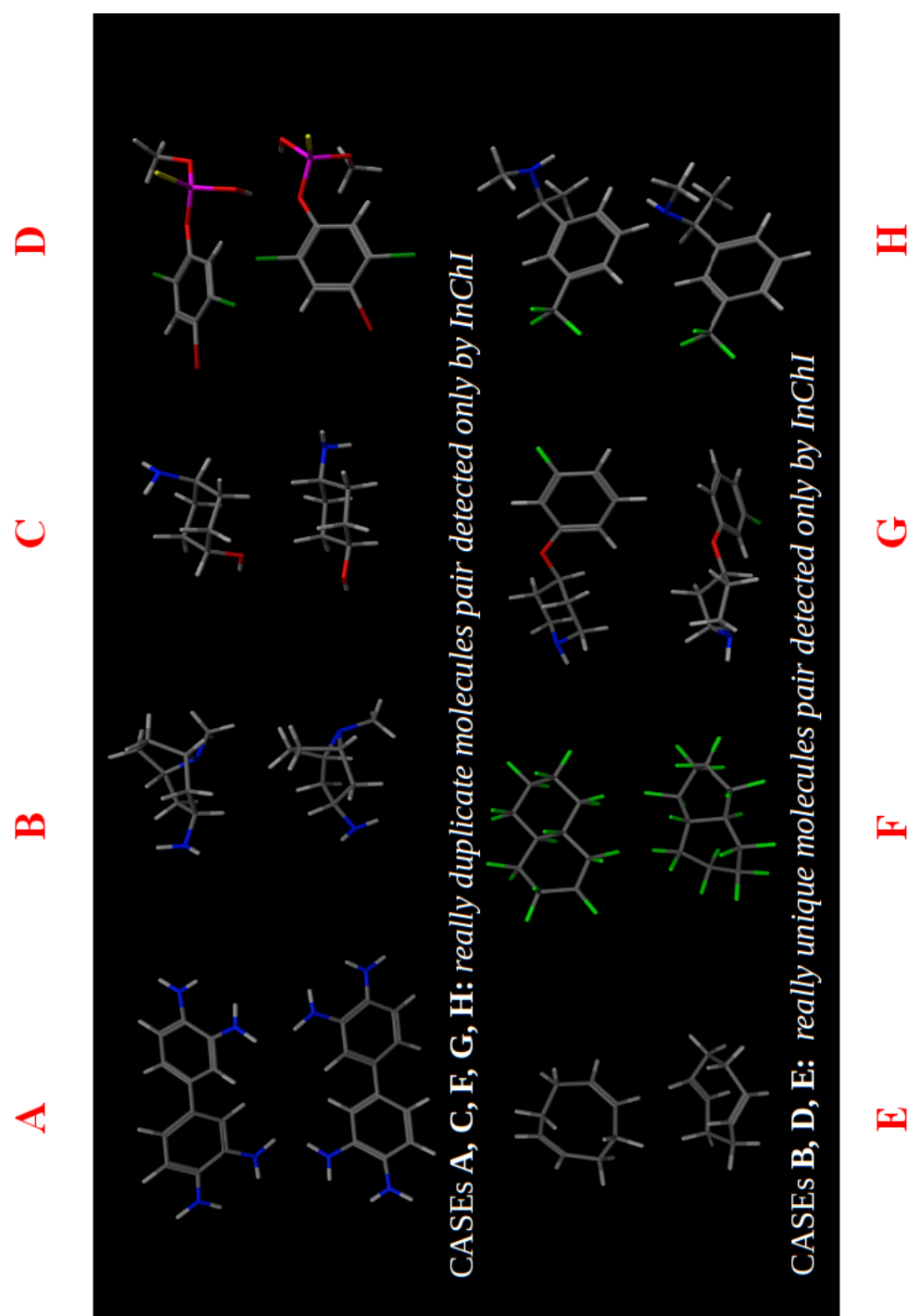


Figure 6.16: SMILES Errors in Redundancy Detection. Example of wrong duplicate/unique molecular pair detected. In each couple the two molecules represent an error case in SMILES description of chemical structure.

Mol	U/D	InChI	BabelCanSmiles	CACTVSusmiles
A	D	InChI=1/C12H14N4c13-9-3-1-7(5-11(9)15)8-2-4-10(14)12(16)6-8/h1-6H,13-16H2	Nc1cccc(cc1N)c1cccc(c(c1)N)N	Nc1=C(N)C=C(C=C1)C2=CC=C(C(N)C=C2)N
A	D	InChI=1/C12H14N4c13-9-3-1-7(5-11(9)15)8-2-4-10(14)12(16)6-8/h1-6H,13-16H2	Nc1cccc(cc1N)c1cccc(c(c1)N)N	Nc1=CC=C(C=C1N)C2=CC=C(N)C(=C2)N
B	U	InChI=1/C8H16N2/c1-10-7-2-3-8(10)5-6(9)4-7/h6-8H,2-5,9H2,1H3/t6-,7-,8-/m0/s1	N[C@@H]1C[C@@H]2CC[C@@H](C1)N2C	CN1[C@H]2CC[C@H]1CC(N)C2
B	U	InChI=1/C8H16N2/c1-10-7-2-3-8(10)5-6(9)4-7/h6-8H,2-5,9H2,1H3/t7-,8-/m0/s1	N[C@@H]1C[C@@H]2CC[C@@H](C1)N2C	CN1[C@H]2CC[C@H]1CC(N)C2
C	D	InChI=1/C6H13NO/c7-5-1-3-6(8)4-2-5/h5-6,8H,1-4,7H2/t5-,6-	N[C@@H]1CC[C@H](CC1)O	N[C@H]1CC[C@H](O)CC1
C	D	InChI=1/C6H13NO/c7-5-1-3-6(8)4-2-5/h5-6,8H,1-4,7H2/t5-,6-	N[C@@H]1CC[C@H](CC1)O	N[C@@H]1CC[C@H](O)CC1
D	U	InChI=1/C7H6BrCl2O3PS/c1-12-14(11,15)13-7-3-5(9)4(8)2-6(7)10/h2-3H,1H3,(H,11,15)/t14-/m0/s1/f/h11H	COP(=S)(Oc1cc(C)c(cc1C)Br)O	CO[P@](O)(=S)OC1=C(C)C=C(Br)C(=C1)Cl
D	U	InChI=1/C7H6BrCl2O3PS/c1-12-14(11,15)13-7-3-5(9)4(8)2-6(7)10/h2-3H,1H3,(H,11,15)/t14-/m1/s1/f/h11H	COP(=S)(Oc1cc(C)c(cc1C)Br)O	CO[P@](O)(=S)OC1=C(C)C=C(Br)C(=C1)Cl
E	U	InChI=1/C8H12/c1-2-4-6-8-7-5-3-1/m1-2,7-8H,3-6H2/t2-1-,8-7-	C1CC=CCCC=C1	C1C\ C=C\ CC\C=C-1
E	U	InChI=1/C8H12/c1-2-4-6-8-7-5-3-1/m1-2,7-8H,3-6H2/t2-1+,8-7+	C1CC=CCCC=C1	C1\ C=C\ CC\C=C-1

Figure 6.17: InChI/BabelCanSmiles/CACTVSusmiles strings comparison. Some strings have been splitted for space reasons; strings' subsets differences are marked in red. U/D state refers to molecule Unique/Duplicate real condition.

The next figures (6.18, 6.19, 6.20, 6.21, 6.22, 6.23), describes in detail specific examples of SMILES strings failures; the represented chemical structures are not correctly nor completely coded into string notation.

Aziridines Stereochemistry (2 enantiomers)

Different InChIs

Same SMILES

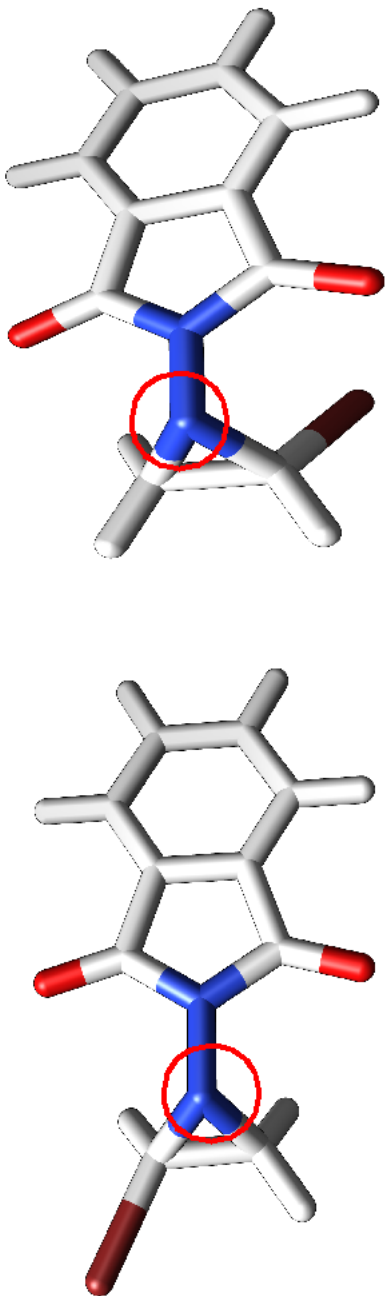

InChI=1/C10H7BrN2O2/c11-8-5-12(8)13-9(14)6-3-1-2-4-7(6)10(13)15/h1-4,8H,5H2/t8-12+/m0/s1
InChI=1/C10H7BrN2O2/c11-8-5-12(8)13-9(14)6-3-1-2-4-7(6)10(13)15/h1-4,8H,5H2/t8-12-/m0/s1
Br[C@@H]1CN1N2C(=O)C3=C(C=CC=C3)C2=O
Br[C@H]1CN1N2C(=O)C3=C(C=CC=C3)C2=O

Figure 6.19: SMILES String failure-1. Aziridines stereoisomers are not distinguished by SMILES.

Stereogenic C: same enantiomer (S)

Same InChI Different SMILES

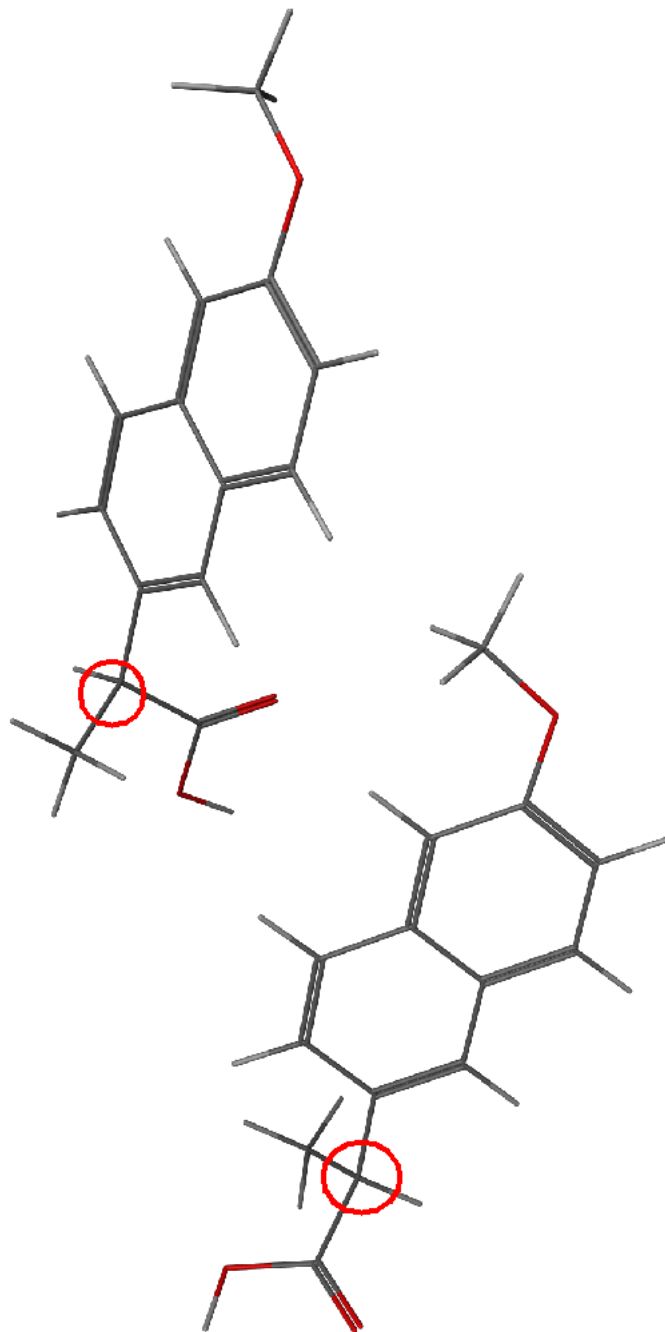


Figure 6.20: SMILES String failure-2. Some stereogenic C centres are not recognized by SMILES.

DiAZO group Cis/Trans Isomerism

Different InChI

Same SMILES

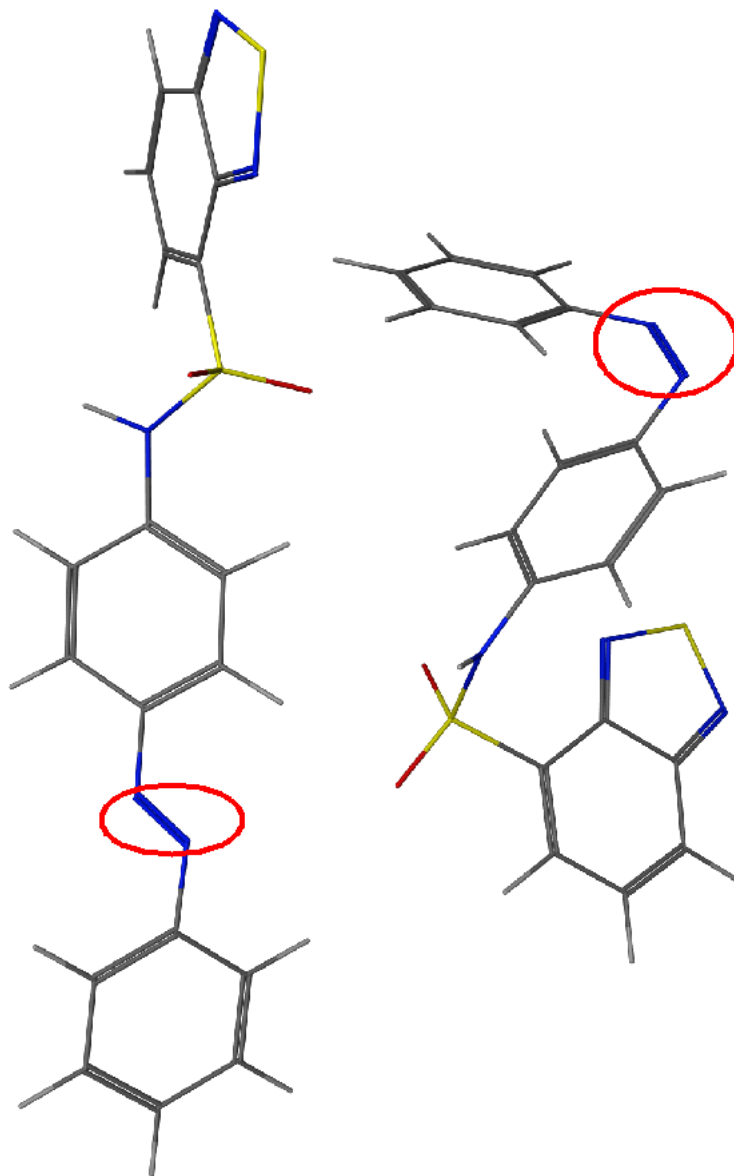


Figure 6.21: SMILES String failure-3. DiAZO group Cis/Trans Isomers are not recognized by SMILES.

Imine group (C=N) E/Z Isomers

Different InChI

Same SMILES

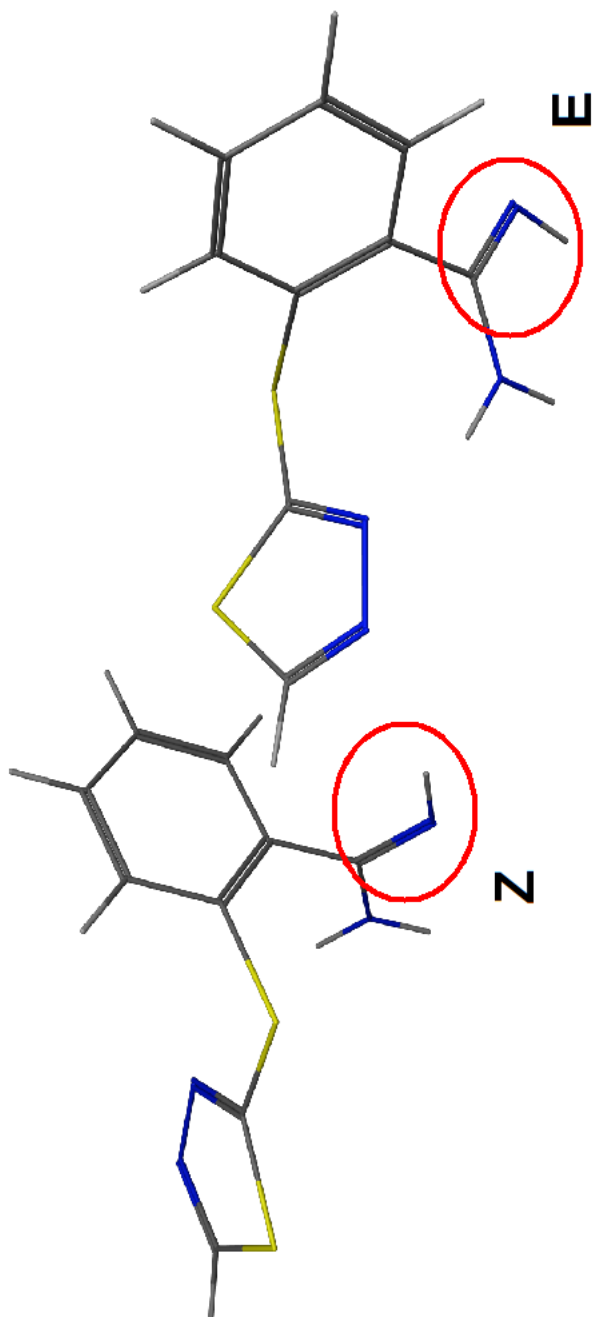
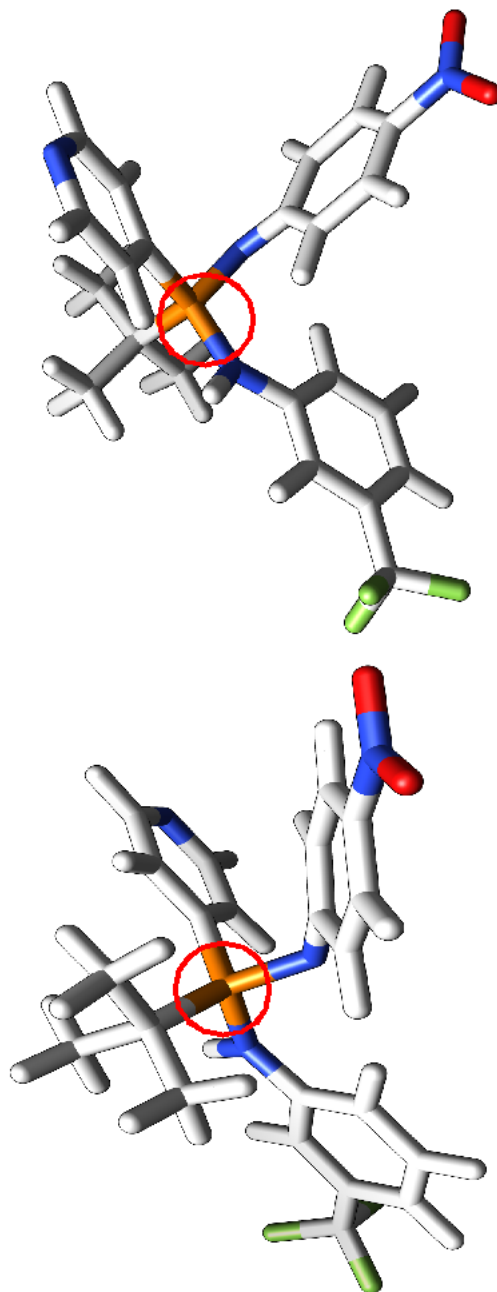


Figure 6.22: SMILES String failure-4. Imine group E/Z Isomers are not recognized by SMILES.

Stereogenic P: Phosphonium ions (2 enantiomers)

Different InChI

Same SMILES



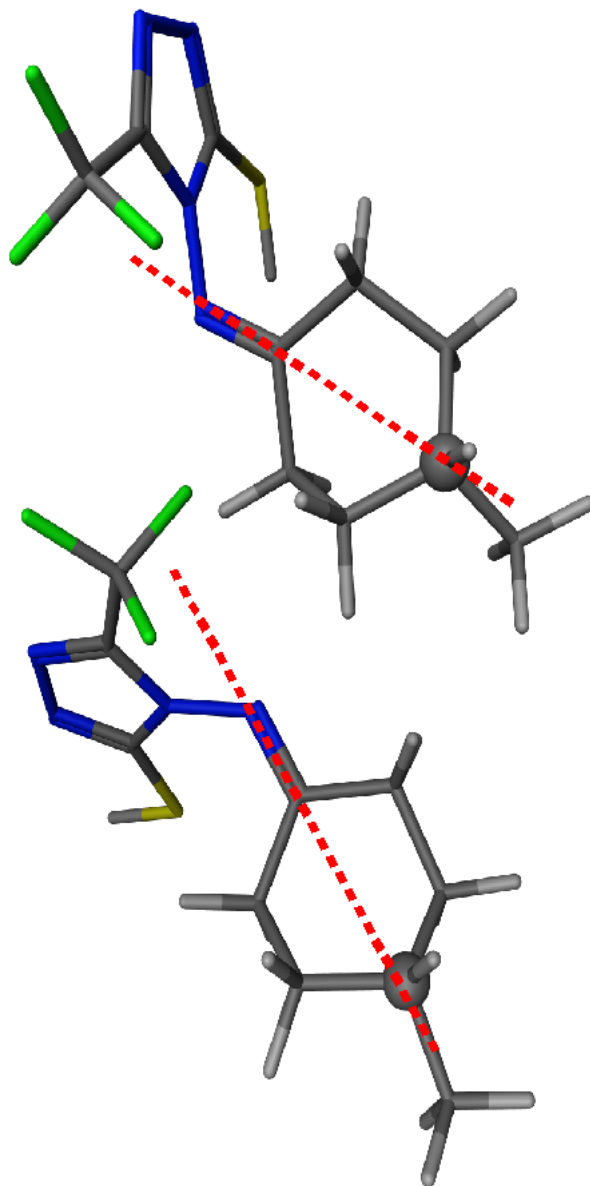
Same situation for Phosphonates, Phosphinates and Phosphinoxides

Figure 6.23: SMILES String failure-5. Stereogenic P centres are not recognized by SMILES.

Axial chirality - Stereogenic axes

Different InChI

Same SMILES



AlkenylCyclicAlkane

Figure 6.24: SMILES String failure-6. Axial Chirality is not recognized by SMILES.

6.7 MMsDusty Pipeline

The previously described redundancy elimination pipeline is available as *MMsDusty* [<http://mms.dsfarm.unipd.it/MMsDusty>], a web-oriented tool that performs an InChI-based redundancy cleaning starting from any SDF file. MMsDusty is organized in three main operating levels:

1. InChI-based orthography check: all structures collected into the SDF input file are re-written in the corresponding InChI strings in order to identify and remove all structures that are not correctly transformable in InChIs;
2. 3D-structure generation step: all structures deriving from step 1 are transformed in 3D structures using the MOE builder tool[38]. All 3D-structures are processed by the *sdwash* utility implemented by MOE: implicit hydrogen are converted into explicit hydrogen atoms and salts and ionic forms are transformed in the corresponding neutral species;
3. InChI-based redundancy-cleaning step: starting from the SDF file coming from step 2, the final InChI-based redundancy-cleaning is performed.

6.8 Conclusion

The analysis of different approaches for redundancy detection and cleaning performed in this work, demonstrates that InChI- and SMILES-based redundancy cleaning processes present different performance, and, in particular, that the InChI-driven pipeline comes very close to 100% redundancy identification and elimination especially when starting from very large and heterogeneous chemical libraries. Concerning strings reproducibility and stability InChI behave in a stable and consistent manner with regards to the sequential atoms numbering of CTs, while in some cases, SMILES generates strings fluctuations depending on CTs atoms numbering. Regarding redundancy identification the two descriptors behave differently. InChI performs linearly and independently on libraries size and chemical heterogeneity, ensuring always 100% identification of duplicate molecules; this is not true for SMILES, which exhibits lacks in redundancy identification depending on libraries size and heterogeneity. Such a pipeline as the one described above, ensures high-quality and non-redundant chemical libraries for chemoinformatics application.

Chapter 7

MMsINC[®]: a large-scale chemoinformatics platform

Contents

7.1	Introduction	91
7.2	Database Creation	92
7.2.1	First Redundancy Washing	93
7.2.2	Step 2: Tautomers Generation	93
7.2.3	Step 3: Ionic States Generation	93
7.2.4	Step 4: Conformer Selection	93
7.2.5	Step 5: Second Redundancy Washing	93
7.2.6	Step 6: Unstable Tautomer Elimination	94
7.2.7	Step 7: Molecular Descriptors calculation	94
7.3	Database Validation: Subsets Analysis	95
7.4	Informatic Structure	96
7.5	Database Querying	96
7.5.1	Identical Structure Search	96
7.5.2	Substructure Search	96
7.5.3	Molecular Scissoring Search	97
7.5.4	Similarity Search	97
7.5.5	PDB-Similarity Search	97
7.5.6	Descriptors Filtering	98
7.6	Results Displaying	98
7.7	Implementation	102
7.8	Conclusion	102

7.1 Introduction

In the post-proteomic era, an important topic is the exploration of proteins pharmacology by ligands chemistry[36]. Related drugs can recognize unrelated

molecular targets leading to side effects and toxicity. This polypharmacology limits bioinformatics attempts to categorize the pharmacological action using protein similarity. A chemo-centric approach thus allows for comparison between targets using ligands chemistry[39]. MMSINC[®] was developed using this chemo-centric approach, integrating chemical structures, properties annotations and specific search functions, in order to create a tool for the analysis of protein pharmacology and toxicology by ligand chemistry. MMSINC[®] is based on a molecular database with richly annotated molecular structures, but its main goals are redundancy absence and high quality of stored chemical entities and data. In fact accuracy in chemical annotation is fundamental to ensure significance to qualitative/quantitative similarity metrics. Structure-and property-based similarities tools allows for establishing chemical relationships inside MMSINC[®] and toward other public databases (PDB, PubChem, Drug-Bank, ZINC, ChemDB, etc.). MMSINC[®] is an input structures source for chemoinformatics and virtual screening applications. MMSINC[®] is accessible through a web interface.

	ZINC	ChemBank	PubChem	ChemDB	MMSINC
Features					
Full download	Y	Y ^a	Y	Y	
Subset download	Y	Y ^a	Y	Y	Y
Size	8M	1.7M	19.6M	5M	4M
Non-redundant (InChI-based)					Y
Chirality	Y	Y		Y	Y
FDA similarities					Y
PDB ligand similarities					Y
Link to PDB		Y	Y		Y
Link to PubChem			N/A	Y	Y
Search					
By exact structure		Y	Y		Y
By similarity		Y	Y	Y	Y
By substructure	Y	Y	Y	Y	Y
By fragment					Y
By molecular descriptors	Y	Y	Y	Y	Y
By assays		Y	Y		

^aOnly for registered users.

Figure 7.1: Public Molecular Databases Comparison.

7.2 Database Creation

MMSINC[®] platform is public and web-based; it was developed through multi-step chemoinformatics pipeline processing of 46 data sources, including commercial vendor catalogues and public repositories (e.g. NCI, <http://cactus.nci.nih.gov>). The first release of the platform was based on about 4 millions unique compounds, obtained by processing of a molecular set of 7.5 million entries. The internal library of MMSINC[®] is non-redundant and with a chemical orthography as accurate as possible. The most probable tautomeric and ionic states at physiological conditions were predicted, with one high-quality stable conformer for each molecular entry.

The platform is now being updated to about 460 millions molecular entries, with one ionic state, up to 5 tautomers and up to 5 high-quality conformers for each molecule.

Database development passes through 7 steps:

1. Step 1: first redundancy washing
2. Step 2: generation of tautomers
3. Step 3: generation of ionic states
4. Step 4: conformer selection
5. Step 5: second redundancy washing
6. Step 6: unstable tautomer elimination
7. Step 7: molecular descriptors computing

7.2.1 First Redundancy Washing

Using the Molecular Operating Environment software suite (MOE, <http://www.chemcomp.com>), were removed all redundant entries using a SMILES-driven process, obtaining 4 millions structures starting from 7.5 millions.

7.2.2 Step 2: Tautomers Generation

Using LigPrep 2.1 tool (<http://www.schrodinger.com>) were generated molecular tautomers for all entries coming from step 1. In order to select and retained only the most stable and favourable tautomers, they were removed in a later phase.

7.2.3 Step 3: Ionic States Generation

The most energetically favourable ionic states were computed at a pH of 7.4, using the *Protonate* tool in MOE. Ionic state prediction is pKa-based and adds 250000 molecular entries.

7.2.4 Step 4: Conformer Selection

The three-dimensional (3D) structures were predicted using Corina 3.4 (<http://www.mol-net.de>) for all entries (ncluding tautomers and ionic states). Generated possible conformers, were obtained by dihedral angles exploration maintaining the molecular topology fixed. The best conformers were selected using an energetic classification.

7.2.5 Step 5: Second Redundancy Washing

SMILES redundancy identification capabilities, as explained in chapter 6, lacks of rules and definitions allowing for complete redundancy removal. So in this step an InChI-driven molecular duplicates identification (as explained in chapter 6) was performed, achieving 0% redundancy in the MMsINC[®] chemical database.

7.2.6 Step 6: Unstable Tautomer Elimination

Unstable tautomers coming from step 2, were energetically sorted using a *MMFF94*[40] forcefield-based criterion applied to 3D structural data generated in Step 4. Once sorted the tautomers, only the up to 5 most stable one of each molecule was retained, adding 1.1 million entries to the data set.

7.2.7 Step 7: Molecular Descriptors calculation

24 molecular properties useful for quantitative structure–activity relationship (QSAR), diversity analysis or combinatorial library design were calculated. Atomic partial charges were computed by *MMFF94* forcefield implemented by MOE; other descriptors were calculated using the MOE tool *QSAR-Descriptor*.

Descriptor category	Descriptor
Physical	MW, Rg, SlogP, logS
Topological	Globularity, Sterimol/B1-4/L
Surface and Volume	ASA/+/-/H, Volume
Pharmacophoric	HB-donor/acceptor, Acid/Basic groups, Chiral centers
Energetic	Potential energy
Drug candidacy	Lipinsky drug like

Figure 7.2: MMSINC[®] Molecular Descriptors.

7.3 Database Validation: Subsets Analysis

Final dataset analysis confirmed the high-quality of chemical structures and data collected in the platform. A as good as possible in-silico description of molecular entities, is a fundamental requirement to ensure high-quality chemoinformatics studies. The analysis divided MMsINC[®] dataset into three pharmaceutically relevant and representative subsets:

- Lipinski[41] Drug-Like Subset:
 - ◇ < 5 hydrogen bond donors
 - ◇ < 10 hydrogen bond acceptors
 - ◇ molecular weight < 500 Da
 - ◇ octanol-water partition coefficient (log P) < 5
- Oprea[42] Lead-Like Subset:
 - ◇ molecular weight 200-350 (optimisation adds ~ 100 Da)
 - ◇ logP 1-3 (optimisation may increase by 1-2 logunits)
 - ◇ single charge (positive charge preferred on secondary or tertiary amine)
- Known Reactive Groups subset (Reactive groups based on the Oprea set):
 - ◇ metals
 - ◇ phospho-
 - ◇ N/O/S-N/O/S single bonds
 - ◇ thiols
 - ◇ azides
 - ◇ esters
 - ◇ Michael' s acceptors

Final dataset analysis highlighted a very good chemical quality of molecular database with 98% of compounds fulfilling Lipinski Drug-Like rules, 91% respecting Oprea Lead-Like rules and 87% being non-reactive.

Subset	Number of Molecules
Full	3.96 Mln
Lipinski Drug-Like	3.89 Mln (98%)
Oprea lead-Like	3.61 Mln (91%)
Reactive Groups free	3.45 Mln (87%)
Unique Fragments	175 k

Table 7.1: MMsINC[®] Chemical Subsets Representation

7.4 Informatic Structure

MMSINC[®] platform is multi-level organized:

- Base level: PostgreSQL containing all data (100 Gbytes):
 - ◊ Parent Molecules: CTs, SMILES, InChI, ID, Molecular Descriptors, Fingerprints
 - ◊ Ionic States: CTs, SMILES, InChI, ID, Molecular Descriptors, Fingerprints
 - ◊ Tautomers: CTs, SMILES, InChI, ID, Molecular Descriptors, Fingerprints
 - ◊ PDB Section: precomputed similarity towards co-crystallized ligands
- Intermediate level: Chemoinformatics Core containing all *Java* (*CDK*, *Chemistry Development Kit*) and *Python* for on-fly computing on molecular entries.
- Top Level: *PHP* web interface for database querying.

7.5 Database Querying

MMSINC[®] allows users to search the database by structural criteria, specifying structures by standard notations (SMILES, InChI, standard molecular formula), by drawing it with the *Java Molecular Editor* (*JME*, by Peter Ertl, <http://www.molinspiration.com/jme>) or by identifying it by its MMsCode. Database could be searched by substructure, similarity or similarity to PDB Ligands, with respect to a query molecule.

7.5.1 Identical Structure Search

Identical structure search performs search for molecules that exactly match query structure. Queries submitted by MMsCode and InChI results in at most one result, since they are unique and unambiguous; SMILES instead is ambiguous, so returns all molecules represented by the query found in MMSINC[®].

7.5.2 Substructure Search

This kind of search allows for identification of molecules containing an atoms-subset (substructure) specified by the query. Substructure could be specified as a SMILES string or as an internal MMsCode. The search process is performed using structural keys, which are bit vectors that indicate with a *1* the presence of a particular structural feature, and with a *0* its absence. MMSINC[®] uses 643-bit structural keys from the PubChem[43] fingerprints. In case of a SMILES-query, MMSINC[®] generates a query structural key dynamically; whereas in case of an MMsCode-query the system fetches the precalculated structural key associated with the identified molecule. Comparison of query key with keys of all molecules in MMSINC[®], allows for identification of all molecules containing all the structural bits defined by the query. This technique is fast but can retrieve false positives, since the key may not completely describe the query

structure. In order to avoid this, the preliminary results are filtered by an exact subgraph containment check using the (CDK) library. If the preliminary key search retrieves too many (more than 30000) molecules to perform the subgraph containment check on-fly, MMsINC[®] only performs the subgraph isomorphism check on the molecules as they are displayed to the user, indicating whether they are false positives.

7.5.3 Molecular Scissoring Search

The molecular scissoring search is an experimental query type based on chemically relevant molecular fragments known as *scaffolds*. The submitted query structure is analyzed to identify all possible scaffolds. The current implementation of the scissoring search can in some rare cases allow the user to select scaffolds that do not exist in the query molecule. Molecular scissoring allows for fast identification of molecules containing particular substructures which are chemically and/or pharmacologically relevant.

7.5.4 Similarity Search

The similarity search retrieves all molecules structurally similar to query molecule. Similarity is measured using the *Tanimoto Similarity Score*[44] on the structural keys describing them. The Tanimoto similarity is the ratio of the number of bits set to 1 in both keys to the number of bits set to 1 in both keys. For two structural keys A, B we have:

$$T(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

To perform this kind of search, beyond the structure is needed a similarity score threshold definition; precalculated structural keys of MMsINC[®] molecules are compared to query structural key, retrieving all molecules with a Tanimoto score greater or equal then the selected threshold. Tanimoto-based similarity searches has been speed up by implementing the technique by Swamidass and Baldi[44]; it allows for bounding the number of 1 in the target structural key required to achieve a similarity score that meets the threshold, considerably reducing the number of molecules for which Tanimoto similarity needs to be pre-calculated.

7.5.5 PDB-Similarity Search

PDB co-crystallized ligands similarity has been precomputed towards all MMsINC[®] molecular structures. This search type starts from a SMILES-query or from a PDB-query. In the first mode, a molecular structure query and a Tanimoto similarity minimum threshold is needed: all the molecules in MMsINC[®] that satisfy the similarity threshold towards the SMILES query, are retrieved. In the second mode, a list of up to five PDB protein identifiers is needed: in this case all MMsINC[®] molecules with the requested similarity towards the PDB ligands list are retrieved. In both cases, the identified PDB ligands are presented with structural diagram and the ligand code. In the ligand report page are summarized all the MMsINC[®] neutral molecules, tautomers, ionic states and FDA approved drugs that are similar to the ligand, with a specified Tanimoto similarity score threshold but >0.70 . The ligand report page also

contains basic information about the ligand, such as its 2D structural diagram, its three-letter code and its name; a table showing all the PDB proteins that interact with this ligand is displayed too.

7.5.6 Descriptors Filtering

In MMSINC[®] is possible to perform searches using one or more molecular descriptors value-range; this allows for filtering of retrieved results.

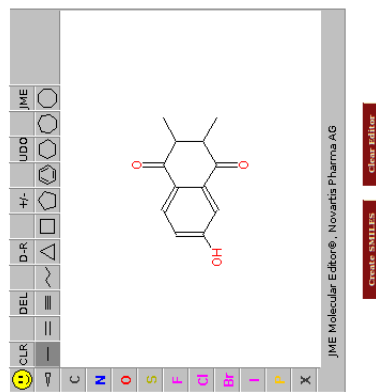
7.6 Results Displaying

Results from any structural query are displayed in pages of up to 20 molecules with their structural diagram and MMsCodes. Search results could be downloaded as SDF, PDB, XYZ format. The report of each one molecules shows basic information about the molecule like the compound type (neutral, tautomer or ionic state), the molecular formula, InChI and SMILES representations, 2D and 3D structural files and images, precalculated descriptors; for neutral molecules all tautomers and ions are listed, while for tautomers and ions the neutral state of the molecule is indicated. Finally, the report has links to the PubChem and ZINC entries for the molecule.

In figures 7.3, 7.4 and 7.5 are showed typical results pages.

MMsINC Search: Structure Search Similarity to PDB ligands

Structure Search



Search

Query Type:

Search using:

Query Data:

Input: SMILES

Molecular Descriptors:

- + Physical Properties
 - Molecular Weight (MW) from to
 - logS from to
 - StopP from to
- Reactive groups
 - Select a value
- + Topological Properties
- + Surface and Volume Properties
- + Pharmacophoric Properties
- + Drug- and Lead-like Properties

Search

Figure 7.3: MMsINC[®] Substructure Search. Descriptors values ranges could be defined before search

Molecule found with a Tanimoto Coefficient superior or equal to **0.85**

Items found 1 - 20 of 520

1 of 26 on this page

Next >>

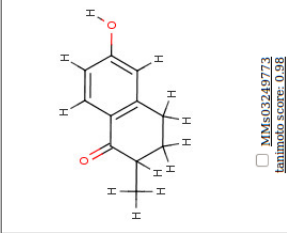
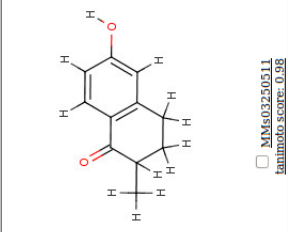
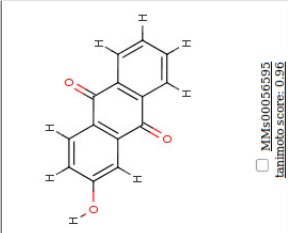
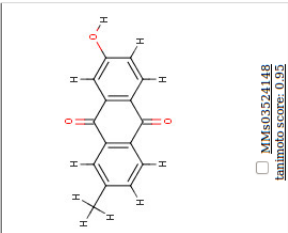
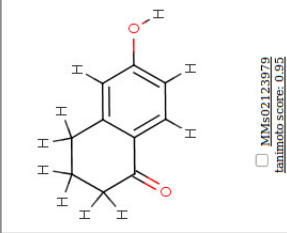
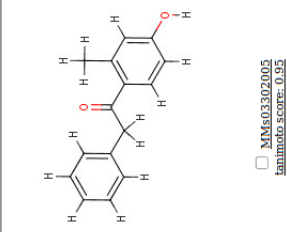
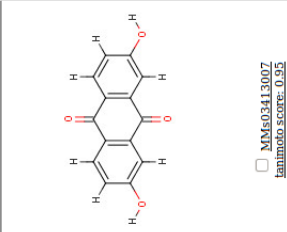
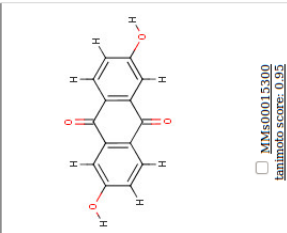
select All	deselect All	add to cart	[cart empty]
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
MMs03249723 tanimoto score: 0.98	MMs03250511 tanimoto score: 0.98	MMs03241448 tanimoto score: 0.95	MMs00015300 tanimoto score: 0.95
			
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
MMs02123979 tanimoto score: 0.95	MMs03302005 tanimoto score: 0.95	MMs03413007 tanimoto score: 0.95	MMs00015300 tanimoto score: 0.95
			

Figure 7.4: MMsINC® Similarity Search: a typical results page with Tanimoto Scores for each one retrieved molecule.

7.7 Implementation

The MMSINC[®] system uses the PostgreSQL RDBMS (<http://www.postgresql.org>) to manage its data, on server running Linux. The system's web application has been developed in PHP, with some components written in Java. MMSINC[®] uses the CDK to perform some of its molecular analyses.

7.8 Conclusion

The MMSINC[®] platform basic aim is to allow for a chemo-centric approach in relating protein pharmacology by ligand chemistry. Chemical structural information and data accuracy, correctness, completeness, and absence of redundancy are primary features. Actually the platform is being updated to about 460 millions of richly annotated high-quality chemical structures and data so represented:

- 42 Mln of parent molecules
- 45 Mln of Tautomers
- 5 Mln of Ionic states
- up to 5 Conformers per molecule

MMSINC[®] platform is integrated with pepMMSMIMIC, a peptidomimetic screening platform, as explained in chapter 8.

Chapter 8

pepMMsMIMIC: a peptidomimetics screening platform

Contents

8.1	Introduction	103
8.2	The pepMMsMIMIC Protocol	103
8.2.1	pepMMsMIMIC workflow	106

8.1 Introduction

Many cellular process are based on Protein–protein interactions, from DNA processing, to cell motility. Protein–protein interactions malfunction or failure could cause cancer or neurodegenerative diseases. Thus, the development of specific drugs able to modulate protein-protein interaction process, requires a good knowledge of how proteins could interact. On the other side peptide drugs have a limited clinical use due to: rapid peptidase-driven degradation, side effects caused by peptides flexibility which allows for multiple target binding, poor absorption because of high molecular weight or transporters lack. Peptidomimetic could bypass these problems because of their chemical structure: despite their small organic molecules structure, are able to incorporate protein surface recognition properties, geometrically and chemically miming aminoacids polymers.

8.2 The pepMMsMIMIC Protocol

pepMMsMIMIC[45] starts from a peptide 3D structure and performs a multi-conformer similarity search among 17 million conformers of available chemicals

collected in the MMsINC[®] database. The multi-conformer library is now being updated to 460 millions structures. Peptidomimetics are searched using a pharmacophore approach starting from 3D structure of any protein–protein or protein/peptide complex; key residues for interaction must be specified. The pharmacophore model allows then for screening of compound libraries. Selected molecules are ranked using two different scoring functions and one consensus scoring approach in order to evaluate electrostatic and shape similarity towards the investigated peptide. pepMMsMIMIC protocol organization is summarized in figure 8.1.

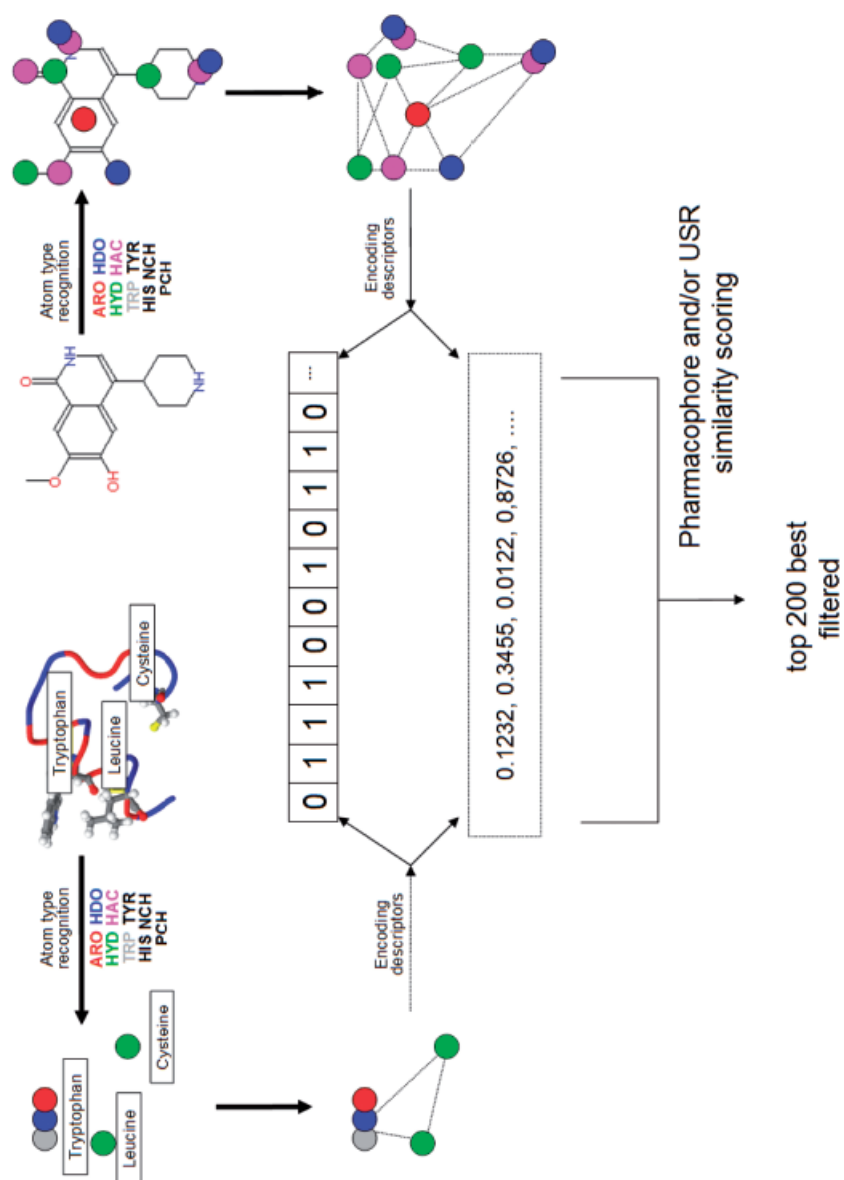


Figure 8.1: pepMMsMIMIC Architecture. Both key residues and compounds from chemical library are coded into a pharmacophoric bitstring. Descriptors similarity analysis allows for peptidomimetics identification.

8.2.1 pepMMsMIMIC workflow

The pepMMsMIMIC workflow is shown in figure 8.2.

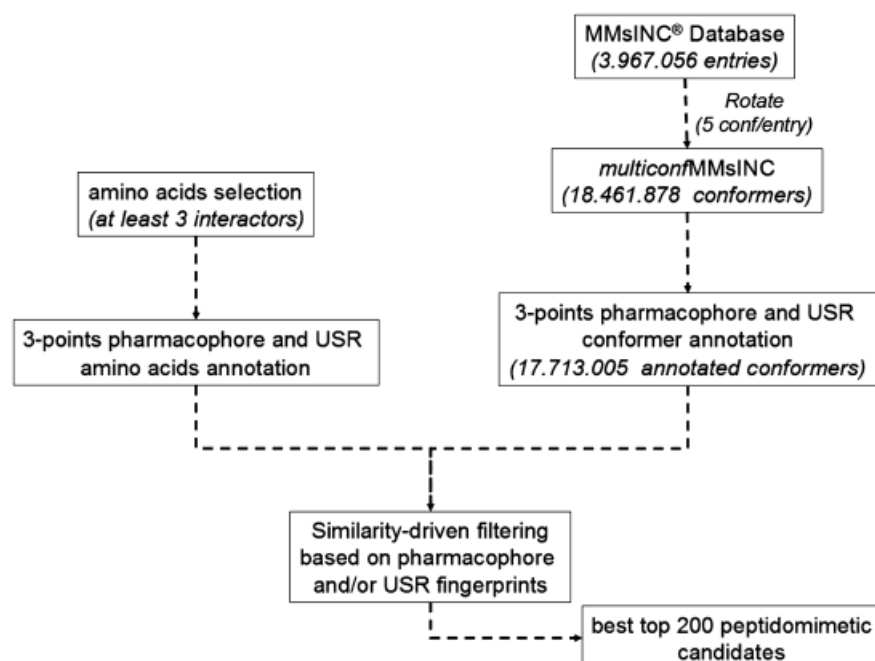


Figure 8.2: The pepMMsMIMIC WorkFlow

The crucial steps of pepMMsMIMIC architecture are detailed as follows.

8.2.1.1 Conformers Generation: MultiConf-MMsINC[®]

The best five lowest-energy conformers of each MMsINC[®] entry (including tautomers and ionic states) was generated by using Rotate ver. 1.0 software (<http://www.mol-net.de>) obtaining an ensemble of 18,461,878 conformers. Conformational analysis was splitted into two phases:

1. cyclic moieties conformational optimization
2. acyclic fragments conformational analysis

Once obtained all conformers for all molecular portions, they have been organized in subsets on the basis of values ranges of dihedral angles; each subset was ordered according to the descending frequency of each dihedral angle. Ordered subsets allowed then for complete molecular conformers reconstruction. Finally conformers have been energetically minimized using the MMF94 forcefield, selecting up to 5 most stable conformers.

8.2.1.2 Pharmacophoric Fingerprint Generation

A pharmacophore is a 3D model of required molecular features necessary for recognition of a ligand by a biological macromolecule. The IUPAC definition

is: an ensemble of steric and electronic features that is necessary to ensure the optimal interactions with a specific biological target and to trigger (or block) its biological response.. Each one *feature* is space-localized by a *centroid*; centroids are defined by contiguous atoms with same labeling, belonging to the same feature (e.g. the six carbon atoms of a benzene ring define an aromatic centroid localized at the centre of the ring). More than one label can be assigned to each atom. Thus, each atom can be part of more than one centroid. Annotation points (features) does not coincide with pharmacophore: an annotation point is a specific and single chemical feature whereas a pharmacophore is a descriptive model of various molecular features organized in the 3D space.

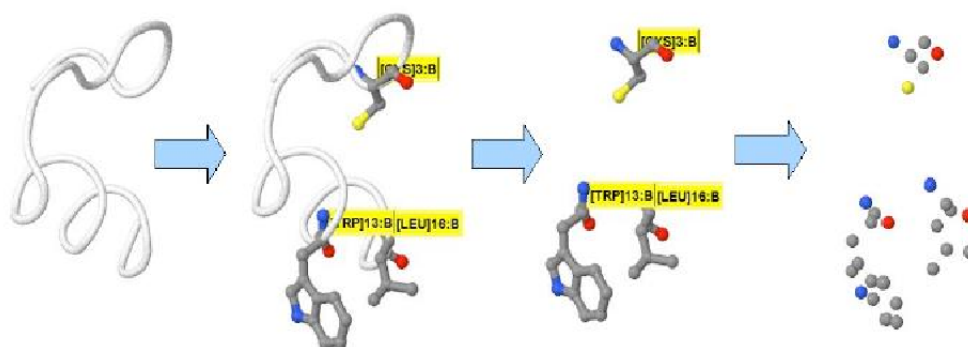


Figure 8.3: Key Residues Coding. Side Chains are translated into ensemble of features (annotation points).

In pepMMsMIMIC the peptide aminoacids (both L and D) are labeled using the following pharmacophoric features:

- Tryptophan (Trp), Tyrosine (Tyr) and Histidine (His) side chains
- H-bond acceptors (HAC)
- H-bond donors (HDO)
- positively ionisable groups (PCH)
- negatively ionisable groups (NCH)
- aromatic (ARO)
- hydrophobic (HYD)

In figure 8.4 are summarized all possible features for each one aminoacid side chain.

Aminoacid side chain	Hydrophobic	Positive ionizable	Negative ionizable	H-bond donor	H-bond acceptor	Aromatic	Other
Alanine	✓						
Arginine	✓	✓		✓			
Asparagine				✓	✓		
Aspartic acid			✓		✓		
Cysteine	✓						
Glutamine				✓	✓		
Glutamic acid			✓		✓		
Histidine				✓	✓	✓	His ring
D-Histidine				✓	✓	✓	His ring
Isoleucine	✓						
Leucine	✓						
Lysine		✓		✓			
Methionine	✓						
Phenylalanine	✓					✓	
Proline	✓						
Serine				✓	✓		
Threonine	✓			✓	✓		
Tryptophan				✓		✓	Trp ring
Tyrosine				✓	✓	✓	Tyr ring
Valine	✓						
O-phosphotyrosine			✓			✓	
Phosphoserine			✓				
Phosphothreonine	✓		✓				
D-Phosphothreonine	✓		✓				
D-Alanine	✓						
D-Arginine	✓	✓		✓			
D-Asparagine				✓	✓		
D-Aspartic acid			✓		✓		
D-Cysteine	✓						
D-Glutamine				✓	✓		
D-Glutamic acid			✓		✓		
D-Histidine				✓	✓	✓	His ring
D-Isoleucine	✓						
D-Leucine	✓						
D-Methionine	✓						
D-Phenylalanine	✓					✓	
D-Proline	✓						
D-Serine				✓	✓		
D-Threonine	✓			✓	✓		
D-Tryptophan				✓		✓	Trp ring
D-Tyrosine				✓	✓		Tyr ring
D-Valine	✓						

Figure 8.4: AminoAcids Side Chains Features. Each one AA could be assigned more than one feature, according with all its chemical features.

Peptide and conformer features are described in terms of three-point pharmacophores, with every possible pair of centroids binned according to the features distances. The pharmacophore fingerprints could be applied for comparison of protein–ligand recognition pathways in their binding sites[46, 47]. This method was successfully applied in pepMMsMIMIC for computing similarity measures between peptides and ligands pharmacophore fingerprints.

8.2.1.2.1 Fingerprint Coding Pharmacophoric fingerprint string is coded through a two step process for both peptide and small molecule conformer.

8.2.1.2.1.1 First Criterion The first criterion encodes triplets of pharmacophoric points into the pepMMsMIMIC bitstring through atom types recognition. All possible three-point combinations (using the nine different centroid types mentioned above) are encoded in the pepMMsMIMIC bitstring. Centroids are defined by contiguous atoms with the same labeling. An in-house developed SMARTS mapping tool (based on the Chemistry Development Kit, CDK, Java libraries)[48, 49] assigns atoms labels. SMARTS is a SMILES extension which

allows for molecular pattern description; SMARTS is able to recognize pharmacophoric schemes common to different molecules.

8.2.1.2.1.2 Second Criterion This second criterion encodes triplets using centroid distances information. According to the FuzCav method (fig. 8.5)[50], the maximum distance cutoff is 14.3 Å with a distance binning defined as this scheme:

[0, 4.8], [4.8, 7.2], [7.2, 9.5], [9.5, 11.9], [11.9, 14.3]

For each class of interaction, meaning each possible combination of features into triplets (i.e. ARO-ARO-PCH, . . . ,HDO-HYD-ARO), distance ranges are coded according to the scheme reported below:

```
ARO-ARO-PCH [0, 4.8][0, 4.8][0, 4.8], . . . ,ARO-ARO-PCH [0, 4.8][0, 4.8][11.9, 14.3]
ARO-ARO-PCH [0, 4.8][4.8, 7.2][0, 4.8], . . . ,ARO-ARO-PCH [0, 4.8][4.8, 7.2][11.9, 14.3]
ARO-ARO-PCH [0, 4.8][7.2, 9.5][0, 4.8], . . . ,ARO-ARO-PCH [0, 4.8][7.2, 9.5][11.9, 14.3]
ARO-ARO-PCH [0, 4.8][9.5, 11.9][0, 4.8], . . . ,ARO-ARO-PCH [0, 4.8][9.5, 11.9][11.9, 14.3]
ARO-ARO-PCH [0, 4.8][11.9, 14.3][0, 4.8], . . . ,ARO-ARO-PCH [0, 4.8][11.9, 14.3][11.9, 14.3]
ARO-ARO-PCH [4.8, 7.2][0, 4.8][0, 4.8], . . . ,ARO-ARO-PCH [4.8, 7.2][0, 4.8][11.9, 14.3]
ARO-ARO-PCH [4.8, 7.2][4.8, 7.2][0, 4.8], . . . ,ARO-ARO-PCH [4.8, 7.2][4.8, 7.2][11.9, 14.3]
```

Each bin is associated with a specific interactions triplet defined by the type of the vertices comprising the triplet and the relative distances between each centroid pairs. Every time a triplet is composed by two aromatic centroids and one positively ionizable centroid, the triplet is associated with the class ARO-ARO-PCH, and the second criterion (based on atom pair distances) is applied to correctly space-locate the triplet inside the ARO-ARO-PCH class in the pepMMsMIMIC bitstring; this rule is applied to all three-points pharmacophores (ARO-ARO-ARO, ARO-NCH-PCH, . . . ,HYD-HYD-HYD) combinations. On these basis, both peptide and small molecules conformers are classified; the only difference is that in peptide pharmacophoric description, more than one annotation point could be assigned to aminoacid side chains, as explained in figure 8.3 (e.g. Arginine could be associated to PCH, HYD e HDO features).

FuzCav method for centroids-pair distance classification

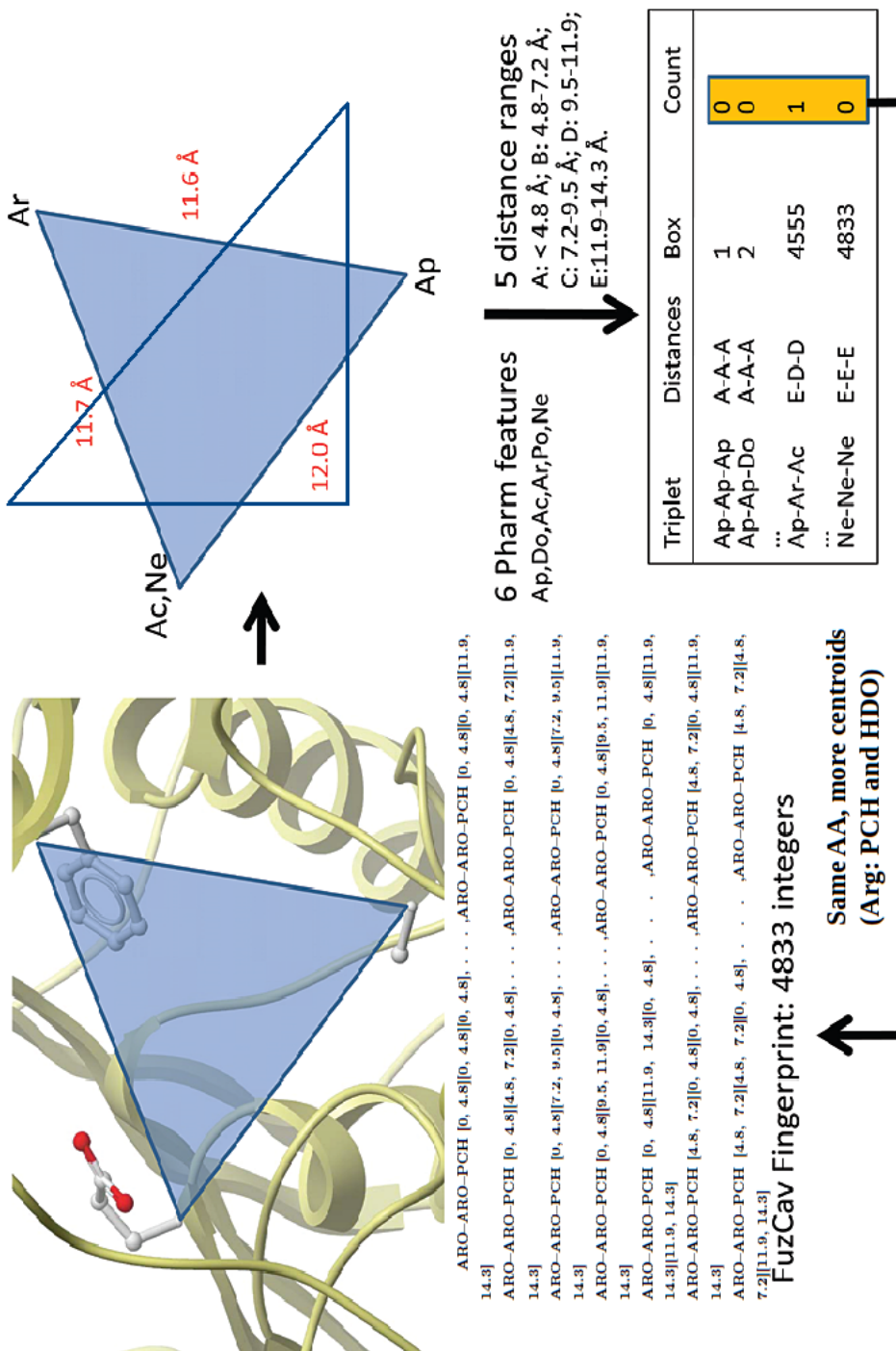


Figure 8.5: Centroids-pair Distance Classification.

The pepMMsMIMIC pharmacophoric fingerprint is a vector with 19.815 possible bits; among them, only 12.448 bits were populated in the MMsINC[®] molecular library. Only those conformers described at least by three spatially distinct features were retained. Pharmacophoric fingerprints were precalculated for 17.713.005 (17.7 Mln) conformers.

8.2.1.3 USR-based Molecular Shape Recognition

The *USR* (*Ultrafast Shape Recognition*) introduced by Ballester and Richards[51], is a fast 3D similarity search method; the central concept is that molecular shape depends only on atoms relative positions. The approach is based on moments of distance distributions, and it has been successfully applied to the fast identification of similarly shaped compounds within large molecular databases. In the USR encoding, the shape of the atomic ensemble (molecule) is characterized by the distributions of atomic distances to four fixed reference locations:

- molecular centroid (ctd)
- the closest atom to ctd (cst)
- the farthest atom to ctd (fct)
- the farthest atom to fct (ftf)

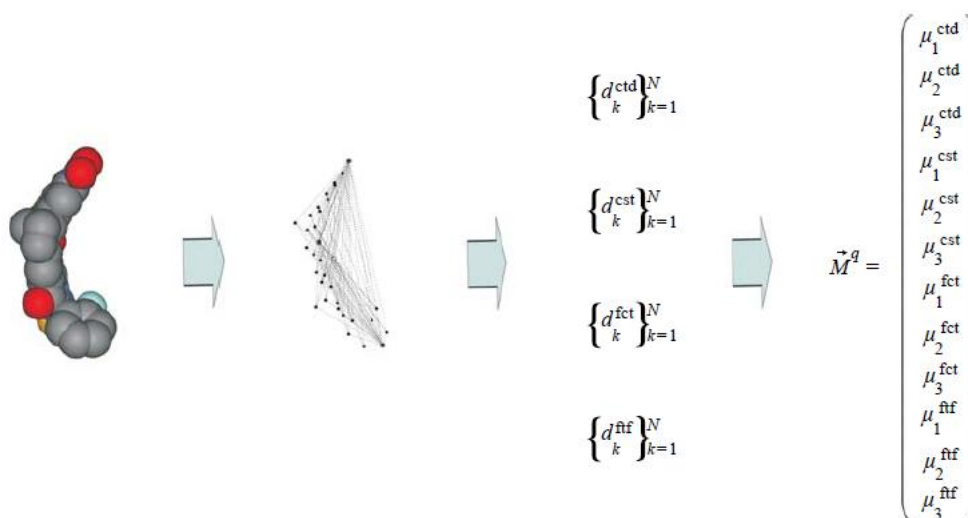


Figure 8.6: USR Coding: atomic distances distribution to 4 reference locations.

The distances distributions to the four reference locations are described by three vectors; in such a way each molecule is associated a vector composed by 12 geometric (molecular shape) descriptors. Vectors similarity analysis allows for molecular shape comparison between the peptide and the multi-conformer

small molecules library. Per ciascuna della 4 locazioni di riferimento, vengono calcolati tre momenti:

- $\mu_1^{ctd,cst,ftf}$ average atomic distance (estimates molecular size)
- $\mu_2^{ctd,cst,ftf}$ atomic distances variance
- $\mu_3^{ctd,cst,ftf}$ atomic distances skeweness (measures distances distribution asimmetry)

The process starts by spatially locating the 4 reference positions; later, monodimensional distance distributions to all 4 reference locations is determined, obtaining a set of monodimensional distributions equal to the molecule atoms number for each one reference location. Since it is not possible to compare molecules with different number of atoms, it is necessary to calculate the moments of discret monodimensional distributions, for each one reference location: in this way 12 geometric molecular descriptors are defined for each molecule. The similarity measure is obtained by the sum of lower absolute differences for respective moments; the Manhattan distance between the vectors of shape descriptors of the query and the currently screened molecule is calculated and divided by the number of descriptors. The resulting dissimilarity measure is transformed into a normalized similarity score by translating the dissimilarity by one unit and inverting the resulting value. The similarity score could vary from 0 to 1 (100% shape similarity). In the Manhattan distance two points are separated by an amount equal to absolute value of sum of coordinates differences; thus reference system translations and reflections does not change distance between two points. Reference locations, distance distribution and shape similarity are computed using in-house developed Python code.

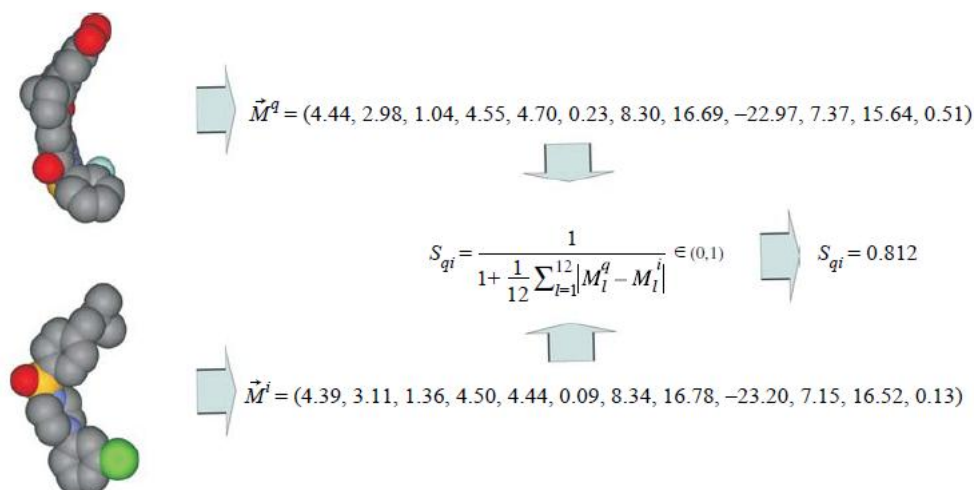


Figure 8.7: USR-based Shape Similarity Score. Similarity values range is 0 - 1(100% shape similarity).

8.2.1.4 Scoring Metrics

Fingerprint -based methods uses similarity indexes, such as Tanimoto or Tversky, to classify compound libraries. Similarity search results are often affected by bit density (distribution) differences between reference and datases, affecting the number and type of peptidomimetic small molecules retrieved. To address this issue it is possible to use weighted similarity indexes, in order to balance complexity differences between reference and database molecules. For example, the bit defining the presence of a TRP centroid (Tryptophan side chain) is always off (value "0") in the small molecule conformer fingerprint, and thus its contribution to the final similarity score must be lower than the one from the peptide TRP-bit (which could be "0" or "1" depending on Tryptophan side chain presence or absence). Opposite the elimination of non-common represented bits in the fingerprint string, causes a decrease in molecular description accuracy. In pepMMsMIMIC, fingerprints similarity is measured using an in-house implemented weighted similarity index (Sw):

$$Sw = c \frac{c}{(c + 2.5m)}$$

where c is the number of common active bits ("1" value) between fingerprint strings of both peptide and library compounds conformers, while m is the number of "1" bits in peptide fingerprint string only. Actually 4 different scoring metrics are available:

1. Shape Score (ShS): based on USR molecular shape similarity score
2. pharmacophoric fingerprint similarity(PFS): based on weighted index Sw
3. ShS-PFS filtering: ShS threshold filter (0.5) followed by PFS weighted index, Sw
4. weighted ShS-PFS combination

$$ShSPSF = (0,4ShS) + (0,6PFS)$$

(hybrid function able to reduce false positive number)

8.2.1.5 Querying pepMMsMIMIC

pepMMsMIMIC is available to the public through a web application at <http://mms.dsfarm.unipd.it/pepMMsMIMIC>. It allows for PDB structures upload and management using *Jmol*[52]. Key residues from PDB structures could be selected just by clicking them in the *Jmol* applet window; user could specify which features consider for each one side chain. In addition it is possible to include in the search process the backbone CO/NH interactors or only side chain interactors. Screening process requires at least 3 selected residues or 3 interactors and once they have been selected they are labeled in the *Jmol* window. By default the protocol returns the top 200 best peptidomimetics identified, based on the scoring metrics selected by user (default scoring metric is the ShS/PFS filtering approach). A screening process takes about 15-20 minutes.

UNIVERSITÀ DEGLI STUDI DI TRIESTE **Molecular Modeling Section**

MMS Molecular Modeling Section

pepMMsMIMIC

Home

About us...

LabMembers

Projects

Publications

eLearning

News

MMsPedia

Lab Agenda

MMS_Lab Access

MMsPLATFORM

Adenosiland

Swimming into peptidomimetic chemical space using pepMMsMIMIC (click on the picture to enjoy it):

PEP MIMIC pepMMsMIMIC

pepMMsMIMIC has been recently accepted for publication into NAR Web Server Special Issue 2011:

[\[Home\]](#)[\[About us...\]](#)[\[LabMembers\]](#)[\[Projects\]](#)[\[Publications\]](#)[\[eLearning\]](#)[\[News\]](#)[\[MMsPedia\]](#)[\[Lab Agenda\]](#)[\[MMS_Lab Access\]](#)[\[eMmsPLATFORM\]](#)[\[Adenosiland\]](#)

Copyright (c) 2005 Molecular Modeling Section. All rights reserved.

Figure 8.8: pepMMsMIMIC Web Page.

[pep.MMs:MIMIC home](#) [Background](#) [Tutorial](#) [FAQ](#) [MMSINC home](#)

Welcome to pep:MMs:MIMIC

- At a glance: pep.MMs:MIMIC is a web-oriented tool that, given a peptide three-dimensional structure, is able to automate a multiconformers three-dimensional similarity search among 17 million of conformers calculated from 3.9 million of commercially available chemicals collected in the MMSINC database.
- For a complete description of the workflow, please read the [Background Section](#).
- Who developed pep:MMs:MIMIC: [Matteo Floris](#) and [Stefano Moro](#).
- How to cite: Floris M., Mascocchi J., Fanton M., and Moro S., *Swimming into peptidomimetic chemical space using pepMMSMIMIC*, Nud. Acids Res. (2011) first published online May 27, 2011, doi:10.1093/nar/ghr287 [open access link]

example peptide: pick residues 9, 13 and 16

Color by: Secondary Structure Chain Rainbow Amicoid Hydrophobicity

Surface: Off Solvent Accessible Solvent Excluded Cavities

Settings: Back Background White Background Wireframe on Wireframe off

Reset Display

1. Upload a valid PDB file

or load an example peptide

[see DOI:10.2210/pdb11ycr/pdb, PDB ID: 1YCR, chain B]

2. Please pick at least 3 residues from the Jmol window

or select from here

[ARG]14:B Side chain Backbone CO Backbone NH

[CYS]15:B Side chain Backbone CO Backbone NH

[CYS]11:B Side chain Backbone CO Backbone NH

3. Select the scoring method

Enter your e-mail address for job completion alert (optional)

Figure 8.9: pepMMSMIMIC User Interface. Once uploaded the PDB structure, key residues and scoring metrics must be selected to start a search process. Selected residues are labeled in the Jmol window.

8.2.1.6 Results Displaying

Based on the user-selected scoring metric, top 200 ranked peptidomimetic candidates are displayed, starting from the higher scored to the lower one; each one candidate is represented by its structural conformer and MMsCode. The report page for a selected candidate is similar to the Clicking on the MMsCode of each peptidomimetic candidate the user will get the molecule report of MMsINC[®]. In the report are listed basic information as compound type (neutral, tautomer or ionic state), molecular formula, precalculated molecular descriptors, InChI and SMILES representations; it is also displayed 2D and 3D molecular rendering using Chemis3D [<http://chemis.free.fr/mol3d/>] Java applet. Parent molecules are linked to their tautomers and ions. Search results are downloadable as SDF format file or as AutoDock input files ready for docking studies.

8.2.1.7 Implementation

The pepMMsMIMIC platform runs a Linux server. The web application is PHP-based with some tool written in Java, Python and CDK.

8.2.1.8 Preliminary Validation

pepMMsMIMIC protocol capabilities and reliability were tested and validated using *Nutlins*, known inhibitors of MD2M/p53 protein-protein interaction. *Nutlins* structures were dispersed into the MMsINC[®]1.1 multiconformers library; the used PDB structure code is 1YCR. All five used *Nutlins* were retrieved by pepMMsMIMIC protocol in top 0.6% of ranked multiconformer library of ~18 millions compounds, using the *ShS-PFS filtering* scoring metric.

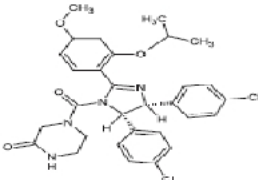
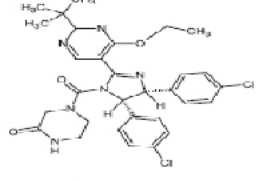
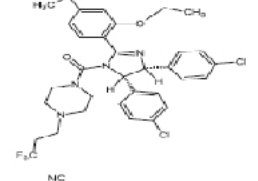
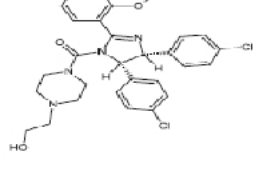
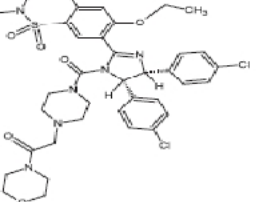
Test set	Shape only ^a	Pharmacophore only ^b	Pharmacophore after Shape ^c	Hybrid ^d
	1 475 800 (8.34)	80 365 (0.45)	4 287 (0.24)	58 751 (1.90)
	1 316 214 (7.44)	220 946 (1.25)	110 684 (0.62)	48 112 (2.72)
	1 469 866 (8.30)	220 945 (1.25)	110 683 (0.62)	62 978 (3.56)
	29 643 (0.17)	80 364 (0.45)	25 82 (0.009)	19 243 (0.11)
	29 647 (0.17)	80 363 (0.45)	25 84 (0.009)	19 244 (0.11)

Figure 8.10: pepMMsMIMIC Validation. 5 Nutlins out of 5 were ranked in top 0.6% of 17 Mln compounds in the MMsINC[®] database, using the *ShS-PFS* filtering scoring metric.

8.2.1.9 Conclusion

The major task in peptidomimetic screening process is the ability to efficiently and compactly represent proteins/peptides molecular structures; the second problem is the structural information transfer from template (peptide) to small molecule libraries. Fingerprints represent a good compromise between quality of chemical information and required computational time. The pepMMsMIMIC protocol is able to correctly identify small molecules acting as peptides, when they are dispersed inside large size molecular databases. The scope of pepMMsMIMIC is to screen chemical libraries in order to elect the most

representative subsets of peptidomimetics with respect to the selected protein target. In this sense its purpose is to act as a chemical library pruning system in order to select a molecular subset of good candidate peptidomimetics, on which perform deeper chemoinformatics or experimental studies.

Chapter 9

Exhaustive Conformational Analysis

Contents

9.1	Introduction	119
9.2	Conformers Population Generation	120
	9.2.1 Exhaustive Systematic Search	121
	9.2.2 Model-based search	123
	9.2.3 Stochastic search	123
9.3	Localized and Exhaustive Conformational Space Analysis	124
	9.3.1 Molecular Ring Systems	124
	9.3.2 Acyclic Flexible Chains	125
	9.3.3 Conformers Generation and Geometric optimization	125
	9.3.4 Protocol Validation	128
9.4	Conclusion	128

9.1 Introduction

Molecules could adopt different conformations by single-bonds rotation; the conformer with the highest affinity towards the target, is selected by the interaction process among all the conformational population entities. The molecular 3D structure is strictly depending on chemical, physical and biological properties so conformational analysis is a crucial requirement to ensure high-quality input structure for chemoinformatics application. Thus it is fundamental to select large represented conformers populations, by systematic exploration of dihedral angles and rotatable bonds and not only by precalculated conformational templates assembling.

Exhaustive conformational analysis allows for:

- computing of conformation-dependent molecular descriptors (volume, solvent accessible area, partial charges)
- geometric or chemical-based similarity analysis
- comparison of different binding modes of ligands to targets

Thus the conformational analysis could improve performances of structure-dependent cheminformatics application, because of it ensures a better chemical information quality than the one provided by single conformation-based applications.

9.2 Conformers Population Generation

The number of experimental solved 3D molecular structures is very low (*367000 X-ray solved structures in the Cambridge Structural Database*) with respect to a commercial chemical space of about 26 millions compounds; thus are required methods for 3D models prediction starting only from a CT. Since ring systems are more rigid than acyclic portions, their conformational space is populated by a very low number of possible conformers and thus the number of low-energy conformers is low too. In this sense rings and open chains need to be treated in a separated manner; then it is necessary to classify conformers into different *conformational clusters* (families) in order to select a single low-energy conformation for each one cluster. The aim of conformational analysis is to generate all energetically accessible conformers, including those corresponding to global and local energetic minimum. The global minimum is unique for a selected molecule, whereas local minimum are more than one. During the interaction process the target and the ligand could reciprocally affect one other conformation to best satisfy the binding interactions. Anywhere the global minimum not always represents the binding conformation; in the same mode different local minimum conformations could interact with the target, making necessary an exhaustive exploration of conformational space.

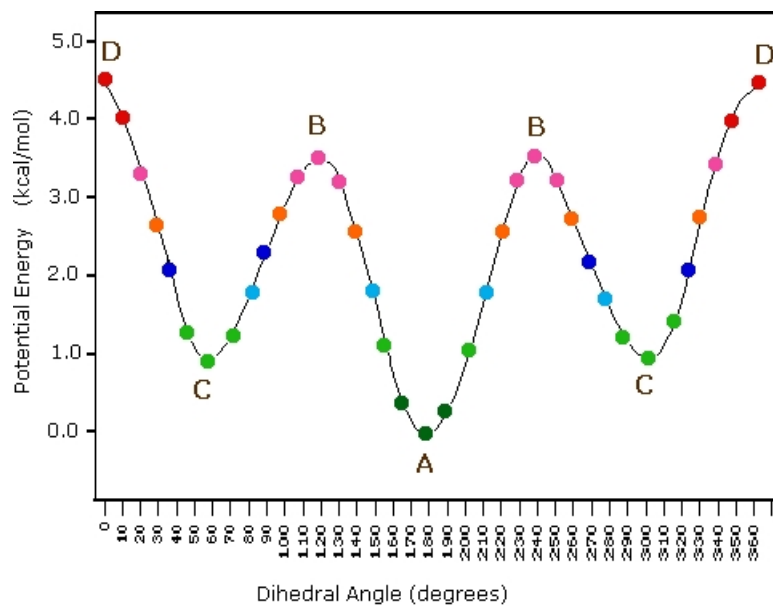


Figure 9.1: Potential Energy Profile according to dihedral angle value. A and C are global minimum and local minimum respectively.

Possible conformational analysis approaches are:

- Exhaustive systematic search
- Model-based search
- Stochastic search

9.2.1 Exhaustive Systematic Search

Systematic search explores conformational space by cyclic modification of 3D molecular structure according to the following scheme:

- Rotatable bonds identification: bond length and angles are not changed
- Fixed degree-increment rotation of rotatable bonds to cover 360° range
- Energetic minimization of conformers

search process ends when all possible conformation of torsional angles have been generated and minimized.

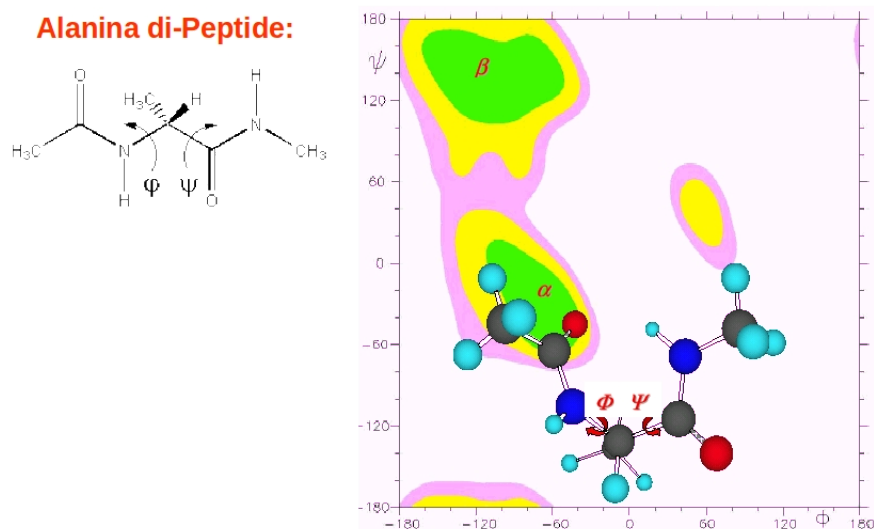


Figure 9.2: Conformational Space Representation. ϕ e ψ angle values are reported for a di-peptide structure.

Since systematic analysis explores all possible dihedral angle values in the range $0-360^\circ$, it is applied only to structures with limited number of rotatable bonds, because of a combinatorial explosion problem. The number N of possible molecular conformers is:

$$N = \prod_{i=1}^n \frac{360^\circ}{\phi_i} = \left(\frac{360^\circ}{\phi} \right)^n$$

where ϕ_i is the increment degree of dihedral angle and n is the number of torsional (dihedral) angles. Considering a fixed degree-increment of 30° , a 5 rotatable bonds molecule could exist in 248832 conformers; a 30 rotatable bonds molecule could exist in about 36 millions conformation. The combinatorial explosion that affects this search methods, make it useless except for very low flexible molecules.

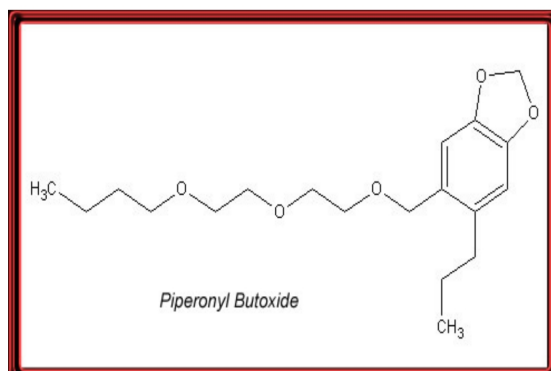


Figure 9.3: Piperonylbutoxide Conformational Space Exploration. 13 torsional angles incremented by 120° generates 1.595.323 conformations.

9.2.2 Model-based search

The Model-based search allows for combinatorial explosion problem addressing because molecular conformers are built by molecular fragments assembly. This approach is based on the following assumptions which restrict application to real cases:

- Each fragment conformation is independent on other fragments conformation
- Possible fragment conformations must be distributed all over the possible conformations observed in whole structures
- Conformational search is possible only if molecular fragments are available

thus this kind of search has limited application due to limited conformational exploration capabilities and to strict dependency on fragments conformational libraries.

9.2.3 Stochastic search

This approach does not perform a complete exploration of conformational space but allows for faster analysis than systematic search. Search process starts from a given conformation and modify the structure randomly to obtain new conformations in a non-predictable way:

- Starting conformation selection
- Random changes in molecular geometry through atomic coordinates/dihedral angles modification
- Energetic minimization of generated conformer
- Last structure comparison with previous generated (RootMeanSquareDeviation) to avoid redundant conformers generation

- selection of a starting conformation for following cycle

Search process end when it met a predefined termination criterion such as RMSD and torsional angles distribution over conformers population. Despite of execution speed, the model-based approach does not ensure for an exhaustive conformational analysis; in addition this method often explores only limited areas of energetic surface, ignoring other local minimum.

9.3 Localized and Exhaustive Conformational Space Analysis

No one of previously described methods allow for exhaustive conformational analysis on large size chemical libraries. To address this issue was tuned a systematic analysis protocol focused on dihedral angle values range which have been experimental observed in X-ray solved molecular structures. It was developed using a 42 millions available compounds library; experimental data on frequency of dihedral angle values are those of *Cambridge Structural Database(CSD)*. The developed protocol is based on CORINA

<http://www.molecular-networks.com/products/corina>

and ROTATE [<http://www.molecular-networks.com/products/rotate>] softwares. Bonds angle and length are kepted unchanged since vibrational motion is not considered: in fact usually bonds angle and length are characterized only by one rigid minimum or a very limited values range. Dihedral angles are instead modified in order to generate conformational models, considering two assumptions:

- Ring system accounts for a limited number of torsional angles
- Required energy minimization of non-bond interaction for acyclic fragments

Since ring systems conformational flexibility is very limited with respect to acyclic portions, molecular structures was splitted into two kind of fragments.

9.3.1 Molecular Ring Systems

Ring systems are analyzed by generation of all possible conformers compatible with ring tension and closure; then they are energetically minimized in a forcefield environment. Conformers are built by changing torsional angles values, according to experimental data in order to focus on restricted dihedral angles value range. In case of multiple ring systems, all possible conformers for each one ring are generated and then they are combinatorially assembled into a 3D models library. Conformational libraries undergo then an energetic minimization process, considering ring substituents in order to avoid atom clashes; if atoms contacts are identified then a focused and reduced conformational analysis is performed according to the following scheme:

- Identification of a strategic rotatable bond
- strategic bond focused conformational analysis in order to reduce/eliminate clashes

- structure optimization by energy minimization

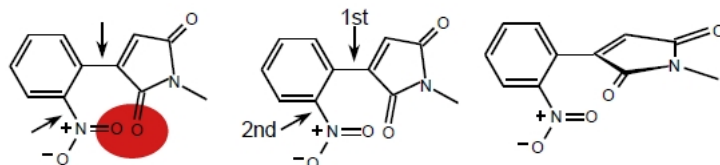


Figure 9.4: Atom Clashes. Non-bond interactions elimination through reduced conformational analysis[\diamond].

Molecular ring systems are processed by CORINA software.

9.3.2 Acyclic Flexible Chains

Flexible chains analysis is more complicated than ring systems case, because of an high degrees of freedom l value which increase with rotatable bonds number; an open chain with 5 rotatable bonds could exists in 6.0466176×10^{12} possible conformers if an increment of 1° is applied to dihedral angles. Starting from such a number of conformational arrangements, the analysis and selection process is a very hard task; thus it is required a more robust exploration protocol able to generate a low number of conformers but an high number of representative classes: in this way, despite of a limited number of generated 3D, exhaustive exploration of conformational space is ensured.

9.3.3 Conformers Generation and Geometric optimization

Despite systematic search approach is not applicable to complex molecules with high degree of freedom, it is the only one method able to guarantee the identification of global minimum and all local minimum of the energetic surface, during a conformational analysis study. This limit was overcome using a search protocol focused on limited value ranges for dihedral angles, and supported by CSD torsional frequencies data. Rotatable bonds exploration is rule and data-based; data are obtained from statistic analysis of X-ray observed conformational preferences for flexible chains. The torsion angle library is derived from CSD. Flexible chains are processed using ROTATE software according to the following scheme:

1. rotatable bonds identification
2. rotamers generation (based on torsion angle library of CSD)
3. conformers elimination if atoms clashes are identified
4. conformers classification
5. 1 conformer per class selection

CSD torsional library contains about 1000 torsional fragments (*Torsional Patterns*); statistic analysis of dihedral angle values frequency for each fragment was performed using 10° steps.

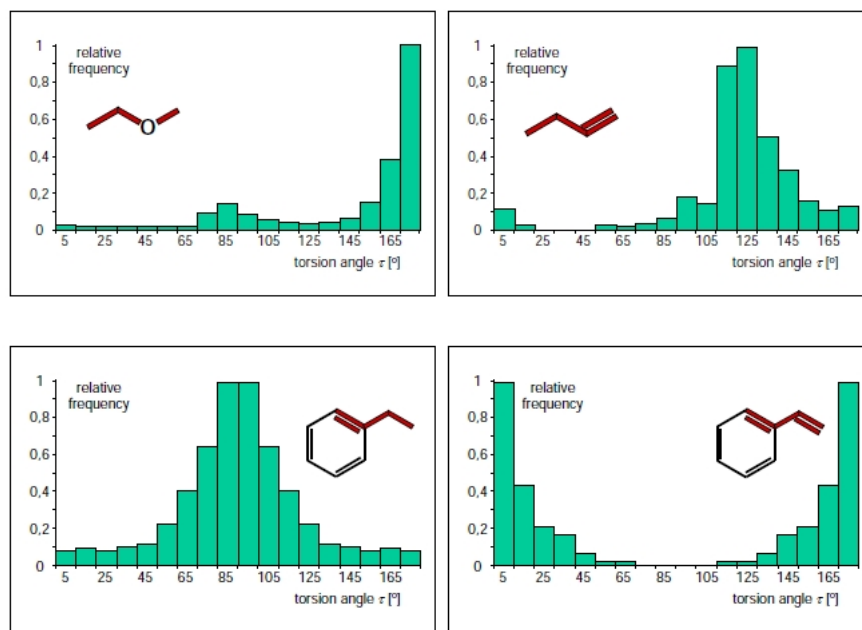


Figure 9.5: Experimental Distribution of Torsional Angle Values.[\diamond]

Starting from frequencies of torsion angle values, it is possible to derive an empiric and dihedral angle-dependent energetic function for each one molecular fragment [53];

$$E(\tau) = -A \cdot \ln f(\tau)$$

where $E(\tau)$ is the symbolic energy value for torsion angle τ , A is an adjustable parameter and $f(\tau)$ is frequency of the torsion angle value τ ; a low frequency value for a specific dihedral angle, corresponds to a high energy state and vice versa.

9.3.3.1 Selection of Initial Torsional Angle

Empiric energetic values are derived from experimental X-ray data, using increments of 10° ; most frequent torsion angle values for the investigated fragment, represent the starting angle values. Fragments dihedral angles are first divided into 12 sub-dihedral angles of 30° and then sorted according to their empiric energy function values; finally the 6 most frequent angles are selected.

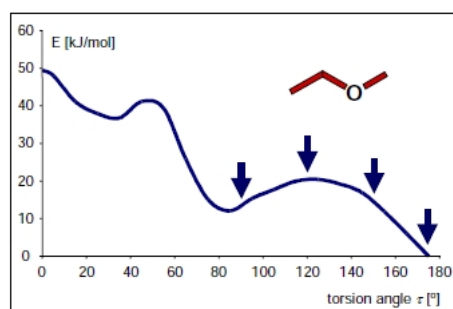


Figure 9.6: Energetic Symbolic Function for dihedral angle Csp3-Csp3-Osp3-Csp3. [◇]

9.3.3.2 Local Minimum Search

Once obtained the preferred torsion angle values set for each fragment of the investigated molecular structure, all possible combination of them are generated; thus each conformer is evaluated using the empiric function and then it undergoes an energetic minimization step in order to obtain the local minimum conformation.

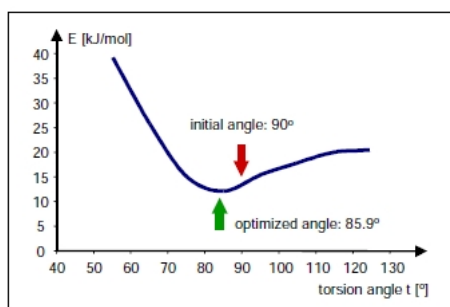


Figure 9.7: Energetic Minimization of Torsion Angle. Searching for local minimum. [◇]

9.3.3.3 Classification of Conformers Population

Since possible conformers number increase with rotatable bonds number, every new generated structure is compared to all previous ones; in this way it is possible to classify similar conformers into families, selecting the best one of each class. Classification is performed on both cartesian or torsional space. In cartesian space two conformers are considered different if their $\text{RMSD}_{XYZ} > 0.3$, reducing so the number of members of each family, maintaining an high structural diversity. In the torsional space instead two conformers are considered different if their $\text{RMSD}_{TA} < 15^\circ$. Also in this case structural diversity is highly represented, whereas number of families members is kept low.

9.3.4 Protocol Validation

Rotate softwares have been validated using the 2-L-Benzylsuccinate: 7776 conformations were generated, but only 3701 were retained after classification of conformers into families.

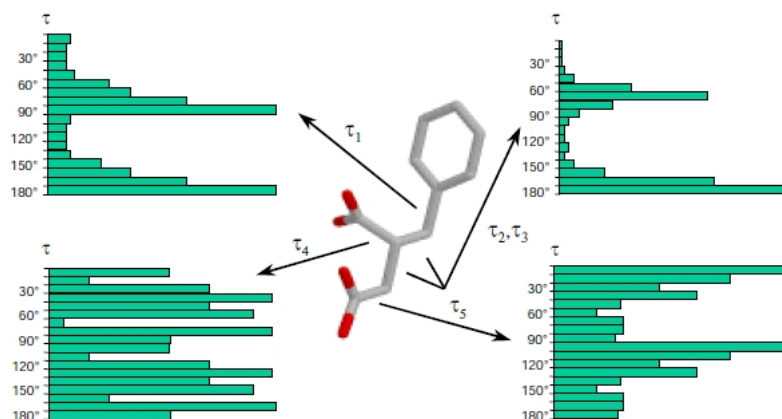


Figure 9.8: Torsion Angle Library for 2-L-Benzylsuccinate

Best RMSD selected conformation with respect to the crystallographic one, for both cartesian space and torsional space classification, are very closer to the X-ray solved structure.

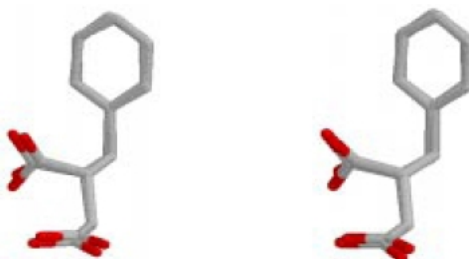


Figure 9.9: Superimposition of X-ray solved structure and predicted conformers. X-ray solved structure of 2-L-Benzylsuccinate is compared with the best rmsd selected conformer in both, cartesian and torsional space. [◇]

9.4 Conclusion

Such a protocol for conformational exploration allows for good quality 3D models prediction, avoiding combinatorial explosion but maintaining a high structural diversity. Torsion angle library of CSD, focus on experimentally determined dihedral angle value ranges. Since high-quality 3d models are required for chemoinformatics application, a multi-conformer version of MMsINC[®] chemical library was generated, starting from about 4 million compounds and obtaining about 17 million conformers (used as screening library in the pep-

MMsMIMIC platform). Recently the protocol was applied to a 92 million compounds dataset obtaining about 2.760.000.000 high-quality conformers.

Aknowledgments

A special thank to:

Pamela for being the best part of my life.

My family for supporting me in every situation and being a strong and continuous reference.

Prof. Stefano Moro for his lessons of life and science, for being teacher and friend.

MMsLab friends and colleagues: Andrea C., Davide S., Silvia P., Antonella C., Magdalena B., Giorgio C., Fabian C., Stefano S., Andrea B.

The MMsINC[®]-adventure friends: Matteo F. and Mattia S.

...and to all person that for better and for worse leave a sign in my life.

"...speak little and listen much, because you have two ears and only one mouth."

Bibliography

- [1] William J. Wiswesser. How the wln began in 1949 and how it might be in 1999. *J. Chem. Inf. Comput. Sci.*, 1982, 22 (2), pp 88-93, 1982.
- [2] William J. Wiswesser. 25, 258-263. *J. Chem. Inf. Comput. Sci.*, 1985.
- [3] William J. Wiswesser. 22, 88-93. *J. Chem. Inf. Comput. Sci.*, 1982.
- [4] IUPAC. International chemical identifier. <http://www.iupac.org/inchi/>, 2000-2011.
- [5] S.M. Welford J.M. Barnard, C.J. Jochum. Rosdal: A universal structure/substructure representation for pc-host communication, in chemical structure information systems: Interfaces communication and standards. W.A. Warr (Ed.), *ACS Symposium Series No. 400*, American Chemical Society, Washington, DC, 1989, pp. 76-81., 1989.
- [6] D. Weininger. 30(3), 237-243. *J. Chem. Inf. Comput. Sci.*, 1990.
- [7] D. Weininger. 28(1), 31-36. *J. Chem. Inf. Comput. Sci.*, 1988.
- [8] R.W. Homer T. Hurst G.B. Smith S. Ash, M.A. Cline. 37, 71-79. *J. Chem. Inf. Comput. Sci.*, 1997.
- [9] A.T. Balaban. 35(3), 339-350. *J. Chem. Inf. Comput. Sci.*, 1995.
- [10] H.P. Schultz. 29(3), 227-228. *J. Chem. Inf. Comput. Sci.*, 1989.
- [11] O. Ivanciuc. Coding the constitution-graph theory in chemistry. *Handbook of Chemoinformatics*, J. Gasteiger (Ed.), Wiley-VCH, Weinheim, Chapter II, Section 4., 2003.
- [12] L. Spialter. 85(13), 2012-2013. *J. Am. Chem. Soc.*, 1963.
- [13] I. Ugi J. Dugundji. 39, 19-64. *Topics Curr. Chem.*, 1973.
- [14] MDL. Mdl molfile format. <http://www.mdli.com/downloads>, 1998.
- [15] W.D. Hounshell A.K.I. Gushurst D.L. Grier B.A. Leland J. Laufer A. Dalby, J.G. Nourse. 32, 244-255. *J. Chem. Inf. Comput. Sci.*, 1992.

- [16] H. L. Morgan. The generation of a unique machine description for chemical structures—a technique developed at chemical abstracts service. *J. Chem. Doc.*, 1965.
- [17] W.C. Herndon. Canonical labelling and linear notation for chemical graphs, in chemical applications of topology and graph theory. *R.B. King (Ed.)*, Elsevier, Amsterdam, pp. 231-242., 1983.
- [18] M.E. Munk X. Liu, K. Balasubramanian. 30, 263-269. *J. Chem. Inf. Compu. Sci.*, 1990.
- [19] V.J.van Geerestein D.M. Bayada, H. Hamersma. 39, 1-10. *J. Chem. Inf. Compu. Sci.*, 1999.
- [20] C.J. Blankley D. Wild. 40, 155-162. *J. Chem. Inf. Compu. Sci.*, 2000.
- [21] D.E. Clark R.A. Lewis, S.D. Pickett. Computer-aided molecular diversity analysis and combinatorial library design. *Reviews in Computational Chemistry, Vol. 16*, K.B. Lipkowitz, D.B. Boyd (Eds.), Wiley-VCH, New York, pp. 8-51, 2000.
- [22] P. Willett. Similarity and clustering in chemical information systems. *Research Studies Press, Letchworth, UK*, 1982.
- [23] G.I. Ouchi W.T. Wipke, S.K. Krishnan. 18, 32-37. *J. Chem. Inf. Comput. Sci.*, 1978.
- [24] L.J. O’Korn G.A. Wilson R.G. Freeland, S.A. Funk. 19, 94-97. *J. Chem. Inf. Comput. Sci.*, 1979.
- [25] J. Lederberg D.H. Smith L.M. Masinter, N.S. Sridharan. 96(25), 7703-7723. *J. Am. Chem.Soc.*, 1976.
- [26] G. Helmchen V. Prelog. 21, 567-654. *Angew.Chem. Int. Ed. Engl.*, 1982.
- [27] V. Prelog R.S. Cahn, C. Ingold. 5, 385-419. *Angew.Chem. Int. Ed. Engl.*, 1966.
- [28] C. Marshall A.P.Johnson P. Mata, A.M. Lobo. 4(4), 657-668. *Tetrahedron: Asymmetry*, 1993.
- [29] C. Marshall A.P.Johnson P. Mata, A.M. Lobo. 34, 491-504. *J. Chem. Inf. Comput. Sci.*, 1994.
- [30] G. Gagnon D. Laramee B. Larouche S. Hanessian, J. Franco. 30, 413-425. *J. Chem. Inf. Comput. Sci.*, 1990.
- [31] J. Gasteiger W.-D. Ihlenfeldt. 35, 663-674. *J. Chem. Inf. Comput. Sci.*, 1995.
- [32] B. Rohde J.M. Barnard, A.P.F. Cook. Storage and searching of stereochemistry in substructure search systems, in chemical information systems beyond the structure diagram. *D. Bawden, E.M. Mitchell (Eds.)*, Ellis Horwood, Chichester, UK, pp. 29-41, 1990.
- [33] J.M. Barnard. 30, 81-97. *J. Chem. Inf. Comput. Sci.*, 1990.

- [34] Dmitrii V. Tchekhovskoi Stephen E. Stein, Stephen R. Heller. The iupac chemical identifier – technical manual. *Physical and Chemical Properties Division National Institute of Standards and Technology Gaithersburg, Maryland, U.S. 20899-8380*, 2006.
- [35] B. D. McKay. Practical graph isomorphism, vol. 30, pp. 45-87. *Congressus Numerantium*, 1981.
- [36] Masciocchi J Frau G Sturlese M Palla P Cedrati F Rodriguez-Tomé P Moro S Fanton M, Floris M. Mmsinc: a large-scale chemoinformatics database. *Nucleic Acids Res.* 37:284-290, 2009.
- [37] Guido van Rossum. Python language. <http://www.python.org/>, 1999-2013.
- [38] Chemical Computing Group. Molecular operating environment. <http://www.chemcomp.com/>, 2012.
- [39] Armbruster B.N. Ernsberger P. Irwin J.J. Kaiser M.J., Roth B.L. and Shoichet B.K. Relating protein pharmacology by ligand chemistry. *Nat. Biotechnol.*, 25, 197–206., 2007.
- [40] Thomas A. Halgren. Merck molecular force field. i. basis, form, scope, parameterization, and performance of mmff94. *Journal of Computational Chemistry*, 1996.
- [41] Dominy B.W. Lipinski C.A., Lombardo F. and Feeney P.J. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Adv. Drug Deliv. Rev.*, 23, 3–25, 1997.
- [42] Oprea T.I. Property distribution of drug-related chemical databases. *J. Comp. Aid. Mol. Des.*, 14, 251–264., 2000.
- [43] Benson D.A. Bryant S.H. Canese K. Chetvernin V. Church D.M. Dicuccio M. Edgar R. Federhen S. Wheeler D.L., Barrett T. Database resources of the national center for biotechnology information. *Nucleic Acids Res.*, 36, D13–D21, 2008.
- [44] Baldi P. Swamidass S.J. Bounds and algorithms for fast exact searches of chemical fingerprints in linear and sublinear time. *J. Chem. Inf. Model.*, 47, 302–317., 2007.
- [45] Fanton M. Moro S. Floris M., Masciocchi J. Swimming into peptidomimetic chemical space using pepmmsmimic. *Nucl. Acids Res.* 39, 261-269, 2011.
- [46] Menard P.R. Cheney D.L. Hulme C. Labaudiniere R.F. Mason J.S., Morize I. New 4-point pharmacophore method for molecular similarity and diversity applications: overview of the method and applications, including a novel approach to the design of combinatorial libraries containing privileged substructures. *J. Med. Chem.*, 42, 3251–3264., 1999.
- [47] Kulkarni A. Karnachi P. Application of pharmacophore engerprints to structure-based design and data mining. In Langer, T. and Hoffmann, R.D. (eds), *Pharmacophores and Pharmacophore Searches*. Wiley-VCH, Weinheim, Germany, pp. 193–206., 2006.

- [48] Kuhn S. Horlacher O. Luttmann E. Willighagen E.L. Steinbeck C., Han Y.Q. The chemistry development kit (cdk): an open-source java library for chemo- and bioinformatics. *J. Chem. Inf. Comp. Sci.*, *43*, 493–500., 2003.
- [49] Kuhn S. Floris M. Guha R. Willighagen E.L. Steinbeck C., Hoppe C. Recent developments of the chemistry development kit (cdk) - an open-source java library for chemo- and bioinformatics. *Curr. Pharma. Design*, *12*, 2111–2120, 2006.
- [50] Weill N. and Rognan D. Alignment-free ultra-high-throughput comparison of druggable protein-ligand binding sites. *J. Chem. Inf. Model.*, *50*, 123–135., 2010.
- [51] Richards W.G. Ballester P.J. Ultrafast shape recognition to search compound databases for similar molecular shapes. *J. Comput. Chem.*, *28*, 1711–1723., 2007.
- [52] Kuhn S. Horlacher O. Luttmann E. Willighagen E.L. Steinbeck C., Han Y.Q. The chemistry development kit (cdk): an open-source java library for chemo- and bioinformatics. *J. Chem. Inf. Comp. Sci.*, *43*, 493–500., 2003.
- [53] P. Murray-Rust. How useful are x-ray studies of conformation? *Molecular Structure and Biological Activity; Griffin, J. F. and Duax, W. L., Ed.; Elsevier Biomed.:New York; pp 117-133.*, 1982.

‡ All images in chapters 2, 3, 4, 5 labeled as "‡" were taken from : "Chemoinformatics: A Textbook". Edited by Johann Gasteiger and Thomas Engel, 2003, Wiley-VCH Verlag GmbH Co. KGaA.

§ All images in chapter 5 and 6 labeled as "§" were taken from : InChI_UserGuide and InChI_TechMan (<http://www.iupac.org/home/publications/e-resources/inchi/download.html>).

◇ All images in chapter 9 labeled as "◇" were taken from: Rotate program manual (http://www.molecular-networks.com/files/docs/rotate/rotate_manual.pdf, <http://www.molecular-networks.com/products/rotate>) and Corina program manual (<http://www.molecular-networks.com/products/corina>, http://www.molecular-networks.com/files/docs/corina/corina_manual.pdf).

