Création et utilisation de chimiothèques optimisées pour la recherche “ in silico ” de nouveaux composés bioactifs.

Aurélien Monge

UNIVERSITE D'ORLEANS

THESE PRESENTEE A L'UNIVERSITE D'ORLEANS

POUR OBTENIR LE GRADE DE

DOCTEUR DE L'UNIVERSITE D'ORLEANS

Discipline : Chimie Informatique et Théorique

PAR

MONGE Aurélien

Création et utilisation de chimiothèques optimisées pour la recherche « in silico » de nouveaux composés bioactifs.

Soutenue le **:** 28 novembre 2006

MEMBRES DU JURY :

- M. Alexandre VARNEK	Président et rapporteur, ULP, Strasbourg
- M. Philippe JAUFFRET	Rapporteur, ENSC, Montpellier
- Mme Sylvaine ROY	Examinateur, CEA, Grenoble
- Mme Christelle VRAIN	Examinateur, LIFO, Orléans
- M. Nicolas BAURIN	Examinateur, Sanofi – Aventis, Paris
- M. Luc MORIN-ALLORY	Directeur de thèse, ICOA, Orléans

A Laetitia, mes parents, mes amis et ma famille.

REMERCIEMENTS

Je tiens à exprimer ma gratitude au Professeur Gérald GUILLAUMET et au Professeur Olivier MARTIN pour m'avoir accordé leurs confiances en m'accueillant dans leur laboratoire.

J'exprime ma profonde reconnaissance au Professeur Luc MORIN-ALLORY, qui a dirigé ce travail de thèse, pour ses conseils scientifiques, son implication et son soutien.

J'adresse ma gratitude au Professeur Alexandre VARNEK et au Docteur Philippe JAUFFRET pour avoir accepté d'être rapporteurs de cette thèse.

Je remercie Sylvaine ROY, le Professeur Christel VRAIN et le Docteur Nicolas BAURIN d'avoir accepté de siéger parmi les membres du jury.

Je remercie également le Docteur Philippe GUEDAT pour les discussions enrichissantes qui ont eu lieu lors de nos collaborations.

J'adresse enfin mes remerciements à l'équipe du laboratoire de Modélisation Moléculaire, pour leur aide, leurs conseils, et leur bonne humeur : le Docteur Christophe MAROT, Alban ARRAULT, le Docteur Maryline BOUROTTE, Laurent ROBIN et le Docteur Eric ARNOULT.

INTRODUCTION

La découverte de médicaments est un processus long et complexe. On estime qu'il faut entre 12 et 15 ans et environ 800 millions d'euros pour la mise au point d'un médicament [1]. La Figure 1, qui présente l'évolution des nouvelles entités chimiques mises sur le marché et les investissements des entreprises pharmaceutiques en R&D aux Etats-Unis ces dernières années, illustre une situation problématique pour le monde de la recherche pharmaceutique : alors que les investissements sont en augmentation constante, le nombre de nouvelles entités chimiques introduites sur le marché n'augmente pas. Cela signifie que le coût de mise au point d'un médicament est, tout comme les investissements, en augmentation constante. L'une des raisons de cette augmentation est le durcissement des critères d'acceptation des médicaments par les organismes gouvernementaux.

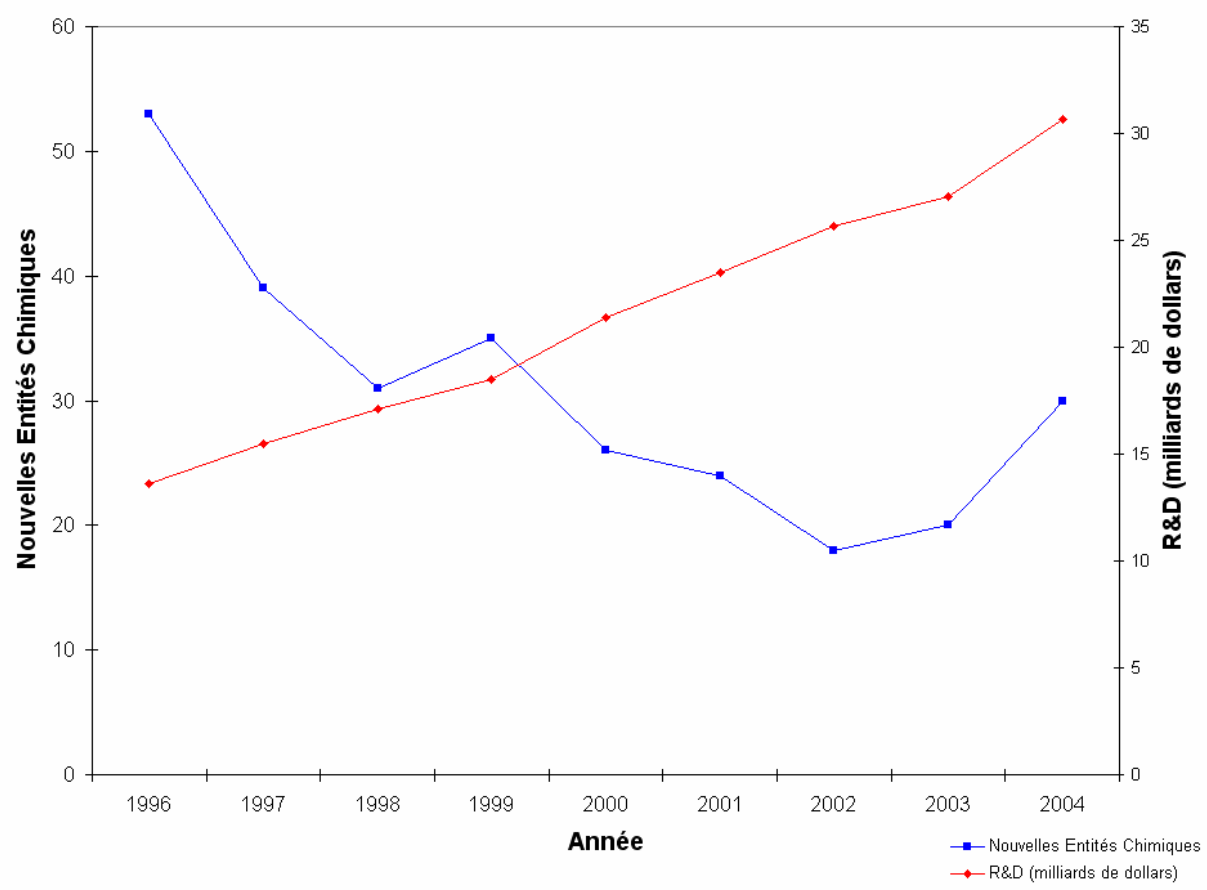


Figure 1. Nouvelles entités chimiques acceptées par la FDA (courbe bleue) et investissement en R&D aux Etats-Unis (courbe rouge) entre 1996 et 2003. D'après [1, 2].

Pour lutter contre cet état de fait, le monde de la recherche pharmaceutique optimise constamment toutes les étapes de son processus de découverte et de mise au point de médicaments. La chemoinformatique est un outil de choix pour diminuer le temps et le coût de développement d'un médicament. Cette discipline peut intervenir à différents niveaux du processus de découverte d'un médicament. Parmi les techniques de chemoinformatique nous pouvons citer le criblage de chimiothèques (techniques de QSAR, de docking et de pharmacophores), la conception de chimiothèques virtuelles, la conception de composés *de novo*, la prédiction de propriétés et l'étude de l'affinité protéine – ligand. La chemoinformatique est aujourd'hui présente dans toutes les étapes de développement d'un médicament. La conception de médicament à l'aide d'outils informatiques à partir de la structure de la cible biologique a par exemple contribué à l'introduction en phase clinique d'environ 50 composés et à la mise sur le marché de nombreux médicaments [3].

Le travail que nous allons présenter porte sur la gestion et l'exploitation de chimiothèques de composés destinés aux tests de criblages, soit réels, soit virtuels.

Dans le premier chapitre nous présenterons les notions liées aux chimiothèques et utilisées pour le reste du travail. Nous aborderons ainsi la notion d'espaces chimiques ainsi que les méthodes pour naviguer dans ces différents espaces chimiques. Nous présenterons également les méthodes les plus utilisées pour visualiser ces espaces chimiques. Nous exposerons ensuite la notion de diversité moléculaire en présentant les descripteurs, les métriques ainsi que les algorithmes utilisés pour les calculs de diversité. La dernière étape que nous aborderons, très importante dans l'exploitation de chimiothèques, est la filtration des composés. Cela inclut la suppression des composés pouvant se révéler être des faux positifs lors des tests biochimiques, la prise en compte des composés « drug-like » et « lead-like » ainsi que des structures privilégiées.

Le deuxième chapitre concerne le travail principal de la thèse, à savoir le développement du logiciel de gestion et d'exploitation de chimiothèque *ScreeningAssistant*. Ce logiciel open source, sous licence GPL, et disponible gratuitement en téléchargement sur le Web, offre par l'intermédiaire d'une interface graphique des fonctionnalités de gestion et de préparation de chimiothèques pour les tests de criblage disponibles que ne proposait aucun logiciel jusqu'à présent. Après une rapide introduction sur la place de l'open source dans la découverte de médicaments, nous présenterons dans un premier temps l'architecture du

logiciel, puis ses fonctionnalités ce qui inclut entre autres la sauvegarde des bases de chaque fournisseur de produits chimiques, l'élimination de doublons, la mesure de la diversité par fingerprints, la filtration de composés pour les tests de criblages, ainsi que la possibilité de générer les conformations des molécules rapidement et très simplement grâce à un système de distribution de calcul développé spécialement pour *ScreeningAssistant*.

Le troisième chapitre présente la création et surtout l'analyse détaillée de la chimiothèque de criblage de l'ICOA comportant 5 millions de références. Ce travail présente l'intérêt d'analyser suivant de nombreux critères les bases de données de 38 fournisseurs de produits chimiques. Une telle analyse n'est, à notre connaissance, pas disponible dans la littérature. Des méthodes d'analyses supplémentaires (des fingerprints, l'obtention de fragments rétrosynthétiques par la méthode RECAP ainsi que l'obtention des chaînes latérales des molécules) seront exposées. L'analyse des bases des différents fournisseurs de produits chimiques constituant notre chimiothèque en terme de diversité sera réalisée grâce aux méthodes d'analyses présentées dans cette partie et à celles présentes dans *ScreeningAssistant*. D'autres propriétés de ces bases seront également étudiées, à savoir le pourcentage de doublons, de composés présents uniquement dans une base, de composés « drug-like » et « lead-like » ainsi que le nombre de structures privilégiées. Nous proposerons en fin du chapitre un classement des bases en fonction d'un score de diversité globale, qui consiste en une moyenne pondérée des huit critères de diversité utilisés dans cette étude. Le même classement - mais en fonction de la diversité relative par composé - sera également présenté.

Le dernier chapitre regroupe différents travaux concrets de préparation d'ensembles de composés destinés à des tests de criblages. Différentes sélections par diversité seront tout d'abord présentées. Nous nous attarderons ensuite sur un cas particulier : la sélection par diversité de composés déjà mis en plaques. Pour résoudre ce problème nous avons développé une bibliothèque d'algorithmes d'optimisation composée d'un algorithme génétique et d'un algorithme de recuit simulé. Ces deux algorithmes ont été comparés pour la sélection de composés déjà mis en plaques. Nous exposerons enfin la sélection de composés dans le but de réaliser une mise en plaques à diversités cumulatives. Cela présente l'intérêt d'avoir des plaques dans lesquels les composés sont classés par diversité décroissante, ce qui permet de tester un nombre limité de plaques tout en couvrant un espace chimique suffisamment important.

Nous présenterons en conclusion un résumé des résultats et des différentes valorisations de ces travaux.

1. Schmid, E.F.; Smith, D.A. Keynote review: Is declining innovation in the pharmaceutical industry a myth? *Drug Discov. Today* **2005**, *10*, 1031-1039.
2. PhRMA Annual Survey, **2005**, http://www.phrma.org/publications/publications/17.03.2005.1142.cfm
3. Jorgensen, W. L. The many roles of computation in drug discovery. *Science* **2004**, *303*, 1813-1818.

Chapitre 1. L'importance des Chimiothèques dans la découverte de médicaments

I. Introduction

Les tests de criblages, virtuels et réels, jouent un rôle important dans la découverte de nouveaux principes actifs. Une étape du processus de criblage a fait l'objet d'une attention particulière ces dernières années : le choix des composés à cribler. C'est en effet une étape importante qui évite de tester inutilement des molécules.

On essaye donc de sélectionner de manière pertinente les ensembles de molécules à cribler, principalement en supprimant les doublons, mais aussi en sélectionnant les molécules les plus diverses, dans le but de couvrir au maximum l'espace chimique. De plus, des filtres sont souvent appliqués pour que les molécules choisies valident un certain nombre de propriétés. On pourra classer ces filtres en trois grandes catégories :

- Ceux destinés à éliminer les molécules avec de mauvaises propriétés ADME-Tox (Absorption, Distribution, Métabolisation, Elimination - Toxicité).
- Ceux destinés à éliminer les molécules pouvant engendrer des faux-positifs lors des tests biochimiques.
- Ceux destinés à réduire l'espace chimique étudié, souvent en se basant sur l'espace chimique des ligands d'origine.

Il existe des bases « prêtes à l'emploi ». Pour le criblage réel, certains fournisseurs proposent leur base optimisée pour le criblage, qui est souvent un ensemble de molécules « drug-like » sélectionné par diversité parmi toutes les molécules du fournisseur en question. Pour le criblage virtuel, il existe par exemple le projet Zinc [1], qui fourni gratuitement des bases destinées au docking.

Utiliser une base prétraitée présente cependant des inconvénients. Tout d'abord, la sélection des composés dépend du projet. Il n'est donc pas évident que la présélection réalisée par un fournisseur soit adaptée à tous les projets. De plus, l'originalité des molécules testées

est également un critère important dans un projet de conception de médicament. En utilisant un ensemble déjà sélectionné et susceptible d'être utilisé par d'autres équipes de recherches, comme cela est le cas avec les bases diverses des fournisseurs ou bien Zinc, les chances de trouver des touches originales sont plus restreintes. Il est donc préférable de faire ses propres sélections de composés, adaptées aux projets traités.

Nous allons présenter dans ce chapitre les notions relatives aux chimiothèques. Après une présentation des notions d'espaces chimiques, nous nous attarderons sur la notion de diversité, puis nous étudierons les méthodes de filtration *in silico* des composés. Les notions présentées ici seront utilisées dans les chapitres suivants.

II. Notion d'espaces chimiques

A. Définitions

Hann et Oprea ont défini quatre types d'espaces chimiques (Figure 2) [2] :

- Virtuel : l'espace chimique virtuel regroupe tous les composés qu'il serait possible de synthétiser. Ce nombre est grossièrement estimé à 10^{60} [3].

- Tangible : l'espace tangible contient toutes les molécules qui peuvent être facilement synthétisées. Une publication récente estime que ce nombre de composés est compris entre 10^{20} et 10^{24} [4]. Ces chiffres ont été obtenus en analysant la base interne de composés commerciaux de Novartis en termes de substituants de moins de 13 atomes lourds. Un substituant a été défini comme étant n'importe quel groupe d'atomes connecté au reste de la molécule par une liaison simple « cassable » et qui ne fait pas partie d'un cycle. A partir de cette étude les nombres de molécules de type R1-X-R2 et R1-X(-R2)-R3 (avec R1, R2, R3 des substituants, et X des scaffolds avec respectivement 2 et 3 point d'ancrages) ont été calculés, pour conduire au résultat cité précédemment.

- Global : l'espace global regroupe tous les composés synthétisés. Il est impossible de connaître exactement le nombre de molécules synthétisées dans le monde, car beaucoup de molécules sont synthétisées pour la recherche industrielle et donc confidentielles. On estime très approximativement ce nombre à 80 millions.

- Réel : l'espace réel correspond à tous les composés possédés par un organisme. Le nombre de composés maximum que possède une société est estimé à 10 millions [5]. Notre laboratoire, l'ICOA, en possède de l'ordre de 10 000.

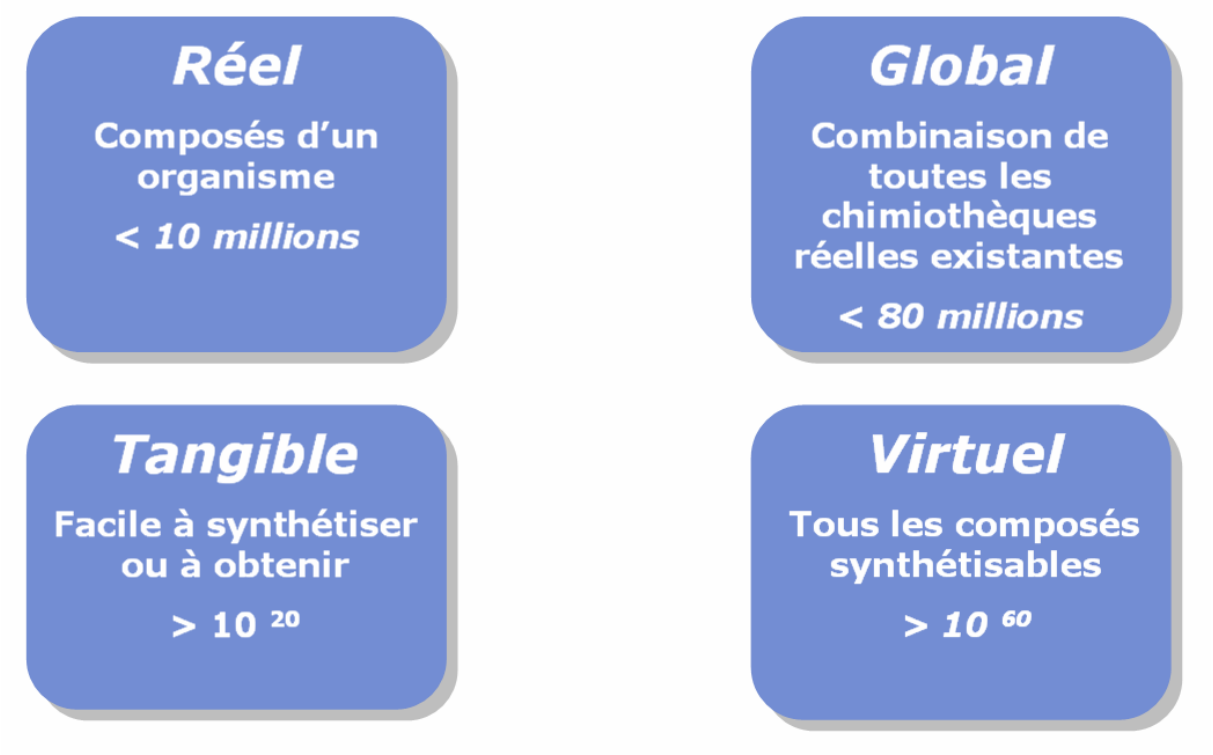


Figure 2. Définitions des quatre grands espaces chimiques.

B. Naviguer dans les différents espaces chimiques

1. Chimiothèques dans l'espace réel et l'espace global

Pour les molécules déjà synthétisées (à savoir celles des espaces chimiques réel et global), il faut obtenir soit les structures pour les tests virtuels, soit les composés sous leur forme physique pour les tests réels.

Les fournisseurs de produits chimiques proposent généralement leurs catalogues gratuitement au format SDF. On retrouve dans ces catalogues des composés issus de la synthèse classique et de la chimie combinatoire. Les fournisseurs peuvent se contenter de revendre des produits collectés dans de multiples laboratoires, et/ou synthétiser des composés en interne [5]. La plupart des librairies disponibles ont une bonne diversité, et proposent des

composés sélectionnés pour leurs propriétés « drug-like ». Certains fournisseurs proposent des librairies focalisées sur certaines familles de cibles biologiques (GPCR, CDK…).

Comme nous l'avons déjà vu en introduction de ce chapitre, il existe des collections regroupant les composés de différents fournisseurs. La plus grande base de données de structures chimiques est le CAS, avec 29 millions d'entrées [6]. Cependant, il n'est pas possible de faire du criblage virtuel sur la base CAS, ni d'acheter directement des composés. La plus grande base de composés commerciaux est la base iResearch de ChemNavigator, qui contient 15 millions de structures uniques provenant de 179 fournisseurs [7]. Il existe également des bases destinées au criblage virtuel et disponibles gratuitement. Nous avons déjà cité l'exemple de la base Zinc. D'autres bases de molécules sont disponibles gratuitement, tels que ChemDB [8], ChemBank [9] et PubMed [10]. Un comparatif de ces bases est disponible sur le site internet de ChemDB [11].

2. Chimiothèques dans l'espace tangible et l'espace virtuel

Par définition les molécules de l'espace tangible et de l'espace virtuel n'existent pas toutes physiquement. Il faut donc générer *in silico* la structure de ces composés, qui devront par la suite être synthétisés (de manière réelle). La taille de ces espaces chimiques étant gigantesque, la génération doit faire entrer en jeu une méthode de focalisation plus ou moins forte pour limiter le nombre de produits possibles. Nous distinguerons deux techniques. La première est la conception de chimiothèques virtuelles focalisées. La deuxième est la conception d'un ligand correspondant au mieux aux critères établis pour une activité biologique donnée. Cette méthode est appelée conception *de novo*.

a. Conception de chimiothèques virtuelles focalisées

La création de chimiothèques par chimie combinatoire virtuelle peut être réalisée avec plusieurs outils [12], parmi lesquels figurent le module CombiLibMaker de Sybyl [13], Analog_Builder inclus dans Cerius2 [14], et QuaSAR-CombiGen inclus dans MOE [15].

Il existe, en plus des filtres « drug-like » souvent utilisés, différentes méthodes pour focaliser une chimiothèque virtuelle. Nous avons fait le choix de regrouper ces méthodes en deux grandes familles.

La première famille englobe les méthodes qui consistent à sélectionner un sous ensemble de composés parmi tous les produits synthétisables avec les fragments de départ.

Une des solutions est de générer un ensemble de composés divers. Le logiciel « ilib diverse », basé sur CombiGen, permet par exemple de générer une chimiothèque diverse en combinant les fragments par un algorithme de Monte Carlo [16].

Une autre solution très employée est de sélectionner les composés ayant potentiellement les meilleures activités biologiques. Pour cela, le docking s'avère être une méthode de choix. Krier [17] a par exemple généré un ensemble de 320 molécules focalisée pour l'inhibition de la phosphodiesterase 4, puis synthétisé les neuf composés ayant les meilleurs scores prédits par FlexX. Ce travail a permis d'isoler un composé ayant un IC50 près de 900 fois supérieur à la zardaverine, l'inhibiteur de référence. Une autre application récente a consisté à créer une base de composés focalisés pour l'inhibition de la formation de PrP^{Sc}, dans le but de traiter notamment la maladie du prion [18]. Un criblage virtuel avec le logiciel GOLD a permis de réduire grandement le nombre de composés à synthétiser. Les auteurs ont finalement découvert 19 composés se liant à $huPrP^{C}$.

En toute logique, la combinaison de la diversité et de l'affinité est aussi utilisée. Chen et son équipe ont ainsi développé un logiciel permettant de générer une chimiothèque virtuelle grâce à un algorithme génétique optimisant trois critères : l'aspect « drug-like », la diversité, et l'affinité [19]. Les propriétés « drug-like » des molécules sont évaluées en utilisant une extension de la « règle des 5 ». L'estimation de la diversité est réalisée par le calcul de distance euclidiennes à partir de 39 descripteurs présentés dans une autre publication comme étant « les meilleurs » pour calculer la diversité moléculaire [20]. Le score d'affinité est obtenu quant à lui grâce à DOCK 4.0. Les auteurs ont utilisé leur logiciel pour générer des bases focalisées pour COX-2 et PPAR-γ.

La deuxième famille regroupe les méthodes qui cherchent à évaluer la pertinence des substituants plutôt que des produits. Cela permet de réduire grandement le nombre de combinaisons possibles. Généralement, ces méthodes se basent sur le docking, en cherchant d'abord à positionner les structures centrales des molécules dans le site actif. Un exemple de méthodologie disponible est PRO_SELECT, qui permet de générer un ensemble de produits à partir de réactifs choisis et de la partie centrale des produits placée à l'intérieur du site actif [21]. La méthode se base sur une liste de réactions possibles en chimie combinatoire, et sur le docking des substituants. CombiDOCK, basé sur DOCK 4.0, détermine les orientations de

chaque partie centrale dans le site actif, puis place les substituants pour calculer un score pour chaque produit [22]. OptiDock fonctionne suivant un principe similaire en utilisant le logiciel FlexX [23]. DREAM++, plus ancien, utilise un principe similaire [24].

b. Conception de novo

La conception *de novo* s'appuie sur les mêmes principes que la conception de chimiothèques virtuelles focalisées, mais avec pour objectif d'obtenir non pas une chimiothèque, mais la structure qui sera identifiée par l'algorithme comme étant la plus active. Le principe est de construire un ligand le mieux adapté à un site actif. De manière générale, on dispose d'une base de fragments, conçue par des spécialistes, ou d'après des algorithmes de rétrosynthèse. Une molécule de départ est alors choisie, ou générée de manière aléatoire. L'algorithme d'optimisation (généralement un algorithme génétique ou un recuit simulé) prend en charge la modification de la molécule afin d'améliorer le critère choisi (similarité à un composé actif déjà existant, score de docking…). L'espace chimique parcouru ainsi est donc très important.

Il existe de nombreux outils de conception *de novo*. Les plus anciens sont LEGEND [25], LUDI [26], SPROUT [27] et HOOK [28]. Le Tableau 1 présente un échantillon de méthodes de conception *de novo*.

Méthode / logiciel	Algorithme d'optimisation	Scoring
TOPAS [29]	Algorithmes génétiques	Fingerprints CATS
LeapFrog [30]	Algorithmes génétiques	A partir de la structure du récepteur ou d'un model COMFA
ADAPT [31]	Algorithmes génétiques	DOCK
LigBuilder [32]	Algorithmes génétiques	Affinité de liaison et biodisponibilité
PRO-LIGAND [33]	Algorithmes génétiques	Champs moléculaires et pharmacophores
CONCERTS [34]	Dynamique moléculaire	Energie potentielle totale du système
GrowMol [35]	Distribution de Boltzman	Energie potentielle totale du système
LEA3D [36]	Algorithmes génétiques	FlexX

Tableau 1. Exemple de méthodes de conception *de novo*.

Un exemple d'application récente de conception *de novo* est LEA3D. Cette méthodologie utilise le docking pour donner un score aux molécules virtuelles générées. Les fragments utilisés pour cette méthode proviennent de la base Comprehensive Medicinal Chemistry [37] et de la base LIGAND de KEGG [38]. Ils ont été générés en séparant les parties cycliques et acycliques des molécules. Les résultats du programme de docking FlexX ainsi que différentes propriétés moléculaires (masse moléculaire, nombre d'atomes…) sont combinés pour donner un score à chaque molécule. Le programme a été appliqué à la génération de substrats naturels de la thymidine monophosphate kinase *Mycobacterium*

tuberculosis, permettant la découverte de 17 inhibiteurs ayant une activité de l'ordre du micromolaire.

C. Visualiser les espaces chimiques

Deux possibilités s'offrent au chemoinformaticien pour visualiser les espaces chimiques. La première est de choisir de représenter les molécules en fonction de deux ou trois descripteurs. La deuxième est de représenter les molécules en fonction de plus de trois descripteurs, en utilisant une méthode pour combiner ces descripteurs en deux ou trois axes significatifs. Plusieurs méthodes de combinaisons de descripteurs sont disponibles. Nous ne nous attarderons ici que sur les deux techniques les plus populaires en chemoinformatique, à savoir l'analyse en composantes principales et les cartes de Kohonen. Bien que ces deux techniques soient les plus utilisées, d'autres techniques, tels que les GTM (Generative Topograpic Map), les HGTM (Hierarchical Generative Topograpic Map) [39], et l'algorithme Stochasitc Proximity Embedding [40] peuvent donner de meilleurs résultats.

1. Analyse en Composantes Principales (ACP)

L'ACP est une méthode statistique qui permet de trouver, par une transformation linéaire, les axes qui représentent au mieux les données dans l'espace. En d'autres termes, cette méthode va permettre de trouver les axes qui expliquent au mieux la dispersion du nuage de points. Si les données sont représentées en fonction de *n* descripteurs, l'ACP va donc permettre de trouver au maximum *n* axes classés en fonction de la variance qu'ils représentent.

L'ACP se révèle être très utile pour la visualisation de données, car elle permet d'obtenir deux ou trois axes qui représentent au mieux la variance globale des données. C'est une méthode très employée pour la visualisation d'espaces chimiques [41, 42, 43, 44].

On notera que l'ACP est aussi très employée pour traiter les descripteurs lors de la mise au point de modèles QSAR ou d'études de similarité. Cette méthode permet en effet de décorréler les variables, réduire leur nombre, et supprimer le bruit [45, 46, 47].

2. Cartes de Kohonen (ou SOM)

Les cartes de Kohonen (aussi appelées Self-Organizing Maps ou SOM), sont des réseaux de neurones non supervisés [48]. Dans le cas d'une représentation d'un espace chimique, la couche d'entrée compte autant de neurones que de descripteurs. Classiquement, la couche de sortie est composée de neurones disposés en carré, qui correspondent à une représentation en 2D de l'espace chimique.

SOM est une méthode d'apprentissage très populaire, notamment dans le domaine de la reconnaissance d'images. Elles trouvent également de nombreuses applications en chemoinformatique [49], en ce qui concerne la prédiction d'activités / propriétés et la visualisation d'espaces chimiques [50]. Gasteiger et son équipe sont d'importants contributeurs du développement de SOM en chemoinformatique, principalement pour la prédiction d'activité par l'analyse de surfaces moléculaires [51, 52, 53]. Cette méthode a été utilisée dans de nombreuses autres applications tels que la prédiction de toxicité [54], la classification de protéines [55, 56] ou bien la création de jeux de données pour les modèles QSAR (jeu d'apprentissage, de cross-validation, et de test) [57].

III. La diversité moléculaire

La diversité moléculaire est très utilisée pour concevoir des ensembles de criblages. Elle repose sur le principe de similarité, qui veut que des composés ayant des structures similaires aient des activités biologiques proches. D'après ce principe, le fait de sélectionner des composés avec des structures diverses pour un criblage doit donc permettre de régulariser le taux de touches et d'augmenter leurs diversités structurelles. Cette règle souffre cependant de nombreuses exceptions, et certains spécialistes de Pfizer sont même arrivés à la conclusion que les sélections par diversités réalisées sur leurs ensembles de criblages n'étaient pas plus efficaces qu'une sélection aléatoire pour trouver des composés actifs [58]. Cependant, de nombreuses autres études réalisées depuis démontrent l'intérêt des sélections par diversité. Il a ainsi été démontré qu'il était nécessaire, pour avoir des touches sur le même nombre de cibles biologiques, de sélectionner 3,5 à 3,7 fois plus de composé avec une sélection aléatoire qu'avec une sélection par diversité utilisant le coefficient de Tanimoto et les fingerprints UNITY. Les auteurs de ce travail suggèrent également qu'un ensemble de criblage conçu avec une limite du coefficient de Tanimoto à 0,7 couvre 90 % des cibles biologiques, ce qui est suffisant pour un premier criblage. Couvrir les 10 % restant demande de fixer la limite du

coefficient de Tanimoto à 0,85, ce qui donne une sélection avec un nombre bien plus important de composés [59]. Martin estime qu'un composé similaire à un composé actif (Tanimoto ≥ 0,85 avec les fingerprints de Daylight) a 30 fois plus de chances d'avoir la même activité biologique que le composé en question par rapport à un composé pris au hasard [60].

La diversité est complètement liée à l'évaluation de la similarité de molécules. Nous allons donc présenter les différents éléments nécessaires à un calcul de similarité à savoir les descripteurs, la métrique et la pondération. Nous étudierons par la suite les grandes familles d'algorithmes de diversité existantes.

Il est à noter qu'une publication récente présente un bilan des techniques de diversité [61], et que d'autres publications traitent également de manière pertinente de ce sujet [62, 63].

A. Descripteurs

Depuis le premier modèle QSAR de Hansch et Fujita [64] la communauté chemoinformatique est sans cesse à la recherche de nouveaux descripteurs moléculaires. L'ouvrage de référence en la matière est le *Hanbook of Molecular Descriptors* de Todeschini. On estime à environ 3100 le nombre de descripteurs moléculaires [61]. Il est communément admis de les classer en fonction de la dimensionnalité de la structure nécessaire pour leur calcul. On parlera donc de descripteurs 1D, 2D et 3D (Tableau 2).

Structure de la molécule	**Informations**	**Exemple de descripteurs**
1D	Formule brute : atomes présents	Masse moléculaire Présence / nombre d'un atome donné
2D	Enchaînement des atomes Type des atomes et des liaisons	Méthodes fragmentales (log P, réfractivité molaire…) Fingerprints
3D	Structure minimisée / Conformations	Surfaces Volumes Pharmacophores

Tableau 2. Exemple de descripteurs en fonction de la dimensionnalité de la structure de départ (modifié à partir de [61]).

Il convient de noter que les descripteurs 3D sont souvent mis de côté pour les calculs de diversité car ils nécessitent au préalable le calcul des structures 3D, ce qui pose deux gros problèmes : le temps de calcul et le choix de la ou des conformations bioactives. Les descripteurs 1D et 2D sont donc les plus utilisés pour les calculs de diversité. Etant donné le grand nombre de descripteurs existant, une des premières difficultés lors d'un calcul de diversité est de choisir ceux à utiliser. Toutes les approches sont possibles, mais de grandes tendances se dégagent dans les recherches publiées.

Une des approches est d'utiliser une technique de réduction de dimensionnalité (déjà présentée dans le contexte de la visualisation d'espaces chimiques dans la partie II.C). Agrafiotis, a ainsi présenté de nombreux travaux dans lesquels l'ACP est utilisée pour combiner un certain nombre de descripteurs [65, 66, 67].

Une autre approche est l'utilisation de fingerprints. C'est le type de descripteurs le plus utilisé pour les études de similarité et de diversité. La raison est que ces descripteurs parviennent généralement à saisir un grand nombre d'informations de la molécule, mais surtout que ces informations sont stockées sous une forme très condensée. La plupart des fingerprints sont codés par des chaînes allant de quelques dizaines à un millier de bits. Ce format est performant en termes d'espace disque (il occupe peu d'espace disque et il est donc facile de sauvegarder les fingerprints de bases de données de millions de molécules) et de performances (les opérations sur les chaînes de bits sont réalisées de manière très performante par les ordinateurs). Un des fingerprints les plus utilisés est le fingerprint MACCS. Il a été mis au point par MDL pour accélérer la recherche sous structurales dans les bases de données. Chaque bit code pour un ou plusieurs éléments sous-structuraux. Il est aujourd'hui utilisé dans sa version 166 bits pour des calculs de similarité / diversité. Daylight propose également un fingerprint très utilisé [68]. La différence entre ces deux fingerprints est que pour MACCS un bit correspond à une propriété, alors que pour Daylight les chaînes de bits sont repliées, et un bit peut donc correspondre à plusieurs propriétés.

Les fingerprints MACCS et Daylight encodent pour de petits fragments sous-structuraux. Une autre stratégie consiste à coder la topologie d'éléments pharmacophoriques, ce qui permet ainsi d'obtenir des fingerprints 2D. C'est le fonctionnement du fingerprint CATS (Chemically Advanced Template Search), qui code chaque atome par un ou plusieurs types prédéfinis : donneur de liaison H (D), accepteur de liaison H (A), chargé positivement (P), chargé négativement (N), lipophile (L) [69]. Cela fait 15 types de distances à coder : DD, DA, DP, DN, DL, AA, AP, AN, AL, PP, PN, PL, NN, NL et LL. Les distances sont codées en fonction du nombre minimal de liaisons séparant deux atomes donnés. Les distances jusqu'à 10 liaisons sont codées ce qui fait 150 distances différentes à coder (15 x 10). Chacune des distances est codée par un entier qui correspond au nombre de fois que la distance est présente dans la molécule. Une autre équipe de recherche a proposé le fingerprint CATS 2 qui diminue la contribution des atomes lipophiles [70]. Ces pharmacophores 2D nous semblent coder des informations importantes des molécules. Cependant, dans les exemples présentés, la similarité entre deux fingerprints est mesurée par des distances euclidiennes, ce qui est à notre avis inadapté. En effet c'est la variation du nombre de distances, et non pas de la distance directement qui est mesurée par la métrique. Une métrique plus adaptée permettrait de mieux tirer parti de ce type de fingerprints.

Nous citerons un dernier type de descripteurs utile pour la recherche par similarité et donc la diversité : les graphes réduits [71, 72, 73, 74]. Ils codent l'enchaînement de groupements prédéfinis. Il existe différents types de graphes réduits. Un exemple est présenté Figure 3.

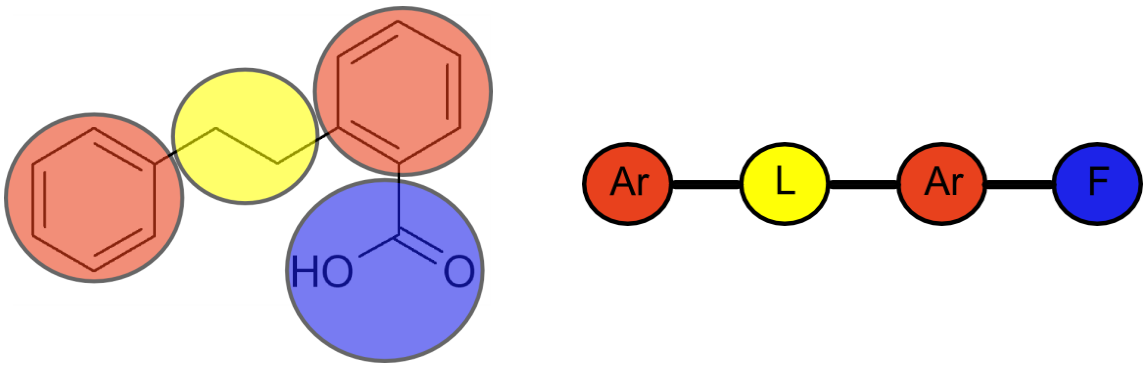


Figure 3. Exemple de molécule convertie en graphe réduit. Le graphe ne garde que l'enchaînement de grands types d'éléments structuraux. *Ar* correspond à un groupe aromatique, *L* à un linker, *F* à un groupe fonctionnel.

On notera que CATS et les graphes réduits ne gardent pas les types atomiques, mais codent les molécules en fonction de types de groupements très généralistes. Ces descripteurs peuvent donc être utilisés pour chercher, à partir d'un composé actif, d'autres composés actifs de familles chimiques très différentes. Cette technique est désignée par le terme anglo-saxon *scaffold hopping*.

B. Métriques

Dans le cas de la similarité chimique, la métrique permet de quantifier la similarité entre deux structures chimiques. On distingue deux cas d'application des métriques : l'application à des variables continues, et l'application à des variables discontinues. Le Tableau 3 présente des métriques utilisées en chemoinformatique.

Métrique	**Variables continues**	**Variables binaires**
Distance de Hamming	$\sum_{j=1}^{j=n} \left\| x_{jA} - x_{jB} \right\|$	$a+b-2c$
Distance Euclidienne	$\sqrt{\sum_{j=1}^{j=n} \left(x_{jA} - x_{jB} \right)^2}$	$\sqrt{a+b-2c}$
Distance de Soergel	$\frac{\sum_{j=1}^{j=n} \left\| x_{jA} - x_{jB} \right\|}{\sum_{j=1}^{j=n} \max\left(x_{jA}, x_{jB} \right)}$	$\frac{a+b-2c}{a+b-c}$
Coefficient de Tanimoto	$\frac{\sum_{j=1}^{j=n} x_{jA} x_{jB}}{\sum_{j=1}^{j=n} x_{jA}^{\ 2} + \sum_{j=1}^{j=n} x_{jB}^{\ 2} + \sum_{j=1}^{j=n} x_{jA} x_{jB}}$	$\frac{c}{a+b-c}$
Coefficient de Dice	$\frac{2\sum_{j=1}^{j=n} x_{jA} x_{jB}}{\sum_{j=1}^{j=n} x_{jA}^{\ 2} + \sum_{j=1}^{j=n} x_{jB}^{\ 2}}$	$\frac{2c}{a+b}$

Coefficient Cosinus	$\frac{\sum_{j=1}^{j=n} x_{jA} x_{jB}}{\sqrt{\sum_{j=1}^{j=n} x_{jA}{}^2 + \sum_{j=1}^{j=n} x_{jB}{}^2}}$	$\frac{c}{\sqrt{ab}}$

Tableau 3. Présentation de métriques utilisées en chemoinformatique (d'après [75]). Pour les variables continues : *n* le nombre de variables, x_{jA} valeur de la variable *j* de la molécule *A*, x_{jB} valeur de la variable *j* de la molécule *B*. Pour les variables binaires (fingerprints) : *a* le nombre de bits activés dans le fingerprint *A*, *b* le nombre de bits activés dans le fingerprint *B*, *c* le nombre de bits activés en commun entre *A* et *B*.

En ce qui concerne les fingerprints, la métrique la plus utilisée est indiscutablement le coefficient de Tanimoto. Ce coefficient, contrairement aux distances de Hamming et Euclidienne, ne considère pas que l'absence commune d'une propriété est un argument en faveur de la similarité de deux molécules. Cette métrique a, entre autres, été analysée par Holliday et son équipe [76]. Cette étude montre que le coefficient de Tanimoto n'est pas efficace pour différencier la similarité des petites molécules. Pour démontrer cela les auteurs ont généré des fingerprints aléatoires et constaté que les valeurs des scores de ces fingerprints comparées deux à deux s'étalaient de 0 à 0,2. Les scores de similarité inférieurs à 0,2 obtenus avec Tanimoto ne sont donc pas fiables.

Un autre inconvénient du coefficient de Tanimoto est le fait qu'il favorise les molécules de grande taille lors des recherches de similarité, et les molécules de petite taille lors des sélections par diversité. Ce biais s'explique par le fait que la densité des bits dans les fingerprints est plus forte pour les molécules de grande taille.

Malgré les défauts du coefficient de Tanimoto et le fait que son efficacité par rapport aux autres métriques varie en fonction des études [77, 78], Tanimoto reste la plus utilisée des métriques pour l'évaluation de la similarité des fingerprints.

C. Pondération

La pondération permet d'affecter des importances différentes aux descripteurs utilisés. C'est la composante des calculs de similarité la moins considérée dans les publications. Nous citerons un travail qui a consisté à étudier l'importance de la pondération dans la recherche par similarité [79]. Dans ce travail, les auteurs ont augmenté les poids des bits d'un fingerprint les plus fréquemment activés pour une famille de composés donnés. Les auteurs ont démontré que cela avait pour conséquence d'augmenter le taux de touches dans les criblages.

D. Méthodes de sélection par diversité

Il existe un très grand nombre d'algorithmes de diversité décrits dans la littérature [63]. Willett classe les méthodes de sélection par diversité en quatre catégories [62] :

- Sélection basée sur les clusters : ce type de méthodes est basé sur le regroupement des composés en clusters, qui doivent contenir des composés chimiquement similaires. La sélection consiste à choisir une molécule dans chaque cluster.

- Sélection basée sur la division : un ensemble de descripteurs est choisi, et des intervalles de valeurs sont définis pour chacun des descripteurs. La combinaison de tous ces intervalles définit des cellules. La sélection consiste à choisir une molécule dans chaque cellule si au moins une molécule est présente dans la cellule.

- Sélection basée sur la diversité : cette méthode consiste à sélectionner directement les molécules les plus diverses par rapport à celles déjà sélectionnées. Cela implique le choix d'une molécule de départ, qui est généralement réalisé arbitrairement. On retrouve dans cette famille de méthodes l'algorithme Maxmin [80], qui est la méthode de sélection par diversité la plus connue.

- Sélection basée sur l'optimisation : cette méthode utilise les méthodes d'optimisation telles que le « D-optimal design », le recuit simulé, ou les algorithmes génétiques pour sélectionner un ensemble de composés les plus divers possible.

Nous souhaitons souligner l'intérêt de combiner des critères de types différents pour la similarité / diversité. C'est à notre avis un bon moyen d'avoir une diversité robuste. Stahl et

son équipe [81] ont ainsi proposé une méthode de clustering prenant en compte à la fois la similarité par des parties très importantes des molécules (Sous-structures Maximales Communes), et la similarité par petits éléments sous-structuraux (fingerprints de Daylight). Cette technique de clustering peut directement être exploitée pour une sélection par diversité.

En plus de la pertinence de la sélection, la vitesse d'exécution est un critère très important dans le choix d'une méthode de sélection dans le cas de grand ensembles de molécules. Les méthodes de diversité sont en effet généralement destinées à être appliquées à des millions de molécules, et les algorithmes doivent donc être suffisamment rapides.

IV. Filtrer les composés indésirables

La filtration des ensembles de molécules permet d'éliminer les molécules qui risqueraient de poser des problèmes dans la suite du processus de mise au point d'un médicament. Nous présenterons dans cette partie les techniques de filtration les plus courantes permettant d'éliminer les faux positifs potentiels, et de sélectionner des composés chimiquement proches des médicaments existants. Les méthodes plus poussées de prédictions ADME-Tox sortent du cadre de notre étude et ne seront pas présentées dans ce document.

A. Les problèmes des faux positifs lors des tests biochimiques

Durant les tests biologiques, il peut arriver qu'un résultat positif ne corresponde pas à l'activité biologique attendue. On qualifie ce phénomène de faux positif.

Les faux positifs lors des tests de criblages sont un réel problème. Dans le meilleur des cas ils compliquent l'analyse des résultats, et dans le pire des cas un faux positif peut être choisi pour continuer temporairement le processus de développement d'un médicament, entraînant une énorme perte de temps, de moyens, et d'argent.

Nous distinguerons trois types de faux positifs : les fonctions réactives, les « warheads », et les « promiscuous aggregating inhibitors ».

1. Les fonctions réactives

Les fonctions réactives sont des fonctions principalement électrophiles du ligand qui permettent la formation de liaisons covalentes entre ce dernier et la cible biologique. La listes de fonctions réactives généralement citée est celle de Rishton [82] (Figure 4).

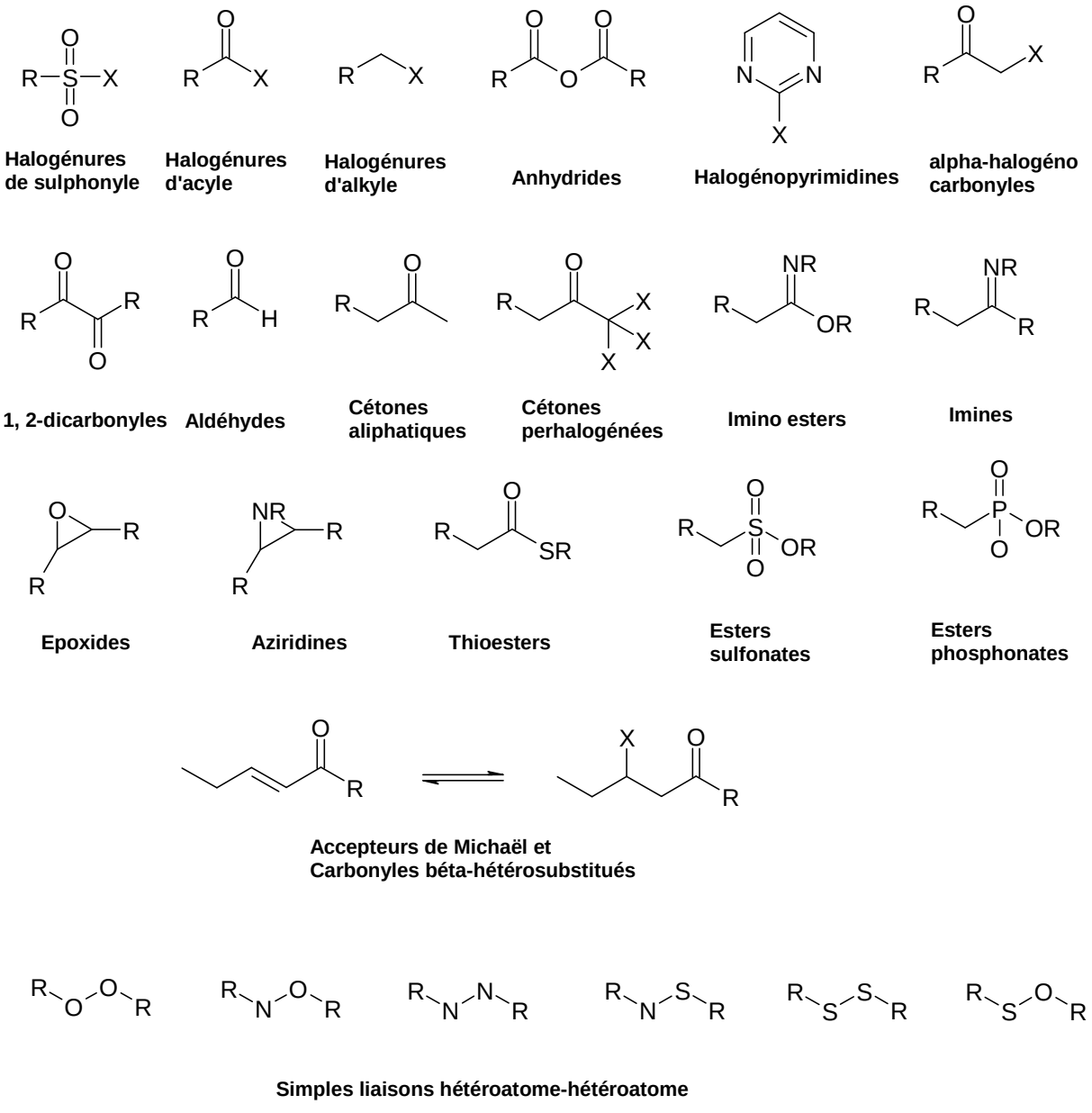


Figure 4. Liste de fonctions réactives [82].

2. Les « warheads »

A l'inverse des fonctions réactives, les « warheads » induisent des faux positifs en formant des liaisons non covalentes, et donc réversibles, avec la cible biologique [82]. Ces molécules peuvent être une catégorie « d'inhibiteurs suicides», des agents chélatant ou des composés polyioniques. Ces composés se distinguent dans les résultats de criblage en ayant une forte relation structure-réactivité [83].

3. Les « promiscuous aggregating inhibitors »

Les « promiscuous aggregating inhibitors » sont des inhibiteurs qui agissent de manière non compétitive, ont une faible relation structure – activité, et une mauvaise sélectivité. Le mécanisme de ce phénomène est longtemps demeuré inexpliqué. Schoichet et son équipe [84] ont récemment proposé un mécanisme commun pour ces inhibiteurs. Ce sont en fait des agrégats, de 50 à 400 nm, de ces inhibiteurs qui seraient responsables de leur activité, en absorbant ou adsorbant l'enzyme cible. Ces résultats ont été appuyés par des travaux complémentaires, qui ont également montré que l'ajout de détergent empêchait la formation d'agrégats [85, 86]. En plus de ces études, un modèle utilisant un partitionnement récursif permettant l'identification *in silico* de composés pouvant être des « promiscuous aggregating inhibitors » a été publié [87].

B. Notion de composés « drug-like » et « lead-like »

Un contributeur majeur dans le domaine de la caractérisation de composés « drug-like » est Lipinski avec la « règle des 5 » [88]. Cette règle est la plus utilisée pour l'identification des composés « drug-like » [89]. D'après cette règle, les composés ne validant pas au moins deux des critères suivants ont de très fortes chances d'avoir des problèmes d'absorption ou de perméabilité :

- masse moléculaire ≤ 500 Da
- log P ≤ 5
- accepteurs de liaisons H ≤ 10
- donneurs de liaisons H ≤ 5

La « règle des 5 » a été mise au point à partir de composés administrables par voie orale ayant passé avec succès la phase II des tests cliniques. Ce n'est donc pas une méthode pour distinguer les composés étant potentiellement des médicaments de ceux n'en étant pas, mais plutôt une méthode pour identifier les composés ayant une faible absorption ou une faible perméabilité. Les résultats publiés par Frimurer [90] illustrent cette idée : la « règle des 5 » accepte 74 % des composés de la base ACD, mais seulement 66 % des composés de la MDDR. L'inefficacité de la « règle des 5 » à distinguer les médicaments des autres composés a également été mis en évidence par Oprea [91].

Deux autres critères introduits par Veber [92], TPSA $\leq$ 140 Å² et nombre de liaisons pouvant tourner[i] $\leq$ 10, sont souvent employés en complément de la « règle des 5 ». Ces limites ont été établies à partir de mesures de la biodisponibilité orale de candidats médicaments.

Plusieurs bilans des avancées récentes dans ce domaine sont disponibles dans la littérature [93, 94, 95, 96]. Parmi les nombreuses méthodes développées depuis la publication de Lipinski, les limites basées sur des propriétés physicochimiques sont très employées [91, 97, 98]. Dans une publication plus récente, les auteurs utilisent des descripteurs basés sur les types des atomes et des liaisons pour identifier les composés « drug-like » [99]. Le comptage des points pharmacophoriques a également été utilisé [100]. Des méthodes d'apprentissage, telles que les machines à vecteurs de support [101] et les réseaux de neurones [102, 103], ont également été utilisées pour la prédiction du caractère « drug-like » des molécules avec de très bons résultats. Cependant, même si cette famille de méthodes donne de bons résultats, le fonctionnement des modèles obtenus, qui sont de véritables « boîtes noires », est difficilement compréhensible. Les chimistes médicinaux préfèrent en général utiliser des règles simples et facilement interprétables.

[i] Nous utiliserons dans le suite de ce document le terme « liaisons pouvant tourner » comme traduction du terme anglo-saxon « rotatable bonds ». Les terminologies françaises et anglaises sont réductrices de la notion. Une définition détaillée de ce type de liaison est donné chapitre 2 partie II. B. 8. a.

Le concept « lead-like » est basé sur le même principe que le concept « drug-like », mais est plus restrictif que ce dernier. Cette notion est liée au fait que l'optimisation d'un composé chef de file conduit généralement à une augmentation de la masse moléculaire, du log P et de la complexité [104, 105]. En conséquence, un filtre « lead-like » doit permettre de sélectionner des composés polaires avec des structures chimiques simples [106]. A la vue de cette définition assez vague, il est évident qu'il est difficile de caractériser de manière objective un composé « lead-like ». Hann et Oprea ont proposé la définition suivante [2] :

- masse moléculaire ≤ 460
- $4 \leq \log P \leq 4{,}2$
- $\log Sw \geq -5$
- liaisons pouvant tourner ≤ 10
- nombre de cycles ≤ 4
- donneurs de liaisons H ≤ 5
- accepteurs de liaisons H ≤ 9

Il existe une autre signification du concept « lead-like ». Dans ce deuxième cas, le terme « lead-like » désigne des composés de faibles masses moléculaires qui seront criblés dans le but d'identifier des activités d'un très faible ordre de grandeur (micro à millimolaire) [89]. Les composés de ce type sont également appelés « fragments ». Une « règle des trois » a été proposée pour ce type de composés [107] :

- masse moléculaire < 300
- $\log P \leq 3$
- donneurs de liaisons H ≤ 3
- accepteurs de liaisons H ≤ 3
- liaisons pouvant tourner ≤ 3
- PSA ≤ 60

C. Structures privilégiées

Une structure privilégiée est une sous-structure de grande taille présente dans les ligands de différents récepteurs biologiques [108]. Généralement, cette sous-structure est la partie centrale de la molécule. Dans la majorité des cas, la structure privilégiée est constituée de deux ou trois cycles liés par des simples liaisons ou par fusion des cycles. Ce sont les groupements fonctionnels présents sur ces structures privilégiés qui sont responsables de la sélectivité des molécules. Cette notion de structures privilégiées peut être mis en relation avec l'étude de Bemis et Murcko qui ont défini la notion de framework (cycles + linkers des molécules, sans types d'atomes et de liaisons), et mis en évidence que 50 % des médicaments de la CMC peuvent être décrits par seulement 32 frameworks.

Le fait qu'une même structure privilégiée puisse être retrouvée dans les ligands de cibles biologiques différentes peut faire penser aux « promiscuous aggregating inhibitors ». Ces deux notions sont cependant bien distinctes : un « promiscuous aggregating inhibitor » produit des faux positifs en formant des agrégats et en inhibant des enzymes variées, alors que de manière générale une molécule contenant une structure privilégiée inhibera spécifiquement une enzyme.

Deux publications récentes listent un certain nombre de structures privilégiées et leurs propriétés [108, 109]. Ce sont des structures bien connues des chimistes médicinaux comme par exemple l'indole, la purine et le biphényl. Parmi les structures privilégiées les plus connues nous pouvons également citer les 1,4-benzodiazépines-2-ones. C'est d'ailleurs en notant que ces structures avaient une affinité pour la cholécystokinine, la gastrine et les récepteurs centraux de la benzodiazépine que Evans et son équipe ont inventé le terme de structures privilégiées.

Le concept de structures privilégiées peut être utilisé pour la sélection de composés pour des tests de criblages. En effet, le fait de choisir des composés comportant des structures privilégiées doit théoriquement permettre de favoriser un bon nombre de touches. Il est également possible de créer des bases de criblages focalisées en choisissant les structures privilégiées des ligands d'une cible donnée. Les structures privilégiées présentent également un autre avantage qui est d'apporter des motifs favorables à la validation des critères « drug-like ». Nous avons en effet vu que les composés « drug-like » avaient majoritairement un

nombre de liaisons pouvant tourner ≤ 10. Les structures privilégiées étant principalement constituées de cycles aromatiques elles sont donc favorables à de bonnes propriétés « drug-like ».

V. Conclusion

Nous avons présenté dans ce chapitre les notions théoriques relatives à l'utilisation de chimiothèques. Nous avons ainsi défini les différents types d'espaces chimiques, puis les méthodes permettant d'explorer et de visualiser ces espaces. La suite de ce chapitre a été consacrée aux notions intervenant lors de la sélection de composés pour le criblage. Nous avons donc tout d'abord présenté les différentes composantes entrant en jeu lors d'une sélection par diversité, à savoir les descripteurs, les métriques et la pondération. Nous avons ensuite présenté les grandes familles d'algorithmes de sélection par diversité. Nous avons enfin abordé la filtration des composés, avec les notions de faux positifs, de composés « drug-like » et « lead-like ».

Les différentes notions présentées dans cette bibliographie ont été utilisées pour concevoir le logiciel *ScreeningAssistant*, comme nous le verrons au chapitre II. Elles ont également été utilisées au chapitre IV, dans différentes sélections de composés pour des tests de criblage.

1. Irwin, J.J.; Shoichet, B.K. ZINC - A Free Database of Commercially Available Compounds for Virtual Screening. *J. Chem. Inf. Model.* **2005**, *45*, 177-182.
2. Hann, M. M.; Oprea, T. I. Pursuing the leadlikeness concept in pharmaceutical research. *Curr Opin Chem Biol* **2004**, *8*, 255-263.
3. Bohacek, R.S.; McMartin, C.; Guida, W.C. The Art and Practiceof Structure-based Drug Design: a Molecular Modelling Perspective. Med. Res. Rev. **1996**, *16*, 3-50;
4. Ertl, P. Cheminformatics Analysis of Organic Substituents: Identification of the Most Common. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 374-380.
5. Schuffenhauer, A.; Popov, M.; Schopfer, U.; Acklin, P.; Stanek, J. Jacoby, E. Molecular Diversity Management Strategies for Building and Enhancement of Diverse and Focused Lead Discovery Compound Screening Collections. *Comb Chem High Throughput Screen* **2004**, *7*, 771-781.
6. Chemical Abstracts Services. http://www.cas.org/cgi-bin/regreport.pl
7. ChemNavigator, 6126 Nancy Ridge Drive, Suite 117, San Diego, CA 92121, USA, www.chemnavigator.com
8. Chen, J.; Swamidass, S.J.; Dou, Y.; Bruand, J.; Baldi, P. ChemDB: a public database of small molecules and related chemoinformatics resources. *Bioinformatics* **2005**, *21*, 4133-4139.
9. ChemBank, http://chembank.broad.harvard.edu/
10. PubMed, http://pubchem.ncbi.nlm.nih.gov/
11. Database System Comparisons, http://cdb.ics.uci.edu/CHEM/Web/cgibin/supplement/Comparison.py
12. Beavers, M. P.; Chen, X. Structure-based combinatorial library design: methodologies and applications. *J Mol Graph Model.* **2002**, *20*, 463-468.
13. Sybyl, Tripos, http://www.tripos.com
14. Cerius2, Accelrys, http://www.accelrys.com/products/cerius2/
15. MOE, Chemical Computing Group, www.chemcomp.com
16. Wolber, G.; Langer, T. CombiGen : A novel software package for the rapid generation of virtual combinatorial libraries. In H.-D. Höltje and W. Sippl, *Rational approaches to drug design*, **2000**, 390-399.
17. Krier, M.; Araujo-Junior, J.X., Schmitt, M.; Duranton, J.; Justiano-Basaran, H.; Lugnier, C.; Bourguignon, J.J.; Rognan, D. Design of small-sized libraries by combinatorial assembly of linkers and functional groups to a given scaffold: application to the structure-based optimization of a phosphodiesterase 4 inhibitor. *J.Med.Chem* **2005**, *48*, 3816-3822.
18. Reddy, T. R. K.; Mutter, R.; Heal, W.; Guo, K.; Gillet, V. J.; Pratt, S.; Chen, B. Library Design, Synthesis, and Screening: Pyridine Dicarbonitriles as Potential Prion Disease Therapeutics. *J. Med. Chem.* **2006**, *49*, 607-615.
19. Chen, G.; Zheng, S.; Luo, X.; Shen, J.; Zhu, W.; Liu, H.; Gui, C.; Zhang, J.; Zheng, M.; Puah, C. M.; Chen, K.; Jiang, H. Focused Combinatorial Library Design Based on Structural Diversity, Druglikeness and Binding Affinity Score. *J. Comb. Chem.* **2005**, *7*, 398-406.
20. Flower, D.R. DISSIM: a program for the analysis of chemical diversity. *Chem. Rev.* **1998**, *16*, 239-253.
21 Murray, C. W.; Clark, D. E.; Auton, T. R.; Firth, M.A.; Li, J.; Sykes, R.A.; Waszkowycz, B.; Westhead, D.R.; Young, S.C. PRO_SELECT: combining structure-based drug design and combinatorial chemistry for rapid lead discovery. 1. Technology. *J. Comput. Aided Mol. Des.* **1997**, *11*, 193-207.
22. Sun, Y.; Ewing, T. J. A.; Skillman, A. G.; Kuntz, I. D. CombiDOCK: Structure-based combinatorial docking and library design. *J. Comput. Aided Mol. Des.* **1998**, *12*, 597-604.
23. Sprous, D. G.; Lowis, D. R.; Leonard, J. M.; Heritage, T.; Burkett, S. N. Baker, D. S.; Clark, R. D. OptiDock: virtual HTS of combinatorial libraries by efficient sampling of binding modes in product space. *J. Comb. Chem.* **2004**, *6*, 530-539.
24. Makino, S.; Ewing, T. J. A.; Kuntz, I. D. DREAM++: Flexible docking program for virtual combinatorial libraries. *J. Comput. Aided Mol. Des.* **1999**, *13*, 513-532.
25. Nishibata, Y.; Itai, A. Automatic creation of drug candidate structures based on receptor structure. Starting point for artificial lead generation. Tetrahedron 1991, 47, 8885-8990.
26. Bohm, H. J. The computer program LUDI: A new method for the de novo design of enzyme inhibitors. J. Comput.-Aided Mol. Des. 1992, 6, 61-78.
27. Gillet, V.; Johnson, A. P.; Mata, P.; Sike, S.; Williams, P. SPROUT: A program for structure generation. J. Comput.-Aided Mol. Des. 1993, 7, 127-153.
28. Eisen, M. B.; Wiley, D. C.; Karplus, M.; Hubbard, R. E. HOOK: A program for finding novel molecular architectures that satisfy the chemical and steric requirements of a macromolecule binding site. Proteins 1994, 19, 199-221.
29. Schneider, G.; Lee, M. L.; Stahl, M.; Schneider, P. De novo design of molecular architectures by evolutionary assembly of drug derived building blocks. *J. Comput.-Aided Mol. Des.* **2000**, 14, 487-494.

30. LeapFrog, 6.8 ed.; Tripos, Inc.: St Louis, MO.
31. Pegg, S. C.; Haresco, J. J.; Kuntz, I. D. A genetic algorithm for structure-based de novo design. *J. Comput.-Aided Mol. De.* **2001**, *15*, 911-933.
32. Wang, R.; Gao, Y.; Lai, L. LigBuilder: A Multi-Purpose Program for Structure-Based Drug Design. *J. Mol. Model*. **2000**, *6*, 498-516.
33. Clark, D. E.; Frenkel, D.; Levy, S. A.; Li, J.; Murray, C. W.; et al. PRO-LIGAND: An approach to de novo molecular design. 1.Application to the design of organic molecules. *J. Comput.-Aided Mol. Des.* **1995**, *9*, 13-32.
34. Pearlman, D.A.; Murcko, M.A. CONCERTS: dynamic connection of fragments as an approach to de novo ligand design. *J. Med. Chem.* **1996**, *39*, 1651-63.
35. Bohacek, R.S.; McMartin, C. Multiple Highly Diverse Structures Complementary to Enzyme Binding Sites: Results of Extensive Application of a de Novo Design Method Incorporating Combinatorial Growth. *J. Am. Chem. Soc.* **1994**, *116*, 5560-5571.
36. Douguet, D.; Munier-Lehmann, H.; Labesse, G.; Pochet, S. LEA3D: a computer-aided ligand design for structure-based drug design. *J.Med.Chem* **2005**, *48*, 2457-2468.
37. CMC, MDL Information Systems, http://www.mdl.com/products/knowledge/medicinal_chem/
38. KEGG, http://www.genome.ad.jp/dbget/ligand.html
39. Maniyar, D. M.; Nabney, I. T.; Williams, B. S.; Sewing, A. Data Visualization during the Early Stages of Drug Discovery. *J. Chem. Inf. Model.* **2006**, *46*, 1806-1818.
40. Agrafiotis, D. K. Stochastic Proximity Embedding. *J. Comput. Chem.* **2003**, 24, 1215-1221.
41. Givehchi, A.; Dietrich, A.; Wrede, P.; Schneider, G. Chemspaceshuttle: A tool for data mining in drug discovery by classification, projection, and 3d visualization. *QSAR Comb. Sci.* **2003**, 22 (5), 549-559.
42. Bayada, D.M.; Hamersma, H.; van Geerestein, V.J. Molecular Diversity and Representativity in Chemical Databases *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 1-10.
43. Oprea, T.I.; Gottfries, J. Chemography: The Art of Navigating in Chemical Space. *J. Comb. Chem.* **2001**, *3*, 157-166.
44. Shi, L. M.; Fan, Y.; Lee, J. K.; Waltham, M.; Andrews, D. T.; Scherf, U.; Paull, K. D.; Weinstein, J. N. Mining and Visualizing Large Anticancer Drug Discovery Databases. *J. Chem. Inf. Comput. Sci.* **2000**, 40, 367-379.
45. Natarajan, R.; Nirdosh, I.; Basak, S. C.; Mills, D. R. QSAR Modeling of Flotation Collectors Using Principal Components Extracted from Topological Indices. *J. Chem. Inf. Comput. Sci.* **2002**, 42, 1425-1430.
46. Xue, L.; Godden, J.; Gao, H.; Bajorath, J. Identification of a Preferred Set of Molecular Descriptors for Compound Classification Based on Principal Component Analysis. J. Chem. Inf. Comput. Sci. 1999, 39, 699-704.
47. Karthikeyan, M.; Glen, R. C.; Bender, A. General Melting Point Prediction Based on a Diverse Compound Data Set and Artificial Neural Networks. *J. Chem. Inf. Model.* **2005**, 45, 581-590.
48. Kohonen, T. *Self-Organizing and Associative Memory*, 3rd ed.; Springer-Verlag: Berlin, **1989**.
49. Manallack, D. T.; Livingstone, D. J. Neural networks in drug discovery: have they lived up to their promise? *Eur. J. Med. Chem.* **1999**, *34*, 195-208.
50. Selzer, P.; Ertl, P. Applications of Self-Organizing Neural Networks in Virtual Screening and Diversity Selection. *J. Chem. Inf. Model.* **2006**, ASAP.
51. Wagner, S.; Hofmann, A.; Siedle, B.; Terfloth, L.; Merfort, I.; Gasteiger, J. Development of a Structural Model for NF-B Inhibition of Sesquiterpene Lactones Using Self-Organizing Neural Networks. *J. Med. Chem.* **2006**, *49*, 2241-2252.
52. Polanski, J.; Zouhiri, F.; Jeanson, L.; Desmaele, D.; d'Angelo, J.; Mouscadet, J-F.; Gieleciak, R.; Gasteiger, J.; Le Bret, M. Use of the Kohonen Neural Network for Rapid Screening of Ex Vivo Anti-HIV Activity of Styrylquinolines. *J. Med. Chem.* **2002**, *45*, 4647-4654.
53. Polanski, J.; Gasteiger, J.; Jarzembek, K. Self-Organizing Neural Networks for Screening and Development of Novel Artificial Sweetener Candidates. *Comb Chem High Throughput Screen* **2000**, *3*, 481-495.
54. Mazzatorta, P.; Vracko, M.; Jezierska, A.; Benfenati, E. Modeling Toxicity by Using Supervised Kohonen Neural Networks *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 485-492.
55. Yang, Z. R.; Chou, K.-C. Mining Biological Data Using Self-Organizing Map. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1748-1753.
56. Otaki, J. M.; Mori, A.; Itoh, Y.; Nakayama, T.; Yamamoto, H. Alignment-Free Classification of G-Protein-Coupled Receptors Using Self-Organizing Maps. *J. Chem. Inf. Model.* **2006**, *46*, 1479-1490.
57. Guha, R.; Serra, J.R.; Jurs, P.C. Generation of QSAR sets with a self-organizing map. *J Mol Graph Model.* **2004**, *23*, 1-14.
58. Spencer, R. W. Diversity Analysis in high throughput screening. J. Biomol. Screening, 1997, 2 , 69-70.
59. Potter, T.; Matter, H. Random or Rational Design? Evaluation of Diverse Compound Subsets from Chemical Structure Databases. *J.Med.Chem* **1998**, *41*, 478-488.

60. Martin, Y. C.; Kofron, J. L.; Traphagen, L. M. Do Structurally Similar Molecules Have Similar Biological Activity? *J. Med. Chem.* **2002**, *45*, 4350-4358.
61. Maldonado, A.G.; Doucet, J.P.; Petitjean, M.; Fan, B.T. Molecular similarity and diversity in chemoinformatics: From theory to applications. *Mol. Divers.* **2006**, *10*, 39-79.
62. Willett, P. Chemoinformatics – similarity and diversity in chemical libraries. *Current Opinion in Biotechnology* **2000**, *11*, 85-88.
63. Aggrafiotis, D. K.; Lobanov, V. S.; Rassokhin, D. N.; Izrailev, S. The Measurement of Molecular Diversity, in Virtual Screening for Bioactive Molecules, Böhm, H. G.; Schneider, G., **2000**, 265-300.
64. Hansch, C.; Fujita, T. rsp Analysis – a Method for the Correlation of Biological Activity and Chemical Structure, J. Amer. Chem. Soc., **1964**, 86, 1616-1626.
65. Agrafiotis, D. K.; Rassokhin, D. N. Design and Prioritization of Plates for High-Throughput Screening. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 798-805.
66. Agrafiotis, D. K. A Constant Time Algorithm for Estimating the Diversity of Large Chemical Libraries. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 159-167.
67. Xu, H.; Agrafiotis, D. K. Nearest Neighbor Search in General Metric Spaces Using a Tree Data Structure with a Simple Heuristic. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1933-1941.
68. Daylight, Fingerprints – Screening and Similarity, http://www.daylight.com/dayhtml/doc/theory/theory.finger.html
69. Schneider, G; Neidhart, W; Giller, T; Schmid, G. Scaffold-Hopping by Topological Pharmacophore Search: A Contribution to Virtual Screening. *Angew. Chem. Int. Ed.* **1999**, *38*, 2894-2896.
70. Nærum, L.; Nørskov-Lauritsen, L.; Olesen, P.H. Scaffold hopping and optimization towards libraries of glycogen synthase kinase-3 inhibitors. *Bioorg. Med. Chem. Lett.* **2002**, *12*, 1525-1528.
71. Gillet, V. J.; Willett, P.; Bradshaw, J. Similarity Searching Using Reduced Graphs. *J. Chem. Inf. Comput. Sci.* **2003**, *0*, 338-345.
72. Barker, E. J.; Gardiner, E. J.; Gillet, V. J.; Kitts, P.; Morris, J. Further Development of Reduced Graphs for Identifying Bioactive Compounds. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 346-356.
73. Harper, G.; Bravi, G. S.; Pickett, S. D.; Hussain, J.; Green, D. V. S. The Reduced Graph Descriptor in Virtual Screening and Data-Driven Clustering of High-Throughput Screening Data. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 2145-2156.
74. Barker, E. J.; Buttar, D.; Cosgrove, D. A.; Gardiner, E. J.; Kitts, P.; Willett, P.; Gillet, V. J. Scaffold Hopping Using Clique Detection Applied to Reduced Graphs. *J. Chem. Inf. Model.* **2006**, *46*, 503-511.
75. Willett, P.; Barnard, J. M.; Downs, G. M. Chemical Similarity Searching. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 983-996.
76. Holliday, J. D.; Salim, N.; Whittle, M.; Willett, P. Analysis and Display of the Size Dependence of Chemical Similarity Coefficients. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 819-828.
77. Whittle, M.; Willett, P.; Klaffke, W.; van Noort, P. Evaluation of Similarity Measures for Searching the Dictionary of Natural Products Database. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 449-457.
78. Whittle, M.; Gillet, V. J.; Willett, P.; Alex, A.; Loesel, J. Enhancing the Effectiveness of Virtual Screening by Fusing Nearest Neighbor Lists: A Comparison of Similarity Coefficients. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1840-1848.
79. Xue, L.; Godden, J. W.; Stahura, F. L.; Bajorath, J. Similarity Search Profiling Reveals Effects of Fingerprint Scaling in Virtual Screening. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 2032-2039.
80. M. S. Lajiness, in *QSAR : Rational Aproaches to the Design of Bioactive Compounds*, C. Silipo, A. Vittoria (Eds.), Elsevier, Amsterdam **1991**, 201-204.
81. Stahl, M.; Mauser, H. Database Clustering with a Combination of Fingerprint and Maximum Common Substructure Methods. *J. Chem. Inf. Model.* **2005**, *45*, 542-548.
82. Rishton, G.M. Nonleadlikeness and leadlikeness in biochemical screening. *Drug Discov. Today* **2003**, *8*, 86-96.
83. Rishton, G.M. Reactive Compounds and In Vitro False Positives. **1999,** presented for Vision in Business, Integrated Drug Discovery, Geneva, Switzerland.
84. McGovern, S.L.; Caselli, E.; Grigorieff, N.; Shoichet B.K. A common mechanism underlying promiscuous inhibitors from virtual and high-throughput screening. *J. Med. Chem.* **2002**, *45*, 1712-1722.
85. Ryan, A. J.; Gray, N. M.; Lowe, P. N.; Chung, C.-w. Effect of Detergent on "Promiscuous" Inhibitors. *J. Med. Chem.* **2003**, *46*, 3448-3451.
86. McGovern, S.L.; Helfand, B.T.; Feng, B.; Shoichet, B.K. A specific mechanism of nonspecific inhibition. *J.Med.Chem* **2003**, *46*, 4265-4272.
87. Seidler, J.; McGovern, S. L.; Doman, T. N.; Shoichet, B. K.; Identification and Prediction of Promiscuous Aggregating Inhibitors among Known Drugs. *J.Med.Chem* **2003**, *46*, 4477-4486.

88. Lipinski, C.A.; Lombardo, F.; Dominy, B.W.; Feeney, P.J. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Adv. Drug. Deliv. Rev.* **2001**, *46*, 3-26.
89. Lipinski, C.A. Lead- and drug-like compounds: the rule-of-five revolution. *Drug Discov. Today* **2004**, *1*, 337-341.
90. Frimurer, T.M.; Bywater, R.; Nærum, L.; Lauritsen, L.N.; Brunak, S. Improving the Odds in Discriminating "Drug-like" from "Non Drug-like" Compounds. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 1315-1324.
91. Oprea, T.I. Property distribution of drug-related chemical databases. *J. Comput. Aided Mol. Des.* **2000**, *14*, 251-264.
92. Veber, D. F.; Johnson, S. R.; Cheng, H.-Y.; Smith, B. R.; Ward, K. W.; Kopple, K. D. Molecular Properties That Influence the Oral Bioavailability of Drug Candidates. *J.Med.Chem* **2002**, *45*, 2615-2623.
93. Lajiness, M. S.; Vieth, M.; Erickson, J. Molecular properties that influence oral drug-like behavior, *Curr. Opin. Drug Discov. Devel.*, **2004**, *7*, 470-477.
94. Walters ,W.P.; Murcko M.A. Prediction of 'drug-likeness'. *Adv. Drug. Deliv. Rev.* **2002**, *54*, 255-271.
95. Clark, D. E., Pickett, S. D., Computational methods for the prediction of 'druglikeness'. *Drug Discov. Today*, **2000**, *5*, 49–58.
96. Muegge, I. Selection criteria for drug-like compounds, *Med. Res. Rev.*, **2003**, *23* ,302–321.
97. Sirois, S.; Hatzakis, G.; Wei, D.; Du, Q.; Chou, K.C. Assessment of chemical libraries for their druggability. *Comput. Biol. Chem.* **2005**, *29*, 55-67.
98. Xu, J.; Stevenson, J. Drug-like Index: A New Approach To Measure Drug-like Compounds and Their. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 1177-1187.
99. Zheng, S.; Luo, X.; Chen, G.; Zhu, W.; Shen, J.; Chen, K.; Jiang, H. A New Rapid and Effective Chemistry Space Filter in Recognizing a Druglike Database *J. Chem. Inf. Comput. Sci.* **2005**, *45*, 856-862.
100. Muegge, I.; Heald, S.L.; Brittelli, D. Simple Selection Criteria for Drug-like Chemical Matter *J.Med.Chem* **2001**, *44*, 1841-1846.
101. Zernov, V. V.; Balakin, K. V.; Ivaschenko, A. A.; Savchuk, N. P.; Pletnev, I. V. Drug Discovery Using Support Vector Machines. The Case Studies of Drug-likeness, Agrochemical-likeness, and Enzyme Inhibition Predictions. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 2048-2056.
102. Ajay, A; Walters, W.P.; Murcko, M.A. Can we learn to distinguish between "drug-like" and "nondrug-like" molecules? *J.Med.Chem* **1998**, *41*, 3314-3324.
103. Sadowski, J.; Kubinyi, H. A scoring scheme for discriminating between drugs and nondrugs. *J.Med.Chem* **1998**, *41*, 3325-3329.
104. Hann, M. M.; Leach, A. R.; Harper, G. Molecular Complexity and Its Impact on the Probability of Finding Leads for Drug Discovery. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 856-864.
105. Oprea, T.I. Current trends in lead discovery: Are we looking for the appropriate properties? *J. Comput. Aided Mol. Des.* **2002**, *16*, 325-334.
106. Davis, A.M.; Teague, S.J.; Kleywegt, G.J. Application and limitations of X-ray crystallographic data in structure-based ligand and drug design. *J. Chem. Inf. Comput. Sci.* **2003**, *42*, 2718-2736.
107. Congreve, M.; Carr, R.; Murray, C.; Jhoti, H. A 'Rule of Three' for fragment-based lead discovery? *Drug Discov. Today* **2003**, *0*, 876-877.
108. DeSimone, R.W.; Currie, K.S.; Mitchell, S.A.; Darrow, J.W.; Pippin, D.A. Privileged Structures: Applications in Drug Discovery. *Comb Chem High Throughput Screen* **2004**, *7*, 473-493.
109. Horton, D.A.; Bourne, G.T.; Smythe, M.L. The Combinatorial Synthesis of Bicyclic Privileged Structures or Privileged Substructures. *Chem. Rev.* **2003**, *103*, 893-930.

Chapitre 2. Conception de *ScreeningAssistant* et utilisation pour la gestion d'une chimiothèque destinée au criblage virtuel.

I. La place de l'open source dans la découverte de médicaments.

A. La chemoinformatique.

Une publication récente retrace l'histoire de la chemoinformatique [1]. Le terme chemoinformatique (cheminformatics) a été introduit pour la première fois par Franck Brown en 1998 [2]. C'est un mimétisme du terme bioinformatique, discipline qui a connu un grand essor il y a une dizaine d'années. Le terme de cheminformatique (cheminformatics) est également employé. Chemoinformatique est légèrement plus employé dans les publications (pubmed donne 72 résultats pour chemoinformatics contre 69 pour cheminformatics) alors que d'une manière générale cheminformatics est plus employé (291 000 résultats dans google pour cheminformatics contre 161 000 pour chemoinformatics), au 23/08/2006. De plus Wikipedia a choisi le terme de cheminformatics, alors que Gasteiger a publié le « Handbook of Chemoinformatics » [3]. Nous avons pour notre part choisi d'utiliser le terme de chemoinformatique dans ce manuscrit.

La définition donnée par Gasteiger de cette science dans son ouvrage est la suivante : « La chemoinformatique est l'utilisation des méthodes informatiques pour résoudre des problèmes chimiques ». L'aspect qui nous intéresse ici se limite à la découverte de médicaments. Les principales grandes suites de logiciels dans ce domaine sont développés par Tripos [4], Accelrys [5], Chemical Computing Group [6], Schrödinger [7], OpenEye [8] et MDL [9]. Ces logiciels sont très bien implantés sur le marché. Les logiciels open source ont du mal à trouver leur place face à ces géants de la chemoinformatique. Curieusement, on retrouve la situation inverse en bioinformatique : les logiciels commerciaux ont du mal à s'imposer face à des alternatives open source ou académiques très utilisées. Des explications ont été avancées pour expliquer les raisons de ce contraste :

- Une histoire économique différente. Les méthodes innovantes en chimie théorique sont généralement valorisées par le développement d'un logiciel commercial. Même les équipes académiques vendent en général les algorithmes donnant de bons résultats,

dans certains cas en passant par la création d'entreprise. En bioinformatique au contraire, les logiciels open source ont atteint un tel stade de développement que le coût de développement d'un logiciel commercial avec des qualités suffisantes serait prohibitif.

- La disponibilité des données. Beaucoup de données, correspondant aux génomes notamment, sont publiques. Du côté de la chimie, même si l'on peut trouver certains jeux de données d'activité et de propriétés disponibles librement [10], la plupart ne sont pas publiés par les auteurs. Cette tendance devrait cependant tendre à s'inverser sous la pression des éditeurs car, en plus de freiner le développement de nouveaux modèles, cet état de fait diminue la qualité des publications scientifiques [11]. Le fait de ne pas publier les données ayant servi à la mise au point d'un modèle va en effet à l'encontre du guide d'éthique de l'ACS : « Un premier rapport de recherche doit contenir des détails et des références à des sources publiques d'informations suffisants pour permettre à des pairs des auteurs de reproduire le travail ».

  Le développement de logiciels open source nécessite dans certains cas de disposer de données (par exemple pour les logiciels de prédiction d'activité / propriétés), et le fait que celles-ci ne soient pas disponibles est donc un frein. En bioinformatique les données sont disponibles dans un petit nombre de formats. En chemoinformatique, il y a un grand nombre de formats. En effet les différentes spécialités utilisant à la fois l'informatique et la chimie (chimie quantique, spectroscopie, QSAR…) ont chacune des formats de fichiers qui leurs sont propres. A titre d'exemple, le logiciel Open Babel [12] permettant de convertir certains formats entre eux, en supporte 70.

  Cette multitude de formats est évidemment un frein pour le développement open source. Le Chemical Markup Language (CML), dérivé du XML, a été développé pour permettre de stocker toutes les informations relatives à la chimie [13, 14, 15, 16, 17, 18]. Cependant, même si son usage tend à se développer, le CML ne s'est pas encore imposé comme un standard [19].

B. Les logiciels open source

Nous n'allons pas donner ici une liste exhaustive des logiciels open source. Une liste de ce type de logiciels peut être trouvée dans la publication de Geldenhuys et al. [20]. On

citera cependant les principales bibliothèques de développement open source ou gratuites (Tableau 4).

Bibliothèques Open source	
Chemical Development Kit	http://almost.cubic.uni-koeln.de/cdk/
JOELib	http://www-ra.informatik.uni-tuebingen.de/software/joelib/
OpenBabel	http://openbabel.sourceforge.net
PerlMol	http://www.perlmol.org
Molecular Modeling Toolkit	http://starship.python.net/crew/hinsen/MMTK/
Bibliothèques Gratuites (pour les académiques)	
Marvin	http://www.chemaxon.com/marvin/
OELib	http://www.eyesopen.com/products/toolkits/oechem.html
CACTVS	http://www2.ccc.uni-erlangen.de/software/cactvs/index.html

Tableau 4. Exemple de bibliothèques de chemoinformatique gratuites.

Pour être considéré comme open source, un logiciel doit avoir son code source disponible, avoir un système de licence permettant une redistribution sans contraintes et permettant la création de travaux dérivés gratuits. Dans une publication récente, Delano [21] a mis en avant les avantages suivants. Ces logiciels sont disponibles gratuitement et immédiatement. Cela évite à l'utilisateur le désagrément d'acheter un logiciel pour s'apercevoir qu'il ne correspond pas à ses besoins. Même si le support technique est souvent une source de revenu pour les logiciels open source, l'utilisateur est libre de choisir qui lui fournira ce support. Outre la gratuité, l'autre aspect de l'open source est, comme son nom l'indique, la mise à disposition du code source. Ainsi, alors que les logiciels non open source se révèlent généralement être de véritables boîtes noires, il est possible de comprendre et de modifier le fonctionnement des logiciels open source. En cas de bug par exemple, pour un logiciel commercial classique, il faut signaler le bug au support technique, et espérer qu'il soit corrigé dans la prochaine version. Dans le cas d'un logiciel open source, le bug peut être corrigé par l'utilisateur, s'il a les compétences techniques requises. L'autre avantage de posséder le code source est que l'on peut adapter le logiciel à ses besoins. Il y a ainsi une

collaboration entre les utilisateurs et les développeurs pour créer des logiciels mieux adaptés aux besoins. D'autres avantages indirects de l'open source peuvent être cités. Alors que dans la partie précédente nous avons parlé de la multitude de formats existants en chemoinformatique, ce type de logiciels privilégie les formats ouverts préexistants, au lieu de chercher à développer d'autres formats propriétaires, limitant ainsi la prolifération de formats. Un autre avantage indirect est que le monde open source maintient une pression sur les grands développeurs privés en étant en compétition avec eux. Si les logiciels commerciaux ne sont pas constamment améliorés, ils courent le risque de se voir égalés voir dépassés par les logiciels open source. Un exemple bien connu de l'informatique grand public est celui des navigateurs internet : alors qu'il y a quelques années Internet Explorer était en situation de quasi monopole, le navigateur open source Firefox a vu son nombre d'utilisateurs augmenter de manière vertigineuse, la principale raison étant que les fonctionnalités d'Internet Explorer n'ayant pas évolué, les innovations sont venues du monde open source. Enfin, on citera comme dernier avantage que l'open source est au monde logiciel ce que les publications sont au monde scientifique : une fois qu'un problème est résolu dans un code open source, n'importe qui peut réutiliser ce code pour résoudre ce type de problèmes.

Bien sûr, les logiciels open source ne présentent pas que des avantages. La qualité des logiciels, de la documentation et du code source est très variable suivant les projets. Les interfaces graphiques sont généralement très primaires dans ce type de logiciels en chemoinformatique, et le plus souvent inexistantes. Ces problèmes peuvent s'expliquer par le fait que les spécialistes de chemoinformatique préfèrent généralement vendre leurs logiciels.

C. Perspectives de l'open source

1. Un peu plus d'organisation

L'un des principaux freins à l'essor des solutions open source en chemoinformatique est « l'éparpillement des projets ». Chaque développeur se concentre sur son projet, quitte à recoder certaines méthodes déjà existantes dans d'autres projets open source. Une tentative récente de centralisation de code open-source a été lancée par plusieurs développeurs de logiciel open source de chemoinformatique, sous le nom de « Blue Obelisk » [22]. Ce projet se décompose en trois grands axes :

- Open source : n'importe qui peut obtenir et réutiliser des algorithmes libres.

- Open standards : n'importe qui peut trouver des standards pour les protocoles et la communication d'informations.
- Open data : n'importe qui peut obtenir et utiliser les données qui sont dans le domaine public.

Le but de ce projet est d'améliorer l'interopérabilité des logiciels. Il est encore trop tôt pour prédire l'efficacité de cette initiative.

2. L'amélioration des interfaces graphiques

Nous avons déjà mentionné le fait que la qualité des interfaces graphiques était un problème pour les logiciels libres. C'est d'ailleurs un des points mis en avant par Gasteiger dans ses perspectives de la chemoinformatique [3]. Mais cela est aussi vrai pour les logiciels commerciaux en chemoinformatique : certains logiciels très reconnus dans le domaine possèdent des interfaces graphiques de mauvaise qualité. Avec l'évolution de l'informatique, cela tend à changer.

Le FORTRAN, conçu en 1957 par IBM, a longtemps été le langage informatique le plus utilisé pour les travaux scientifiques [23]. Il existe encore aujourd'hui dans un certain nombre de logiciels de chemoinformatique contenant du code en FORTRAN. Les langages C (en 1972) et C++ (en 1983) ont ensuite été développés par les laboratoires Bell, et sont devenus à leur tour des références. Enfin, Sun Microsystems a introduit le langage Java en 1995. Java est aujourd'hui le langage de programmation le plus populaire. Il a de nombreux avantages comme une très grande portabilité, et une intégration dans les navigateurs internet sous forme d'applets. Les points sur lesquels il a été critiqué à ses débuts, à savoir principalement ses performances, s'améliorent à chaque version. L'affichage tridimensionnel, très important en chemoinformatique, peut être mis en place, suivant que l'on recherche la simplicité de développement ou les performances, soit par Java3D, soit par JOGL (qui permet d'utiliser les librairies OpenGL).

L'évolution des techniques informatiques rend la conception de logiciels, et en particulier d'interfaces graphiques, de plus en plus simple. De plus, les outils de développements sont aujourd'hui facilement accessibles (un bon nombre sont gratuits). C'est d'ailleurs un autre bon exemple de la pression que peut exercer le monde open source sur les logiciels commerciaux. Ainsi Eclipse, un logiciel libre, est le plus utilisé des environnements

de développement Java. Afin de ne pas voir ses outils complètement délaissés, Sun a récemment décidé de mettre gratuitement à disposition des développeurs ses outils de développement (Java Studio Creator et Java Studio Enterprise). De même, Microsoft a sorti des versions gratuites de ses outils de développements (C#, C++, Visual Basic…) afin de continuer à attirer les développeurs vers ses systèmes. La gratuité de ces logiciels offre ainsi un vaste choix d'outils de développement aux programmateurs de logiciels open source qui ont souvent des budgets restreints.

II. Le logiciel ScreeningAssistant

A. Le projet

Les tests de criblages, qu'ils soient réels ou virtuels, consistent à mesurer ou estimer l'activité biologique d'un grand nombre de molécules. Cela nécessite donc de pouvoir gérer ces molécules, ce qui implique de maintenir à jour la liste de molécules que l'on veut cribler et de pouvoir choisir les molécules qui seront testées pour un projet donné. Nous réalisons au laboratoire des tests de criblages virtuels, principalement de docking. Nous utilisons pour ces tests une sélection de composés soit commerciaux, soit de l'ICOA, soit enfin de la Chimiothèque Nationale. Nous utilisions précédemment MOE [24] pour gérer ces bases, en combinaison avec une interface permettant de filtrer les composés [25]. MOE dispose d'un tableur moléculaire performant, cependant il n'est pas conçu pour traiter les problèmes directement liés à la gestion de chimiothèques. Il faudrait en effet par exemple que pour chaque nouvelle molécule ajoutée à la liste, le système puisse détecter si elle existe déjà, et si c'est le cas ne pas ajouter la structure, mais garder la nouvelle référence. Ce type d'opération n'est pas disponible par défaut dans MOE, ce qui rendait la gestion de bases de criblages difficile. Il a donc été décidé de mettre en place un système adapté à la gestion de chimiothèques destinées au criblage. Au début de ce travail de thèse le cahier des charges du logiciel a été défini comme suit :

Principaux points du cahier des charges de *ScreeningAssistant*
- le logiciel doit être gratuit ou très abordable
- le logiciel doit être utilisable par un chimiste médicinal (sans connaissances particulières en chemoinformatique)
- un seul exemplaire de chaque structure doit être stocké (cela implique une gestion efficace des doublons)
- la mise à jour des structures de la base à partir des fichiers SDF des fournisseurs doit être automatique
- la visualisation des propriétés (diversité, doublons, pourcentage de composés « drug-like »…) des bases des fournisseurs doit être possible
- le logiciel doit permettre de générer des ensembles de molécules de manière pertinente (cela implique le calcul de propriétés physicochimiques, la prédiction des caractères « drug-like » et « lead-like » des composés, la génération d'ensemble divers…)

Tableau 5. Cahier des charges de *ScreeningAssistant*.

Nous sommes également conscients que d'autres structures de recherche connaissent ce problème de gestion de base de criblage. En fait seuls les grands groupes pharmaceutiques ont les moyens de développer en interne des systèmes de gestion de chimiothèques performants. Les laboratoires universitaires n'ont souvent ni les moyens humains, ni les moyens financiers pour un tel développement. De même, les petites entreprises de biotechnologies, n'ont souvent ni les moyens ni une connaissance suffisantes en chemoinformatique pour ce développement. En partant de ce constat, nous avons décidé d'élargir les objectifs du projet et de développer un logiciel qui permettrait non seulement de gérer nos bases de composés destinées au criblage, mais qui pourrait également être utilisé dans cette optique par d'autres organismes. Cet objectif fixe des nouvelles contraintes pour le projet. Il faudra ainsi que l'interface graphique et la documentation soient de qualité suffisante pour être utilisées par n'importe quel utilisateur avec des connaissances de chimie médicinale, mais sans connaissances particulières en chemoinformatique. En outre, le logiciel devra être financièrement accessible.

B. Présentation technique

1. Architecture

Nous avons vu qu'il n'existe pas de logiciel complètement adapté pour la gestion d'une base de composés destinés au criblage, et que chaque organisme développe une solution de gestion en interne basée sur des outils plus ou moins bien adaptés à cette tâche. En simplifiant, nous ferons ressortir trois grandes composantes de ce système :

- un Système de Gestion de Bases de Données (SGBD). Les trois plus connus sont Oracle [26], MySQL [27] et SQLServer [28]. Ces logiciels permettent d'utiliser le langage structuré de requêtes (SQL) pour interroger et manipuler les bases de données relationnelles.

- un logiciel permettant la manipulation informatique de structures chimiques. Ils servent à lire les fichiers contenant des structures de molécules, traiter ces structures, calculer des descripteurs… Nous avons déjà vu une liste d'un certain nombre de logiciels gratuits de ce type (Tableau 4). D'autres logiciels commerciaux existent, comme le Daylight Toolkit [29]. Les outils Isis/Base et Isis/Host [30] permettent ce type d'opérations et sont en plus spécialement conçu pour gérer les chimiothèques, ce qui en fait une référence dans ce domaine. On notera également que le logiciel Activity Base [31] permet la gestion de bases de données chimiques. Pour gérer les chimiothèques de grandes tailles, ces logiciels peuvent interagir directement avec Oracle. L'outil PipelinePilot [32], qui permet de combiner l'utilisation de nombreux logiciels de chemoinformatique, peut aussi être utilisé pour la gestion de chimiothèques.

- un logiciel permettant d'effectuer des opérations poussées de modélisation moléculaire. En effet les logiciels permettant la manipulation informatique de structures chimiques ne disposent en général pas de fonctions de modélisation moléculaire avancées. Les opérations basiques utilisées dans une chimiothèque sont la conversion en 3D, la génération de conformations, ainsi qu'éventuellement le calcul de descripteurs spécifiques. Pour le passage rapide en 3D, les programmes les plus utilisés sont Corina [33] et Concord [34]. La génération de conformations peut quant à elle être réalisée par la plupart de grandes suites logicielles de modélisation, à savoir entre-autres Sybyl [35], MOE [24] et Omega [36] de la suite de logicielles de

OpenEye. Le calcul de descripteurs spécifiques ainsi que d'autres opérations complexes peuvent également être réalisés par ces suites logicielles.

Des exemples de systèmes de gestions de chimiothèques ont été publiés dans la littérature. La base de composés de criblage de Novartis est ainsi gérée en utilisant les logiciels Oracle, Isis/Host et Isis/Base, et Sybyl [37]. Les créateurs du projet ZINC (une base de données regroupant des composés commerciaux, principalement dédiée au docking) utilisent entre autres MySQL pour stocker les données, la bibliothèque OEChem pour manipuler informatiquement les molécules, Omega pour créer les conformations, LigPrep [38] pour donner le bon état de protonation aux structures, la suite CACTVS et le logiciel de Molinspiration [39] pour la correction d'erreurs, l'uniformité, et le calcul de propriétés. Ces deux exemples montrent qu'il existe de nombreuses manières de gérer les chimiothèques.

Comme nous l'avons déjà énoncé précédemment notre objectif est de développer un logiciel gratuit. Nous avons donc tenté de privilégier au maximum l'intégration de composants gratuits dans notre système.

Nous n'avons pas continué d'utiliser MOE pour gérer les chimiothèques, comme cela était le cas précédemment au laboratoire. Ce choix a été réalisé non pas parce que ce logiciel est commercial (bien que cela eut été un problème), mais parce que le développement d'un système avec MOE aurait reposé sur un code SVL. Malgré le fait que ce langage vectoriel se révèle être très efficace pour les opérations sur les molécules, il est totalement lié à MOE. Cela implique un certain nombre d'inconvénients, notamment que les fonctions disponibles (autant celles développées par les auteurs du langage que celles développées par d'autres contributeurs), en dehors de la gestion de molécules, sont plus limitées qu'un langage classique. Il a semblé à priori qu'un programme développé en Java ou en C++ (les deux langages les plus utilisés à l'heure actuelle) serait un meilleur choix. Le Java ayant notre préférence du fait de sa grande portabilité et de sa facilité de développement.

Cependant, le choix du langage a été complètement lié à celui de la bibliothèque de chemoinformatique choisie. Nous avons évidemment fait notre choix parmi celles disponibles en open source ou gratuitement. Les deux qui nous ont semblé le plus adapté à nos besoins sont JOELib et le CDK. Notre choix s'est porté sur JOELib, qui est basé sur un portage

d'OpenBabel en Java, car cette bibliothèque dispose d'un grand nombre de descripteurs, ainsi que d'un type de fingerprint. Notre projet a donc été développé en Java.

Au début du projet, trois SGBD ont été considérés : Oracle, MySQL, et PostgreSQL. Nous avons d'abord éliminé Oracle car les deux autres bases sont gratuites et suffisamment puissantes pour répondre à nos besoins. Parmi ces deux bases nous avons choisi MySQL, car sa communauté d'utilisateurs est plus développée, ce qui permet de trouver plus rapidement des réponses en cas de problèmes techniques.

Enfin, un certain nombre d'autres logiciels ont été utilisés lors du développement. Pour la génération des structures 3D, Corina est utilisé. Les conformations sont générées avec Omega. Le choix de ces deux logiciels peut paraître surprenant au vu de notre objectif de développer un logiciel gratuit. Cependant il n'existe pas, à notre connaissance, de logiciels gratuits suffisamment performants pour ces deux opérations. L'inconvénient d'intégrer ces deux logiciels commerciaux à notre projet peut être relativisé par deux points. Premièrement, Omega est gratuit pour les laboratoires académiques, et Corina est disponible avec un coût relativement restreint pour une licence définitive. Deuxièmement, aucune de ces applications n'est indispensable pour l'utilisation de *ScreeningAssistant*. En effet il n'est pas nécessaire de disposer de la structure 3D des molécules si l'on travaille sur un projet de criblage réel. Si l'on travaille sur un projet de criblage virtuel nécessitant les structures 3D ou les conformations, il est toujours possible d'utiliser *ScreeningAssistant* pour toutes les étapes de gestion et de sélection de chimiothèque, d'exporter les composés choisis en SDF, et de générer les structures 3D ou les conformations avec les applications dont dispose l'utilisateur. Nous avons également utilisé le logiciel commercial Marvin pour la visualisation des structures. Ce logiciel est gratuit pour les laboratoires académiques. De plus, Si le logiciel Marvin n'est pas installé sur l'ordinateur de l'utilisateur, un programme de visualisation moins évolué permet tout de même de visualiser les structures.

Outre ces logiciels commerciaux, nous avons utilisé le programme InChI [40] pour la génération d'un code unique des structures chimiques. Nous reviendrons en détails sur ce logiciel plus loin. Un outil permettant d'améliorer l'aspect de l'interface graphique ainsi qu'un autre facilitant le placement des composants graphiques ont aussi été utilisés [41]. Enfin, pour visualiser les caractéristiques des chimiothèques, nous avons utilisé la bibliothèque de génération de graphiques JFreeChart [42].

2. Qu'est-ce qu'un doublon ?

Si certaines notions utilisées dans ce document sont difficiles à définir, comme celle des composés « drug-like » ou la diversité, ce n'est à priori pas le cas du terme de doublons : un doublon est une molécule qui, dans l'ensemble de composés considéré, a une structure identique à une autre molécule de cet ensemble. Dans la pratique cependant, il est nécessaire d'apporter des précisions à cette définition.

La notion de doublon est donc absolue d'un point de vue structural, mais elle est relative pour les biologistes, ainsi que pour certains chimistes. Deux molécules pourront être considérées comme des doublons absolus si elles ont les mêmes atomes, les mêmes liaisons entre atomes, et la même stéréochimie. Pour identifier les doublons dans une base de donnée il faut donc une méthode permettant de caractériser de manière unique une molécule. Nous présenterons plus loin le logiciel que nous avons choisi pour effectuer cette tâche.

La notion de doublon peut être complexifié par la prise en compte de différents facteurs :

- Les sels : la prise en compte ou non des contre-ions dans les sels pour la caractérisation des doublons peut être sujet à débat. Dans le cas d'un criblage virtuel de type de docking, les contre-ions ne sont pas pris en compte. Le contre-ion peut cependant avoir une influence indirecte sur l'activité biologique. Nous avons fait le choix, après discussion avec des chimistes, de ne pas prendre en compte les contres-ions pour l'identification des doublons. Donc, dans une chimiothèque, deux sels différents de la même molécule seront considérés comme des doublons.

- L'état de protonation : le fait que deux molécules identiques soient représentées avec des états de protonations différents peut faire qu'elles ne soient pas reconnues par les logiciels comme étant des doublons. Il faut donc procéder à une standardisation des états de protonations avant la génération d'un code unique pour la molécule. Nous avons défini les états de protonations à pH physiologique avec JOELib.

- Les tautomères : les formes tautomères correspondent à des représentations limites de la même molécule. Mais, si la même molécule apparaît dans la base sous deux formes différentes, beaucoup de logiciels ne les considèrent pas, à

tort, comme des doublons. Deux solutions existent pour gérer ce cas particulier : générer tous les tautomères de chaque structure et les prendre en compte pour la recherche de doublons, ou bien utiliser un code unique qui est capable de prendre en compte les formes tautomères. Nous avons utilisé cette dernière solution.

- La stéréochimie : la stéréochimie doit évidement être prise en compte. Comme nous le verrons par la suite, elle est plus ou moins bien gérée en fonction des logiciels qui génèrent un code unique. Un autre problème lié à la stéréochimie est sa représentation : les logiciels ne prennent en compte de manière fiable que la stéréochimie explicite (c.-à-d. qui utilise la représentation liaisons avant / arrière). Il est fréquent de trouver dans les chimiothèques des structures avec des stéréochimies non définies ou mal définies, ce qui fausse la détection des doublons. De plus les glucides sont souvent représentés avec une stéréochimie implicite. Il faut faire un choix pour le traitement des centres asymétriques dont la stéréochimie n'est pas définie : un carbone dont la stéréochimie n'est pas définie est-il identique ou bien différent des carbones R ou S. Certains logiciels permettent de définir comment traiter ce type de carbones. Nous avons fait le choix de considérer que les carbones R, S et indéfinis sont trois types de carbones bien différents.

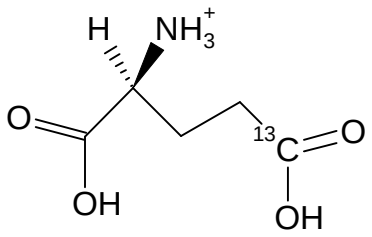
- Les isotopes : les isotopes sont également à prendre en compte lors de la recherche des doublons, et le code unique utilisé doit donc gérer cette propriété. Deux molécules ne se différenciant que par l'état isotopique d'un atome seront donc considérées comme étant des structures différentes par notre système.

- La mauvaise représentation des molécules : des erreurs peuvent se produire au moment de la représentation de la structure par le chimiste. En plus des erreurs de stéréochimie dont nous avons parlé, d'autres erreurs plus grossières sont fréquemment rencontrées dans les chimiothèques (oubli ou rajout d'une liaison, erreur sur l'ordre d'une liaison…) : cela pose un réel problème lors de l'identification des doublons.

La recherche de doublons repose donc sur l'utilisation d'un code unique efficace. Nous allons présenter le code unique que nous avons choisi d'utiliser dans notre système, et le comparer à d'autres logiciels.

3. Identification des molécules par un code unique

Nous avons comparé les qualités des codes uniques générés par quatre logiciels : InChI, MOE, OEChem et Marvin. InChI est un projet débuté par l'IUPAC en 2000 qui est aujourd'hui arrivé à maturité. C'est un logiciel gratuit qui a pour objectif de caractériser les molécules par une ligne de texte. Il a l'avantage par rapport au numéro CAS de permettre de générer un code unique sans qu'il soit besoin de se connecter à un serveur centralisé. De plus il est possible de générer la structure à partir du code InChI, bien que cette fonctionnalité soit encore peu implémentée dans les logiciels actuels. Ce code prend en charge les stéréochimies sp2 et sp3, les isotopes, et les formes simples de tautomérisme. Il a de très bonnes fonctionnalités par rapport aux autres logiciels que nous avons testés, qui utilisent des approches basées sur un code SMILES unique. Une comparaison des fonctionnalités de ces quatre logiciels est disponible dans le Tableau 6. Il en ressort que le code InChI est celui qui a le plus de fonctionnalités. C'est le seul à gérer les formes tautomères simples, et les charges positives mobiles. La Figure 5 présente un exemple de code InChI.



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

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

Figure 5. Structure de l'acide (S)-Glutamique, marqué au ^{13}C et dans un état de protonation particulier, comme fourni dans la documentation du code InChI (à gauche), détails des différentes parties du code (à droite), et le code généré (en bas).

Fonctionnalité	InChI	MOE	OEChem	Marvin
Stéréochimie sp3	X	X	X	X
Stéréochimie sp2	X	X	X	X
Tautomérie simple C(O)-[NH]	X			
Tautomérie céto-énolique				
Représentation des NO2 :	X	X		
N(=O)=O et [N+](=O)-[O-]				
Détection d'une charge positive mobile *	X			

* Exemple de la documentation InChI :

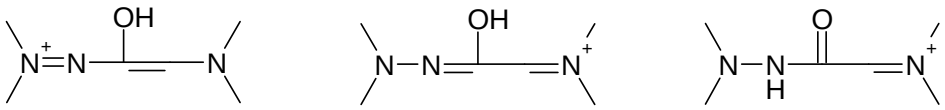


Tableau 6. Comparaison des codes uniques de quatre logiciels.

En plus de ces tests de fonctionnalités, qui ont été réalisés sur quelques exemples bien précis, nous avons voulu tester les performances de ces codes pour détecter la présence de doublons dans des chimiothèques. Il apparaît que, comme attendu, le nombre de doublons trouvés varie en fonction des logiciels. Nous avons comparé attentivement toutes les structures des doublons isolés par ces logiciels afin d'identifier les points forts et les lacunes de chaque logiciel. Nous avons utilisé pour cela quatre chimiothèques. Les résultats de cette étude sont disponibles dans le Tableau 7.

Chimiothèque (composés)	InChI	MOE	OEChem	Marvin
Prestwick (1120)	1 117	1 116	1 116	1 102
ICOA (3272)	3 213	3 214	3 216	3 195
NCI (260071)	244 321	242 776	243 165	244 584
ChemBridge (425953)	425 941	425 944	425 944	425 944

Tableau 7. Nombre de composés uniques trouvés dans chaque chimiothèque par différents logiciels.

La base Prestwick ne contient aucun doublon réel. Cependant, étant donné que nous avons supprimés les contre-ions dans notre étude, deux doublons doivent être trouvés. De plus, deux diastéréoisomères ont un carbone asymétrique non défini, ce qui entraine la détection d'un autre doublon. Tous les logiciels trouvent ces trois doublons. InChI n'en trouve aucun autre. MOE, OEChem et Marvin en trouvent un autre, qui est dû à la présence de deux stéréoisomères ; l'un a tous ses carbones asymétriques définis correctement (Loracarbef), l'autre a une erreur dans la définition d'un carbone asymétrique (Cefaclor) : une liaison « en avant » ne part pas du carbone mais de l'azote. InChI arrive à distinguer ces deux structures, mais pas MOE, OEChem et Marvin. Enfin, Marvin a des problèmes pour gérer la stéréochimie dans son code unique ce qui entraîne un certain nombre de faux doublons. Pour cette raison nous n'approfondirons pas les performances de Marvin dans la suite de ce comparatif.

Nous avons continué à comparer les résultats des logiciels InChI, MOE, OEChem sur une base un peu plus grande : celle de l'ICOA. Comme n'importe quelle base de composés internes, certaines structures comportent des erreurs qui ont été commises lors de l'entrée des structures. Par exemple, deux structures identiques existent, mais l'une d'entre elles a été a été dessinée avec quatre doubles liaisons dans un cycle aromatique de type benzène ! Seule InChI a été capable détecter cette erreur et d'identifier ces deux doublons. Une des structures de la base a un atome de deutérium noté « D ». MOE a transformé cet atome en hydrogène, et détecté ce composé comme étant identique à un autre composé de la base. MOE est le seul à considérer ces deux molécules comme des doublons. La chimiothèque de l'ICOA contient aussi des sucres représentés par la projection d'Haworth. De ce fait, la stéréochimie peut

difficilement être réinterprété informatiquement en utilisant la structure 2D. InChI trouve deux doublons pour ce type de molécules, alors que MOE et OEChem n'en trouvent aucun. Les autres différences dans le nombre de doublons identifiés sont dues, comme nous l'avons déjà vu avec la base Prestwick, à des carbones asymétriques non définis ou mal définis.

Nous avons également étudié le nombre de doublons trouvés dans de grandes bases telles que NCI et ChemBridge. La base NCI a beaucoup de doublons ; elle est une bonne illustration du fait que le nombre de doublons trouvés dans une base est dépendant du logiciel utilisé. ChemBridge à quant à elle a peu de doublons et, en conséquence, le nombre de doublons trouvés par les différents logiciels n'est pas très différent.

Les résultats de ces tests fait ressortir le classement suivant par ordre de performances décroissantes pour la génération d'un code unique : InChI, MOE, OpenEye et Marvin. InChI est donc le plus performant, et c'est ce logiciel qui sera utilisé dans *ScreeningAssistant* pour générer un code unique.

Il est important de signaler que les structures avec une stéréochimie mal ou non définie sont une des raisons majeures de la présence de doublons. Ces erreurs de structures se produisent quand les composés sont entrés dans les bases des fournisseurs. Comme la stéréochimie est utilisée pour l'analyse de doublons, ces erreurs peuvent fausser cette étape. Deux choix peuvent être fait pour gérer les centres asymétriques non définis : ils peuvent être considérés comme différents des carbones R et S (c'est le choix de InChI par défaut), ou peuvent être considérés comme identiques à la fois aux carbones R et S (c'est le choix de MOE, OEChem et Marvin). Nous pensons que le choix d'InChI est le plus adapté pour notre système.

4. Insertion et traitement des composés

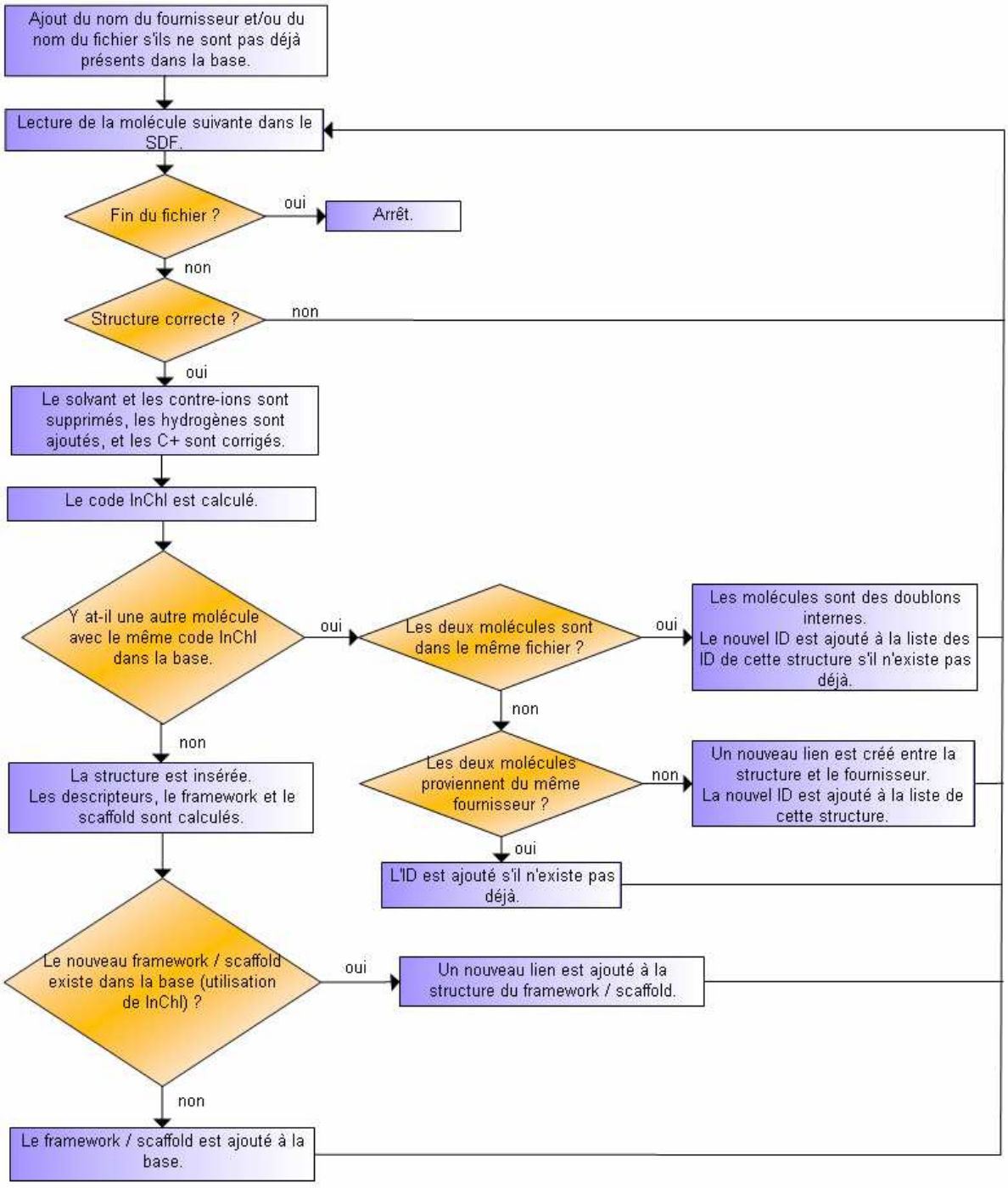


Figure 6. Algorithme utilisé pour l'insertion des composés dans la base.

Le logiciel est conçu pour gérer une chimiothèque dont les composés proviennent de différents fournisseurs. Le principe est donc de détecter les doublons et de ne garder qu'un

exemplaire d'une même structure. On gardera cependant les différentes références correspondant à une même structure.

Les composés sont insérés dans la base à partir de fichiers SDF, comme décrit dans l'algorithme Figure 6. Tout d'abord, le nom du fournisseur et/ou du fichier sont ajoutés dans la base de données s'ils ne sont pas déjà présents. Chaque molécule est lue dans le SDF. La molécule n'est pas traitée si une erreur importante dans la structure la rend ininterprétable par JOELib. Ensuite, le solvant et les éventuels contres ions sont supprimés si nécessaire. Cette étape est réalisée en ne gardant que le plus grand ensemble d'atomes liés présent dans la structure (c.-à-d. la plus grande molécule). Si des carbones avec des charges positives sont présents, ils sont corrigés en mettant la charge à 0. Les hydrogènes sont ajoutés si nécessaire, les bases sont protonées et les acides déprotonés en utilisant les règles de protonations à pH physiologique implémentées dans JOELib. Ces choix ont été faits afin de garder des molécules prêtes à être utilisées par les logiciels de criblages virtuels. Un code unique est calculé pour la molécule en utilisant InChI. Afin de détecter la présence de doublons, le code de hachage MD5 de l'InChI de cette molécule est comparé aux codes de hachage MD5 d'InChI indexés de tous les composés de notre base [i]. Si deux structures ont les mêmes codes MD5, leurs codes InChI sont alors comparés pour vérifier si ce sont bien de vrais doublons, même si la probabilité que deux molécules aient le même code MD5 sans avoir le même code InChI est très faible (il y a 2^{128} possibilités de codes MD5, la probabilité de collision est donc très faible). L'utilisation de codes de hachage MD5 permet de rechercher les doublons dans une table plus petite avec des champs de taille fixe, ce qui est beaucoup plus rapide.

Si la structure n'est pas présente dans la base, elle est insérée. Les descripteurs, le framework, et le scaffold (voir définitions dans la partie II.B.8 de ce chapitre) sont calculés automatiquement. Les listes des frameworks et des scaffolds uniques sont stockées dans la base de données et, comme pour les molécules, ils sont identifiés par leur code InChI. Seuls les nouveaux frameworks et les nouveaux scaffolds sont ajoutés à la base.

Si la structure est déjà présente dans la base, elle peut être soit un doublon, soit la même structure si le fichier est une mise à jour d'un fichier précédent du même fournisseur (les fournisseurs donnent souvent des fichiers de mise à jours de bases qui contiennent à la fois les anciennes et les nouvelles structures). Une solution pour déterminer si un composé est

[i] Un code de hachage est le résultat de la conversion par une fonction de hachage d'un grand ensemble en un ensemble plus petit. La fonction MD5 permet ainsi de convertir un ensemble de données de n'importe quelle taille en une chaîne de 32 caractères hexadécimaux (128 bits).

un vrai doublon serait d'utiliser l'identifiant de ce composé chez le fournisseur (donnée qui est stockée dans notre base). Si cet identifiant original est déjà présent dans la base, alors le composé provient d'un fichier de mise à jour et ne doit pas être considéré comme un doublon (nous comptabilisons à titre d'information le nombre de doublons par fournisseurs). Cependant, cela signifie que si le fournisseur change son système d'identifiants (ce qui se produit de temps en temps) beaucoup de faux doublons seront considérés. En prenant cela en compte, nous avons choisi de ne comptabiliser les doublons que dans les mêmes fichiers. Si le nom de fichier contenant le composé à insérer est le même que le nom de fichier du composé déjà présent dans la base alors ces composés sont considérés comme des doublons. Si les deux composés sont différents alors les noms de leurs fournisseurs sont comparés. Si les composés sont issus du même fournisseur, alors l'identifiant du composé à insérer est ajouté à la liste des identifiants de cette structure s'il n'est pas déjà présent. Si les composés ne proviennent pas du même fournisseur, l'identifiant original de ce fournisseur est ajouté.

5. Base de données

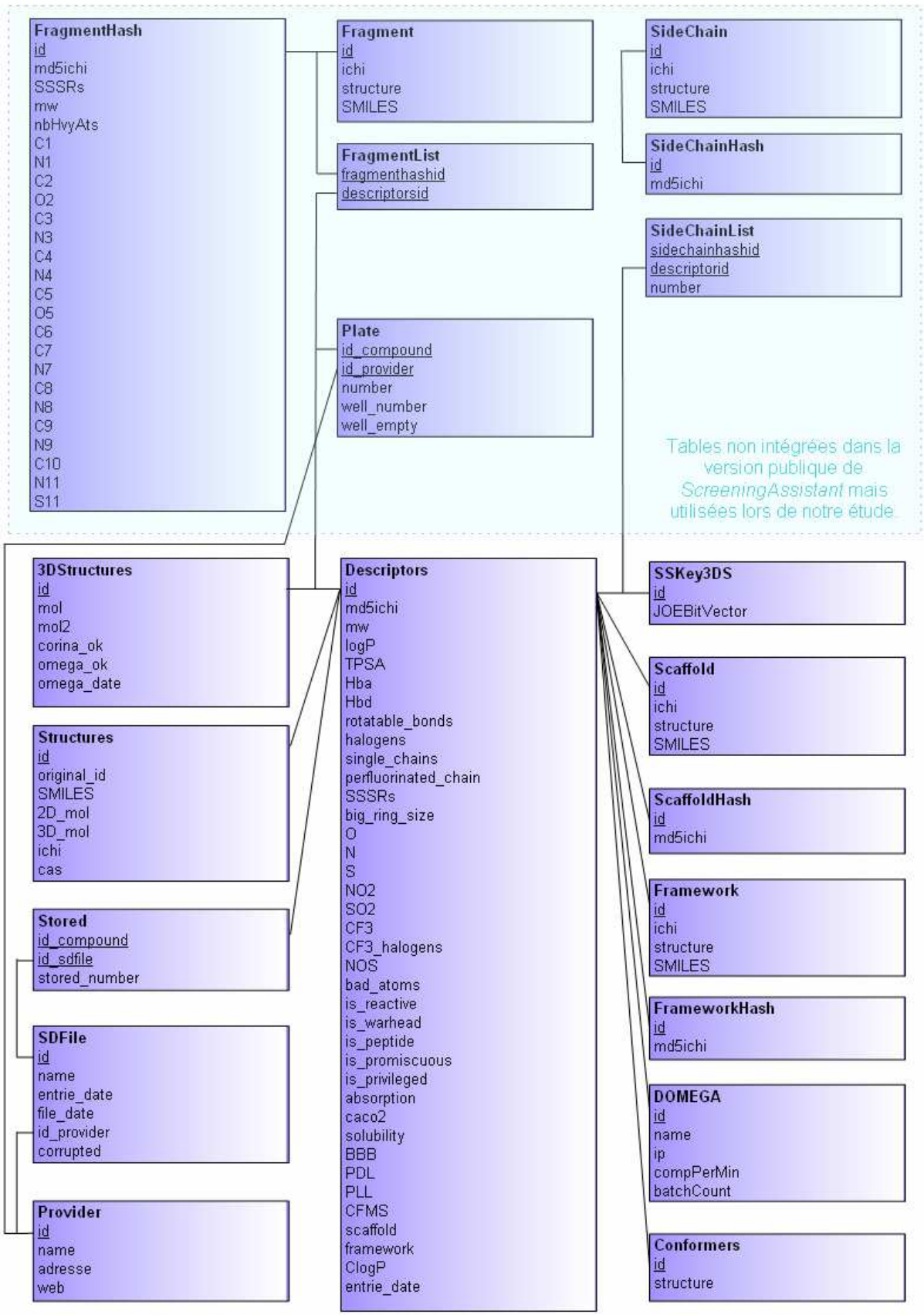


Figure 7. Structure de la base de données utilisée par *ScreeningAssistant*.

Les données sont stockées dans une base MySQL. La structure de cette base est présentée Figure 7. La table principale est *Descriptors*. Comme son nom l'indique, celle-ci regroupe les valeurs des principaux descripteurs, mais aussi différentes informations comme la présence de différents types de fonctions réactives, de peptides et de structures privilégiées entre autres. Les valeurs des scores PDL et PLL (que nous décrirons par la suite) sont également présentes dans cette table. D'autres champs correspondant à la perméabilité Caco-2, la solubilité, la capacité à franchir la barrière hémato-encéphalée et la valeur du ClogP ont été prévus pour un usage futur. Nous avons également intégré à cette table le MD5 du code InChI. Celui-ci étant de taille fixée, il ne perturbe pas les performances des requêtes effectuées sur cette table. Ce champ a été indexé afin de permettre une recherche très rapide des molécules en fonction du MD5 de leur code InChI, ce qui permet de détecter les doublons lors de l'insertion de molécules de manière efficace. D'une manière générale, tous les codes MD5 de notre base ont été indexés. De même, le code InChI stocké dans la table *Structure* est lui aussi indexé. Pour la gestion des Scaffolds et des Frameworks, les codes InChI ont été placés dans des tables à part (*ScaffoldHash* et *FrameworkHash*), toujours dans un souci de performances.

On notera également que le schéma de structure de la base contient sept tables qui ne sont pas intégrées directement dans la version actuelle de *ScreeningAssistant*. Nous avons été amenés à utiliser ces tables pour réaliser différents travaux. Les chaînes et les fragments ont été utilisés pour réaliser l'étude de la diversité des bases commerciales, que nous présenterons dans ce chapitre. Nous avons rajouté les informations concernant les liaisons qui ont été coupées afin de créer chaque fragment (C1, N1, C2…). Cela a été réalisé pour permettre d'utiliser ces fragments afin de créer des molécules virtuelles, pour des applications de type *de novo*. Même si cette fonctionnalité n'est pas présente dans *ScreeningAssistant*, nous avons jugé utile de prévoir ce type d'application lors de la conception de la base de données. Enfin, la table *Plate* a été utilisée pour la sélection de composés déjà mis en plaques, présentée dans le chapitre III.

L'avantage du serveur MySQL est qu'il peut être utilisé sur PC Windows très simplement par l'intermédiaire d'EasyPHP [43]. Pour une utilisation plus poussée et plus fiable, une version Linux de MySQL doit être utilisée. Pour notre part, le serveur MySQL est installé sur un PC équipé d'un processeur AMD 64 3800+ et de 4Go de RAM, avec un système d'exploitation Linux 64 (Soit une machine d'environ 1000 € et donc accessible à tout laboratoire).

6. Interface graphique

Un soin particulier a été apporté à la conception de l'interface graphique du logiciel (Figure 8). C'est assez rare pour un logiciel gratuit, pour la simple et bonne raison que les logiciels de chemoinformatique gratuits sont développés par des chercheurs académiques, et qu'un travail de conception d'interface graphique est difficilement valorisable dans le domaine de la recherche. Cependant l'interface graphique est un élément important qui peut faire que l'utilisateur va adopter ou non un logiciel.

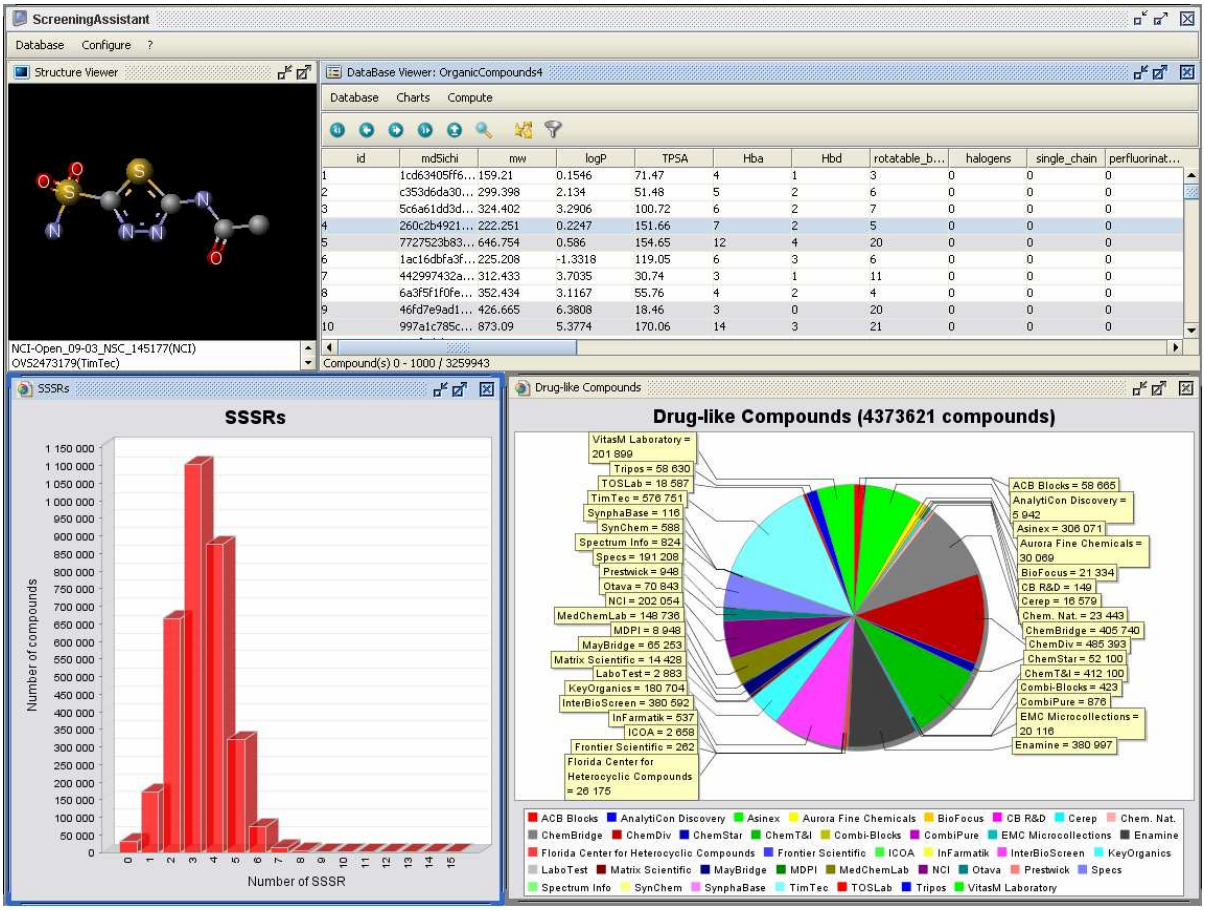


Figure 8. Interface graphique du logiciel *ScreeningAssistant*.

Java est un langage portable, et donc par définition les interfaces graphiques de Java le sont aussi. Pour cette raison, Java ne peut pas utiliser les composants graphiques des systèmes d'exploitation : il utilise donc ses propres composants de la bibliothèque Swing. Ces

composants avaient pour inconvénient d'avoir un aspect peu attrayant (cela s'est amélioré dans la dernière version du langage). Nous avons donc cherché à améliorer cet aspect. Nous nous sommes basé pour cela sur les outils JGoodies [41] qui ont été développés dans cette optique. Ces outils sont intervenus à trois niveaux :

- Changement de l'aspect de l'interface, notamment avec l'ajout d'effet 3D sur la barre des menus.
- Utilisation du layout (sorte de grille utilisée pour disposer les composants dans les fenêtres graphiques) JForm.
- Application des conseils du site Internet de JGoodies pour le design des interfaces graphiques.

Du point de vue graphique, nous avons également implémenté la possibilité de visualiser les résultats d'analyse des bases. Les graphiques sont générés par la bibliothèque JFreeChart [42].

Le dernier aspect important de l'interface graphique est la visualisation des structures des composés chimiques. Nous avons pour cela choisi d'utiliser Marvin qui propose un outil très bien conçu. Cependant, Marvin n'est gratuit que pour les académiques. Nous avons donc également intégré un autre outil de visualisation destiné aux utilisateurs ne disposant pas d'une licence de Marvin. Notre choix s'est porté vers le module de visualisation de JOELib.

7. Les fingerprints pour la mesure de la diversité

Les fingerprints[ii] sont des descripteurs largement utilisés pour des évaluations de diversité ou de similarité. Nous avons choisi d'utiliser les SSKey-3DS [44]. Ces fingerprints sont constituées de 32 bits codant pour la présence (ou l'absence) de 32 fragments, et de 22 bits qui codent pour le nombre de donneurs de liaisons H, le nombre de liaisons aromatiques, et la fraction de liaisons pouvant tourner (Figure 9). Nous avons utilisé notre propre version corrigée des fingerprints implémentés dans JOELib pour cette étude (voir Annexes).

[ii] Nous avons gardé le terme anglosaxon fingerprint, car la traduction française « empreinte digitale » n'est généralement pas employé en chemoinformatique. Nous utiliserons ce mot au masculin.

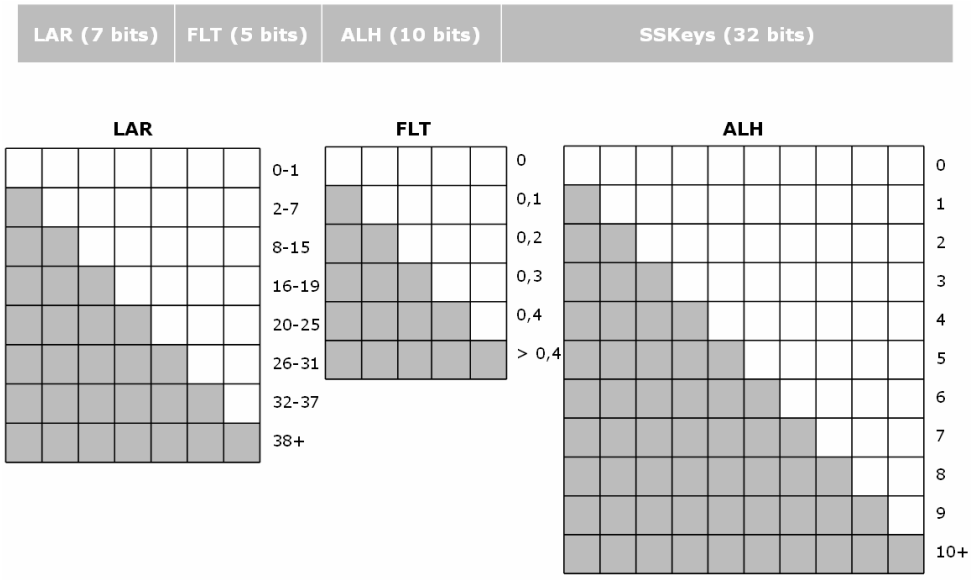


Figure 9. Constitution de la fingerprint SSKey-3DS (d'après [44]). ALH correspond au nombre d'accepteurs de liaisons H, LAR au nombre de liaisons aromatiques, et FLT à la fraction de liaisons pouvant tourner. Pour chacun de ces descripteurs chiffrés, un tableau contenant toutes les solutions possibles est représenté. Par exemple, le tableau LAR correspond aux différents codages du nombre de liaisons aromatiques. Si la molécule a 0 ou une liaison aromatique, aucun bit des 10 bits réservés au codage de cette propriété n'est activé, si la molécule a entre 2 et 7 liaisons aromatiques, un bit est activé, et ainsi de suite. Si la molécule a 38 liaisons aromatiques ou plus, les 10 bits de LAR sont activés.

D'après les auteurs de la publication décrivant cette méthode, les fingerprints de petites tailles sont moins sensibles que les fingerprints complexes à des changements de structures mineures et donc, sont plus performantes pour reconnaître des composés avec des activités biologiques similaires. *ScreeningAssistant* utilise l'algorithme Stochastic Clustering Analysis (SCA) [45] pour identifier le nombre de clusters comme cela a été fait dans une étude de la base NCI [46]. A la place d'utiliser le score de Carhart, nous avons utilisé le coefficient de Tanimoto qui est le plus utilisé pour comparer la similarité de fingerprints :

$$\textit{Coefficient de Tanimoto} = \frac{C}{A+B-C} \qquad \textbf{Équation 1}$$

avec *A* le nombre de bits à 1 dans le fingerprint *A*, *B* est le nombre de bits à 1 dans le fingerprint *B* et *C* est le nombre de bits à 1 dans les fingerprints *A* et *B*.

L'algorithme SCA identifie les molécules qu'il considère comme étant des centres de clusters. Le nombre de centres de clusters est égal au nombre de clusters. L'algorithme SCA permet donc d'obtenir le nombre de clusters d'un ensemble de molécules, ce qui donne une information de la diversité de ces molécules.

L'algorithme procède de la manière suivante :

- La première molécule de la base est considérée comme étant un centre de cluster.
- Ensuite, pour chacune des molécules suivantes, la similarité avec chaque centre de cluster déjà identifié est évaluée. Si pour une molécule tous les scores de similarité sont inférieurs à une valeur de seuil fixée au préalable, cela signifie que les composés ne se trouvent dans aucun cluster existant. Un nouveau cluster est donc créé en ajoutant la molécule à la liste des centres de clusters.

Nous avons choisi de fixer la valeur du seuil de similarité à 0,8 dans *ScreeningAssistant*. Nous avons choisi cette valeur après une série de tests. Elle permet de générer assez de clusters pour analyser des bases de très petites tailles, tout en donnant des clusters qui ont un sens chimique (c.-à-d. avec une bonne similarité entre les composés d'un même cluster). Des exemples fictifs de mesure d'espaces chimiques par l'intermédiaire de clusters sont présentés Figure 10.

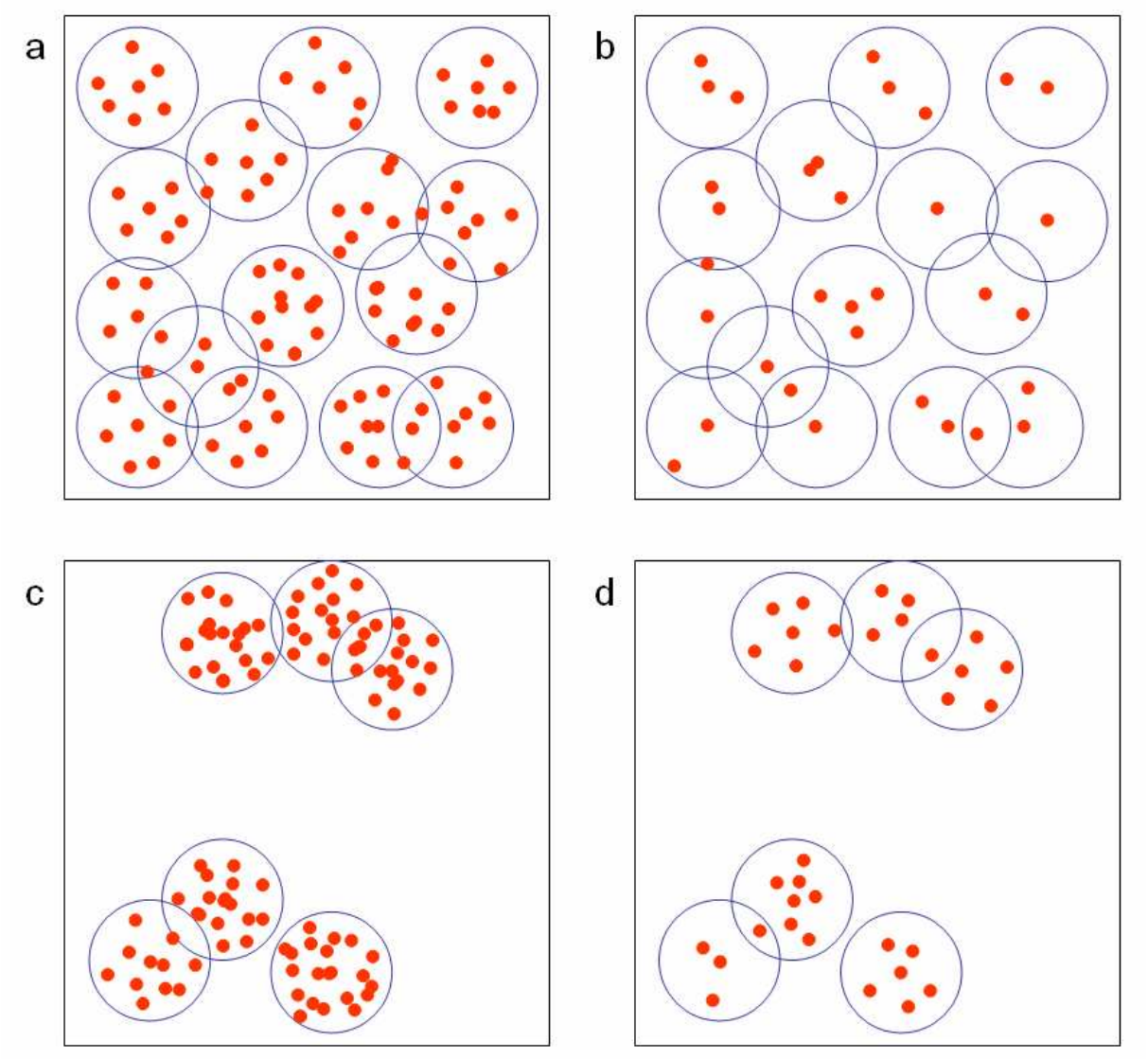


Figure 10. Exemples fictifs d'espaces chimiques hyper dimensionnels projetés en deux dimensions. Les espaces sont analysés avec l'étape de diversité de l'algorithme Stochastic Clustering Analysis. Les points représentent les molécules, et les cercles les clusters. Le nombre de clusters donne une estimation de la diversité. Chaque cluster a au moins un composé : celui se trouvant en son centre. Un composé peut se trouver à l'intérieur de plusieurs clusters. L'ensemble **a** a plus de composés que l'ensemble **b**, mais les deux bases ont le même nombre de clusters (cette observation est également valable pour les ensembles **c** et **d**). L'ensemble **c** a le même nombre de composés que l'ensemble **a**, mais les composés de **c** couvrent une partie plus petite de l'espace chimique, et le nombre de clusters de **c** est plus petit que celui de **a** (cette observation est aussi valable pour **b** et **d**).

Le principal avantage de cet algorithme réside en sa capacité à estimer la diversité de millions de composés très rapidement. Son seul inconvénient est de ne pas estimer le nombre de composés au sein de chaque cluster. Cependant, comme un cluster a au moins un composé cette méthode donne une bonne estimation de l'espace chimique couvert. Il faut également noter que le nombre de clusters trouvé est à priori dépendant des choix du seuil de similarité et de la première « sonde ». Afin d'illustrer cette dépendance, nous avons étudié l'impact de l'ordre des composés sur le nombre de clusters obtenu par l'algorithme SCA (Tableau 8).

Classement :	défaut	1	2	3	4	5	6	7	8	9	10
Chim. Nat.	1997	2028	2071	2111	2149	2026	2058	2076	2038	2004	2027
NCI	10623	11379	11062	11589	11413	10864	11179	10948	10990	10595	10716

Tableau 8. Nombre de clusters trouvés par l'algorithme SCA pour la Chimiothèque Nationale et la base NCI classées en fonction de différents classements.

La chimiothèque nationale (14946 produits) et la base NCI (244321 produits) sont choisies comme librairies représentatives pour faire cette évaluation. Le nombre de clusters est calculé pour chaque base avec les composés ordonnés par défaut, et par 10 autres critères. Le clustering a ensuite été lancé pour chaque version de chaque base. Pour la chimiothèque nationale, le nombre moyen de clusters est de 2053, avec une déviation standard de 46 (2,3 %). La base NCI a quant à elle un nombre moyen de clusters de 11033, avec une déviation standard de 331 (3 %). Ces résultats donnent un ordre d'idée de la précision de la méthode. Une solution pour obtenir des clusters identiques pour un même ensemble de molécules est de classer les composés par masses [47].

Nous avons programmé cet algorithme en Java. Le nombre de clusters d'une base de 2,6 millions de molécules uniques est calculé en moins d'une heure sur un PC PIV 3GHz avec 2Go de RAM.

8. Les frameworks et les scaffolds comme mesure de la diversité

Alors que les fingerprints analysent les molécules par des petits fragments sous-structuraux, les frameworks permettent d'avoir une information sur la forme générale du squelette de la molécule. Ces deux notions sont donc complémentaires pour estimer la diversité. Nous utiliserons également les notions de scaffolds pour caractériser la diversité d'ensembles de molécules. Les frameworks et les scaffolds ont été introduits par Bemis et Murcko [48, 49] pour analyser les caractéristiques structurales des médicaments.

Les frameworks correspondent à une modification de la structure moléculaire qui ne garde que les cycles connectés entre eux par des linkers. Pour obtenir le graphe correspondant aux frameworks, les hydrogènes sont supprimés et tous les atomes de la molécule sont remplacés par des atomes non typés. Ensuite les atomes qui sont connectés à un seul fragment sont supprimés. Cette étape est répétée jusqu'à ce qu'il ne reste plus aucun atome répondant à ce critère à supprimer (Figure 11). Cette représentation d'une structure a l'avantage de donner une information sur la forme générale de la molécule en deux dimensions, et peut donner des informations utiles pour le clustering [50].

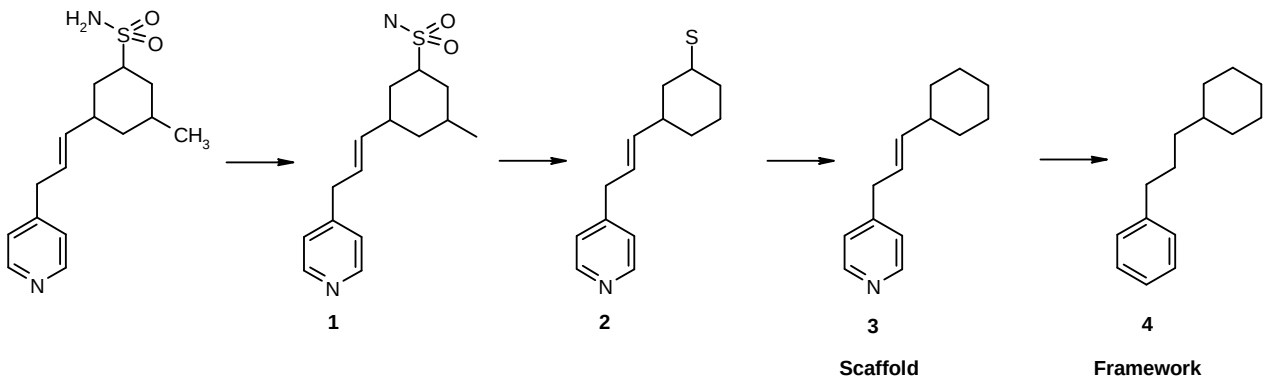


Figure 11. Les étapes de l'algorithme de génération de framework : 1) les hydrogènes sont supprimés, 2) les atomes avec une seule liaison sont supprimés successivement, 3) Le scaffold est obtenu, 4) tous les types d'atomes sont définis en tant que C et tous les types de liaisons (sauf les types aromatiques) sont définis en tant que simples liaisons, ce qui permet d'obtenir le framework.

Dans notre implémentation de l'algorithme, les atomes non typés sont remplacés par des atomes de carbone et les liaisons non typées sont représentées par des simples liaisons, mais contrairement à la méthode de Bemis les liaisons aromatiques sont différenciées des liaisons non aromatiques. C'est à notre avis une distinction importante à faire, car cela change radicalement les familles auxquelles appartiennent les composés. L'avantage de cette représentation est qu'elle peut être stockée en tant que structure, et représentée par un visualiseur moléculaire. De plus, le code InChI peut ainsi être calculé pour le framework, ce qui nous permet de gérer une liste de frameworks uniques.

Nous pensons que les frameworks et les scaffolds sont des techniques simples et efficaces pour regrouper les composés par familles. Le nombre de ces familles dans un ensemble de composés est lié à la diversité de l'ensemble. Il est cependant évident que la diversité n'est pas une notion univoque, et que l'utilisation de différentes méthodes pour la mesurer donne des résultats différents.

La Figure 12 illustre cette idée, et montre que deux composés non similaires en termes de fingerprints peuvent avoir un framework identique. De plus, des composés avec des frameworks différents peuvent être considérés comme similaires en termes de fingerprints. Ces exemples montrent que les fingerprints et les frameworks sont deux notions complémentaires.

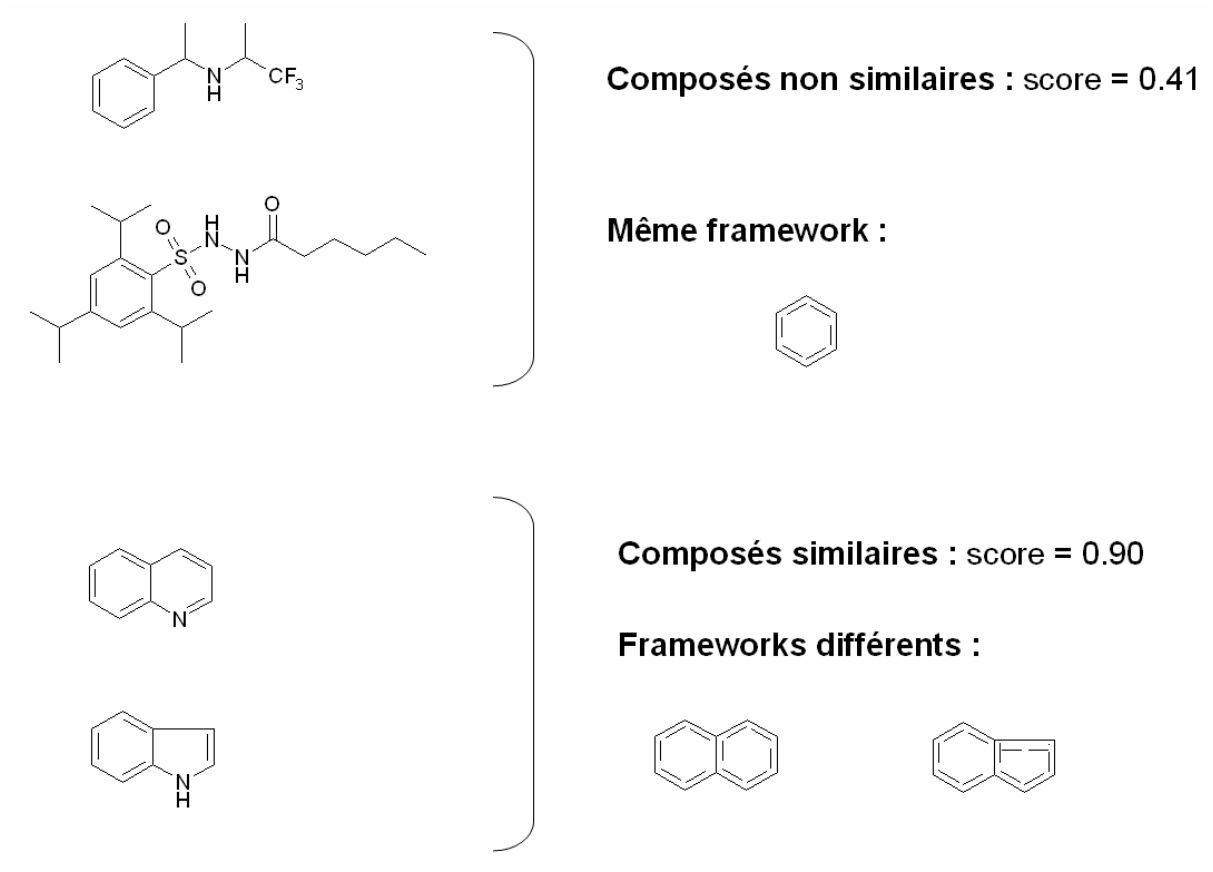


Figure 12. Illustration des différences d'information données par la diversité et par les frameworks : des composés non similaires peuvent avoir le même framework (en haut), et des composés similaires peuvent avoir des frameworks différents (en bas).

Les scaffolds sont également calculés par *ScreeningAssistant*. Du fait qu'ils conservent les types de liaisons et les types atomiques, ils permettent de faire des sous familles au sein d'un même framework.

9. Filtration des composés pour les tests de criblages

a. Propriétés « drug-like » et « lead-like »

La règle de Lipinski [51] est la plus utilisée pour caractériser les composés « drug-like » [52]. Nous rappelons que cette règle a pour but d'identifier les composés posant des problèmes d'absorption et de perméabilité, et qu'elle a été établie à partir d'une liste de

composés ayant passé avec succès les tests cliniques de phase II. D'autres règles ont depuis été introduites.

Nous avons choisi d'utiliser des règles « drug-like » basées sur des limites physicochimiques. Cela donne un filtre « drug-like » plus facilement interprétable. Cependant, pour éviter les inconvénients des limites fixes, nous avons choisi d'utiliser des limites progressives pour notre score.

Avant toute chose, on ne considèrera que les composés ne contenant pas d'autres atomes que C, O, N, S, P, F, Cl, Br, I, Na, K, Mg, Ca, ou Li comme pouvant être « drug-like » (et à fortiori « lead-like »). Nous avons choisi dans un premier temps de définir des règles utilisant des limites fixes, en se basant sur une autre étude du laboratoire [53] qui s'appuie sur des règles publiées [54, 55] pour évaluer les proportions de composés « drug-like » dans les bases commerciales :

- $100 \leq$ masse moléculaire ≤ 800 g.mol^{-1}
- Log P ≤ 7
- accepteurs de liaisons H ≤ 10
- donneurs de liaisons H ≤ 5
- liaisons pouvant tourner ≤ 15
- atomes d'halogènes ≤ 7
- nombre de cycles (Smallest Set of Smallest Rings) ≤ 6
- pas de cycle de plus de 7 membres

Les définitions suivantes seront utilisées :

- accepteurs de liaisons H : atomes d'azote, d'oxygène, de phosphore et de soufre, sauf dans les cas suivants : oxygène et soufre aromatiques, azote aromatique connecté à trois autres atomes, azote de valence 5, soufre de valence 6 ou 7.
- donneurs de liaisons H : hétéroatomes avec au moins un atome d'hydrogène et sans charge négative.

- Liaisons pouvant tourner : la définition de JOELib est utilisée : « Nombre de liaisons pouvant tourner, avec les atomes étant des atomes lourds avec des ordres de liaisons de 1, et avec une hybridation qui n'est pas 1 (sp). De plus la liaison ne fait pas partie d'un cycle.

- Log P : le Slog P sera utilisé [56]. On notera que Lipinski a utilisé le ClogP pour la mise au point de la « règle des 5 ». Les résultats peuvent varier légèrement en fonction du programme de calcul du log P utilisé.

Pour ce qui est de caractériser un composé « lead-like », différentes règles existent. Nous avons choisi de nous baser sur celles proposée par Hann et Oprea [57] : masse moléculaire ≤ 460, $-4 \leq \log P \leq 4{,}2$, logSw ≥ -5, liaisons pouvant tourner ≤ 10, nombre de cycles ≤ 4, donneurs de liaisons H ≤ 5 et accepteurs de liaisons H ≤ 9. Nous considèrerons pour notre part qu'un composé est « lead-like » s'il est « drug-like », avec le nombre d'accepteurs de liaisons H ≤ 9, la masse moléculaire ≤ 460, $-4 \leq \log P \leq 4{,}2$, les liaisons pouvant tourner ≤ 10 et les cycles ≤ 4.

Sur la base de ces critères, nous avons établi des scores « drug-like » et « lead-like » progressifs. Pour chaque critère, une pénalité est calculée. Pour huit critères, cette pénalité varie de 0 à 1 et est calculée à partir de fonctions empiriques basées sur les valeurs de seuils citées précédemment. Ces fonctions sont décrites dans le Tableau 9 et un exemple est donné dans la Figure 13.

	Pénalités “drug-like”	Pénalités “lead-like”
Donneurs de l. H	≤ 3,5: 0	-
	> 3,5 and < 6,5: 0,3333 * P – 1,1667	
	≥ 6,5: 1	
Accepteurs de l. H	≤ 7: 0	≤ 6,3: 0
	> 7 and < 13: 0,1667 * P – 1,1667	> 6,3 and < 11,7: 0,1852 * P – 1,1667
	≥ 13: 1	≥ 11,7: 1
L. pouvant tourner	≤ 10,5: 0	≤ 7: 0
	> 10,5 and < 19,5: 0,1111 * P – 1,1667	> 7 and < 13: 0,1667 * P – 1,1667
	≥ 18,5: 1	≥ 13: 1
Nb de cycles (SSSR)	≤ 4,2: 0	≤ 2,8: 0
	> 4,2 and < 7,8: 0,2778 * P – 1,1667	> 2,8 and < 5,2: 0,4167 * P – 1,1667
	≥ 7,8: 1	≥ 5,2: 1
Taille maximale de cycle	≤ 6: 0	-
	> 6 and < 9,1: 0,3226 * P – 1,9355	
	≥ 9,1: 1	
Nb d’halogènes	≤ 4,9: 0	-
	> 4,9 and < 9,1: 0,2381 * P – 1,1667	
	≥ 9,1: 1	
Masse moléculaire	≤ 100: 1	≤ 100: 1
	> 100 and < 150: -0,02 * P + 3	> 100 and < 150: -0,02 * P + 3
	≥ 150 and ≤ 350: 0	≥ 150 and ≤ 322: 0
	> 350 and < 800: 0,0022 * P – 0,7778	> 322 and < 588: 0,0038 * P – 1,2105
	≥ 800: 1	≥ 588: 1
Log P	≤ -5: 1	≤ -5: 1
	> -5 and < -1,5: -0,2857 * P – 0,4286	> -5 and < -1,5: -0,2857 * P – 0,4286
	≥ -1,5 and ≤ 4,5: 0	≥ -1,5 and ≤ 2,94: 0
	> 4,5 and < 7,5: 0,3333 * P – 1,5	> 2,94 and < 5,46: 0,3968 * P – 1,667
	≥ 7,5: 1	≥ 5,46: 1

P correspond à la propriété considérée ; - signifie que la pénalité « lead-like » est identique à la pénalité « drug-like ».

Tableau 9. Fonctions utilisées dans les scores “drug-like” et “lead-like”.

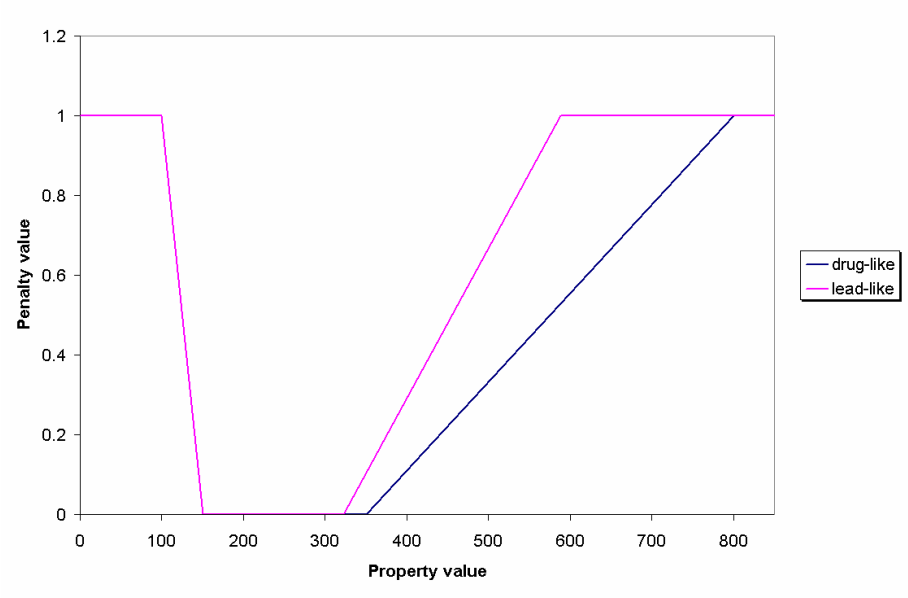


Figure 13. Représentation graphique des fonctions de penalties “drug-like” et “lead-like” pour la masse moléculaire.

Pour les donneurs de liaisons H, les accepteurs de liaisons H, les liaisons pouvant tourner, le nombre de cycles, la taille maximale des cycles et les halogènes, nous définissons une zone intermédiaire de pénalité de 60 % de la valeur limite. Par exemple, la valeur limite pour les accepteurs de liaisons H est de 10. En conséquence, la fonction de pénalité intermédiaire pour cette propriété va s’étendre de 7 (10 – 30 %) à 13 (10 + 30 %). Si une molécule a moins de 7 accepteurs de liaisons H, la pénalité pour cette valeur sera de 0, et si la molécule a plus de 13 accepteurs de liaisons H, la pénalité maximale de 1 sera appliquée. Pour la masse moléculaire et le log P, les fonctions de pénalités sont basées sur les distributions publiées de ces propriétés pour des médicaments. Ainsi, pour la masse moléculaire, la zone intermédiaire basse de pénalité s’étend de 100 à 150 $g.mol^{-1}$ (basé sur la distribution des masses moléculaires des médicaments mis sur le marché [58]). La limite haute de pénalité s’étend de 350 à 800 pour le score « drug-like » (500 – 30 % et l’ancienne limite de 800 est conservée car elle est déjà très permissive) et de 322 à 588 pour le score « lead-like » (ancienne limite : 460 [57]). Pour le log P, la zone intermédiaire de pénalité

basse s'étend de -5 à -1,5, la zone haute de pénalité s'étend de 4,5 à 7,5 pour la fonction « drug-like » (basé sur la répartition des médicaments commercialisés [58]) et de 2,9 à 5,5 pour la fonction « lead-like ». (ancienne limite : 4,2).

Toutes ces fonctions sont le résultat soit de la distribution des propriétés de médicaments, soit de limites déjà proposées. Nous verrons par la suite qu'elles sont capables d'identifier efficacement des composés qui présentent une mauvaise absorption ou une solubilité basse. Ces règles sont cependant empiriques, et des études futures devraient permettre de les affiner.

Dans la suite de ce document, nous appellerons Progressive « Drug-Like » (PDL) le score « drug-like » progressif, et Progressive « Lead-Like » (PLL) le score « lead-like ».

Cette méthode possède deux avantages par rapport à des méthodes basées sur des limites fixes. Premièrement, les effets de seuils sont évités. Par exemple, ces effets peuvent être importants pour le log P, qui peut être calculé par différentes méthodes, et conduire à des résultats légèrement différents. Deuxièmement, ces scores progressifs permettent de classer les molécules en fonction de leurs propriétés « drug-like » et « lead-like ».

Le score d'un composé est obtenu par la somme de ces pénalités. Un score faible (≤ 1) indique qu'une molécule peut être considérée comme « drug-like » ou « lead-like ». Un score ≥ 2 signifie que le composé n'est pas « drug-like » ou « lead-like ».

Il est important d'insister sur le fait qu'il n'existe pas de règles « drug-like » et « lead-like » absolues. Ces règles sont grandement dépendantes du projet étudié. Bien que nous ayons choisi des paramètres pour chacune des règles, notre système nous permet de les changer très simplement afin d'extraire un nouveau jeu de composés avec des propriétés différentes. En plus des paramètres classiques, nous pouvons éliminer les molécules avec des sous structures indésirables du jeu de données.

Nous avons comparé deux approches « drug-like », à savoir la « règle des 5 » de Lipinski et notre score, en utilisant la base Prestwick. Cette base a été choisie pour cette analyse car 85 % des composés de cette base sont des médicaments sur le marché [59]. La version utilisée pour notre étude contient 876 composés, parmi lesquels 92 % (804) sont acceptés par la « règle des 5 » et 8 % (72) sont rejetés. C'est un résultat qui semble normal pour des produits qui sont principalement utilisés par administration orale.

Notre score « drug-like » est plus restrictif ; il accepte 85 % (744) des produits, et en rejette 15 % (132). Tous les composés acceptés par le PDL sont aussi acceptés par la « règle des 5 », mais 7 % des composés de la base sont rejetés par le PDL et acceptés par la « règle des 5 ». A partir de ces résultats, il est possible de créer trois groupes de composés :

- le groupe A (744 composés) qui contient les produits acceptés par les deux approches,
- le groupe B (60 composés) contenant les produits acceptés par la « règle des 5 » mais rejetés par le PDL,
- Le groupe C (72 composés) contenant les structures refusées par les deux méthodes.

Le problème est de déterminer si le PDL a identifié des composés, ceux du groupe B, avec des problèmes potentiels de solubilité ou d'absorption non identifiés par la « règle des 5 », ou bien si notre filtre est simplement trop restrictif. Comme nous n'avons pas les valeurs expérimentales de ces propriétés pour ces composés, nous avons choisi d'utiliser les valeurs de solubilité prédites par MOE 2005.06 [60], et la surface polaire topologique (Topological Polar Surface Area ou TPSA) introduit par Ertl [61] pour la prédiction de l'absorption. Nous avons considéré que les composés avec une solubilité dans l'eau < 1 μM [62] ont une faible solubilité, et que les composés avec TPSA > 140 Å² ont une mauvaise absorption. La limite basée sur la TPSA est basée sur le travail de Palm qui a établi une bonne corrélation sigmoïdale entre la surface polaire dynamique et le transport passif de médicaments ($r^2 = 0.94$ [63]).

En utilisant ces critères, 7 % des composés du groupe A sont rejetés ; ce qui prouve que ce groupe est principalement formé de composés avec de bonnes propriétés. Pour le set C, 96 % sont rejetés, une preuve des mauvaises propriétés des produits de ce groupe.

Le groupe B contient 68 % de composés rejetés. Cela montre que les composés de ce groupe ont en majorité de mauvaises propriétés, et le PDL peut donc être une bonne méthode pour filtrer les composés par rapport à la « règle des 5 ». Grâce aux limites progressives, notre score est capable de détecter les composés qui ont plusieurs valeurs de propriétés justes en dessous des limites de la « règle des 5 ». Il est probable que ces produits aient des problèmes de solubilité ou d'absorption.

L'importance des limites progressives peut être illustrée à travers un exemple. Dans notre groupe B, le composé avec la valeur la plus haute (donc la pire) de notre score

progressif (2,6) est la néamine (numéro CAS 3947-65-7). Ce composé est très polaire (log P = -5,1), mais il a un seul critère qui ne vérifie pas la « règle des 5 » (donneurs de liaisons H = 8), et un critère qui est à la limite (accepteurs de liaisons H = 10). En conséquence, la « règle des 5 » est validée pour ce composé, mais la probabilité qu'il ait des problèmes d'absorption est forte (TPSA = 210). La neamine est en fait un composant de la neomycine commerciale, un antibiotique à usage topique ou gastrointestinal. La neomycine est connue pour avoir une mauvaise perméabilité intestinale et un mauvais passage cutané. Ceci est un bon exemple d'un composé, qui passe les filtres binaires de la « règle des 5 », mais qui peut être identifié grâce à des limites progressives.

b. Critères supplémentaires

Nous avons défini le PDL et le PLL pour identifier les composés « drug-like » et « lead-like ». Cependant, comme nous l'avons vu dans le premier chapitre, d'autres caractéristiques sont à prendre en compte pour sélectionner des composés destinés à être évalués par des tests biochimiques.

Nous avons donc rajouté la possibilité de prendre en compte un certain nombre de critères :

- présence d'une fonction réactive
- présence d'un « warhead »
- la molécule est un « promiscuous aggregating inhibitor »
- présence d'une chaîne alkyle > $-(CH_2)_6CH_3$
- présence d'une chaîne perfluorée
- pas d'atome d'azote ou d'oxygène

Nous compléterons les définitions du chapitre I :

- fonctions réactives : ces fonctions peuvent générer un faux positif lors des tests biochimiques en se liant de manière covalente à la cible biologique. Nous utiliserons la version modifiée par Oprea [64] de la liste publiée par Rishton [65]. En plus de cette liste, nous considèrerons les vinyl sulfones comme des fonctions réactives, en raison de leur fort caractère électrophile.

- « warheads » : les « warheads » peuvent également provoquer des faux positifs lors des tests biochimiques mais, contrairement aux fonctions réactives, cela est dû à une liaison non covalente avec la cible [66].

- « promiscuous aggregating inhibitors » : ces composés forment des agrégats qui inhibent certaines enzymes. Dans notre filtre, nous utilisons la liste de 48 « promiscuous aggregating inhibitors » publiée par Seidler [67].

Ces propriétés sont utilisées pour calculer un autre score que nous appellerons Cleaning For My Screening (CFMS). Pour ce calcul, nous utiliserons comme base le score PDL ou PLL et les critères choisis parmi les six présentés. Pour chacun des six critères que l'on choisira d'utiliser et qui sera validée pour une molécule, une pénalité de 2 sera ajouté au score PDL ou PLL. Le score résultant donnera le CFMS. Les molécules ayant un CMS ≤ 1 seront considérées comme acceptables. Celles avec un CFMS ≥ 2 ont de grandes chances d'avoir des problèmes, que ce soit en termes de solubilité, d'absorption, ou de faux positifs. On notera que la pénalité de 2 appliquées aux molécules pour chacun des six critères supplémentaires est très forte, ceci afin de les éliminer de la sélection.

On notera également que la sélection de composés contenant un groupement nitro est parfois considérée comme problématique car ce groupement risque de causer un faux positif lors des tests de criblages à hauts débits [55]. Bien que le groupement nitro ne soit pas dans notre liste de fonctions pouvant entraîner des faux positifs, tous les composés comportant un groupement nitro sont marqués et peuvent être mis à l'écart facilement par le logiciel si l'utilisateur le désire.

c. Les structures privilégiées

Nous avions défini dans le chapitre I une structure privilégiée comme une structure de taille importante (qui peut être considérée comme un squelette) présente dans des ligands de récepteurs divers [68]. Cette notion est liée à celle de molécule « drug-like », car les structures privilégiées sont des sous structures de tailles importantes extraites de médicaments. Il est donc admis que les composés contenant une structure privilégiée voient leurs chances d'avoir

de bonnes propriétés ADME-Tox augmentées. De plus une molécule contenant une structure privilégiée voit également ses chances d'avoir une activité biologique augmentées.

ScreeningAssistant permet d'identifier les structures privilégiées. Cette opération est réalisée par défaut à l'insertion des structures. Un numéro est attribué à chaque structure afin d'identifier quelle structure privilégiée a été identifiée dans la molécule (voir partie II en Annexe). Si aucune structure privilégiée n'est trouvée pour la molécule, le numéro attribué est -1. La liste des structures privilégiées utilisées est extraite de la littérature [68, 69, 70, 71, 72, 73, 74] (Figure 14).

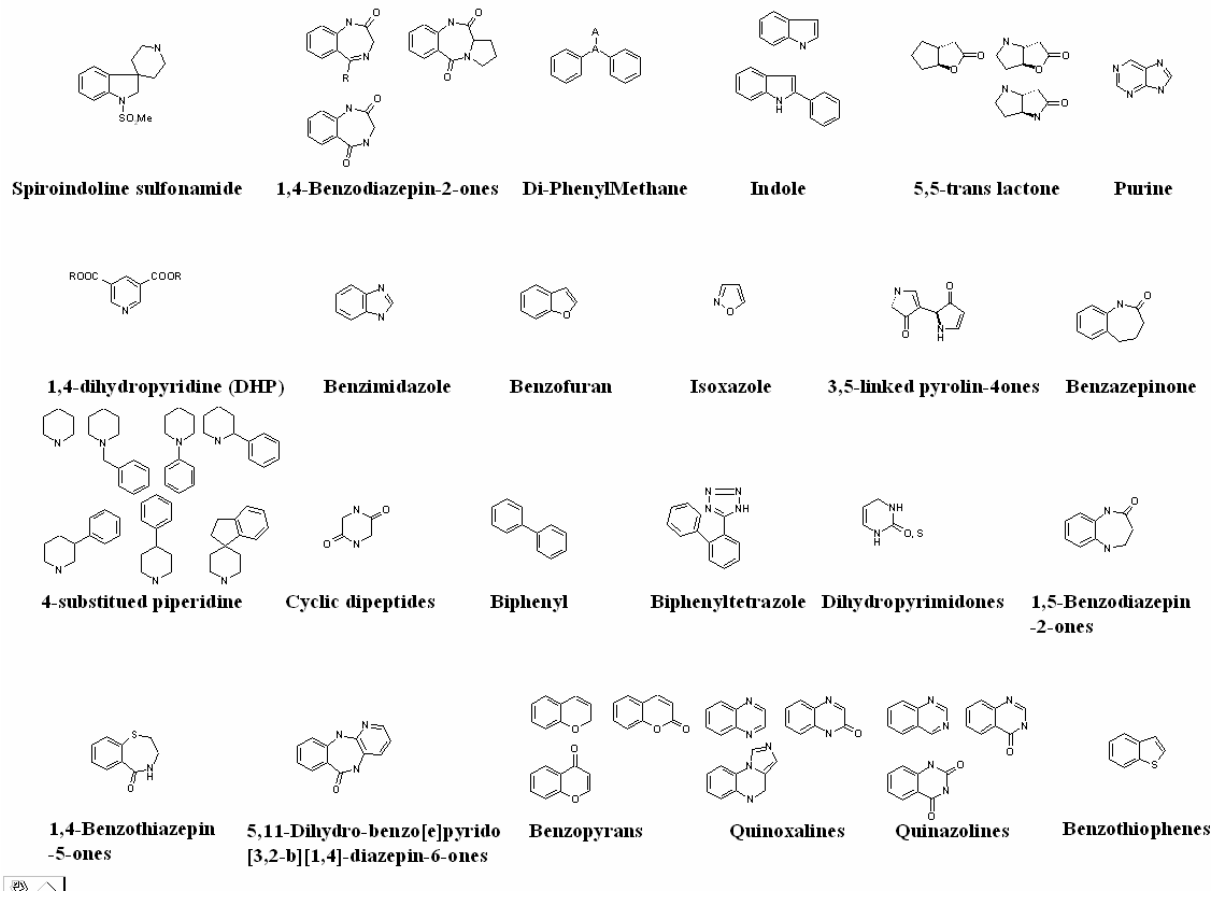


Figure 14. Liste des structures privilégiées reconnues par *ScreeningAssistant*.

10. Génération des conformations.

Par défaut, *ScreeningAssistant* gère les structures en 2D, ce qui est tout à fait adapté à la préparation d'ensembles de molécules pour le criblage réel, ou bien destinées aux tests

QSAR 2D. Cependant, pour réaliser des criblages par pharmacophores et par certains logiciels de docking, il est nécessaire de disposer des conformations des molécules. On notera que certain logiciels de docking génèrent eux même les conformations et se contentent donc d'une seule structure 3D par molécule.

Nous avons interfacé deux logiciels commerciaux à *ScreeningAssisant* pour qu'il puisse générer les structures 3D et les conformations : Corina et Omega [75]. Corina a été choisi pour « déployer » les structures en 3D. Pour utiliser Corina avec *ScreeningAssistant* il suffit de placer l'exécutable de Corina dans un sous répertoire du répertoire d'installation de *ScreeningAssistant*. L'utilisateur peut ainsi minimiser l'intégralité d'une chimiothèque à partir de l'interface de *ScreeningAssistant*. Les structures 3D générées sont stockées dans la base. Si une molécule dispose d'une structure 3D calculée par Corina, celle-ci est affichée à la place de la structure par défaut dans la fenêtre de visualisation des molécules. De même, les structures peuvent être exportées soit avec les coordonnées par défaut, soit avec les coordonnées générées par Corina.

Pour la génération des conformations, Omega sera utilisé. Les tests que nous avons effectués ont montrés que le taux de succès de la génération de conformations avec Omega était plus important si ce dernier effectuait la minimisation à partir de structures « déployées » par Corina. Nous avons donc fait le choix, pour le logiciel *ScreeningAssistant*, de générer les conformations à partir des structures 3D de Corina stockées dans la base.

La génération des conformations par Omega pose deux problèmes techniques : les informations générées occupent beaucoup d'espace disque, et les calculs sont très longs. Nous avons résolu le premier problème en stockant les conformations sous forme compressée dans la base de données. La distribution des calculs a permis de résoudre le problème des temps de calculs. La génération de conformations se prête en effet très bien à la distribution, puisque les calculs sont indépendants entre les molécules.

Omega offre la possibilité d'utiliser PVM [76] sous Linux pour distribuer les calculs. PVM est un logiciel qui permet à un ensemble d'ordinateurs en réseau et avec de systèmes d'exploitation différents d'être utilisés comme un grand ordinateur parallèle. Cependant cette solution n'était pas adaptée à nos besoins, car nous avons développé *ScreeningAssistant* pour qu'il soit utilisable par un chimiste médicinal. La solution proposée par OMEGA pose alors deux problèmes : Omega ne permet pas d'utiliser PVM sous Windows, et sa mise en place se révèle être difficile pour un non informaticien. Nous avons résolu ces problèmes en

développant, en Java [77], un système complet de distribution des calculs d'Omega. Nous sommes arrivés à créer un système de distribution performant, simple d'utilisation et robuste.

Le programme est composé d'un serveur, intégré dans l'interface de *ScreeningAssistant*, et d'un client, qui se présente sous la forme d'un petit logiciel (Figure 15). Un client est ajouté au système en entrant simplement l'adresse IP du serveur gérant le calcul.

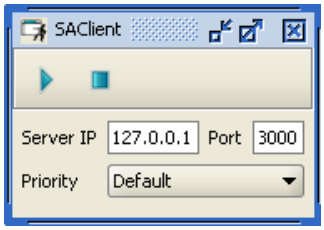


Figure 15. Logiciel client de distribution des calculs de conformations.

Les priorités des calculs peuvent être définies à la fois sur les clients et sur le serveur. Si une priorité autre que celle par défaut est spécifiée sur un client, elle prend le pas sur celle spécifiée par le serveur. Cela signifie qu'il est possible d'utiliser toutes les machines à disposition comme clients, et d'exploiter de cette façon les ressources inutilisées des laboratoires ou des entreprises. Il est ainsi possible d'exploiter des machines utilisées ponctuellement pour d'autres calculs ou des machines destinées à la bureautique : il suffit de spécifier dans l'interface graphique du client ou du serveur que le calcul de génération des conformations a une priorité basse.

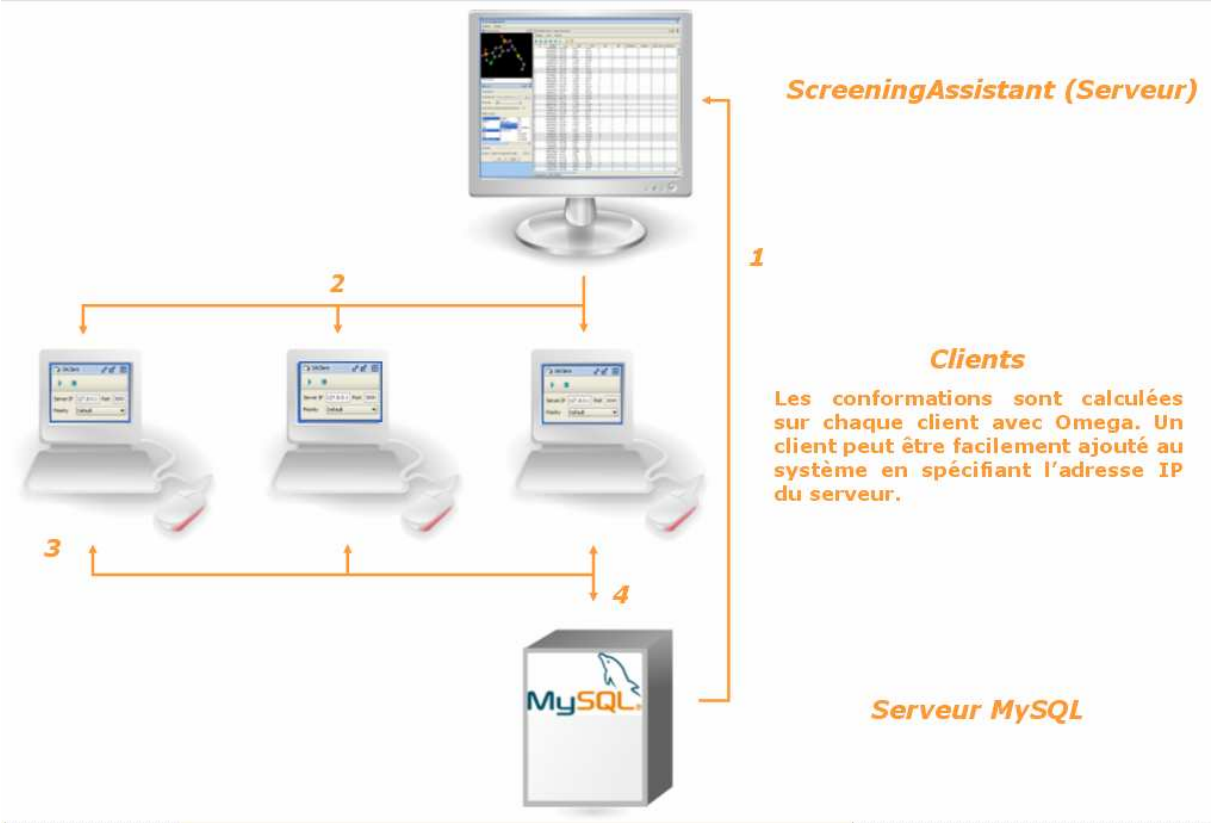


Figure 16. Fonctionnement du système de calculs distribués de conformations : *1.* Le serveur récupère les ID des molécules du prochain groupe à traiter. *2.* Le serveur envoie cette liste d'ID au premier client demandant des molécules à traiter, l'état des molécules de ce groupe est mis « en cours de traitement » et la date de début de calcul est stockée dans la base de données. Le serveur revient à la première étape. *3.* Le client récupère les structures des molécules à traiter dans la base MySQL, calcule les conformations, insère les conformations dans la base et met l'état des molécules à « traité ». *4.* Le client demande un autre jeu de molécules au serveur.

ScreeningAssistant utilisant une base MySQL, nous avons exploité cette dernière pour la gestion de la distribution. Chaque structure peut se voir attribuer trois états : non traitée, en cours de traitement, et traitée. En complément d'information, quand l'état d'une structure est mis en cours de traitement, la date et l'heure du début de traitement sont stockées dans la base de données. Le fait de stocker ces simples informations permet à notre système d'avoir une très bonne tolérance aux pannes : si n'importe lequel des ordinateurs du système est arrêté, il suffit de le relancer pour que le calcul continue. Le serveur envoie d'abord aux clients les molécules non traitées, puis il renvoie les molécules en cours de traitement par ordre

chronologique de début de traitement. Cela permet de relancer les calculs n'ayant pas pu, pour une raison ou pour une autre, se terminer correctement. La Figure 16 présente les étapes de fonctionnement du système.

Omega est utilisé avec les options *–fromCT false*, pour débuter le calcul à partir de la structure générée par Corina, et *finalopt true*, pour minimiser les conformations générées.

III. Conclusion

Nous avons présenté dans ce chapitre le développement du logiciel *ScreeningAssistant* qui permet la gestion de chimiothèques destinées au criblage. Le code source de ce programme comporte 11 000 lignes de code, disponibles sur le site SourceForge.net. Les principales fonctionnalités du logiciel, à savoir la prise en compte des doublons, le traitement des structures, la filtration des composés ont été présentées. Nous avons présenté le score PDL conçu pour identifier les composés « drug-like », le score PLL pour identifier les composés « lead-like », et le score CFMS pour identifier soit les composés « drug-like », soit les composés « lead-like », en tenant compte de critères supplémentaires comme la présence de faux positifs potentiels. Le logiciel est d'une part utilisé pour gérer la chimiothèque virtuelle de 5 millions de références de notre laboratoire, et d'autre part disponible gratuitement sur internet. Il a été développé sous licence GPL afin de permettre à n'importe quel développeur de lui ajouter des fonctionnalités.

Dans le chapitre suivant, nous présenterons l'analyse des bases des 38 fournisseurs regroupées au sein de la base de criblage virtuel de notre laboratoire.

1. Chen, W.L. Chemoinformatics: Past, Present, and Future. *J. Chem. Inf. Model.* **2006**, ASAP.
2. Brown, F. Chemoinformatics: What is it and How does it Impact Drug Discovery. *Annu. Rep. Med. Chem.* **1998**, 33, 375-384.
3. Handbook of Chemoinformatics; Gasteiger, J., Ed.; Wiley-VCH: Weinheim, Germany, **2003**.
4. Tripos, www.tripos.com
5. Accelrys, www.accelrys.com
6. Chemical Computing Group, www.chemcomp.com
7. Schrödinger, www.schrodinger.com
8. Open Eye, www.eyesopen.com
9. MDL, http://www.mdli.com
10. Cheminfomatics.org, www.cheminformatics.org
11. W.L. Jorgensen. QSAR/QSPR and Proprietary Data. *J. Chem. Inf. Model.*, **2006**, 46, 937-937.
12. Open Babel, http://openbabel.sourceforge.net
13. Murray-Rust, P.; Rzepa, H. S. Chemical Markup, XML, and the Worldwide Web. 1. Basic Principles. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 928-942.
14. Murray-Rust, P.; Rzepa, H. S. Chemical Markup, XML and the World-Wide Web. 2. Information Objects and the CMLDOM *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 1113-1123.
15. Gkoutos, G. V.; Murray-Rust, P.; Rzepa, H. S.; Wright, M. Chemical Markup, XML, and the World-Wide Web. 3. Toward a Signed Semantic Chemical Web of Trust. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 1124-1130.
16. Murray-Rust, P.; Rzepa, H. S. Chemical Markup, XML, and the World Wide Web. 4. CML Schema. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 757-772.
17. Murray-Rust, P.; Rzepa, H. S.; Williamson, M. J.; Willighagen, E. L. Chemical Markup, XML, and the World Wide Web. 5. Applications of Chemical Metadata in RSS Aggregators. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 462-469.
18. Holliday, G. L.; Murray-Rust, P.; Rzepa, H. S. Chemical Markup, XML, and the World Wide Web. 6. CMLReact, an XML Vocabulary for Chemical Reactions. *J. Chem. Inf. Comput. Sci.* **2006**, *46*, 145-157.
19. Liao, Y.M.; Ghanadan, H. Communicating Chemistry: The Chemical Markup Language. *Anal. Chem.* **2002**, *74*, 389-390.
20. Geldenhuys, W.J.; Gaasch, K.E.; Watson, M.; Allen, D.D.; Van der Schyf, C.J. Optimizing the use of open-source software applications in drug discovery. *Drug Discov. Today* **2006**, *11*, 127-132.
21. DeLano, W.L. The case for open-source software in drug discovery. *Drug Discov. Today* **2005**, *10*, 213-217.
22. Guha, R.; Howard, M.T.; Hutchison, G.R.; Murray-Rust, P.; Rzepa, H.; Steinbeck, C.; Wegner, J.; Willighagen, E.L. The Blue Obelisk-Interoperability in Chemical Informatics. *J. Chem. Inf. Model.* **2006**, *46*, 991-998.
23. IBM Archives: Valuable resources on IBM's history:
http://www-03.ibm.com/ibm/history/history/decade_1950.html
24. MOE, Chemical Computing Group, http://www.chemcomp.com/
25. Arnoult, E. ; Mozziconacci, J. C. ; Baurin, N. ; Marot, C. ; Morin-Allory, L. Structural analysis of molecular databases and selection of drug-like compounds for virtual screening applications. Congrès annuel de la société française de biochimie et biologie moléculaire, 2003, Lyon.
26. Oracle, http://www.oracle.com
27. MySQL AB, http://www.mysql.com
28. SQL Server, Microsoft, www.microsoft.com/sql/
29. Daylight, www.daylight.com
30. Isis/Host and Isis/Base, MDL Information Systems, Inc., www.mdl.com
31. Activity Base, idbs, www.id-bs.com/activitybase/
32. Pipeline Pilot, SciTegic, www.scitegic.com
33. CORINA, Molecular Networks, http://www.mol-net.de/software/corina/
34. Concord, Tripos, http://www.tripos.com
35. Sybyl, Tripos, http://www.tripos.com
36. http://www.eyesopen.com/products/applications/omega.html
37. Schuffenhauer, A.; Popov, M.; Schopfer, U.; Acklin, P.; Stanek, J. Jacoby, E. Molecular Diversity Management Strategies for Building and Enhancement of Diverse and Focused Lead Discovery Compound Screening Collections. Comb Chem High Throughput Screen 2004, 7, 771-781.
38. LigPrep, Schrödinger, http://www.schrodinger.com
39. Molinspiration, http://www.molinspiration.com
40. The IUPAC International Chemical Identifier. http://www.iupac.org/inchi

41. JGoodies, http://www.jgoodies.com
42. JFreeChart, www.jfree.org/jfreechart/
43. EasyPHP, http://www.easyphp.org/
44. Xue, L.; Godden, J.W.; Bajorath, J. Database Searching for Compounds with Similar Biological Activity Using Short Binary Bit String Representations of Molecules. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 881-886.
45. Reynolds, C. H.; Druker, R.; Pfahle, L. B. Lead Discovery Using Stochastic Cluster Analysis (SCA): A New Method for Clustering Structurally Similar Compounds. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 305-312.
46. Voigt, J. H.; Bienfait, B.; Wang, S.; Nicklaus, M. C. Comparison of the NCI Open Database with Seven Large Chemical Structural Databases *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 702-712.
47. Baurin, N.; Aboul-Ela, F.; Barril, X.; Davis, B.; Drysdale, M.; Dymock, B.; Finch, H.; Fromont, C.; Richardson, C.; Simmonite, H.; Hubbard, R. E. Design and Characterization of Libraries of Molecular Fragments for Use in NMR Screening against Protein Targets. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 2157-2166.
48. Bemis, G.W.; Murcko, M.A. The Properties of Known Drugs. 1. Molecular Frameworks. *J.Med.Chem* **1996**, *39*, 2887-2893.
49. Bemis GW, Murcko MA. Properties of known drugs. 2. Side chains. *J.Med.Chem* **1999**, *42*, 5095-5099.
50. Harper, G.; Bravi, G. S.; Pickett, S. D.; Hussain, J.; Green, D. V. S. The Reduced Graph Descriptor in Virtual Screening and Data-Driven Clustering of High-Throughput Screening Data. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 2145-2156.
51. Lipinski, C.A.; Lombardo, F.; Dominy, B.W.; Feeney, P.J. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Adv. Drug. Deliv. Rev.* **2001**, *46*, 3-26.
52. Lipinski, C.A. Lead- and drug-like compounds: the rule-of-five revolution. *Drug Discov. Today* **2004**, *1*, 337-341.
53. Mozziconacci, J.C., Arnoult, E., Baurin, N., Marot, C., Morin-Allory, L., *Preparation of a molecular database from a set of 2 million compounds for virtual screening applications : gathering, structural analysis and filtering*, 9th Electronic Computational Chemistry Conference, World Wide Web, march 2003.
54. Walters ,W.P.; Murcko M.A. Prediction of 'drug-likeness'. *Adv. Drug. Deliv. Rev.* **2002**, *54*, 255-271.
55. Charifson, P.S.; Walters W.P. Filtering databases and chemical libraries *J. Comput. Aided Mol. Des.* **2002**, *16*, 311-323.
56. Wildman, S.A.; Crippen, G.M. Prediction of physicochemical parameters by atomic contributions *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 868-873.
57. Hann, M. M.; Oprea, T. I. Pursuing the leadlikeness concept in pharmaceutical research. *Curr Opin Chem Biol* **2004**, *8*, 255-263.
58. Wenlock, M.C.; Austin, R.P.; Barton, P.; Davis, A.M.; Leeson P.D. A Comparison of Physiochemical Property Profiles of Development and Marketed Oral Drugs *J. Med. Chem.* **2003**, *46*, 1250-1256.
59. The Prestwick Chemical Library, http://www.prestwickchemical.com/chem_lib.htm
60. Hou, T. J.; Xia, K.; Zhang, W.; Xu, X. J. ADME Evaluation in Drug Discovery. 4. Prediction of Aqueous Solubility Based on Atom Contribution Approach. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 266-275.
61. Ertl, P.; Rohde, B.; Selzer, P. Fast Calculation of Molecular Polar Surface Area as a Sum of Fragment-Based Contributions and Its Application to the Prediction of Drug Transport Properties. *J. Med. Chem.* **2000**, *43*, 3714-3717.
62. Baurin, N.; Baker, R.; Richardson, C.; Chen, I.; Foloppe, N.; Potter, A.; Jordan, A.; Roughley, S.; Parratt, M.; Greaney, P.; Morley, D.; Hubbard, R.E. Drug-like Annotation and Duplicate Analysis of a 23-Supplier Chemical Database Totalling 2.7 Million Compounds *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 643-657.
63. Palm, K.; Stenberg, P.; Luthman, K.; Artursson, P. Polar molecular surface properties predict the intestinal absorption of drugs in humans, *Pharm. Res.*, **1997**, 14, 568-571.
64. Oprea, T.I. Property distribution of drug-related chemical databases. *J. Comput. Aided Mol. Des.* **2000**, *14*, 251-264.
65. Rishton, G.M. Reactive compounds and in vitro false positives in HTS. *Drug Discov. Today* **1997**, *2*, 382-384.
66. Rishton, G.M. Nonleadlikeness and leadlikeness in biochemical screening. *Drug Discov. Today* **2003**, *8*, 86-96.
67. Seidler, J.; McGovern, S. L.; Doman, T. N.; Shoichet, B. K.; Identification and Prediction of Promiscuous Aggregating Inhibitors among Known Drugs. *J.Med.Chem* **2003**, *46*, 4477-4486.
68. DeSimone, R.W.; Currie, K.S.; Mitchell, S.A.; Darrow, J.W.; Pippin, D.A. Privileged Structures: Applications in Drug Discovery. *Comb Chem High Throughput Screen* **2004**, *7*, 473-493.
69. Horton, D.A.; Bourne, G.T.; Smythe, M.L. The Combinatorial Synthesis of Bicyclic Privileged Structures or Privileged Substructures. *Chem. Rev.* **2003**, *103*, 893-930.

70. Sheridan, R.P. Finding Multiactivity Substructures by Mining Databases of Drug-Like Compounds. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1037-1050.
71. Zhou, C.; Guo, L.; Morriello, G.; Pasternak, A.; Pan, Y.; Rohrer, S.P.; Birzin, E.T.; Huskey, S.E.; Jacks, T.; Schleim, K.D.; Cheng, K.; Schaeffer, J.M.; Patchett, A.A.; Yang, L. Nipecotic and iso-nipecotic amides as potent and selective somatostatin subtype-2 receptor agonists. *Bioorg. Med. Chem. Lett.* **2001**, *11*, 415-417.
72. Gurrath, M. Peptide-Binding G Protein-Coupled Receptors: New Opportunities for Drug Design. *Curr. Med. Chem.* **2001**, *8*, 1605-1648.
73. Mason, J.S.; Morize, I.; Menard, P.R.; Cheney, D.L.; Hulme, C. Labaudiniere, R.F. 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* **1999**, *42*, 3251-3264.
74. Barakat, K.J.; Cheng, K.; Chan, W.W.; Butler, B.S.; Jacks, T.M.; Schleim, K.D.; Hora, D.F.; Hickey, G.J.; Smith, R.G.; Patchett, A.A.; Nargund, R.P. Synthesis and biological activities of phenyl piperazine-based peptidomimetic growth hormone secretagogues. *Bioorg. Med. Chem. Lett.* **1998**, *8*, 1431-1436.
75. Omega, http://www.eyesopen.com/
76. Parallel Virtual Machine, http://www.csm.ornl.gov/pvm/pvm_home.html
77. Jim Farley, *Java Distributed Computing*, O'Reilly & Associates.

Chapitre 3. Création et analyse de la base de criblage virtuel de l'ICOA

Nous avons utilisé *ScreeningAssistant* pour créer une base de criblage virtuel, qui sera utilisée principalement pour des tests par docking. Cette base contient 5 millions de références (3,3 millions de structures uniques) correspondant aux produits de l'ICOA, de la chimiothèque nationale du CNRS, et de catalogues de produits commerciaux.

Nous allons présenter ici une analyse aussi que complète que possible de notre chimiothèque et des différentes bases la constituant, en termes de propriétés « drug-like », « lead-like » et de diversité. L'estimation de la diversité a été réalisée en utilisant autant de méthodes que possibles, à savoir quatre types de fingerprints, les frameworks, les scaffolds, les chaînes latérales et les fragments rétrosynthétiques (RECAP). Cela permettra au final d'avoir un aperçu de la diversité des bases commerciales, ce qui est très utile pour choisir un fournisseur lors de la constitution d'une base de criblage réel. Une analyse de bases commerciales avec des méthodes aussi diverses n'est, à notre connaissance, pas disponible dans la littérature. Le manque d'études poussées dans ce domaine est souligné par Bradley [1]. On trouve cependant des études publiées, mais qui sont limitées soit dans le nombre de fournisseurs considérés, soit dans les méthodes de comparaisons utilisées [2, 3, 4, 1, 5, 6, 7, 8]. Une version préliminaire de ce travail a fait l'objet d'une publication [9].

I. Méthodes non présentes par défaut dans *ScreeningAssistant*.

A. Fingerprints

Nous avons vu que la version publique de *ScreeningAssistant* utilise les SSKey3DS pour estimer la diversité d'un ensemble de composés. Afin d'avoir les résultats les plus représentatifs possibles de la diversité des bases étudiées, nous avons utilisé d'autres fingerprints :

- MACCS : ces fingerprints sont largement utilisées et codent principalement pour la présence de fragments donnés [10].

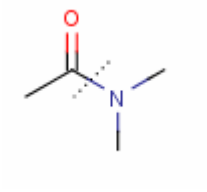
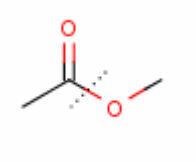
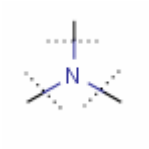
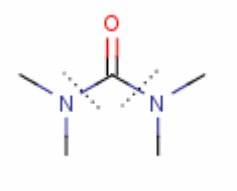
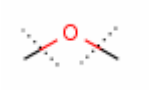
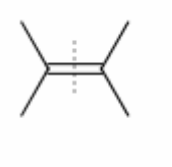
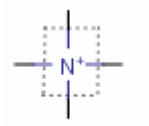
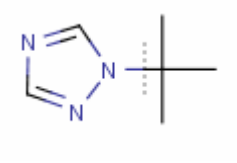
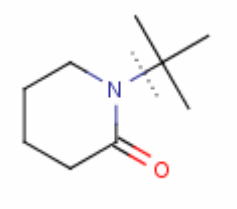
- Typed Graph Distances (TGD) : codent pour des distances topologiques entre les paires de pharmacophores. Il est assigné à chaque atome le type acide, base, donneur de liaison H, accepteur de liaison H, à la fois donneur et accepteur de liaison H, hydrophobe, aucun. Les informations sont codées sous forme de triplets (*u*, *v*, *d*) avec *u* et *v* des types atomiques et *d* la distance topologique entre les deux atomes.)

- Typed Graph Triangles (TGT) : sont conçus de la même manière que les TGD, mais codent des triangles pharmacophoriques. Les types d'atomes utilisés sont : donneur de liaison H ou base, accepteur de liaison H ou acide, à la fois donneur et accepteur de liaisons hydrogènes, hydrophobe. Les distances sont codées par des vecteurs (*u*, *v*, *w*, *d*, *e*, *f*) avec *u*, *v*, *w* les types atomiques, et *d*, *e*, *f* les distances topologiques.

Pour estimer la similarité entre deux fingerprints nous utiliserons l'étape de dissimilarité de l'algorithme SCA avec le coefficient de Tanimoto, comme décrit précédemment.

B. Fragments rétrosynthétiques

La rétrosynthèse virtuelle a deux principales applications : celle qui nous intéresse dans le présent travail - à savoir l'analyse de bases de données - et la synthèse virtuelle (combinatoire ou *de novo*). Nous avons choisi la méthode RECAP [11] pour fragmenter les composés de notre base, car c'est une méthode reconnue et utilisée dans plusieurs travaux [12, 13]. RECAP utilise 11 règles de rétrosynthèse (Tableau 9).

Structure	nom	SMARTS
	1 : Amide	C(!@[#7]([#6,#1])[#6,#1])(=O)[#6]
	2 : Ester	C(!@O[#6])(=O)[#6,#1]
	3 : Amine	[#7](-!@[#6])([#6,#1])[#6]
	4 : Urée	C(!@N([#6,#1])[#6,#1])(=O)N([#6,#1])[#6,#1]
	5 : Ether	O(-!@[#6!R])[#6]
	6 : Alcènes	C(=!@C([#6,#1])[#6,#1])([#6,#1])[#6,#1]
	7 : Azote quaternaire	[N](!@[#6])([#6])([#6])[#6]
	8 : Azote aromatique – carbone aliphatique	C(!@n)([#6,#1])([#6,#1])[#6,#1]
	9 : Azote de lactame – carbone aliphatique	C(!@[N](-@[C](=O)-@[C])-@[C])([#6,#1])([#6,#1])[#6,#1]

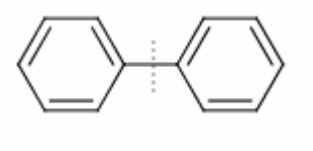
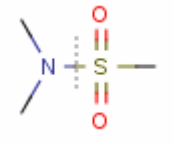
	10 : Carbone aromatique – carbone aromatique	c-!@c
	11 : Sulphonamide	S(!@[#7]([#6,#1])[#6,#1])(=O)(=O)-[#6]

Tableau 10. Règles de fragmentation RECAP utilisées pour notre étude.

En appliquant ces règles à un ensemble de molécules nous obtenons une liste de fragments. Nous ne garderons que les fragments uniques. Le compte de ces différents fragments nous permet d'obtenir une information sur la diversité des composés de la base, différente de celles obtenues par les fingerprints, les frameworks, et les scaffolds.

Nous utilisons encore une fois le code InChI pour ne garder que les fragments uniques. Les structures des fragments sont stockées dans la table *Fragment* (Figure 7). La table *FragmentHash* ne contient que le MD5 du code InChI (dans le but d'accélérer les requêtes, notamment la recherche de doublons parmi les fragments), ainsi que le nombre de cycles, la masse et le nombre d'atomes lourds du fragment. Cette table contient également 20 champs composés d'une lettre, correspondant à un nom d'atome, et d'un nombre, correspondant au numéro de la règle RECAP utilisée pour obtenir le fragment en question (Tableau 10). Ces informations ne sont pas utilisées pour l'analyse de la base. Nous avons créé cette structure de donnée pour pouvoir reconstruire les molécules. Cela laisse la possibilité de développer dans le futur un programme de synthèse *de novo* de composés.

C. Chaînes latérales

Les chaînes latérales correspondent aux parties des molécules supprimées lors de la création des scaffolds [14]. Comme le montre la Figure 17, l'atome d'ancrage de la chaîne est conservé. Cela permet de différencier des chaînes latérales qu'il ne serait pas possible de distinguer autrement.

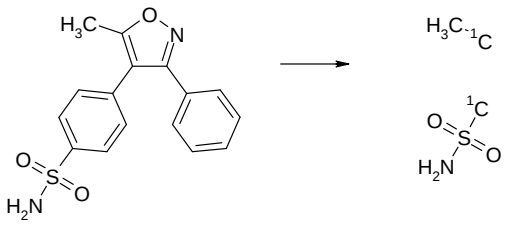


Figure 17. Chaînes latérales du valdecoxib. Les atomes des cycles auxquels sont liées les chaînes latérales sont conservés et marqués virtuellement par un isotope 1.

Le nombre de chaînes latérales d'un ensemble de composés donne une information sur leur diversité qui est complémentaire de celle donnée par le nombre de frameworks et de scaffolds.

II. Résultats

A. Propriétés générales des bases

1. Origine des composés présents dans la chimiothèque virtuelle

Les composés présents dans la base proviennent de 38 fournisseurs différents. La base totalise 5 millions de référence, qui correspondent à 3,3 millions de structures uniques. Les composés sont en très grande majorité issus de catalogues de produits commerciaux. Ces catalogues proposent des molécules synthétisés par chimie combinatoire et par chimie classique, mais aussi quelques composés d'origine naturelle ou semi-naturelle. On trouve quelques bases d'initiatives publiques, notamment les bases NCI et Chimiothèque Nationale. La base Chimiothèque Nationale (Chim. Nat.) regroupe des composés issus de plusieurs laboratoires publics français. Nous avons inclus la chimiothèque réelle de l'ICOA dans cette étude. On notera cependant que cette chimiothèque fait partie de la Chimiothèque Nationale.

L'origine des composés constituant la base est résumée dans le Tableau 11. Les bases contenant le plus de références sont les bases TimTec (658 422), ChemDiv (553 150) et Chem T&I (484 881). Les bases contenant le plus de composés ne sont pourtant pas forcément celles qui couvrent le plus de diversité, comme nous le verrons par la suite.

2. Doublons

Le pourcentage de doublons dans les bases varie de 0 à 6 %. *ScreeningAssistant* ne garde pas les contre-ions pour un composé (seul le plus grand fragment est conservé), ce qui veut dire que deux molécules avec des contre-ions différents seront considérées comme étant des doublons. Cela peut engendrer une légère surestimation du nombre des doublons. La base NCI est celle qui contient le plus de doublons. On notera que seulement trois bases ne contiennent aucun doublons : ACBBlocks, CB R&D et CombiPure. Le pourcentage moyen de doublons toutes bases confondues est de 1,1 %. On pourrait en effet s'attendre à un nombre important de doublons dans les grandes bases, mais le pourcentage de doublons n'est pourtant pas lié à la taille de la base. On peut en effet prendre comme exemple ChemDiv qui contient plus de 550 000 composés et qui n'a que 0,01 % de doublons.

3. Structures exclusives

Une propriété intéressante d'une base de données chimique est le pourcentage de composés exclusifs (c'est-à-dire que l'on ne retrouve que dans cette base). Ce pourcentage peut aller de 16 à 100 %. C'est la base de VitasM Laboratory qui a le moins de composés exclusifs. Seuls AnalytiCon Discovery et BioFocus proposent des bases totalement exclusives. Il n'y a pas de corrélations entre la taille des bases et le pourcentage de composés uniques. On remarquera que parmi les bases de plus de 300 000 composés, les pourcentages de composés exclusifs restent assez faibles, sauf pour la base Enamine qui, alors qu'elle contient près de 430 000 molécules, propose 85 % de composés exclusifs. La moyenne de composés exclusifs par base de données est de 58,8 %.

Les grandes variations de composés exclusifs entre les bases s'expliquent par le fait que certaines bases sont la compilation de plusieurs autres.

Fournisseurs	Web	Date	Composés	Doublons (%)	Exclusives (%)
ACB Blocks	http://www.acbblocks.com	07/05	61 237	0,00	97,5
AnalytiCon Discovery	http://www.ac-discovery.com	07/05	8 653	0,15	100
Asinex	http://www.asinex.com	07/05	345 782	0,02	38,2
Aurora Fine Chemicals	http://www.aurorafinechemicals.com	07/05	31 512	0,51	17,7
BioFocus	http://www.biofocus.com	03/04	23 712	0,01	100
CB R&D	http://www.cbrd.net	07/05	176	0,00	28,4
Cerep	http://www.cerep.fr	07/05	20 078	0,00	97,3
ChemBridge	http://chembridge.com	07/05	425 941	0,00	25,5
ChemDiv	http://www.chemdiv.com	07/05	553 150	0,01	53,8
ChemStar	http://www.chemstaronline.com	07/05	60 051	0,35	22,2
ChemT&I	http://www.chemti.com	07/05	484 881	0,15	43,5
Chim. Nat.	http://chimiotheque-nationale.enscm.fr	07/05	26 330	1,15	78,4
Combi-Blocks	http://www.combi-blocks.com	07/05	1 055	0,85	60,3
CombiPure	http://www.combipure.com	07/05	910	0,00	97,9
EMC Microcollections	http://www.microcollections.de	03/05	23 936	3,86	99,8
Enamine	http://www.enamine.net	06/05	428 271	0,03	84,6
Florida Center for Heterocyclic Compounds	http://ark.chem.ufl.edu	07/05	29 515	3,42	74,9
Frontier Scientific	http://www.frontiersci.com	07/05	611	3,48	45,5
ICOA	http://www.univ-orleans.fr/icoa/chimiotheque	07/05	3 213	1,77	22,6
InFarmatik	http://www.infarmatik.com	07/05	541	1,81	71,2
InterBioScreen	http://www.ibscreen.com	04/05	425 676	0,17	56,6

KeyOrganics	http://www.keyorganics.ltd.uk	07/05	187 079	0,07	87,0
LaboTest	http://www.labotest.com	07/05	3 097	2,15	20,5
Matrix Scientific	http://www.matrixscientific.com	07/05	14 963	1,35	52,5
MayBridge	http://www.maybridge.com	07/05	69 138	0,20	75,4
MDPI	http://www.mdpi.org	2004	10 193	4,32	75,6
MedChemLab	http://mosmedchemlabs.com	07/05	179 248	0,02	43,7
NCI	http://dtp.nci.nih.gov	09/03	244 406	6,02	89,1
Otava	http://www.otava.com.ua	04/05	76 819	0,39	25,4
Prestwick	http://www.prestwickchemical.com	07/05	1 117	0,27	42,7
Specs	http://www.specs.net	07/05	219 452	0,03	25,9
Spectrum Info	http://www.spectrum.kiev.ua	07/05	1 179	3,60	59,0
SynChem	http://www.synchem.com	07/05	590	1,01	56,8
SynphaBase	http://www.synphabase.com	07/05	147	5,16	85,0
TimTec	http://www.timtec.net	11/05	658 422	0,23	23,7
TosLab	http://www.toslab.com	07/05	23 235	0,09	43,2
Tripos	http://leadquest.tripos.com	07/05	65 288	0,04	95,2
VitasM Laboratory	http://www.vitasmlab.com	07/05	226 325	0,10	16,0

Tableau 11. Analyse des 38 jeux de molécules constituant notre chimiothèque. Le pourcentage de structures exclusives correspond aux structures n'étant présentes que chez le fournisseur en question. La date d'obtention de la base est indiqué sous la forme mois/année.

B. Composés « drug-like » et « lead-like »

1. « Drug-like » et « lead-like »

La Figure 18 fait ressortir les propriétés « drug-like » (PDL ≤ 1) et « lead-like » (PLL ≤ 1) des molécules. Cette figure montre également tous les composés avec un CFMS ≤ 1. Pour calculer ce score nous nous sommes basé sur le PDL et nous avons utilisés tous les critères possibles du CFMS (voir Chapitre 2.II.B.9.b). Pour rappel, ce score peut être grossièrement considéré comme un score « drug-like » prenant en compte les composés pouvant se révéler être des faux positifs lors des tests biochimiques. Pour l'ensemble de notre

chimiothèque, il y a 2,8 millions de molécules avec un PDL ≤ 1), 1,3 millions avec un PLL ≤ 1 et 2,7 millions avec un CFMS ≤ 1.

En moyenne, 86 % des composés ont un PDL ≤ 1 et 79 % ont un CFMS ≤ 1. Deux bases ressortent comme étant les moins « drug-like » selon cette analyse : Combi-Blocks et Frontier Scientific. Cela est cohérent avec le fait que ces deux fournisseurs proposent des réactifs plutôt que des produits finaux. On notera également que Spectrum Info, qui propose à la fois des réactifs et des composés destinés au criblage, et SymphaBase, qui propose des réactifs, ont des pourcentages de composés avec de mauvaises propriétés « drug-like » plus forts que la moyenne.

Nous désignerons par le terme « bases commerciales de grandes tailles » les bases de plus de 200 000 composés, à l'exclusion de la base NCI qui n'est pas réellement une base commerciale. Pour ce type de bases, la moyenne de composés avec un PDL ≤ 1 est de 88 %, et la moyenne des composés avec un CFMS ≤ 1 est de 84 %. Ces bases sont donc plus « drug-like » que l'ensemble des bases. Ceci peut aisément s'expliquer par le fait que ce sont des bases conçues pour des tests de criblages à haut-débits, et qu'à ce titre les fournisseurs intègrent dans leurs bases un maximum de composés « drug-like », et plus particulièrement validant les critères de Lipinski.

Les bases ACB blocks, CombiPure, InFarmatik et SynChem sont celles qui présentent les meilleures propriétés « drug-like ». D'une manière générale on notera que les bases ayant des résultats les plus éloignés de la moyenne sont des bases de petites tailles. Ces résultats sont en corrélation avec la faible diversité de ces bases.

Les écarts entre le pourcentage de composés avec un PDL ≤ 1 et le pourcentage de composés avec un CFMS ≤ 1 sont faibles. Cela s'explique par le fait que le critère ayant le plus d'influence dans les critères supplémentaires introduit par le score CFMS est la présence de fonctions réactives. Ces fonctions sont présentes en nombre limité dans les bases destinées au criblage. Les différences sont par contre plus marquées parmi les bases contenant des réactifs.

Les pourcentages de composés avec un PLL ≤ 1 sont plus faibles, avec en moyenne 50 % de composés passant ce filtre. Cela est principalement dû à la limite contraignante fixée au log P. Etant donné que dans les tests de criblages les composés « lead-like » sont aussi importants que les composés « drug-like », nous considérons que, idéalement et d'une

manière générale, la moitié des composés « drug-like » d'une base destinée au criblage devraient être « lead-like ». Nous rediscuterons de ce point dans le chapitre IV. Suivant le critère que nous venons de fixer, la base TimTec a un bon ratio « drug-like » / « lead-like », tout comme Asinex, ChemBridge, ChemStar, Chem T&I, Specs et VitasM. D'autres bases ont par contre un nombre de composés « lead-like » un peu faible par rapport à leur nombre de composés « drug-like ». Ce sont les bases Cerep, ChemDiv, Enamine, InterBioScreen, KeyOrganics, MedChemLab, et Tripos.

On remarquera que certaines bases (CB R&D, Combi-Blocks, Frontier Scientific, LaboTest, Matrix Scientific, Spectrum Info, Synchem et SynphaBase) ont un profil particulier : les pourcentages de composés avec PLL ≤ 1 sont proches des pourcentages de composés avec un PDL ≤ 1, et supérieurs aux pourcentages de composés avec un CFMS ≤ 1. Ce sont en fait des bases proposant des réactifs, et donc des composés de petites tailles. Etant donné la définition des composés « lead-like », cette constatation semble tout à fait logique.

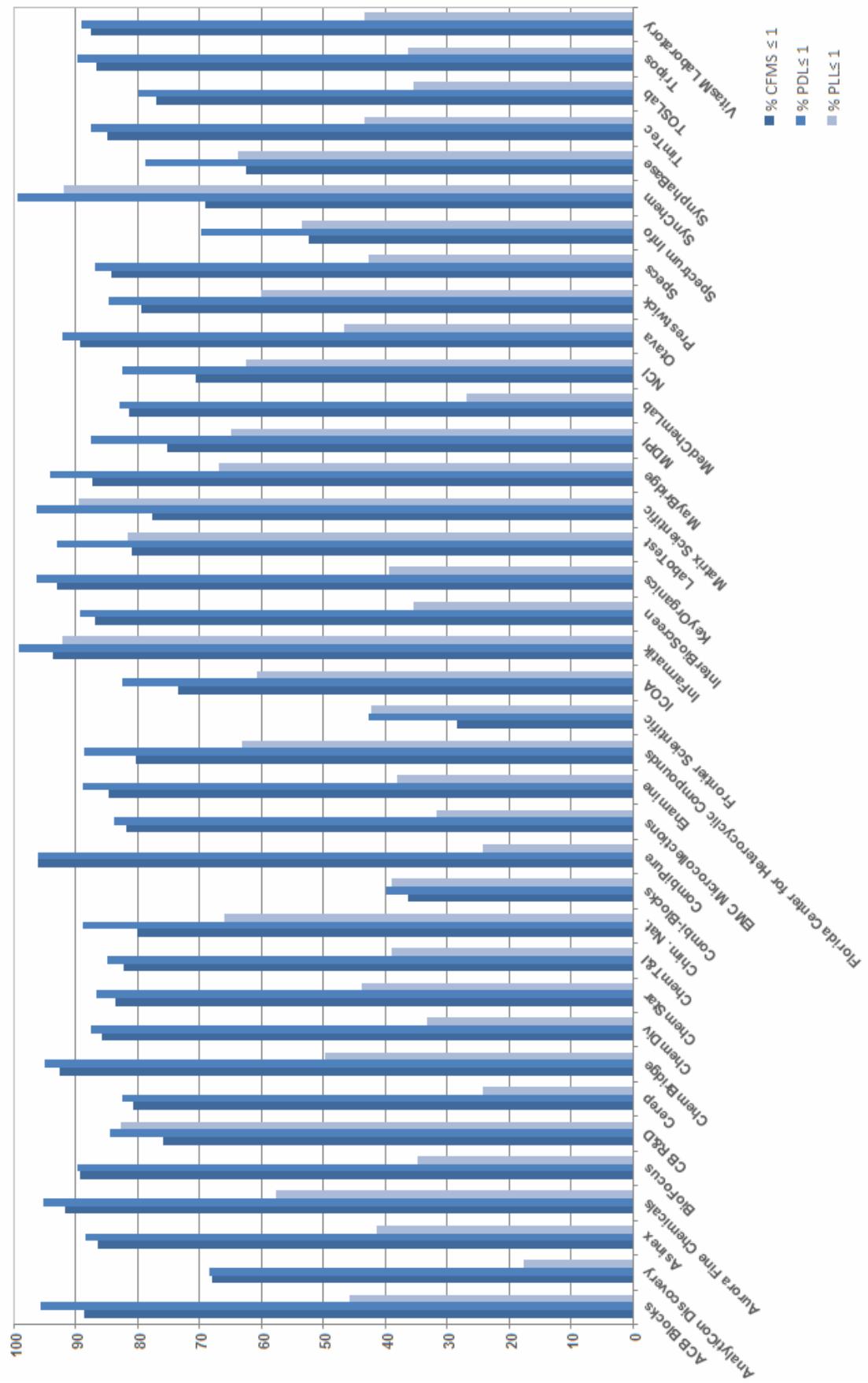


Figure 18. Pourcentage de molécules ayant un score PDL ≤ 1, PLL ≤ 1 et CFMS ≤ 1.

2. Structures privilégiées

La présence de nombreux composés avec une structure privilégiée dans un ensemble de molécules devrait théoriquement favoriser un bon taux de touches positives lors des tests de criblages. Le pourcentage de composés contenant une structure privilégiée peut également donner une indication sur les propriétés « drug-like » d'une base (Figure 19). Les bases ont en moyenne 24 % de composés avec une structure privilégiée. On notera tout d'abord que la base Prestwick a un pourcentage de structures privilégiées qui est dans la moyenne haute, ce qui tend à vérifier le bon fonctionnement du filtre. Les deux bases, toutes tailles confondues, arrivant de loin en tête du classement suivant ce critère sont ACB Blocks et AnalytiCon Discovery. InterBioScreen et ChemDiv sont les bases de grandes tailles contenant le plus de structures privilégiées. La base CB R&D ne possède quant à elle aucune structure privilégiée, mais cela n'est pas choquant étant donnée qu'elle ne compte que 176 structures au total.

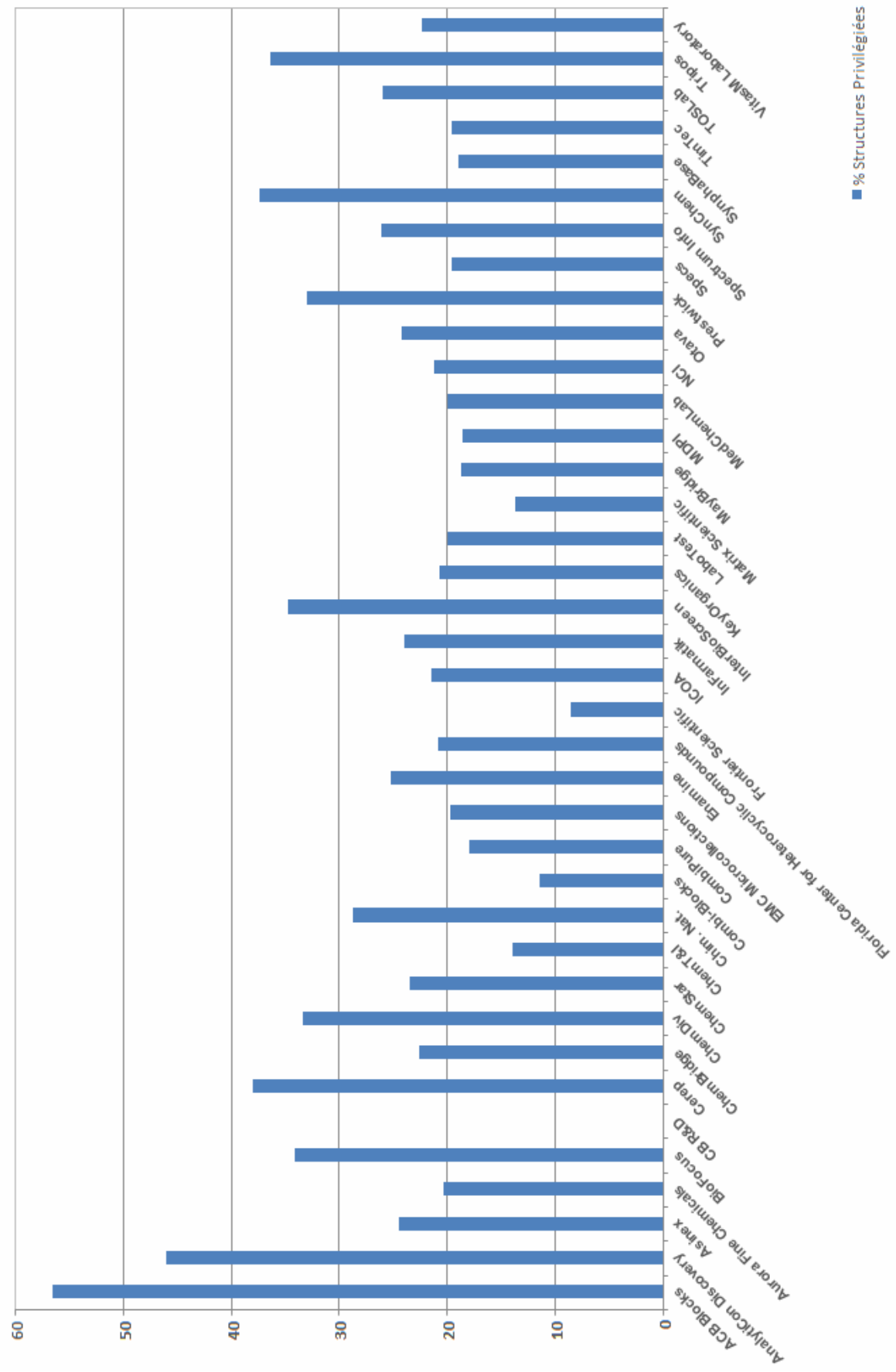


Figure 19. Pourcentage de structures privilégiées.

C. Diversité basée sur les « fingerprints »

Nous avons utilisé les quatre fingerprints cités précédemment pour évaluer la diversité des bases. Les résultats sont représentés dans la Figure 20. Les résultats sont présentés sous formes de pourcentages de clusters de la base ayant la diversité maximale (pour un type de fingerprints donné). Dans tous les cas la base la plus diverse est la NCI, et les pourcentages exprimés dans le graphique correspondent donc au pourcentage de clusters de la base NCI pour un fingerprint donné. Cela simplifie la représentation des données, et donc la lecture.

D'une manière générale, les SSKey3DS donnent des résultats comparables aux fingerprints MACCS, et les TGD des résultats comparables aux fingerprints TGT. Cela semble normal puisque les fingerprints SSKey3DS et MACCS sont du même type (basées principalement sur des fragments), et TGD et TGT sont également du même type (basées sur des pharmacophores).

La base NCI est donc nettement la plus diverse, quelque soit le fingerprint utilisé. Cette base est particulière dans sa conception, puisque les composés qu'elle contient ont des origines très variées. Le résultat est que cette base est beaucoup plus diverse que la base TimTec qui contient pourtant deux fois et demie plus de composés. NCI est cependant un cas à part, et d'une manière générale, plus une base contient de composés, plus sa diversité est grande. Parmi les autres bases, la plus grande, TimTec, est aussi la plus diverse.

Les coefficients de corrélation entre la taille des bases et leurs diversités mesurées par les différentes fingerprints sont présentés Tableau 12. A la vue de ces résultats il est clair qu'il existe une corrélation entre la taille des bases et leur diversité. Cette corrélation est plus forte avec les fingerprints basées sur des fragments qu'avec les fingerprints pharmacophoriques. Les coefficients de corrélation sont très hétérogènes puisqu'ils varient de 0,87 (MACCS) à 0,54 (TGT). Nous pouvons donc en déduire que, même si les conclusions générales sont semblables quelles que soient les fingerprints utilisées, il existe malgré tout des différences notables entre les différentes fingerprints.

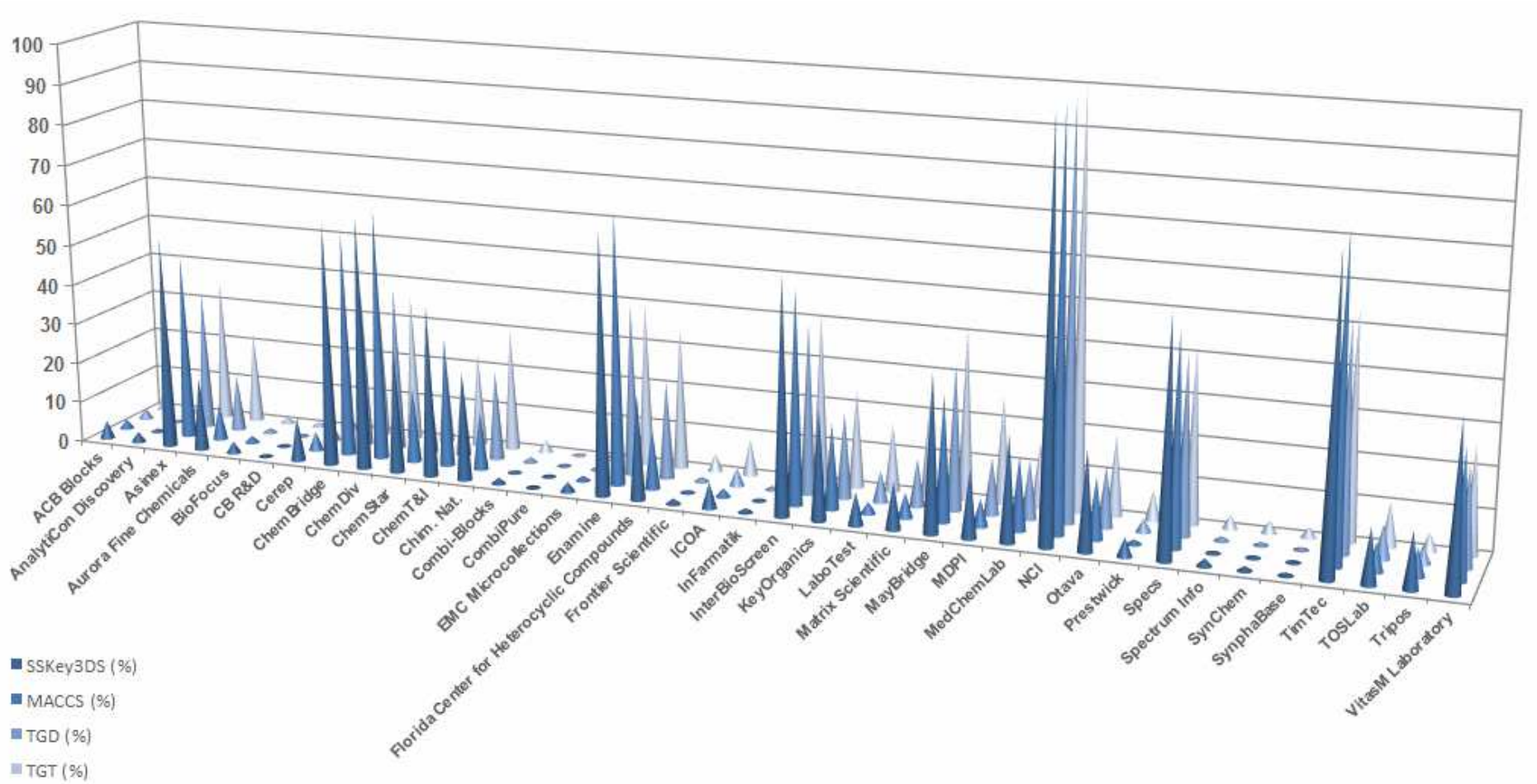

100
90
80
70
60
50
40
30
20
10
0
ACB Blocks
AnalytiCon Discovery
Asinex
Aurora Fine Chemicals
BioFocus
CB R&D
Cerep
ChemBridge
ChemDiv
ChemStar
ChemT&I
Chim. Nat.
Combi-Blocks
CombiPure
EMC Microcollections
Enamine
Florida Center for Heterocyclic Compounds
Frontier Scientific
ICOA
InFarmatik
InterBioScreen
KeyOrganics
LaboTest
Matrix Scientific
MayBridge
MDPI
MedChemLab
NCI
Otava
Prestwick
Specs
Spectrum Info
SynChem
SynphaBase
TimTec
TOSLab
Tripos
VitasM Laboratory
SSKey3DS (%)
MACCS (%)
TGD (%)
TGT (%)

Figure 20. Diversités des bases évaluées en utilisant quatre fingerprints différentes. Les valeurs sont exprimées en diversité relative (c.-à-d. en pourcentage de la base avec la diversité maximale, à savoir NCI pour les quatre fingerprints).

Fingerprints	**r^2**
SSKey3DS	0,81
MACCS	0,87
TGD	0,70
TGT	0,54

Tableau 12. Etude de la corrélation entre la diversité des bases et le nombre de composés qu'elles contiennent.

D. Diversité basée sur la fragmentation

1. Frameworks, Scaffolds et chaînes latérales

La Figure 21 représente les frameworks, les scaffolds et les chaînes latérales de chaque base. Les résultats sont exprimés en pourcentage de représentativité de la base totale. Les tailles des bases sont corrélées au nombre de frameworks avec un r^2 de 0,86, au nombre de scaffolds avec un r^2 0,88 et au nombre de chaînes latérales avec un r^2 de 0,65. Nous voyons donc que les chaînes latérales sont beaucoup moins corrélées avec la taille des bases que les frameworks et les scaffolds.

Pour la totalité de notre base virtuelle, il y a 98 000 frameworks, 600 000 scaffolds et 40 000 chaînes latérales. Il est cependant très difficile de donner un schéma général à partir de la figure. Quelques points sont quand même à noter. Le plus frappant est que la base NCI est celle qui a de loin le plus grand nombre de chaînes latérales (38 % de la base), alors qu'elle est loin d'avoir le plus grand nombre de frameworks et de scaffolds. Cela montre une fois de plus le profil très particulier de cette base. La base qui a le plus de chaînes latérales après NCI est TimTec. On note que cette base a un profil assez équilibré puisque les pourcentages de représentativité de la base totale en termes de frameworks et de scaffolds sont très proches de celui des chaînes latérales. On notera que c'est la base Enamine est la base la plus diverse en

termes de frameworks et de scaffolds, et qu’elle a un nombre de chaînes latérales tout à fait correct.

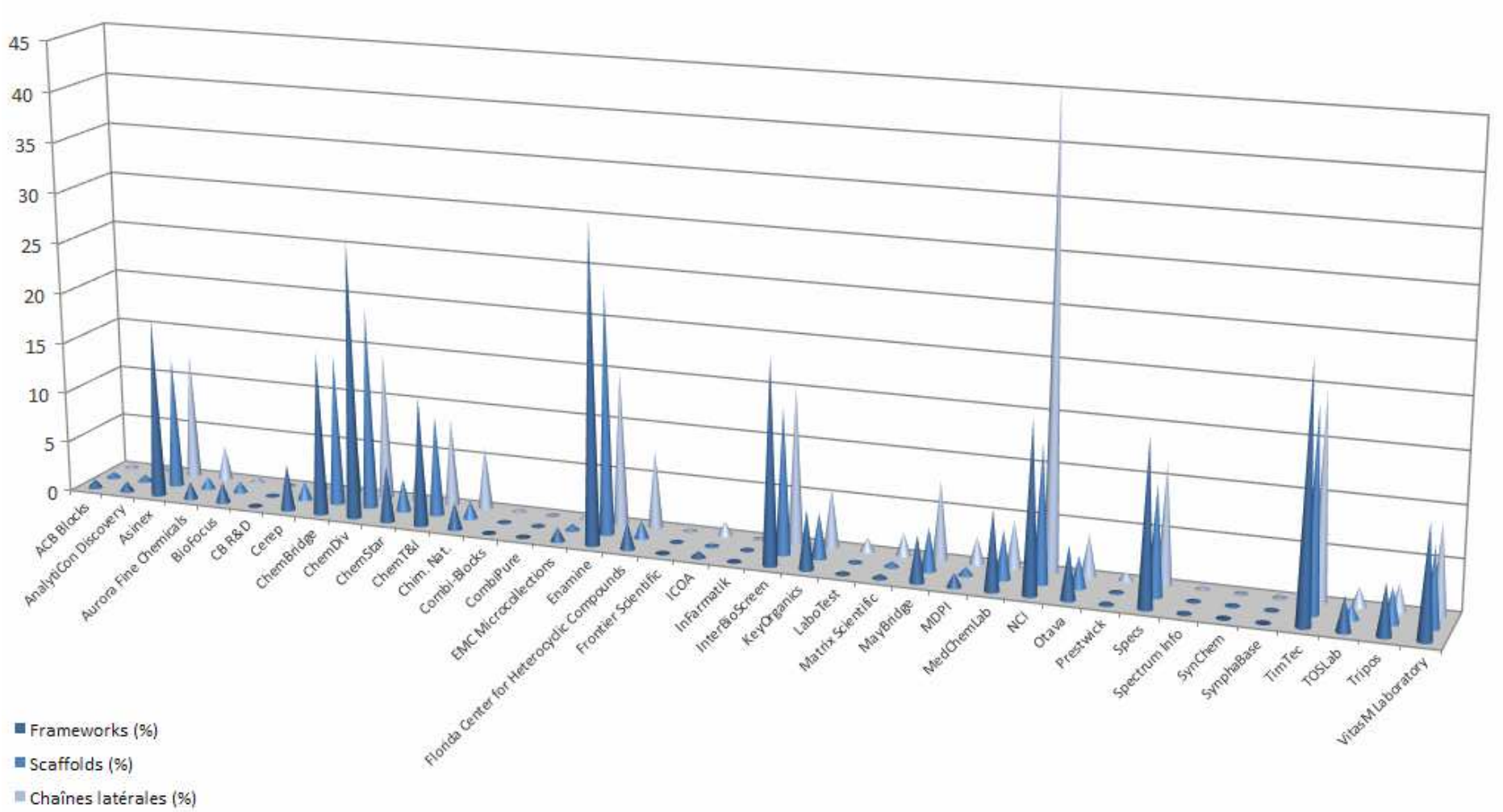

45
40
35
30
25
20
15
10
5
0
ACB Blocks
AnalytiCon Discovery
Asinex
Aurora Fine Chemicals
BioFocus
CB R&D
Cerep
ChemBridge
ChemDiv
ChemStar
ChemT&I
Chim. Nat.
Combi-Blocks
CombiPure
EMC Microcollections
Enamine
Florida Center for Heterocyclic Compounds
Frontier Scientific
ICOA
InFarmatik
InterBioScreen
KeyOrganics
LaboTest
Matrix Scientific
MayBridge
MDPI
MedChemLab
NCI
Otava
Prestwick
Specs
Spectrum Info
SynChem
SynphaBase
TimTec
TOSLab
Tripos
VitasM Laboratory
Frameworks (%)
Scaffolds (%)
Chaînes latérales (%)

Figure 21. Diversités des bases évaluées en utilisant les frameworks, les scaffolds et les chaînes latérales. Les valeurs sont exprimées en pourcentage de la base totale (par exemple, la base NCI possède la moitié des chaînes latérales présentent dans la base totale).

Fragments	**r^2**
Frameworks	0,86
Scaffolds	0,88
Chaînes latérales	0,65

Tableau 13. Etude de la corrélation entre les frameworks, les scaffolds, chaînes latérales des bases et le nombre de composés qu'elles contiennent.

2. RECAP

Les composés de toutes les bases ont été fragmentés en utilisant les règles RECAP. Le nombre de fragments obtenus pour chaque base est présenté dans la Figure 22. Pour l'ensemble des composés de la base, il y a en moyenne 2,7 fragments par composé. L'ensemble de notre chimiothèque comporte environ 350 000 fragments.

La moyenne du nombre de fragments par base est de 19 202. La base comportant le plus de fragments est TimTec. On trouve ensuite NCI et ChemDiv. NCI n'est donc pas la base la plus diverse en termes de fragments, comme cela était le cas en mesurant la diversité à l'aide de fingerprints. Son nombre de fragments est tout de même très bon, puisqu'elle a 84 % du nombre de fragments de TimTec, alors qu'elle a deux fois et demie moins de composés. La corrélation entre le nombre de fragments d'une base et son nombre de composés est bonne, avec un r^2 de 0,86. Les bases de grandes tailles s'éloignant le plus de la droite de régression linéaire sont NCI, Specs et Chem T&I. NCI et Specs ont un nombre de fragments très important au vu du nombre de composés dans ces bases. Chem T&I a, quant à elle, un faible nombre de fragments différents au vu du nombre de composés de la base.

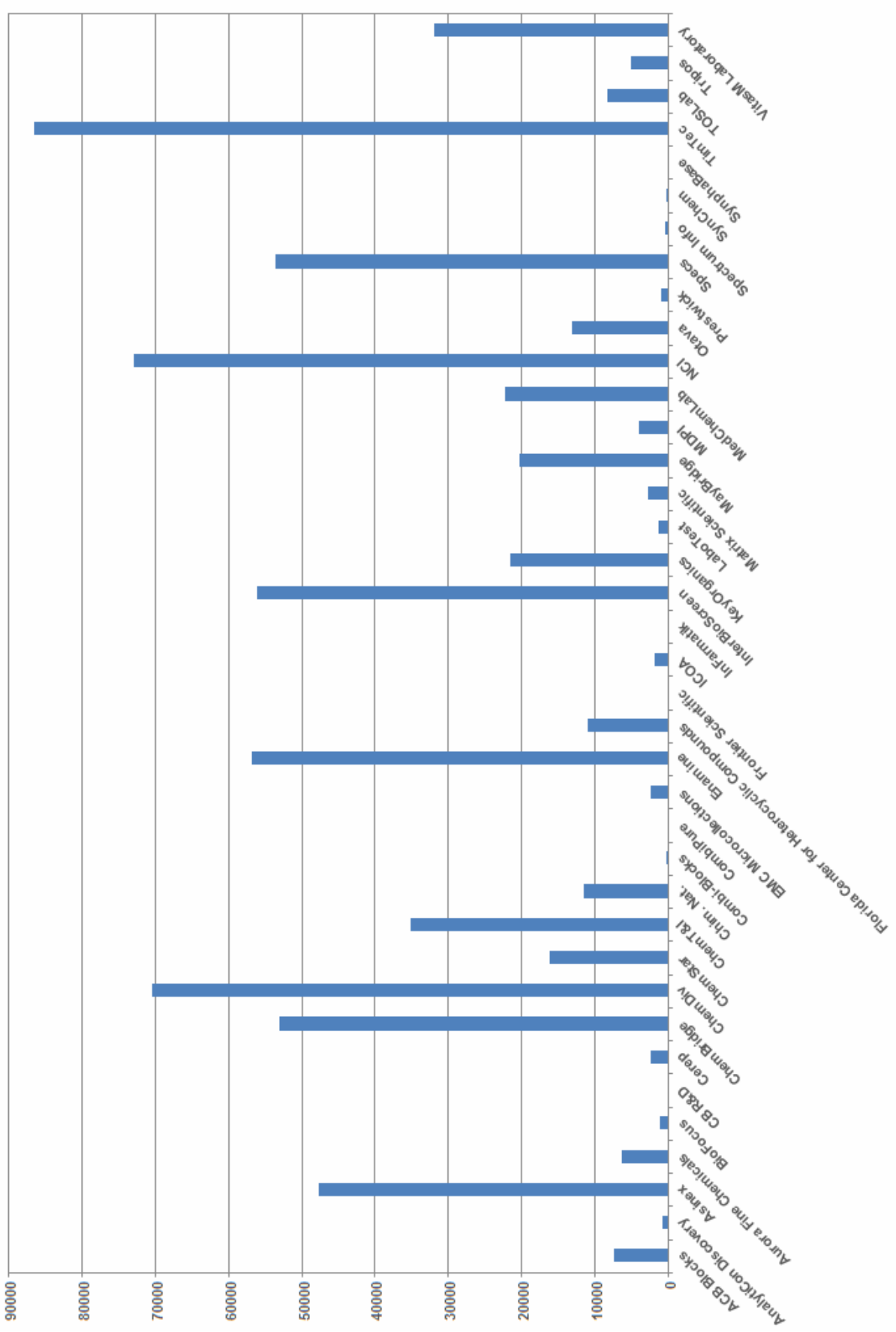


Figure 22. Nombre de fragments obtenus.

E. Estimation de la diversité globale des bases

1. Méthode

Nous avons estimé la diversité chimique des différentes bases constituant notre chimiothèque virtuelle en utilisant des fingerprints, les scaffolds, les frameworks, les chaînes latérales et RECAP. Nous avons pu noter des tendances globales, mais également des disparités de résultats entre les différentes méthodes. Notre objectif est donc de proposer une analyse de la diversité des bases combinant l'ensemble des méthodes utilisées. Nous allons, pour chaque base, faire une moyenne pondérée des différentes diversités mesurées.

La seule difficulté de cette étape est de déterminer les coefficients de pondération. Nous avons pour notre part fait les choix suivants :

- Tout d'abord les résultats de la mesure de la diversité par chaque méthode seront standardisés. Les résultats seront exprimés en pourcentage par rapport à la valeur de la base la plus diverse par la méthode en question.

- Les méthodes utilisées seront réparties en deux grandes familles : les fingerprints, et les méthodes basées sur des sous structures de tailles relativement importantes, à savoir les scaffolds, les frameworks, les chaînes latérales et RECAP. Chacune de ces grandes familles aura le même poids, à savoir un coefficient de 1/2.

- Au sein de la famille des fingerprints, nous distinguons deux sous-familles : les fingerprints basées sur des petites sous structures, et les fingerprints basées sur des pharmacophores. Etant donné que nous avons utilisé deux fingerprints dans chaque famille, nous appliquerons un coefficient de 1/8 aux résultats de chaque fingerprint afin d'obtenir un coefficient global de 1/2 pour les méthodes basées sur les fingerprints.

- Nous diviserons l'autre grande famille en trois sous-familles : une famille représentant les squelettes des molécules (comprenant les frameworks et les scaffolds), une famille représentant les chaînes latérales, et une famille représentant les fragments rétrosynthétiques. Nous utiliserons donc, au sein de cette famille, les coefficients de 1/12 pour les frameworks, de 1/12 pour les squelettes, de 1/6 pour les chaînes latérales, de 1/6 pour les fragments rétrosynthètiques. Cela donnera donc un poids de

0,5 à cette grande sous famille correspondant aux sous structures de tailles relativement importantes.

Nous utiliserons donc la formule suivante pour calculer la diversité des bases :

$$D_{Totale} = \frac{1}{8} \times D_{SSKey-3DS} + \frac{1}{8} \times D_{MACCS} + \frac{1}{8} \times D_{TGD} + \frac{1}{8} \times D_{TGT} + \frac{1}{12} \times D_{Frameworks} + \frac{1}{12} \times D_{Scaffolds} + \frac{1}{6} \times D_{Chaînes\ Latérales} + \frac{1}{6} \times D_{Fragments}$$

Équation 2

2. Résultats

La Figure 23 présente la diversité globale des bases, calculée suivant l'équation 2. La base NCI arrive en tête du classement. Cela traduit ce que nous avons vu dans nos analyses, à savoir que c'est une base très diverse. La base commerciale la plus diverse est TimTec. Ce résultat doit être nuancé par le fait que c'est la base qui comporte le plus de composés et qui est la plus récente de notre base. Enamine, ChemDiv, InterBioScreen, ChemBridge, Specs et Asinex suivent TimTec en termes de diversité globale. Les bases les moins diverses sont celles qui contiennent le moins de composés, c'est-à-dire celles qui proposent des réactifs pour la chimie combinatoire.

La diversité globale est corrélée à la taille des bases avec un r^2 de 0,74. Mais il est aussi intéressant d'étudier la diversité relative, celle apportée par un composé de chaque base. Ce critère est particulièrement intéressant lorsque les produits de tests sont limités ; on cherche alors à obtenir le maximum de diversité avec le minimum de produits (Figure 24). Alors que le classement précédent favorisait les bases de grandes tailles, celui-ci favorise les bases de petites tailles. Ainsi la première base est celle qui a le moins de composés, à savoir SymphaBase (147 composés). On notera que la base Prestwick (1 117 composés), qui est contrairement à SymphaBase destinée au criblage, arrive en deuxième position. Les bases commerciales de grandes tailles destinées au criblage, pénalisées par leur grand nombre de composés, arrivent en fin de classement. La base NCI se trouve quant à elle au milieu du classement. En effet, même si cette dernière comporte un grand nombre de composés, ils sont suffisamment divers et originaux pour qu'elle ne se retrouve pas en fin de classement avec les grandes bases commerciales.

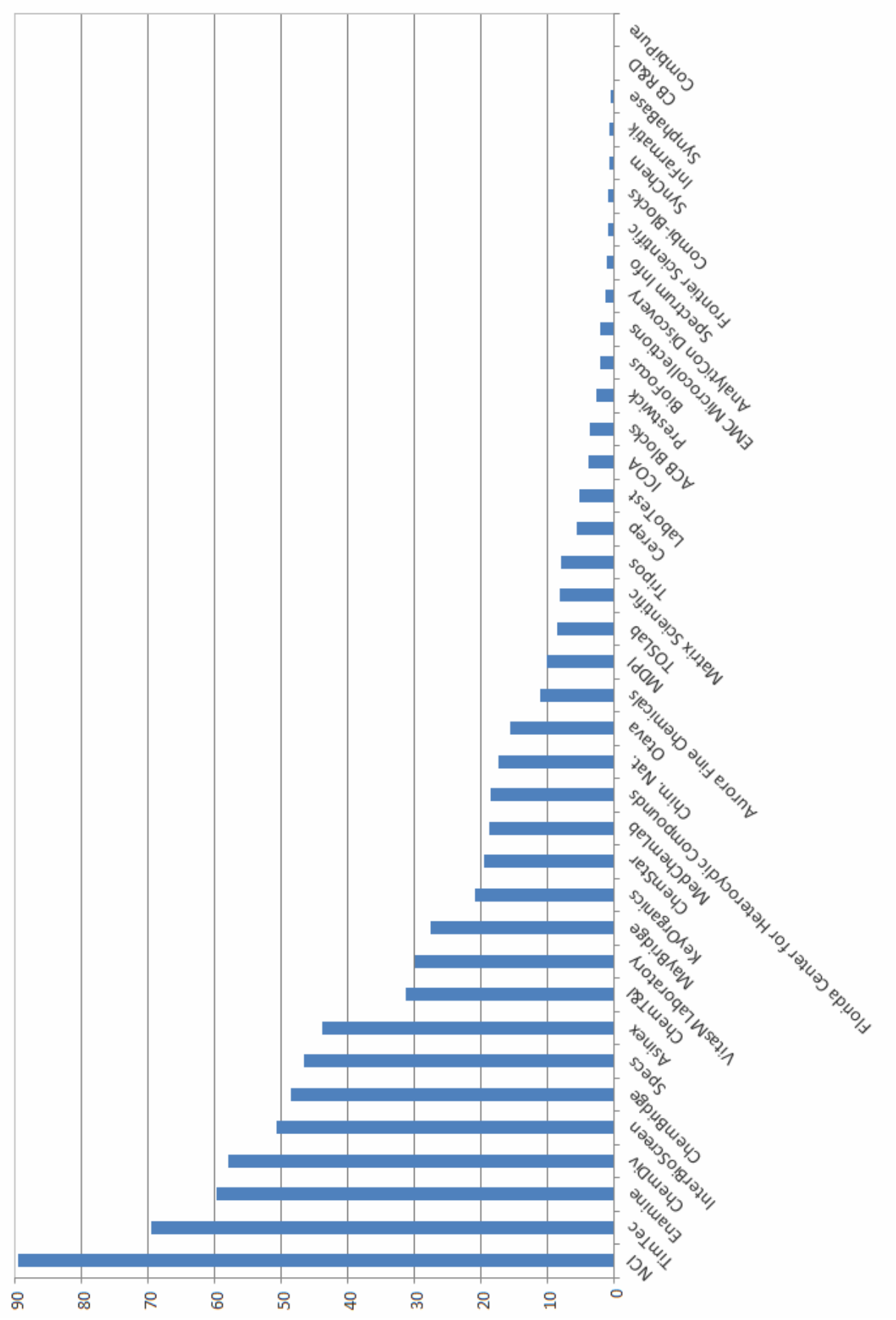


Figure 23. Classement des bases en fonction de leur diversité globale.

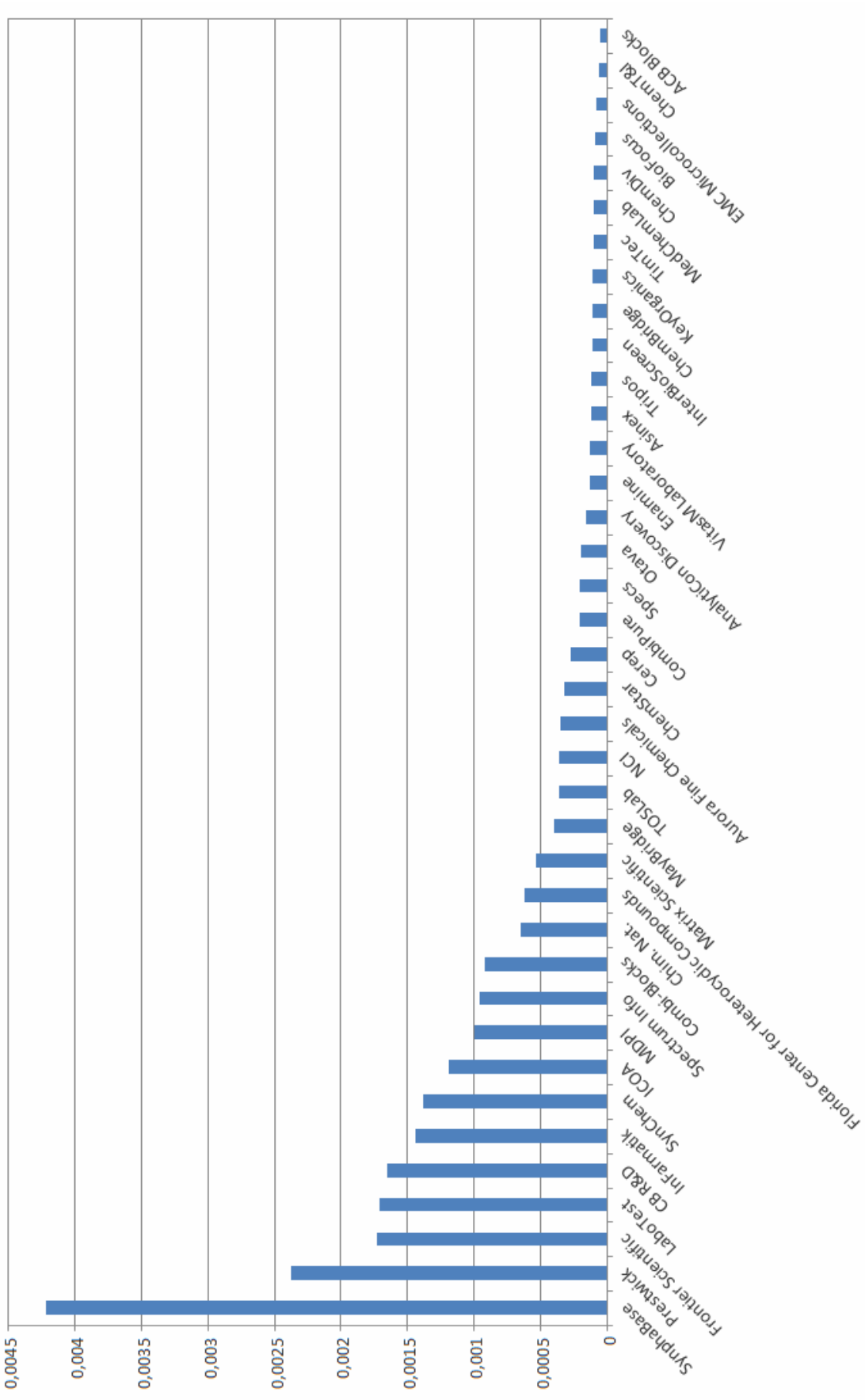


Figure 24. Classement des bases en fonction de la diversité relative.

III. Conclusion

Nous avons présenté l'analyse des bases académiques ou commerciales disponibles pour des tests de criblage. Ces bases ont été analysées en termes de propriétés « drug-like », « lead-like » et de structures privilégiées. La diversité a également été étudiée par quatre fingerprints, frameworks, scaffolds, chaînes latérales et fragment RECAP. A partir de ces différentes analyses de diversité nous avons défini un score de diversité globale qui nous a servi à classer les bases par diversité. Un classement des bases en fonction de la diversité globale par composé a également été établi.

1. Bradley, M.P. An overview of the diversity represented in commercially-available databases *J. Comput. Aided Mol. Des.* **2002**, *16*, 299-300.
2. Voigt, J. H.; Bienfait, B.; Wang, S.; Nicklaus, M. C. Comparison of the NCI Open Database with Seven Large Chemical Structural Databases *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 702-712.
3. Mozziconacci, J.C., Arnoult, E., Baurin, N., Marot, C., Morin-Allory, L., *Preparation of a molecular database from a set of 2 million compounds for virtual screening applications : gathering, structural analysis and filtering*, 9th Electronic Computational Chemistry Conference, World Wide Web, march 2003.
4. Baurin, N.; Baker, R.; Richardson, C.; Chen, I.; Foloppe, N.; Potter, A.; Jordan, A.; Roughley, S.; Parratt, M.; Greaney, P.; Morley, D.; Hubbard, R.E. Drug-like Annotation and Duplicate Analysis of a 23-Supplier Chemical Database Totalling 2.7 Million Compounds *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 643-657.
5. Sirois, S.; Hatzakis, G.; Wei, D.; Du, Q.; Chou, K.C. Assessment of chemical libraries for their druggability. *Comput. Biol. Chem.* **2005**, *29*, 55-67.
6. Cummins, D.J.; Andrews, C.W.; Bentley, J.A.; Cory, M. Molecular Diversity in Chemical Databases: Comparison of Medicinal Chemistry Knowledge Bases and Databases of Commercially Available Compounds. *J. Chem. Inf. Comput. Sci.* **1996**, *36*, 750-763.
7. Krier, M.; Bret, G.; Rognan, D. Assessing the Scaffold Diversity of Screening Libraries. *J. Chem. Inf. Model.* **2006**, *46*, 512-524.
8. Baurin, N., *Etude et développement de techniques QSAR pour la recherche de molécules d'intérêt thérapeutique , Texte imprimé : criblage virtuel et analyse de chimiothèques*, Thèse, Université d'Orléans, France, **2002**.
9. Monge, A. Arrault, A., Marot, C, Morin-Allory, L. Managing, Profiling and Analyzing a Library of 2.6 Million Compounds Gathered from 32 Chemical Providers. *Mol. Divers.* **2006**, DOI : 10.1007/s11030-006-9033-5
10. MACCS II Manual. MDL Information Systems, Inc.
11. Lewell, X. Q.; Judd, D. B.; Watson, S. P.; Hann, M. M. RECAP-Retrosynthetic Combinatorial Analysis Procedure: A Powerful New Technique for Identifying Privileged Molecular Fragments with Useful Applications in Combinatorial Chemistry. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 511-522.
12. Fechner, U.; Schneider, G. Flux (1): A Virtual Synthesis Scheme for Fragment-Based de Novo Design. *J. Chem. Inf. Model.* **2006**, *46*, 699-707.
13. Todorov, N. P.; Buenemann, C. L.; Alberts, I. L. Combinatorial Ligand Design Targeted at Protein Families. *J. Chem. Inf. Model.* **2005**, *45*, 314-320.
14. Bemis GW, Murcko MA. Properties of known drugs. 2. Side chains. *J.Med.Chem* **1999**, *42*, 5095-5099.

Chapitre 4. Application de *ScreeningAssistant* à des projets concrets

I. Sélection d'ensembles de composés par diversité

A. Introduction

La finalité du logiciel *ScreeningAssistant* est de permettre la sélection de composés pour des tests de criblages. Nous avons été amené à sélectionner des composés à la fois pour le criblage virtuel et pour le criblage réel. La sélection pour le criblage virtuel s'est effectuée dans le cadre des projets de criblage du laboratoire. La sélection pour le criblage réel a, quant à elle, été effectuée en collaboration avec des sociétés de biotechnologies françaises et suisses. Le criblage virtuel ayant, de manière générale, un meilleur débit que le criblage réel, les ensembles de composés à sélectionner pour ce type de projet est souvent de plusieurs centaines de milliers de composés, et les algorithmes de diversité utilisés doivent donc pouvoir gérer un grand nombre de composés. D'un autre coté, le fait de travailler sur la conception de chimiothèques réelles nous a permis de rencontrer d'autres types de problèmes. Nous avons ainsi par exemple dû sélectionner des plaques pour compléter une chimiothèque existante, problème qui n'est pas géré par les algorithmes classiques de diversité.

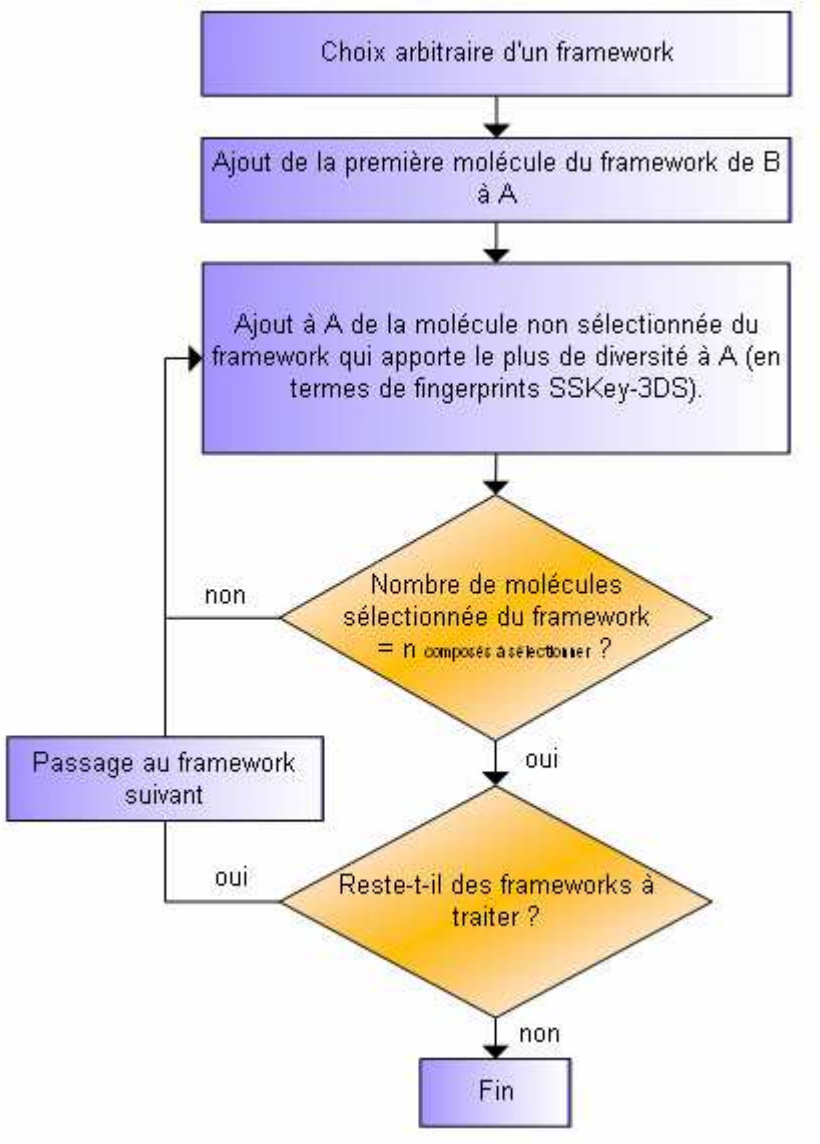
B. Génération d'ensemble de molécules destinées au docking

Nous allons présenter la sélection d'un ensemble de molécules destinées à être testées par docking sur la cible PPAR- γ. Cette cible est impliquée dans différentes pathologies, notamment le diabète non insulino dépendant. Le point de départ de la sélection est une version de notre base virtuelle contenant 2,6 millions de structures uniques. Il faut tout d'abord noter que les ligands connus de cette cible ne sont pas « drug-like ». C'est pourquoi nous prendrons comme base de notre sélection, non pas les molécules « drug-like », mais simplement les molécules qui n'engendrent pas de faux positifs. En plus de cela, nous allons calquer l'espace chimique de notre sélection sur l'espace chimique de 187 ligands PPAR- γ :

- $290 \leq$ masse moléculaire ≤ 670
- $\log P \leq 8$

- $70 \leq TPSA \leq 150$
- $2 \leq$ accepteurs de liaisons H ≤ 11
- $6 \leq$ liaisons pouvant tourner ≤ 22
- $1 \leq$ nombre de cycles ≤ 6
- pas de cycles de plus de 7 membres
- pas de NO_2

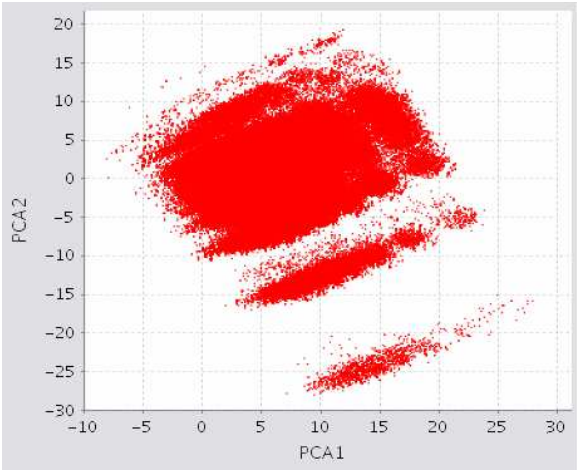
Cette étape est réalisée rapidement et de façon interactive par l'interface graphique de *ScreeningAsssistant*. L'application de ces filtres réduit notre sélection à environ un million de molécules. Or les moyens informatiques à notre disposition limitent nos capacités de tests virtuels à environ 500 000 molécules (nous nous sommes fixé un mois de temps de cacul sur un cluster de 8 PCs). Nous allons donc utiliser un algorithme de diversité pour sélectionner un ensemble de 500 000 composés. Cet algorithme va traiter les composés par frameworks, et les proportions de composés entre les frameworks du groupe de composés initial et de la sélection seront identiques. Au sein de chaque framework, les composés seront choisis en utilisant les fingerprints SSKey-3DS (Figure 25). Cet algorithme a été implémenté dans *ScreeningAssistant*.



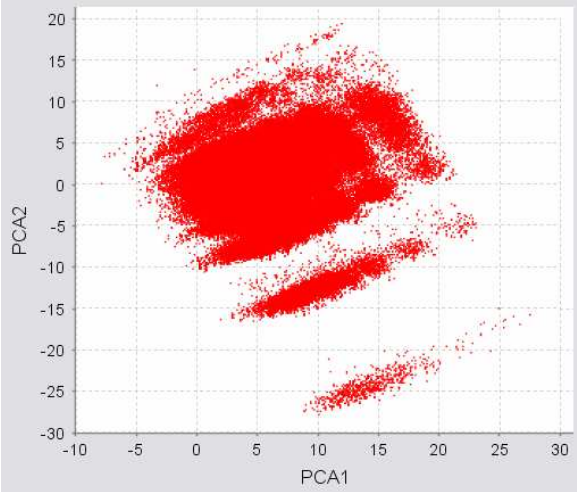
$$n_{\text{composés à sélectionner}} = n_{\text{composés du framework de B}} \times \frac{n_{\text{composés à sélectionner au total}}}{\text{n}_{\text{composés de B}}}$$

Figure 25. Algorithme de sélection d'un ensemble par diversité. L'algorithme traite les composés par frameworks, et utilise les fingerprints SSKey-3DS pour choisir les composés au sein d'un framework. *A* correspond à l'ensemble à sélectionner, et *B* à l'ensemble des composés de départ.

L'espace chimique avant et après sélection par diversité est représenté Figure 26. Cela nous permet de constater que les deux espaces chimiques sont similaires, et donc qu'il n'y a pas (en gardant en tête les limitations de cette méthode de contrôle) d'erreur flagrante dans la sélection par diversité.



a) Molécules de notre chimiothèque ayant des propriétés physicochimiques similaires aux ligands PPAR- γ (environ un million).



b) Molécules sélectionnées par diversité parmi les molécules de notre chimiothèque ayant des propriétés physicochimiques similaires aux ligands PPAR- γ (environ 500 000).

Figure 26. Nous avons sélectionné un ensemble représentatif (environ 18 000 molécules) de notre base par diversité et calculé pour chacune le log P, la masse ainsi que les fingerprints SSKey3DS. Les deux premières composantes principales de cet ensemble représentatif seront utilisées pour visualiser les molécules de notre base en deux dimensions (20 % de la variance). Nous comparons ainsi l'ensemble des molécules sélectionnées pour la cible PPAR-γ dans notre chimiothèque (a), avec les molécules sélectionnées par diversité dans cet

ensemble (b). On constate que les molécules sélectionnées par diversité sont représentatives de l'espace chimique de départ.

C. Génération d'ensembles de molécules destinées au criblage réel

1. Introduction

Lors de toute sélection la notion de l'espace chimique à considérer se pose. Généralement dans les cas où les ligands sont connus, on utilisera un espace chimique calqué sur celui de ces ligands connus. Mais dans les cas plus généraux cette technique n'est pas applicable. On s'oriente donc habituellement vers les notions d'espaces chimiques « drug-like » et « lead-like ». Certains préféreront en effet se focaliser sur un espace « lead-like », afin de laisser une plus grande liberté à l'optimisation. D'autres préféreront au contraire l'espace « drug-like » pour couvrir une gamme plus large de l'espace chimique. Une publication récente [1] traite de ce problème. Afin de faire un choix sur l'espace chimique à considérer pour les criblages, il faut avant tout s'intéresser à la définition d'une touche intéressante. Une touche doit avant tout être un composé non réactif, dont la structure et la pureté ont été vérifiés, avec une activité à une concentration en général inférieure à 20 μM en criblage à haut débit. Il est donc important de filtrer les composés pouvant engendrer de faux positifs. De plus, on pourra utiliser le concept d' « efficacité » de ligand pour classer les hits [2]. Cette notion s'appuie sur la définition de l'affinité de liaison par atome définie par Kutz et al. [3]. On peut ainsi calculer l'énergie libre de liaison du ligand à partir de la constante de dissociation K_d (on utilisera la valeur de l'IC_{50} pour le K_d) :

$$\Delta G = -R \times T \times \ln K_d \qquad \textbf{(Équation 3)}$$

L'IC_{50} est la moitié de la concentration d'un inhibiteur nécessaire pour obtenir 50 % d'inhibition d'une enzyme, d'un recepteur, d'une cellule ou d'un microorganisme. On peut à partir de l'Equation 3 déduire l'énergie libre de liaison par atome en divisant l'énergie libre de liaison du ligand par son nombre d'atomes lourds :

$$\Delta g = \frac{\Delta G}{N_{\text{atomes lourds}}}$$ (**Équation 4**)

Les composés avec les plus fortes valeurs de Δg seront les plus prometteurs dans le processus d'optimisation de touches.

Cette technique de sélection favorisera les composés les moins complexes. Il est important également de prendre en compte d'autres paramètres tels que les propriétés ADME-Tox. Il apparaît qu'il est plus facile d'augmenter l'affinité d'un composé que d'optimiser ses propriétés ADME-Tox [4].

La notion d' « efficacité » par atome et de sélection de composés avec de bonnes propriétés ADME-Tox tend à prendre le dessus sur la méthode standard qui consiste à sélectionner les touches avec les plus fort IC_{50}. On peut ainsi être tenté de ne sélectionner que des composés « lead-like », même si ces derniers ont des valeurs d'IC_{50} plus faibles que les composés « drug-like ». Si l'on considère les propriétés ADME et l'efficacité par atome, cela n'est pas gênant. Cependant, le revers de la médaille est que si les composés « lead-like » laissent plus de champ libre pour l'optimisation, ils la rendent aussi plus compliquée. Les composés « drug-like », même s'ils laissent moins de place à l'optimisation, présentent certains avantages. D'une part, ils sont souvent plus simples à optimiser. D'autre part, ils présentent dès le départ une activité importante. De plus, dans les grandes compagnies pharmaceutiques, les composés « drug-like » sont souvent issus d'un processus d'optimisation de lead. Ils ont donc déjà un certain niveau de spécificité biologique, évitant ainsi les faux positifs. En outre, l'espace chimique du lead à partir duquel le composé est issu a souvent été exploré.

En résumé, il n'est pas possible de choisir à priori entre un espace « lead-like » ou un espace « drug-like » pour le criblage. Il faut choisir les molécules en tenant compte de l' « efficacité » par atome. Les composés « lead-like » laissent plus de liberté pour la phase d'optimisation, alors que d'un autre côté, les composés « drug-like » sont, dès le départ, plus actifs et faciles à optimiser.

2. Conception de la base

Nous allons présenter dans cette partie le travail de conception de la base de criblage d'une société pharmaceutique (Hybrigenics). La base finale sera de 100 000 composés.

Lors du début de ce travail la société disposait déjà d'une chimiothèque de molécules provenant de trois sources : Prestwick (820 composés), le laboratoire de pharmacochimie de la communication cellulaire de Strasbourg (3200 composés), et enfin Tripos (6953 composés). Les composés issus de Tripos avaient été sélectionnés précédemment par notre laboratoire à partir d'une version filtrée de la base LeadQuest.

La conception de la base totale s'est effectuée en plusieurs étapes et s'est étalée sur une période de temps relativement longue (environ 2 ans et demi). Les différentes étapes du travail traduisent donc l'évolution de nos techniques (filtres, algorithmes de diversité…), et notamment du logiciel *ScreeningAssistant*. Au départ le travail a été réalisé avec *ScreeningAssistant* et MOE, puis au fil des travaux on arrivera rapidement à un usage exclusif de *ScreeningAssistant* et d'algorithmes développés dans le laboratoire. Ces travaux, ont permis de réfléchir, avec des chimistes médicinaux, à des améliorations du logiciel *ScreeningAssistant*.

a. Sélection de 5 500 composés de ChemBridge

Le premier travail a été de sélectionner 5 500 composés qui complètent au mieux la chimiothèque de la société pharmaceutique en terme de diversité, et ce à partir d'un fichier de 16486 composés du fournisseur ChemBridge, présélectionnés par cette société. Ce travail a été réalisé avec MOE. Les fonctions réactives et les propriétés « drug-like » sont évaluées avec le script MOE « Evaluation of Druglikeness » disponible sur le site web SVL Exchange [5].

Tout d'abord les molécules déjà présentes dans la sélection des composés de la LeadQuest de Tripos sont éliminés. Cela représente 13 molécules. Les molécules contenant des fonctions réactives sont ensuite écartées, ce qui correspond à 1 574 molécules. Enfin, une sélection « drug-like » est réalisée, éliminant ainsi les molécules qui valident l'un des critères suivants :

- nombre de donneurs de liaisons H > 5

- nombre d’accepteurs de liaisons H > 10
- nombre de cycles de plus de 7 atomes
- nombre d’halogènes > 7
- nombre de cycles supérieur >6
- présence d’atomes non organiques (autres que C, H, O, N, S, P, Br, F, Cl et I)
- absence d’atome d’azote ou d’oxygène
- nombre de liaisons pouvant tourner > 15

La liste de molécules qui nous a été communiquée avait déjà été préfiltrée, et les filtres de donneurs et accepteurs de liaisons H, de nombre d’halogènes, d’atomes non organiques et d’absence d’atome d’azote ou d’oxygène n’éliminent aucune molécule. Les autres filtres permettent de supprimer 36 molécules de la liste. Il reste donc 14 861 molécules après traitement. La comparaison visuelle de ces molécules avec les 6 953 molécules Tripos déjà sélectionnées est réalisée par une analyse en composante principale sur trois axes des descripteurs de surface (VSA) de MOE (Figure 27).

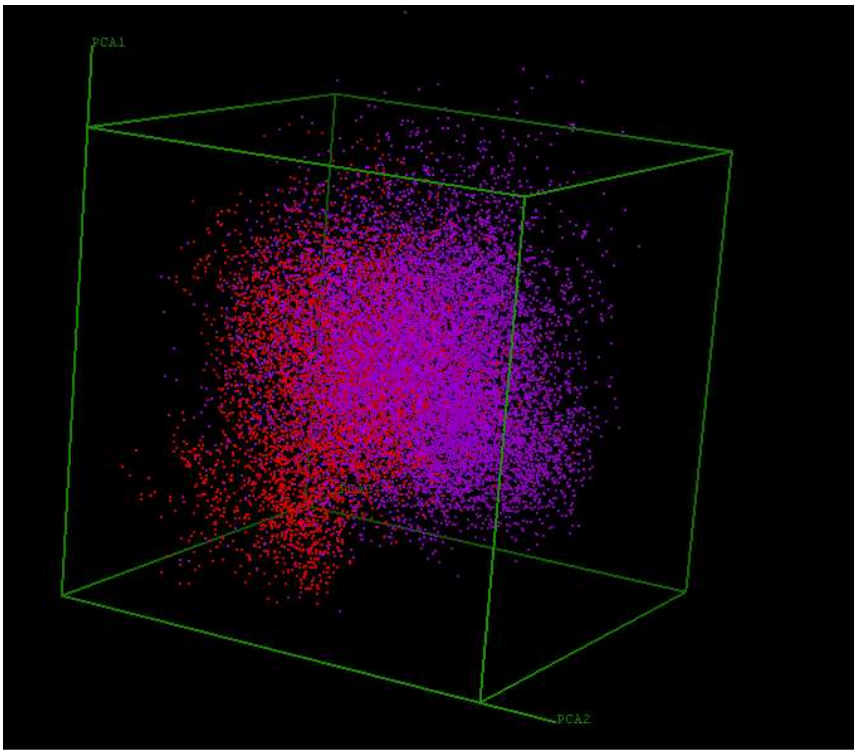


Figure 27. Comparaison de la base de 14 861 molécules générée à partir de produits Chembridge (en rouge) avec la base des 6 953 molécules générée à partir de produits Tripos (en violet).

L'étape suivante a consisté à sélectionner les 5 500 molécules Chembridge les plus diverses à la fois par rapport à l'ensemble des 14 861 molécules Chembridge ayant passé nos filtres, et par rapport aux molécules déjà présentes dans la chimiothèque de la société.

Les 14 861 molécules Chembridge et les molécules de la société ont été combinées ensemble avec MOE, et un classement par diversité a été réalisé avec les clés MACCS. Les 5 500 premières molécules Chembridge de ce classement ont été retenues.

b. Sélection de 10 000 composés de la base VitasM

Suite à ce travail, la société pharmaceutique a souhaité rajouter 10 000 composés du fournisseur VitasM. On partira de 173 803 composés VitasM. Pour ce travail nous utiliserons *ScreeningAssistant*. Nous avons appliqué les filtres suivants :

- les critères « drug-like » utilisés lors de la précédente sélection

- log P ≤ 4,2
- 170 ≤ masse molaire ≤ 450
- nombre de donneurs de liaisons H ≤ 4
- nombre d'hétéroatomes (N, O et S) ≤ 6
- nombre d'halogènes (CF_3 compte pour un seul halogène) ≤ 2
- pas de NO_2
- pas de BOC

Ainsi, 146 946 composés sont « drug-like », et 50 967 sont « lead-like ». Nous continuerons notre étude avec ces composés « lead-like ».

La suite du travail s'est déroulée de la manière suivante :

- les doublons ont été supprimés.
- 30 000 composés vérifiant le mieux les critères « lead-like » ont été sélectionnés.
- parmi ces composés, l'équipe informatique de la société pharmaceutique en a sélectionné 3000 de manière aléatoire, ceci dans le but d'intégrer une diversité purement aléatoire et donc non biaisée par des critères chimiques.
- de notre côté nous avons classé par diversité les 8 000 composés les plus divers parmi l'ensemble de 30 000. Cette sélection s'est faite sans tenir compte des 3 000 composés sélectionnés par la société pharmaceutique. Les 1 000 derniers composés de cette liste de 8 000 sont destinés à compenser les éventuels doublons entre les 3 000 composés sélectionnés aléatoirement et les 7 000 premiers de notre classement par diversité.

La suppression de doublons parmi les 50 967 molécules a été effectuée avec la version bêta 1.12 du programme InChI. Les doublons sont répartis en 47 familles (une famille regroupant les composés ayant le même code unique) dont une famille qui comporte 307 membres. Nous nous sommes rendu compte que la raison pour laquelle ces composés avaient

été regroupés dans la même famille est que le code InChI n'avait pas été calculé pour ces composés. Il s'agit en fait d'une erreur, ou plus exactement d'une incompatibilité entre JOELib et InChI. En effet JOELib code les structures au format MOL en utilisant le type 4 pour les liaisons aromatiques, au lieu d'alterner les simples et doubles liaisons. Cette notation est reconnue par un très grand nombre de logiciels de chemoinformatique, mais ne respecte pas les spécifications de MDL. Dans sa définition, le type 4 pour les liaisons n'est en effet destiné qu'aux requêtes. Nous avons soumis ce problème aux développeurs du code InChI. Ils l'ont corrigé dans les versions suivantes du programme.

Nous avons ensuite classé les molécules de la base VitasM, non présentes dans les bases de la société pharmaceutique et Maybridge, par score croissant et isolé les 30 000 premières (on rappelle que plus le score est petit plus la molécule est lead-like). Le score CFMS maximum atteint pour ces 30 000 molécules est de 0,49. Le score CFMS maximal atteint pour les 50 967 molécules est de 2,47 (on rappelle qu'une molécule avec un score CFMS ≥ 2 n'est absolument pas « lead-like »).

Les 30 000 molécules ainsi obtenues sont classées par diversité suivant leurs clés MACCS, et les 8 000 premières sont sélectionnées.

La Figure 28 permet de visualiser la diversité des 30 000 molécules filtrées par rapport à la base de la société pharmaceutique, et à une autre base qu'ils venaient d'acquérir, à savoir la base MayBridge.

Nous avons utilisé une analyse en composantes principales, sur deux axes, de 52 descripteurs de surface P_VSA de MOE. Les deux axes représentés codent 33 % de la variance.

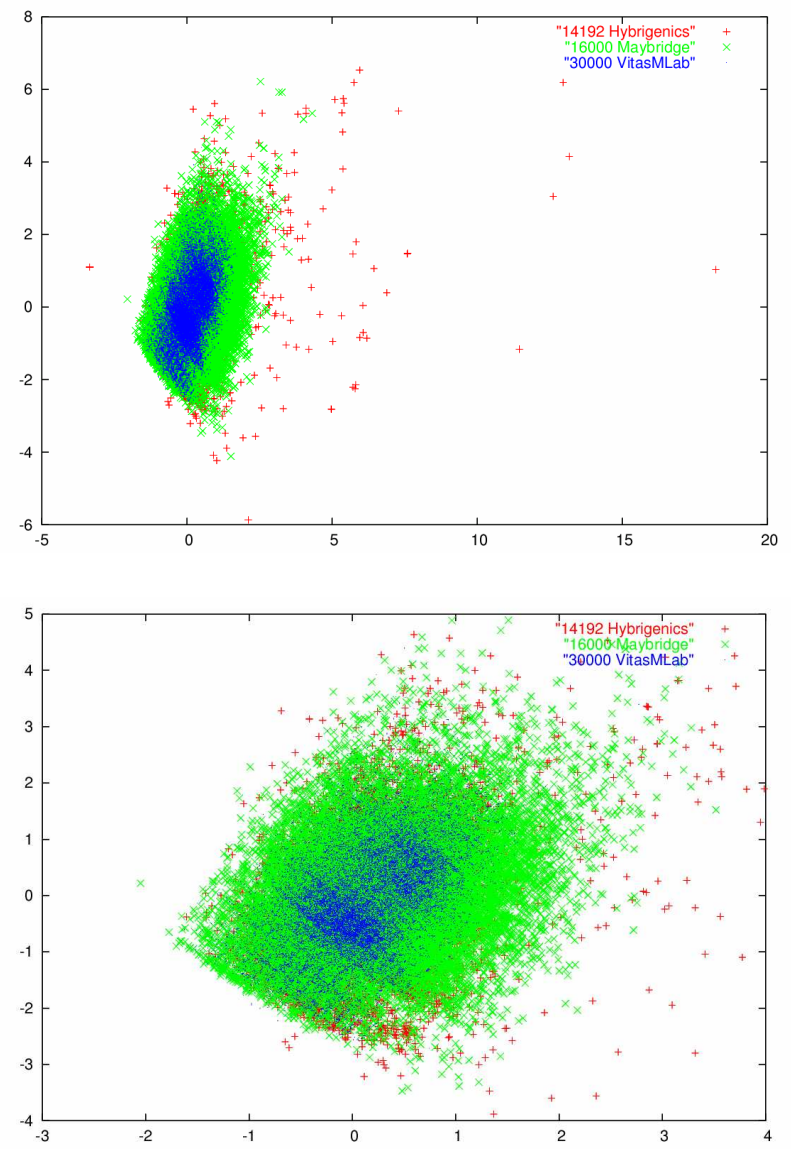


Figure 28. Représentations des 30 000 molécules VitasM (en bleu) et des molécules que possède déjà Hybrigenics (en rouge et vert). Le second graphique est un agrandissement du premier.

Il apparaît clairement sur ces graphiques que les composés sélectionnés dans la base VitasM couvrent un espace chimique plus restreint, bien que cette base soit de taille bien supérieure aux deux autres. Cela s'explique en partie par le fait que le jeu de molécules issu de VitasM est, contrairement aux deux autres bases, « lead-like ». En conséquence, l'espace chimique couvert par ces composés est plus restreint. Cela traduit la volonté de la société pharmaceutique de se tourner vers un espace « lead-like », mais sans pour autant éliminer

l'espace « drug-like », ce qui correspond bien à la tendance actuelle du choix des touches que nous avons évoqué au début de cette partie.

c. Sélection de 35 000 composés ChemDiv

Le dernier travail réalisé sur la chimiothèque de la société pharmaceutique a été de porter le nombre de composés à 100 000. Les composés seront choisis dans la base du fournisseur ChemDiv. Les nombres de composés de ces deux bases sont présentés dans le Tableau 14.

Bases	Nombre total de composés
Hybrigenics (12/05/2006)	65 335
ChemDiv (25/04/2006)	637 388

Tableau 14. Nombre de composés dans les bases de la société pharmaceutique et de ChemDiv.

La base de la société pharmaceutique a été construite en plusieurs étapes, par l'ajout de composés sélectionnés par diversité comme nous l'avons vu dans ce document, mais également par l'ajout de bases dans leurs intégralités. La présence de doublons est donc inévitable. Dans cette base, le logiciel *ScreeningAssistant* a identifié 1 249 doublons. La chimiothèque de la société pharmaceutique possède donc 64 086 composés uniques. Etant donné que nous voulons arriver à une base 100 000 composés et que quelques composés seront sans doute manquants chez le fournisseur, nous allons porter le nombre de composés à sélectionner à 37 000.

Nous avons, pour ce travail, utilisé un filtre destiné à supprimer les composés potentiellement mutagènes. Ce filtre, disponible sous le logiciel MOE [5], est basé sur des toxicophores [6]. Les auteurs ont testé ce filtre sur un ensemble de composés avec une erreur de classification de 15 %, ce qui correspond à l'erreur de reproductibilité des tests Ames entre différents laboratoires.

Une fois les composés potentiellement mutagènes supprimés, la base ChemDiv est insérée dans *ScreeningAssistant*. Le nombre de structures à ce stade (c.-à-d. sans composés mutagènes et sans doublons) est de 568 119.

Pour la sélection des composés par diversité, nous n'avons considéré que les composés passant notre filtre « drug-like » CFMS.

Nous avons choisi de conserver le principe de l'algorithme de diversité présenté dans la partie traitant de la sélection de composés pour le docking. Cependant de profondes modifications lui ont été apportées, principalement pour permettre de sélectionner un ensemble par diversité qui complète au mieux la diversité d'une base existante.

Nous appellerons base *A* la base que nous souhaitons compléter et base *B* celle à partir de laquelle nous allons choisir des composés (dans notre cas ChemDiv). Nous traiterons les composés par familles de frameworks. On complétera tout d'abord les frameworks de *A*, puis on ajoutera à *A* de nouveaux frameworks. L'algorithme fonctionnera de la manière suivante :

1. Pour chaque framework de *B* qui existent également dans *A*, tant que le nombre de composés à ajouter n'est pas atteint, on ajoute le composé du framework de *B* qui complète au mieux les composés déjà présents dans *A* (en termes de diversité par fingerprints). Le nombre de composés à ajouter se calcule de la manière suivante :

$$n_{\text{composés à sélectionner}} = n_{\text{composés du framework de B}} \times \frac{n_{\text{composés à sélectionner au total}}}{\text{n}_{\text{composés de B}}} \qquad \textbf{(Équation 5)}$$

2. Pour les frameworks de *B* qui n'existent pas dans *A*. On procède de la même manière que l'étape précédente, mais le nombre de composés à sélectionner se calcule de la manière suivante :

$$n_{\text{composés à sélectionner}} = n_{\text{composés du framework de B}} \times \frac{n_{\text{composés à sélectionner au total}} - n_{\text{composés sélectionnés à l'étape 1}}}{\text{n}_{\text{composés de B}}} \qquad \textbf{(Équation 6)}$$

Le seul paramètre à fournir à l'algorithme est le nombre de composés à sélectionner. Du fait du fonctionnement même de l'algorithme, un ensemble est sélectionné d'un ordre de grandeur comparable à la taille demandée. Dans le cas présent, nous souhaitons sélectionner 37 000 composés. Après plusieurs essais nous sommes arrivés une sélection de 37 060 composés. En ne comptant que le nombre de composés uniques de l'ancienne base de la société pharmaceutique, cela donne au final une base de 101 146 composés.

Nous avons vérifié la pertinence de notre sélection en étudiant la diversité de l'ensemble sélectionné. Nous avons utilisé les deux descripteurs de notre algorithme, à savoir les frameworks et les fingerprints SSKey-3DS. Etant donné que cette sélection a été la première réalisée avec cet algorithme, cela a également été un moyen d'évaluer la pertinence de ce dernier.

	Base Hybrigenics (59 829)	Base Hybrigenics + 37 060 ChemDiv	ChemDiv (493 303)
Clusters	3 890	5 107	5 585
Frameworks	6 105	23 293	21 682
Composés « drug-like » (CFMS)	93 %	96 %	87 %

Tableau 15. Analyse des sélections réalisées en considérant uniquement les composés « drug-like » et non mutagènes. Les valeurs de diversité correspondent au nombre de clusters trouvés pour l'ensemble.

Il apparaît clairement d'après le Tableau 15 que la sélection des 37 060 composés de ChemDiv remplit pleinement les objectifs fixés en termes de pourcentage de composés « drug-like » et de diversité par rapport à la base ChemDiv. En effet, alors que les composés « drug-like » de la société pharmaceutique représentent en nombre 19 % des composés « drug-like » de ChemDiv, la base de la société pharmaceutique couvre 91 % de la diversité de ChemDiv et 107 % des frameworks de ChemDiv (tous les frameworks de ChemDiv ont été sélectionnés, et d'autres étaient déjà présent dans la base de la société pharmaceutique). Au vu de ces résultats, l'objectif d'arriver à une base comparable en terme de diversité à la base

ChemDiv est atteint. Le pourcentage de composés « drug-like » de la base de la société pharmaceutique finale est également très bon.

II. Sélection par diversité de composés déjà mis en plaques

La sélection de plaques est un problème qui relève de l'optimisation. Nous avons choisi d'étudier l'efficacité de méthodes d'optimisation sur cette problématique suivant des critères de diversité. Nous allons tout d'abord traiter des méthodes d'optimisations naturelles, puis nous exposerons l'implémentation des méthodes utilisées pour la sélection, avant de présenter les résultats des différentes méthodes.

A. Méthodes d'optimisation naturelles

L'optimisation est un processus consistant à rendre quelque chose le meilleur possible. En science, on cherchera à trouver la meilleure solution possible à un problème donné. L'exemple de problème le plus fréquemment employé pour tester des algorithmes d'optimisation est le problème du voyageur de commerce. Il s'agit de trouver le trajet le plus court possible que pourrait emprunter un voyageur de commerce, sachant qu'il doit passer une seule fois dans plusieurs villes données. C'est Euler qui le premier introduisit un problème de ce type en 1759. Le problème d'Euler consistait à faire parcourir à un cavalier une seule fois toutes les cases d'un échiquier. Ce n'est cependant qu'aux alentours de 1931 que ce problème fit son apparition en mathématiques. Un algorithme d'optimisation doit répondre à deux exigences contradictoires, à savoir trouver une solution qui soit la meilleure possible, et trouver cette solution le plus rapidement possible. Le problème est qu'il existe souvent un grand nombre de solutions possibles pour un problème donné. Ainsi si l'on prend notre exemple du voyageur de commerce, il y a (N-1)! possibilités de parcourir les villes. Pour 30 villes cela fait 8.8×10^{30} solutions possibles. Si l'on suppose que l'on peut évaluer un million de solutions par seconde grâce à un ordinateur, il faudrait 8.8×10^{24} secondes pour résoudre notre problème. A titre de comparaison, l'âge de la terre est de 1.4×10^{17} secondes. Cet exemple montre qu'il est généralement impossible d'évaluer toutes les solutions d'un problème donné.

Les méthodes d'optimisation vont donc devoir, non pas parcourir toutes les solutions possibles, mais parcourir l'espace des solutions afin d'arriver, sinon à la solution idéale, au moins à une solution très proche de celle-ci. Le point de départ dans l'espace des solutions est en principe une ou des solutions aléatoires. Les différences entre les méthodes résident dans la manière de parcourir l'espace des solutions afin de converger vers la ou les meilleures. Dans ce domaine, la nature est une source d'inspiration très riche. Ainsi, toutes les techniques que nous allons présenter sont basées sur des phénomènes naturels.

Avant de présenter quelques unes de ces méthodes, il est intéressant de s'attarder sur le théorème « No free lunch » (NFL). Introduit en 1995 par Wolpert et Macready, ce théorème affirme que les performances moyennes de toutes les méthodes d'optimisation (y compris une recherche aléatoire) sur tous les problèmes sont identiques [7, 8]. Ce théorème a été le point de départ d'une grande controverse dans le domaine de l'optimisation combinatoire, certains allant même jusqu'à utiliser NFL pour réfuter la théorie de l'évolution de Darwin et favoriser celle du « Dessein intelligent » [9, 10, 11]. En se focalisant sur l'aspect purement informatique, il convient de reformuler NFL. Ce dernier affirme comme nous l'avons dit qu'il n'existe pas de méthode d'optimisation « miracle », qui pourrait résoudre tous les problèmes. Cela veut dire que chaque méthode a ses points forts et ses points faibles, et qu'il faut plus se focaliser sur le domaine d'application de la méthode qu'essayer de trouver une méthode capable de résoudre tous les types de problèmes. Le NFL n'est donc pas en contradiction avec le fait que les méthodes que nous allons présenter se révèlent être très efficaces sur certains problèmes donnés.

1. Algorithmes génétiques

Les algorithmes génétiques sont basés sur le principe de la sélection naturelle, développé par Charles Darwin en 1838. Ils ont été étudiés pour la première fois par John Holland dans le milieu des années 70 [12]. Les solutions du problème sont représentées sous forme de chromosomes. Les gènes de ces chromosomes correspondent aux variables du problème. Les valeurs de ces variables sont appelées allèles. L'algorithme s'exécute de la manière suivante :

1- Un nombre prédéfini de chromosomes sont générés avec des allèles aléatoires. Cet ensemble de chromosomes constitue la première génération. La qualité de la solution est estimée grâce à un score, dépendant du problème traité.

2- La génération suivante est générée. Dans un premier temps un certain pourcentage des chromosomes est conservé. Pour chaque nouveau chromosome généré, deux parents sont choisis parmi les chromosomes conservés. Un crossover est alors effectué sur les deux parents pour donner deux chromosomes enfants. On procède ensuite à une étape de mutation. Le nombre de gènes mutés est défini à partir d'un taux de mutation prédéfini. Les positions des gènes mutés sont quand à elle définies aléatoirement.

3- L'étape précédente est répétée jusqu'à la fin de l'évolution. Plusieurs critères peuvent être utilisés pour cela :

   - Le nombre de générations prédéfini a été atteint.
   - Le score maximal est stable sur un grand nombre de générations.
   - Le score prédéfini a été atteint.

Nous allons maintenant nous attarder sur certains composants des algorithmes génétiques.

a. La sélection des parents

La sélection des parents consiste à choisir, parmi les chromosomes conservés lors de l'étape de sélection naturelle, les deux qui vont être utilisés pour donner naissance à deux nouveaux chromosomes. Il existe de nombreuses méthodes pour sélectionner les parents [13]. Les plus connus sont les suivantes :

- *Les chromosomes sont choisis deux à deux par ordre de score*. Les deux chromosomes avec les meilleurs scores sont choisis en premier, puis les deux chromosomes suivants, et ainsi de suite. Cette méthode est très simple, mais ne traduit pas la complexité de ce phénomène dans la nature.
- *Les chromosomes sont choisis aléatoirement*. A l'inverse de la méthode précédente qui ne laisse aucune part au hasard, dans celle-ci les chromosomes sont choisis de manière

totalement aléatoire. Les chromosomes ayant les meilleurs scores ne sont pas privilégiés dans le choix, ce qui une fois encore n'est pas représentatif du mécanisme naturel.

- *On choisi les chromosomes aléatoirement avec pondération*. La probabilité qu'a un chromosome d'être choisi dépend de son score. Ce type de méthode combine les avantages des deux précédentes, en faisant intervenir un choix aléatoire tout en privilégiant les chromosomes avec les scores les plus élevés. On assimile ce choix à une roue de la fortune, la taille des segments de la roue étant dépendante de la valeur du score. On distingue deux sous-méthodes pour donner un poids aux chromosomes lors de la sélection :
    - *La pondération par rang*. Les chromosomes sont classés en fonction de leurs scores. La probabilité de sélection des chromosomes est pondérée par leur position dans le classement, le premier étant celui qui est le plus favorisé.
    - *La pondération par score*. Dans ce cas ce sont les scores et non plus les classements qui sont pris en compte pour la pondération. Avec cette méthode les écarts entre les valeurs des scores entrent en jeux pour la sélection.
- *Les chromosomes sont choisis par des tournois*. Pour ce type de sélection, pour chaque sélection d'un parent, deux chromosomes (ou plus) sont choisis parmi ceux ayant passé la sélection naturelle. Le meilleur est gardé comme parent. L'opération est répétée pour chaque parent à sélectionner.

b. La reproduction

Lors de l'étape de reproduction les parents mélangent leurs gènes afin de donner naissance aux enfants. Cette étape est appelée crossover. C'est une étape importante du fonctionnement des algorithmes génétiques, et différentes méthodes existent [14]. La méthode la plus simple est le crossover à un point. Dans ce cas les parents échangent une partie de leurs gènes en fonction d'un point déterminé aléatoirement, comme le montre la Figure 29.

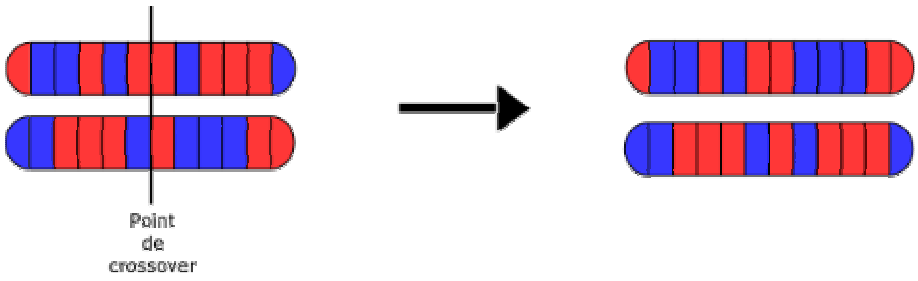


Figure 29. Crossover un point : les chromosomes échangent les gènes qui se trouvent à droite d'un point de crossover choisi aléatoirement.

L'inconvénient du crossover un point est que les gènes qui sont très éloignés dans un même chromosome ont plus de chances de se trouver séparés que deux gènes qui sont très proches.

Ce problème peut être limité par l'utilisation du crossover deux points qui fonctionne suivant le même principe que le crossover un point (Figure 30). On notera l'existence de crossover multi points qui utilisent encore plus de points de crossover.

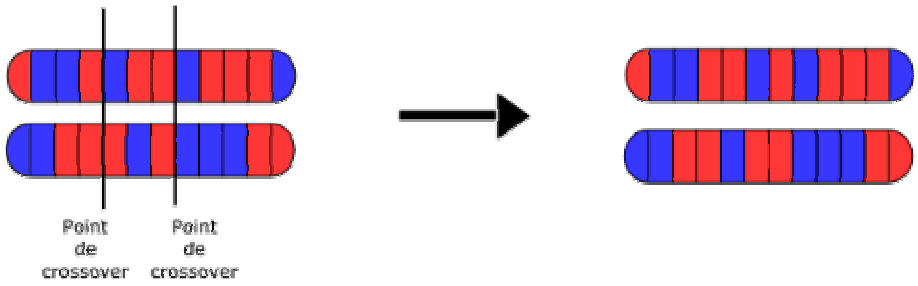


Figure 30. Crossover deux points : deux points sont définis aléatoirement et les gènes entre ces deux points sont échangés entre les deux chromosomes.

Le crossover uniforme fonctionne quand à lui sans utiliser de point de crossover. Dans ce cas la génération des enfants se fait en passant aléatoirement les gènes d'un parent ou de l'autre (Figure 31).

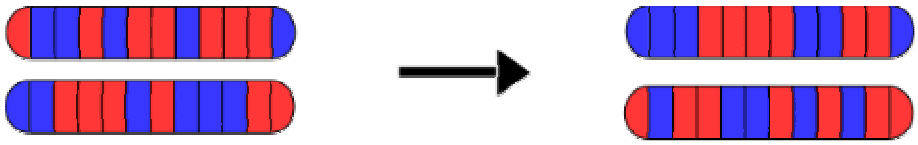


Figure 31. Crossover uniforme : un certain nombre de gènes choisis aléatoirement sont échangés.

c. Mutation

Le principe de la mutation est de modifier de manière aléatoire un certain nombre de gènes dans la population. Le but est d'éviter une convergence prématurée de l'algorithme, lui permettant ainsi de sortir des minimums locaux. Les meilleures solutions sont généralement conservées intactes, et ne sont donc pas soumises à la mutation. Le nombre de gènes mutés dépend du taux de mutation prédéfini. Les positions des gènes mutés sont choisies aléatoirement (Figure 32).

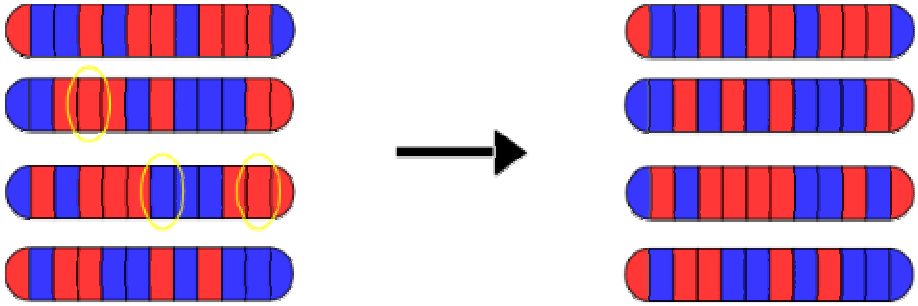


Figure 32. Exemple d'une mutation de 10 % sur une population. Le premier chromosome est le meilleur, et n'est pas modifié. Trois gènes sont donc modifiés parmi les trois gènes restants.

d. Fonction de score

La fonction de score est l'élément qui permet aux algorithmes génétiques de prendre en compte un problème donné. La conception d'une fonction de score permettant une évaluation pertinante des solutions d'un problème sous forme chiffrée est indispensable à la réussite du calcul d'optimisation.

2. Recuit Simulé

Le recuit simulé a été proposé comme technique d'optimisation globale par Kirkpatrick et son équipe en 1983 [15]. Cette méthode se base sur le phénomène de recuit des matériaux : le fait de chauffer un matériau, puis de baisser lentement sa température permet d'obtenir une structure cristalline de meilleure qualité. La première étape de l'algorithme est la génération d'une solution aléatoire. Comme pour les algorithmes génétiques, un score est calculé pour chacune des solutions générées par l'algorithme. C'est ce score qui sera optimisé par l'algorithme. Une fois la solution aléatoire générée, on procède au processus de chauffage, cela va correspondre dans notre algorithme à une modification aléatoire des variables. C'est le critère d'acceptation qui va permettre de décider si la nouvelle solution générée va remplacer ou non l'ancienne. Le critère d'acceptation le plus fréquemment utilisé est celui de Metropolis :

$$r = e^{\frac{score_{ancien} - score_{nouveau}}{T}} \qquad \textbf{(Équation 7)}$$

r est comparé à un nombre aléatoire compris entre 0 et 1, et T est la pseudo température du système. Si la condition est validée, alors la nouvelle solution remplace l'ancienne. Dans le cas contraire, l'ancienne solution est conservée. Plus la température sera élevée, plus une solution avec un mauvais score aura de chance d'être acceptée, et à l'inverse, à mesure que la température du système diminue, un score doit être de plus en plus satisfaisant pour que la solution soit acceptée.

Différents schéma d'évolution de température peuvent être utilisés, tout en gardant à l'esprit que c'est le fait de diminuer lentement la température qui permettra de maximiser les chances de s'approcher du minimum global. Nous pouvons citer trois fonctions de refroidissement [13] :

- refroidissement linéaire : $T_n = T_0 - n\frac{T_0 - T_n}{N}$ **(Équation 8)**
- refroidissement géométrique : $T_n = 0{,}99 \times T_{n-1}$ **(Équation 9)**

- refroidissement Hayjek optimal : $T_n = \frac{c}{\log(1+n)}$ **(Équation 10)**

Comme pour les algorithmes génétiques, l'arrêt de l'algorithme peut se faire soit lorsque l'on considère que la convergence est atteinte, soit au bout d'un nombre d'itérations prédéterminées.

3. Optimisation par essaims particulaires

Cette technique d'optimisation a été introduite par Kennedy et Eberhart en 1995 [16]. Elle trouve ses fondements dans le comportement des animaux se déplaçant en essaims, tels que les insectes. Ce sont en fait les déplacements des individus qui sont modélisés. Cette méthode est principalement destinée à optimiser des variables continues. L'algorithme a été mis au point à partir de travaux précédents portant sur la simulation numérique de vol d'oiseaux ou de mouvements de bancs de poissons [17, 18]. De plus la base sociologique de cet algorithme est très forte. Ainsi, le sociobiologiste Wilson [19] suggère que le partage des informations entre individus, pour trouver de la nourriture par exemple, donne un avantage au groupe dans l'évolution. On peut séparer ce phénomène en deux niveaux. Le niveau supérieur permet notamment la résolution de problèmes, et le niveau inférieur correspond au comportement des individus [20], comportement qui peut se résumer en trois étapes : l'évaluation, la comparaison, et l'imitation.

Dans le cas des essaims particulaires, deux paramètres sont utilisés pour déterminer la vitesse des particules : le meilleur score rencontré par l'individu, p_{local}, et le meilleur score qu'une particule de l'essaim a trouvé, p_{global}. On peut considérer p_{global} comme étant la mémoire de l'individu. L'influence de p_{global} sur le mouvement de l'individu peut s'assimiler à la nostalgie : chaque individu veut retourner vers le moment le plus agréable de son passé. Pour sa part, g_{best} s'assimile plutôt à une norme du groupe, que chaque individu cherche à atteindre.

A chaque itération, pour chacune des particules (donc des solutions) on calcule l'équation de mouvement suivante [21] :

- La vitesse (ou déplacement) :

$$V_{k+1} = a \times V_k + b_1 \times (X_{local} - X_k) + b_2 \times (X_{global} - X_k)$$ (**Équation 11**)

- La valeur de chaque variable : $X_{k+1} = X_k + V_{k+1}$ (**Équation 12**)

a est la constante d'inertie, *b1* et *b2* deux variables aléatoires positives.

4. Colonies de fourmis

Les algorithmes de colonies de fourmis ont été introduits par Marco Dorigo [22] lors de sa thèse. Ils imitent le comportement des fourmis pour trouver de la nourriture. Ils peuvent sembler similaires à l'optimisation par essaims particulaires, mais sont cependant bien différents, et utilise notamment les notions de chemins et de phéromones. En effet, les fourmis errent pour trouver de la nourriture. Quand elles en trouvent, elles marquent le chemin entre la colonie et la nourriture. D'autres fourmis vont être attirées par ces phéromones, et vont renforcer le marquage du chemin. Parmi plusieurs chemins menant à de la nourriture, les plus courts auront une concentration en phéromones plus forte et seront donc plus marqués. Etant donné qu'ils sont basés sur une notion de chemins, ces algorithmes trouvent leurs applications principales dans les graphes.

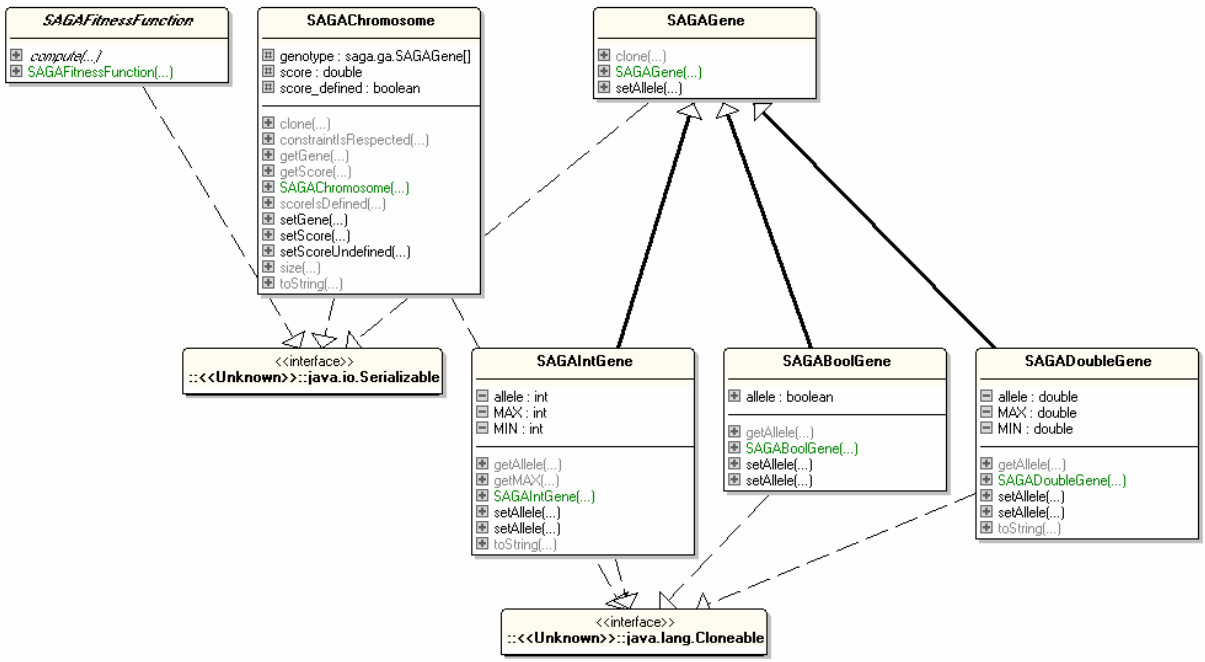
B. Implémentations des méthodes d'optimisations

Nous avons implémenté et testé les algorithmes génétiques et le recuit simulés. Les résultats obtenus par ces deux méthodes ont été satisfaisant, et nous n'avons pas eu besoin de tester les deux autres méthodes envisagées.

1. Algorithmes génétiques

Nous avons développé une bibliothèque d'optimisation. Cette bibliothèque devait dans un premier temps être utilisable pour la sélection de plaques. Cependant, elle devait également être suffisamment générale pour être utilisable dans d'éventuels projets futurs de chemoinformatique comme notamment le QSAR (sélection de descripteurs, optimisation de paramètres de réseaux de neurones ou de machine à support de vecteurs) et la conception *de-*

novo. Il a dans un premier temps été envisagé d'utiliser une bibliothèque d'algorithmes génétiques existante. Cette bibliothèque devait être en Java afin de pouvoir être intégrée facilement à d'autres logiciels du laboratoire, principalement *ScreeningAssistant*. Notre choix s'était porté sur la librairie JGAP [23]. Cependant les résultats de nos tests ont étés décevants sur des problèmes d'optimisation simples, sans que la cause de ces mauvais résultats puisse être trouvée. Nous avons donc décidé de développer notre propre algorithme génétique, afin de disposer d'une bibliothèque dont nous connaîtrions parfaitement le code. Les résultats avec cette nouvelle bibliothèque se sont montrés très satisfaisants, et ce code a été adopté pour la suite de nos travaux. Le diagramme UML de cette bibliothèque est présenté Figure 33.



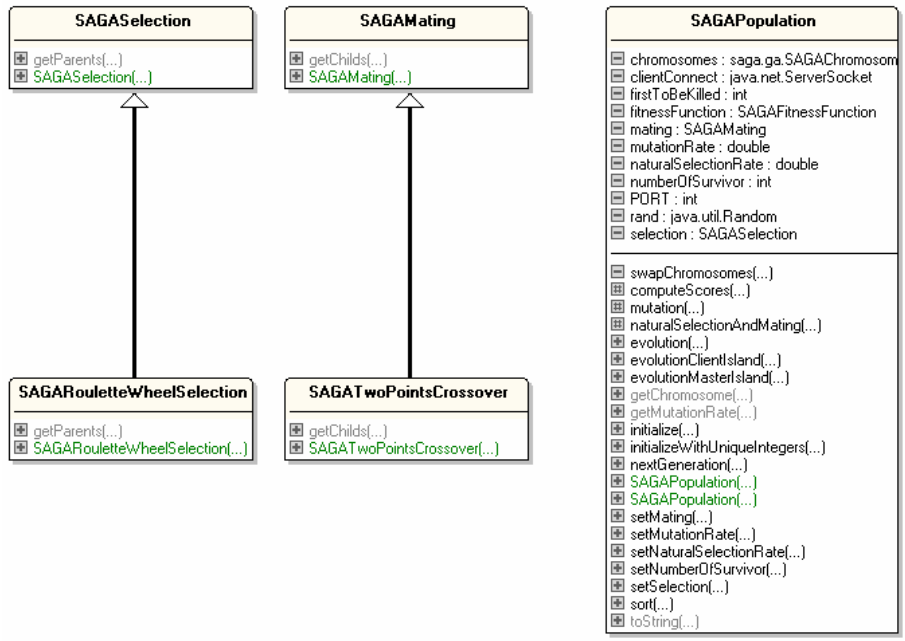


Figure 33. Diagramme UML de la bibliothèque d'algorithmes génétiques.

Les noms de toutes les classes débutent par *SA*, pour *ScreeningAssistant*, et *GA*, pour Genetic Algorithms. La classe *SAGAPopulation* est la classe principale de la bibliothèque, et permet de contrôler l'évolution de l'algorithme. Elle stocke un tableau de *SAGAChromosome*, qui représente donc la population à un moment donné. *SAGAChromosome* stocke quand à elle un tableau de *SAGAGene*, qui correspond au génotype. Trois classes héritent de la classe abstraite *SAGAGene*. Ces classes permettent de manipuler des données de type booléen, entier, ou double (réel).

La classe abstraite *SAGAFitnessFunction* définit une structure générale pour calculer le score d'une solution. Il faudra donc, pour chaque problème, implémenter une classe enfant de *SAGAFitnessFunction* qui permet de calculer le score pour une solution (c.a.d. un chromosome) du problème donné. Cette classe sera passée au constructeur de *SAGAPopulation*.

Les classes abstraites *SAGASelection* et *SAGAMating* permettent d'implémenter respectivement la sélection des parents, et la reproduction. Pour ces deux étapes, nous avons choisi d'implémenter des méthodes couramment utilisées. Ainsi la sélection des parents se fait par la méthode de la roue de la fortune avec une pondération basée sur les scores, et la reproduction par un crossover deux points.

Nous avons décrit les différents éléments constitutifs de notre algorithme génétique. Il y a cependant un facteur à prendre en compte, et que nous n'avons pas encore abordé : les contraintes sur les solutions. Nous avons pour l'instant comme unique contrainte dans notre système les intervalles qui sont définis pour les valeurs des gènes de types entiers et réels. Certains problèmes vont cependant nécessiter des solutions un peu plus sophistiquées. Supposons que notre algorithme génétique serve à sélectionner des descripteurs pour un modèle QSAR, ou bien des molécules pour la sélection d'un ensemble de composés divers. Le problème pourra être codé en attribuant un identificateur de type entier à chacun des éléments, les descripteurs dans un cas, et les molécules dans l'autre (on aura alors dans le premier cas un modèle QSAR avec un nombre de descripteurs prédéfini). Chacun des éléments ne devra être présent qu'une seule fois dans une même solution, car il est évidemment impensable d'avoir un modèle QSAR utilisant deux fois le même descripteur, ou bien un ensemble de molécules divers dans lequel on retrouve des doublons. Nous voyons donc que notre système doit permettre d'établir des contraintes sur les solutions. Dans nos exemples la contrainte est l'unicité des allèles. Il existe plusieurs manières de gérer ces contraintes. Ainsi JGAP [24] utilise la notion de supergène. Il s'agit d'un gène définissant une contrainte et regroupant tous les gènes concernés par la contrainte. Cette méthode a l'inconvénient de compliquer la structure des données. Nous avons pour notre part implémenté la gestion de la contrainte de manière un peu différente. Cette dernière s'effectue en étendant la classe Chromosome et en redéfinissant la méthode *constraintIsRespected*. Cette méthode sera appelée après chaque modification de chromosome c'est à dire pendant les étapes d'initialisation, de crossover et de mutation. Si le chromosome généré pendant une étape donnée ne respecte pas la contrainte, une autre tentative est effectuée.

On notera enfin que les performances des algorithmes génétiques peuvent être améliorées significativement par une implémentation parallèle. Dans ce cas ont utilise des sous-populations qui évoluent de manière indépendante, mais en échangeant des individus entre elles occasionnellement. Plusieurs études montrent même une augmentation superlinéaire de la vitesse d'exécution des algorithmes génétiques parallèles. Ces résultats nécessitent quelques précisions. En effet si l'on exécute sur un même processeur tous les processus d'un programme parallèle, le temps d'exécution ne peut pas être inférieur au temps d'exécution de la même tâche par un programme en série. En fait, la superlinéarité de la vitesse d'exécution peut généralement s'expliquer par le fait que les algorithmes génétiques exécutent moins de travail et ont une pression de sélection plus importante [25].

La parallélisation des algorithmes génétiques est donc très intéressante. Même si l'utilisation d'algorithmes génétiques parallèles sort du cadre de ce travail, certaines implémentations ont été réalisées afin de permettre une éventuelle parallélisation. Ainsi les classes *SAGAChromosome*, *SAGAGene* et *SAGAFitnessFunction* implémentent l'interface *Serializable*, et elles sont donc transmissibles par communication réseau.

2. Recuit Simulé

En nous basant sur les classes définies pour les algorithmes génétiques, nous avons développé un algorithme de recuit simulé (Figure 34).

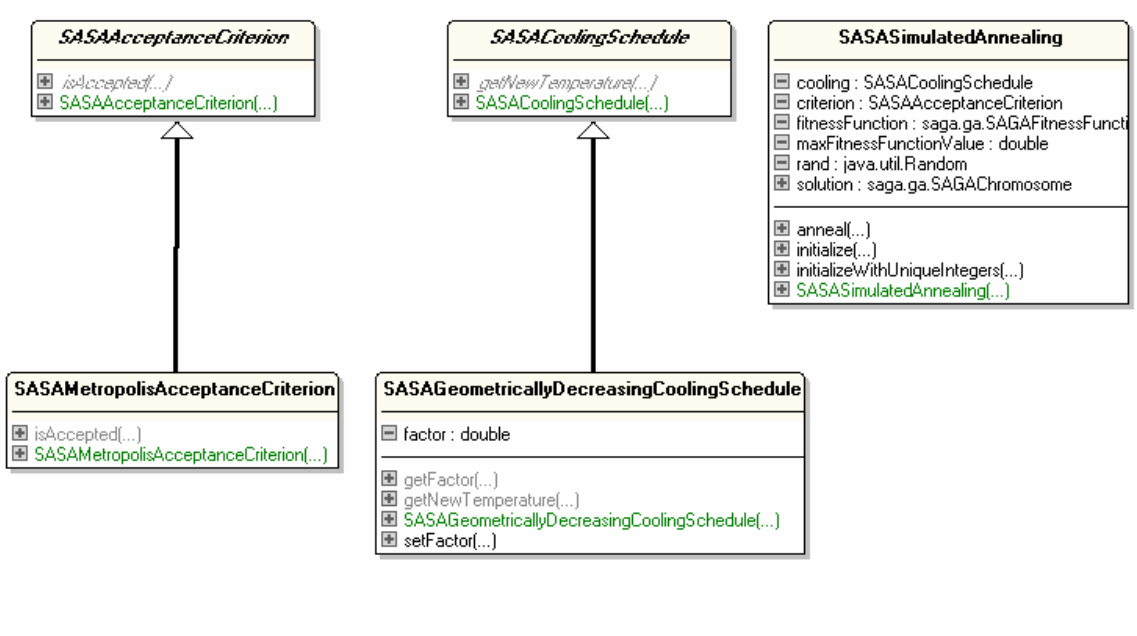


Figure 34. Diagramme UML de la bibliothèque de recuit simulé.

La classe principale est ici *SASASimulatedAnnealing*, qui prend en charge le déroulement de l'expérience de recuit simulé. Nous avons conservé la classe *SAGAChromosomes* pour gérer les solutions. Bien que la notion de chromosomes n'entre pas en jeu dans les algorithmes de recuit simulé, notre classe est bien adaptée pour gérer la solution qui sera optimisée par le recuit simulé. Comme pour les algorithmes génétiques, les solutions pourront être de types entier, booléen ou double, et des contraintes pourront être

appliquées. La création d'une expérience de recuit simulé nécessite de définir le schéma de refroidissement (classe abstraite *SASACoolingSchedule*) et le critère d'acceptation (classe abstraite *SASAAcceptanceCriterion*). Nous avons implémenté le critère d'acceptation de Metropolis, et un schéma de refroidissement géométrique.

C. La sélection de plaques

Alors que la sélection de composés par diversité a fait l'objet de nombreuses recherches, la sélection de plaques par diversité n'a donné lieu, à notre connaissance, qu'à une seule publication [26]. Dans cette étude, les auteurs proposent une méthode générale utilisant le recuit simulé pour la sélection de groupes de composés. Une application directe de cette méthode est la sélection de plaques.

La sélection de plaques peut être utilisée pour créer ou agrandir une chimiothèque à partir de composés déjà mis en plaques, ou bien encore pour choisir des plaques pour des tests à hauts débits. La taille de l'espace des solutions est :

$$N_S = \frac{n!}{(n-k)!k!} \qquad \textbf{(Équation 13)}$$

n étant le nombre total de plaques et *k* le nombre de plaques à sélectionner. Par rapport à une sélection par diversité classique, la sélection de plaques nécessite d'évaluer la diversité qu'apporte un groupe indissociable de composés (ceux de la plaque) à un autre groupe de composés (ceux déjà sélectionnés).

Nous avons utilisé pour cette étude 125 plaques contenant des produits du fournisseur InterBioScreen.

1. Critères de sélection par diversité

Quelque soit l'algorithme utilisé pour la sélection de plaques, il faut définir le type de diversité que l'on souhaite. Afin que notre comparaison ne soit pas dépendante du type de

diversité utilisé, nous avons choisi deux types de diversité différents, déjà présentés dans ce manuscrit. Le premier consiste à maximiser le nombre de frameworks. Le deuxième, consiste à maximiser le nombre de clusters obtenus à partir des fingerprints SSKey-3DS avec l'algorithme SCA.

De plus, des filtres basés sur d'autres critères peuvent être utilisés. On peut ainsi privilégier les composés « drug-like » ou encore ceux ayant des propriétés physicochimiques ou des sous-structures voulues.

2. Algorithmes

Nous avons comparé les performances de trois algorithmes différents pour traiter le problème de la sélection de plaques :

- Classement : cet algorithme classe les plaques en fonction de la diversité qu'elles apportent à la sélection. Au départ la sélection est vide. On cherche la plaque qui a les composés les plus divers. On ajoute cette plaque à la sélection. On cherche ensuite la plaque qui apporte le plus de diversité à la sélection et on l'ajoute. On répète cette dernière étape jusqu'à ce que l'on obtienne le nombre de plaques souhaité. Dans le cas de la diversité par fingerprint on utilise l'algorithme SCA pour évaluer la diversité et dans le cas de la diversité par frameworks on compte le nombre de frameworks.

- Algorithmes génétiques : nous utiliserons la bibliothèque d'algorithmes génétiques décrite précédemment. Elle utilise la sélection aléatoire avec pondération (roue de la fortune) et le crossover double point. Les deux chromosomes avec les meilleurs scores survivent dans la génération suivante.

- Recuit simulé : nous utiliserons également la bibliothèque de recuit simulé présentée précédemment, avec un schéma de refroidissement géométrique, et le critère d'acceptation de Metropolis.

3. Résultats

Afin d'étudier l'influence du nombre de composés sélectionnés sur les performances des algorithmes, nous sélectionnerons dans chaque cas 30, 60 et 90 plaques. Les données ont

été importées dans une base MySQL en utilisant une version spéciale de *ScreeningAssistant*, modifiée pour gérer les composés mis en plaques. L'algorithme est directement connecté à la base MySQL.

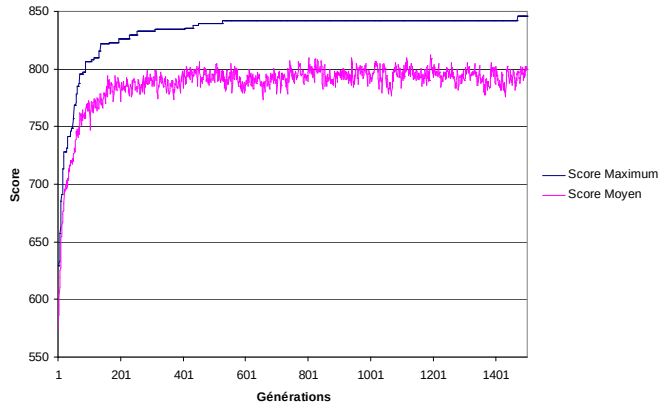
Le taux de sélection naturelle pour les algorithmes génétiques est de 50 % et le taux de mutation de 5 %. Entre 10 et 20 chromosomes sont utilisés en fonction des cas, et le nombre de générations est de 1500 pour la sélection de 30 plaques, et de 2000 pour les sélections de 60 et 90 plaques.

Pour le recuit simulé, la pseudo-température initiale est fixée à 800 et le nombre de pas à 2500.

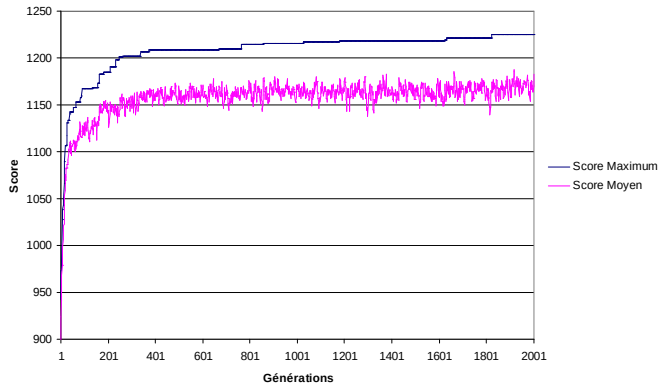
a. Frameworks

La comparaison de l'évolution des scores pour les algorithmes d'optimisation mimant un phénomène naturel fait apparaître que, si la qualité des solutions des premières itérations est meilleure pour les algorithmes génétiques, le recuit simulé atteint plus rapidement sa solution optimale (Figure 35).

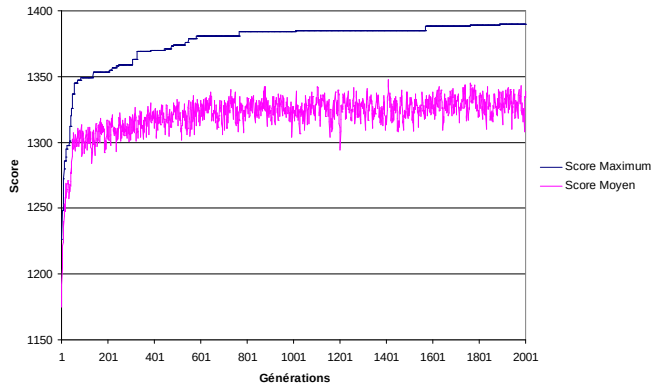
Algorithmes génétiques



30 plaques

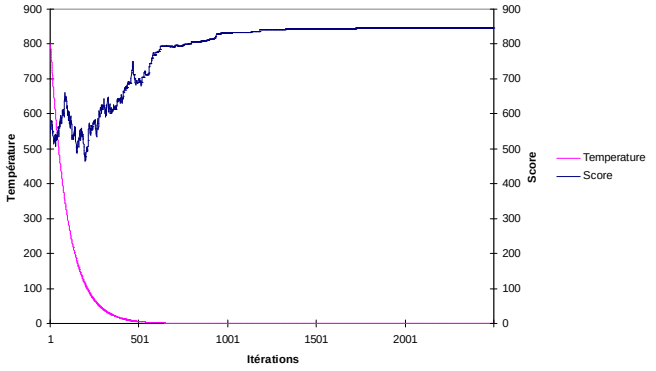


60 plaques

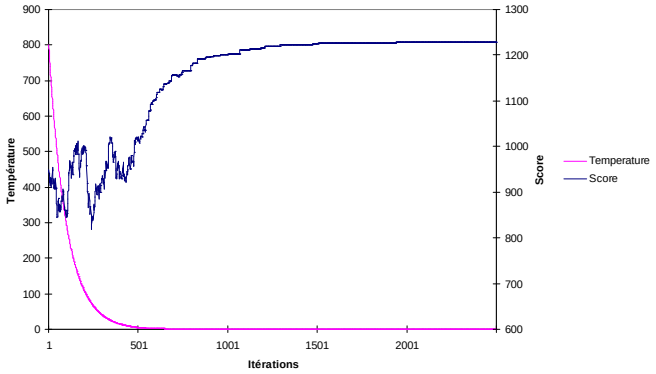


90 plaques

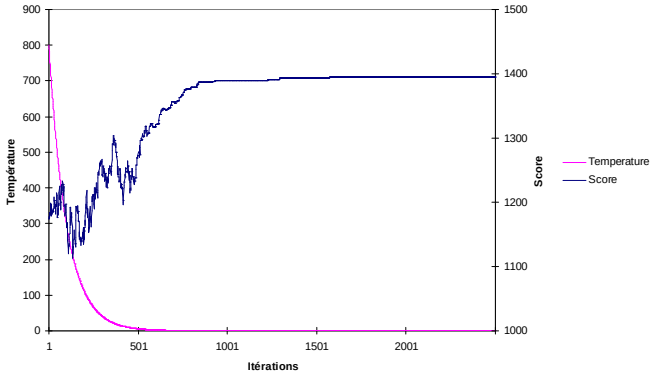
Recuit Simulé



30 plaques



60 plaques



90 plaques

Figure 35. Evolutions du nombre de frameworks pour les algorithmes génétiques (colonne de gauche), et le recuit simulé (colonne de droite) pour des sélections de 30, 60 et 90 plaques.

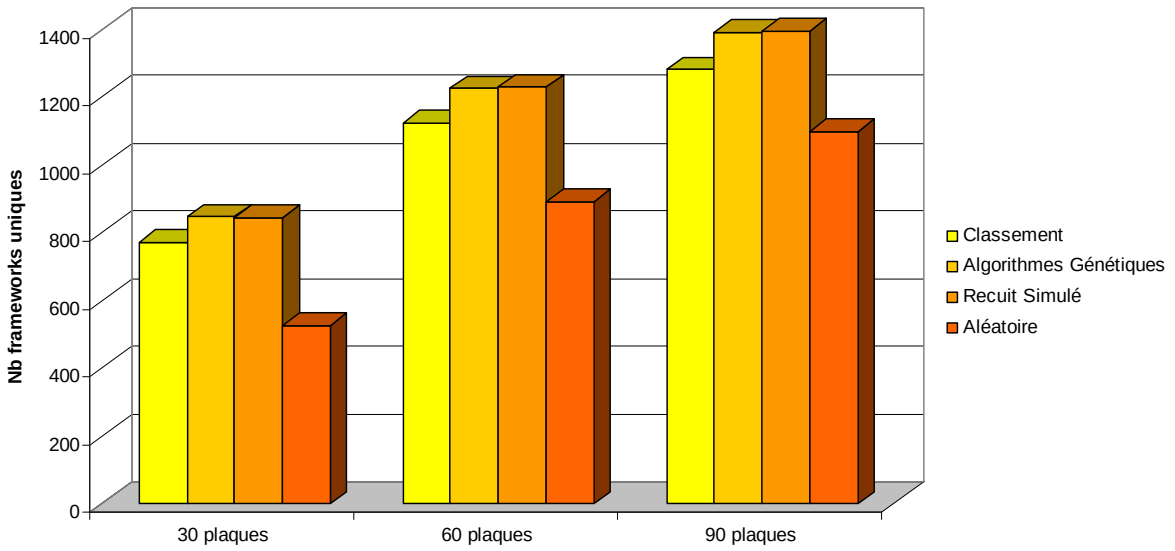


Figure 36. Comparatifs des trois méthodes utilisées pour la sélection de plaques en optimisant le nombre de frameworks. Les résultats des sélections aléatoires sont également présentés comme références.

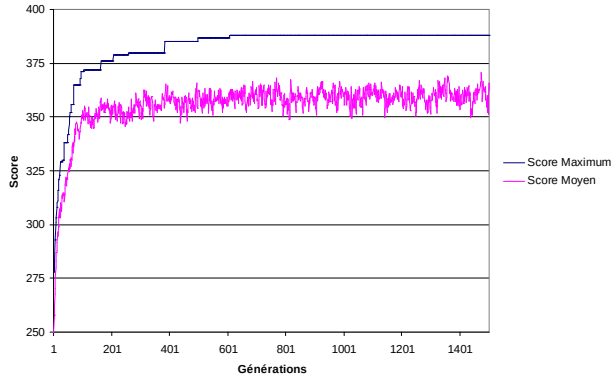
Le comparatif des trois algorithmes montre que tous donnent des résultats meilleurs qu'une sélection aléatoire (Figure 36). Cependant les algorithmes basés sur des méthodes d'optimisation naturelles donnent de meilleurs résultats qu'un simple classement. Il n'y a pas de différences notables entre les résultats des algorithmes génétiques et le recuit simulé. Le recuit simulé a cependant deux avantages par rapport aux algorithmes génétiques. Il est d'une part plus simple à paramétrer car les seuls critères à définir sont la température de départ et le nombre de pas. Ces paramètres sont simples à régler. D'autres parts, le nombre de pas nécessaires à la convergence est inférieur pour le recuit simulé.

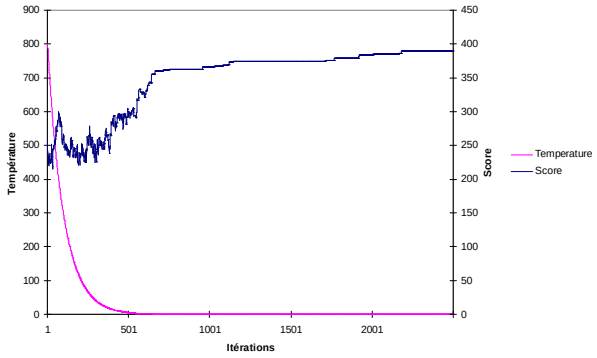
b. SSKey-3DS

Si compter le nombre de frameworks uniques pour un groupe de composés est rapide, compter le nombre de clusters est plus lent. Le temps d'exécution est donc un paramètre à

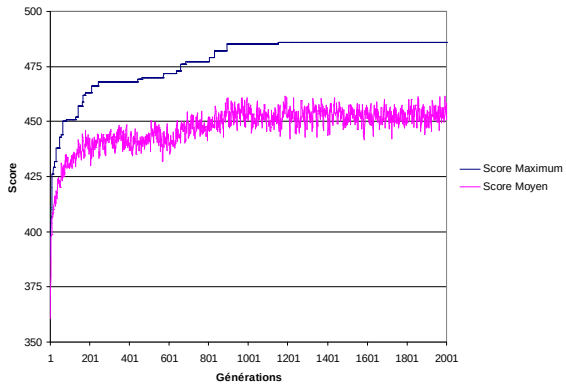
prendre en compte. Ce dernier est en faveur du recuit simulé. En effet lors de l'exécution de cet algorithme, le nombre de clusters est calculé une fois par pas. En revanche, pour les algorithmes génétiques, le nombre de cluster est calculé plusieurs fois par génération (autant de fois qu'il y a de nouveaux chromosomes à cette génération).

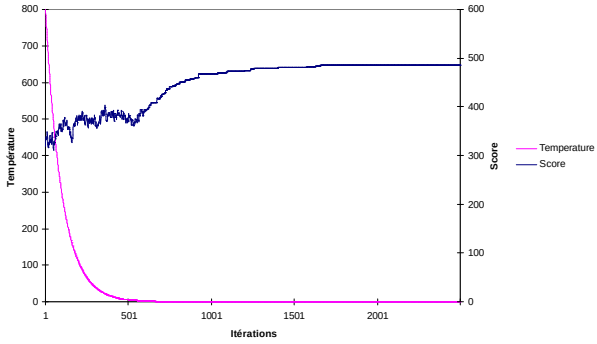
Algorithmes génétiques **Recuit Simulé**



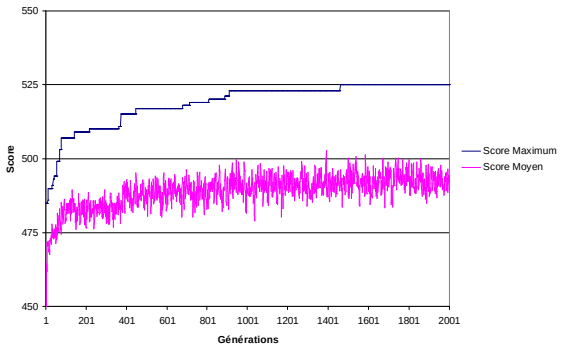


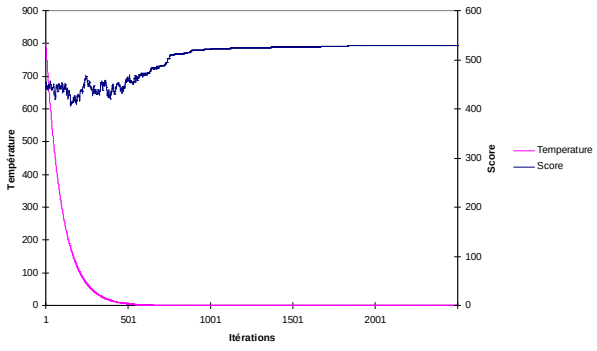
30 plaques 30 plaques





60 plaques 60 plaques





90 plaques 90 plaques

Figure 37. Evolutions du nombre de clusters pour les algorithmes génétiques (colonne de gauche), et le recuit simulé (colonne de droite) pour des sélections de 30, 60 et 90 plaques.

Bien qu'ayant des schémas d'évolution différents, les deux méthodes convergent approximativement au même nombre de pas vers leur solution maximale (Figure 37).

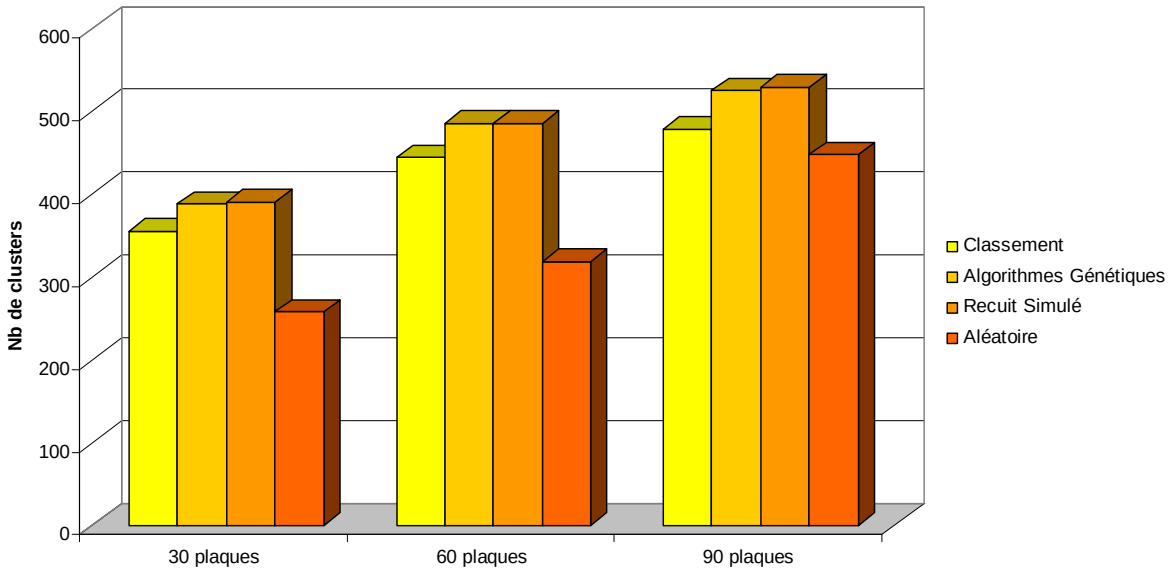


Figure 38. Comparatifs des trois méthodes utilisées pour la sélection de plaques en optimisant le nombre de clusters. Les résultats des sélections aléatoires sont également présentés comme références.

Les résultats obtenus en utilisant les SSKey3DS sont très comparables à ceux obtenus avec les frameworks. Les trois méthodes étudiées donnent de meilleurs résultats qu'une sélection aléatoire (Figure 38). Les méthodes d'optimisations basées sur des phénomènes naturels sont celles qui donnent les meilleurs résultats. Il n'y a pas de différences notables entre les résultats des algorithmes génétiques et ceux du recuit simulé.

4. Conclusion du comparatif de sélection de plaques

La sélection de plaque est un problème très particulier, et est donc très peu étudié dans la littérature. Une solution à base de recuit simulé a déjà été publiée. Nous avons développé une bibliothèque d'optimisation offrant la possibilité d'utiliser les algorithmes génétiques et le recuit simulé. Cette bibliothèque pourra également être utilisée pour d'autres types d'applications (par exemple la sélection de descripteurs pour le QSAR).

Les résultats de ce travail montrent que les algorithmes génétiques fonctionnent aussi bien que le recuit simulé pour ce type de problèmes, même si le recuit simulé a l'avantage du temps de calcul et de la facilité de réglage des paramètres. Nous avons utilisé des diversités basées sur les frameworks et les SSKey3DS pour notre étude. En pratique, nous utiliserons une diversité basée à la fois sur les frameworks et les SSKey3DS.

III. Sélection de composés à diversité cumulative pour la mise en plaques

La chimiothèque de l'ICOA regroupe toutes les structures synthétisées au sein de l'institut. Cette chimiothèque est intégrée dans la chimiothèque nationale du CNRS. Ce projet a pour principal objectif de valoriser les composés chimiques synthétisés par les différentes équipes de chimistes du CNRS. La mise en plaque des composés de l'ICOA facilitera la réalisation de tests biochimiques par les différents partenaires. De toute évidence, il est pertinent de concentrer la diversité sur un nombre le plus réduit possible de plaques. Pour l'ICOA, l'objectif est de disposer de 6 plaques représentatives de l'espace chimique couvert par sa chimiothèque.

Chacune des plaques comportera 80 produits. Le problème consiste à sélectionner des ensembles successifs de 80 produits qui ajoutent chacun un maximum de diversité par rapport à l'ensemble déjà sélectionné. Ainsi, on pourra choisir, pour des tests onéreux ou lents, de tester moins de 6 plaques tout en gardant une représentativité maximale de la chimiothèque de l'ICOA. Il existe diverses méthodes pour sélectionner un échantillon de molécules diverses, le plus connu étant l'algorithme Maxmin [27]. Nous avons également cherché à évaluer les performances de quelques algorithmes permettant de résoudre ce type de problèmes, sachant que nous souhaitons que la sélection par diversité prenne en compte d'une part la notion de frameworks, et d'autre part la notion de fingerprints.

Nous utiliserons comme descripteurs notre version modifiée des frameworks, ainsi que les fingerprints SSKey3DS. La diversité d'un ensemble sera évaluée en comptant le nombre de squelettes différents présents dans cet ensemble et le nombre de clusters générés par l'algorithme SCA avec les SSKey3DS.

Nous utiliserons trois familles d'algorithmes différents pour cette étude.

A. Algorithmes

1. Maxmin

Maxmin est l'algorithme de diversité le plus employé. Il fait partie de la famille des algorithmes de dissimilarité maximale. Les algorithmes de ce type fonctionnent de la manière suivante pour sélectionner *n* composés :

1. Initialisation de la sélection avec un composé choisi arbitrairement (souvent le premier) dans la base.
2. Calcul de la dissimilarité entre chaque composé restant dans la base et les composés de la sélection.
3. Sélectionne le composé de la base le plus dissimilaire à la sélection, et l'ajoute à la sélection.
4. Retour à l'étape 2 s'il y a moins de *n* composés dans la sélection.

Il y a plusieurs méthodes pour évaluer le composé le plus dissimilaire par rapport à la sélection. Dans le cas de Maxmin, pour chaque composé, la dissimilarité avec la sélection est donnée par la dissimilarité minimum entre ce composé et chaque composé de la sélection. Le composé le plus dissimilaire à la sélection est le composé pour lequel cette valeur de dissimilarité maximale est la plus grande.

Dans notre implémentation de l'algorithme Maxmin, avant chaque ajout d'un composé à la sélection, on distingue deux cas :

- soit il existe dans la base des composés dont les frameworks ne sont pas présents dans la sélection, et dans ce cas le composé à ajouter sera choisi par diversité uniquement parmi ceux-ci,
- soit il n'existe pas de composés dans la base dont les frameworks ne sont pas présents dans la sélection, et dans ce cas le composé à ajouter sera choisi par diversité parmi l'ensemble des composés de la base.

Ensuite, le choix du composé est réalisé comme décrit dans l'étape 3 de l'algorithme de diversité maximale. La diversité sera évaluée en utilisant les fingerprints SSKey-3DS et le coefficient de Tanimoto.

2. AddTheBest

Nous avons implémenté un algorithme qui sélectionne pour chaque nouvelle insertion, le composé qui apporte le plus de diversité à la sélection. Plus précisément l'algorithme fonctionne de la manière suivante :

1. La sélection est initialisée avec le premier composé de la base.
2. Pour chaque composé de la base on évalue la diversité de l'ensemble constitué de ce composé et de la sélection. Le composé apportant la plus grande diversité est ajouté à la sélection.
3. Retour à l'étape 2 s'il y a moins de *n* composés dans la sélection.

3. Algorithmes génétiques : traitement plaque par plaque et global

Nous avons cherché à évaluer les performances d'une méthode d'optimisation naturelles pour générer un ensemble divers de petite taille. Quelques tests rapides de comparaisons des algorithmes génétiques et du recuit simulé de type Monte-Carlo ont montré que les algorithmes génétiques étaient légèrement plus performants dans ce cas. Nous utiliserons donc ceux-ci pour ce problème.

Deux approches ont été considérées : d'une part le traitement successif plaque par plaque, d'autre part un traitement global du nombre de plaques souhaitées. Le traitement successif plaque par plaque a pour avantage de ne faire optimiser à l'algorithme que des solutions de 80 gènes, donc plus faciles à optimiser. Le revers de la médaille est que l'algorithme ne peut plus agir sur une plaque déjà créée. A l'inverse, une sélection globale permet à l'algorithme d'agir sur tous les composés à la fois. Par contre la taille des chromosomes est importante, et le problème à résoudre est donc plus complexe.

B. Résultats

Lors de la réalisation de ce travail, 2332 composés de la base ICOA étaient stockés en quantité supérieure à 10 mg. Parmi ces composés, 2310 sont détectés comme étant uniques, et parmi ces ceux-ci, 1671 sont « drug-like » d'après notre score CFMS. Ces 1671 composés constitueront l'ensemble de travail.

La diversité sera évaluée en tenant compte à la fois des fingerprints SSKey3DS et des frameworks. La diversité par fingerprints sera évaluée en comptant le nombre de clusters générés par l'algorithme SCA. Nous avons choisi de donner la même importance aux fingerprints et aux frameworks. Il y a 474 frameworks et 575 clusters dans l'ensemble de composés étudiés. Pour donner la même importance aux frameworks et aux clusters nous utiliserons le score suivant :

$$Score = 1{,}2 \times nb_{Frameworks} + nb_{Clusters} \quad \textbf{(Équation 14)}$$

Les ensembles générés par les différentes méthodes ont été comparés en utilisant ce score. Les clusters sont comptés 10 fois pour chaque mesure avec à chaque fois un ordre aléatoire des composés, car l'algorithme SCA est dépendant de l'ordre des composés. Les résultats obtenus sont présentés Figure 39.

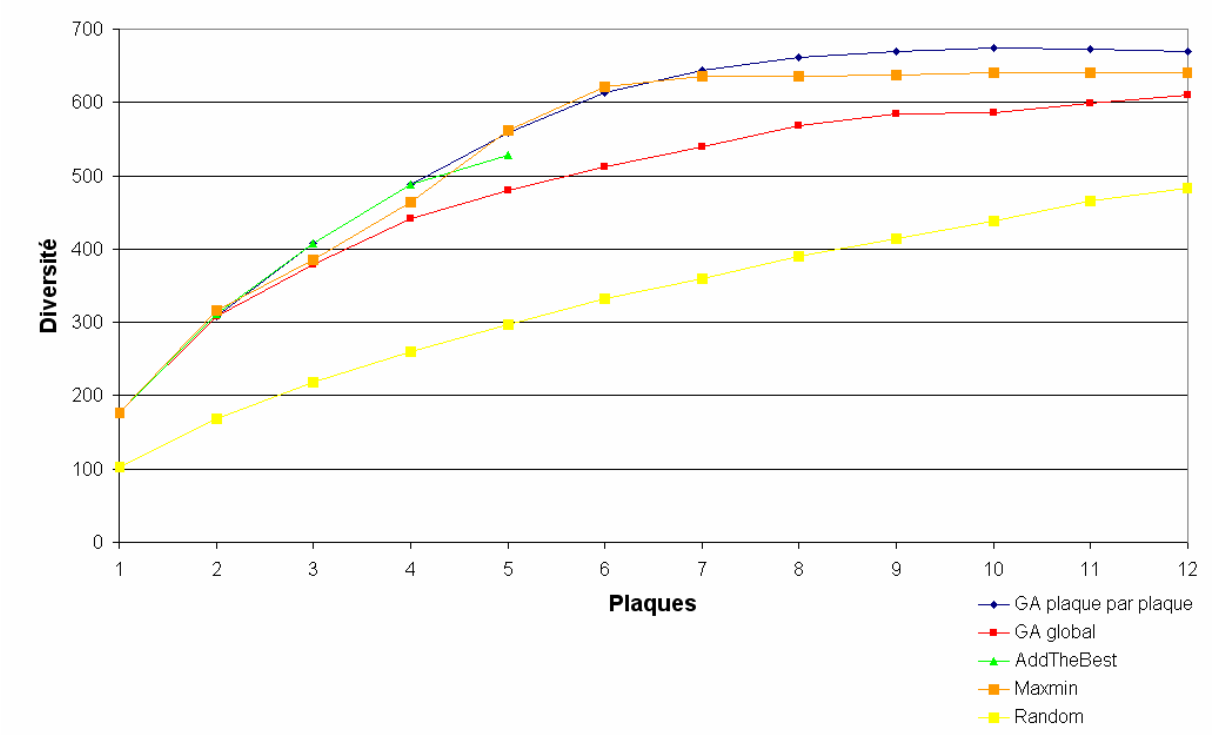


Figure 39. Diversité des plaques générées par les différents algorithmes. La courbe jaune est la référence. Elle correspond à des ensembles de composés choisis aléatoirement. La courbe de la méthode AddTheBest s'arrête à la plaque 5, pour des raisons de temps de calcul.

Il est à noter que tous les commentaires de cette partie concernant les temps de calculs, sont valables uniquement pour le problème étudié, à savoir la sélection d'un nombre restreint de molécules. En effet, pour des ensembles de molécules beaucoup plus grands, la plupart des méthodes présentées ici seraient inapplicables car trop coûteuses en temps de calcul.

1. Maxmin

Cette courbe est la moins régulière de toutes. Cela s'explique par le fait que, contrairement à toutes les autres méthodes, ce n'est pas le score de diversité qui est directement optimisé par Maxmin. Pour cette méthode le score n'est utilisé que dans un but de contrôle de diversité. En plus de l'irrégularité de la courbe liée à l'incertitude du score de diversité, cela signifie également que les résultats de la méthode Maxmin sont légèrement pénalisés.

La courbe de Maxmin est très proche de celle des algorithmes génétiques avec traitement plaque par plaque, dépassant même cette dernière par endroits. Nous considérons donc que, étant donné que la courbe Maxmin est la seule méthode pour laquelle le score de diversité n'est pas surestimé, les résultats de Maxmin sont du même ordre que ceux des algorithmes génétiques avec traitement plaque par plaque.

La méthode Maxmin est la plus rapide des méthodes que nous avons étudiées.

Figure 40. Vitesse d'exécution de l'algorithme Maxmin.

Le Figure 40, représentant la vitesse d'exécution de l'algorithme Maxmin, montre bien les deux phases de cet algorithme. Jusqu'à la quatrième plaque, il reste de nouveaux frameworks à ajouter. L'algorithme ne considérant dans un premier temps que les composés avec des frameworks non sélectionnés, l'algorithme est $O(FK)$[i], avec F le nombre de composés de la base d'origine dont les frameworks ne sont pas déjà sélectionnés, et K le nombre de composés déjà sélectionnés. La deuxième phase débute à la plaque 5 pour laquelle

[i] La notation de « O », appelée notation de Landeau, est utilisée pour indiquer à quelle vitesse une fonction augmente ou diminue. Cette notation est très utilisée pour définir la complexité d'un algorithme. Des explications détaillées sont disponibles à l'adresse suivante : http://fr.wikipedia.org/wiki/Notations_de_Landau

il n'y a plus de nouveaux frameworks à sélectionner. L'algorithme devient alors *O(NK)*, avec *N* le nombre de composés déjà sélectionnés, et *K* le nombre de composés de la base d'origine.

Cet algorithme s'avère être très intéressant de par la diversité de l'ensemble qu'il génère. Le temps d'exécution est très bon, surtout si la sélection est arrêtée avant que tous les frameworks n'aient été sélectionnés.

2. AddTheBest

Cet algorithme permet d'obtenir de très bon résultats jusqu'à la quatrième plaque, puis ces résultats diminuent en dessous de ceux de Maxmin et de la sélection par algorithmes génétiques avec traitement plaque par plaque. Mais ce qui ressort du graphique est que nous nous sommes arrêtés à une sélection de cinq plaques avec cet algorithme. La raison est que le temps de calcul augmente très rapidement. Cela s'explique par le fait que l'algorithme SCA doit être exécuté un grand nombre de fois (*O(NK)* avec *N* le nombre de composés non sélectionnés, et *K* le nombre de composés sélectionnés), sachant que la durée d'exécution de SCA augmente également avec la taille des composés sélectionnés. AddTheBest n'est donc pas une méthode intéressante, ses résultats étant moyens et son temps de calcul le rendant impossible à utiliser pour une sélection de plus de quelques plaques.

3. Algorithmes génétiques : traitement plaque par plaque et global

Nous avons vu dans la partie précédente que l'utilisation d'algorithmes génétiques était une méthode efficace pour sélectionner des plaques. Nous avons donc voulu comparer leur efficacité à celle d'autres algorithmes pour une sélection d'un petit nombre de composés. Deux approches on été choisies. D'une part sélectionner successivement les plaques (donc des groupes de 80 composés), et d'autre part la sélection de l'ensemble des composés souhaités. La première méthode donne des plaques de diversité croissante, alors que la deuxième donne un ensemble de plaques dont les composés sont les plus divers par rapport à l'ensemble d'origine. Aucune des deux méthodes ne s'occupe de classer les composés au sein d'une même plaque comme les méthodes Maxmin et AddTheBest. Cela n'est aucunement gênant car une plaque est testée dans son intégralité.

Le fait de tester ces deux méthodes nous permettra d'évaluer les performances des algorithmes génétiques dans deux cas différents. Dans le premier cas les algorithmes génétiques ne traiteront que 80 gènes à la fois soit un espace de solution restreint. Cela donne donc une taille de chromosome acceptable pour la minimisation. Par contre une plaque déjà conçue ne peut plus être modifiée, ce qui limite le champ d'action de l'algorithme. Dans le deuxième cas il y a autant de gènes que de composés à sélectionner. Pour 6 plaques à sélectionner cela représente 480 gènes. Cela donne une taille de chromosome conséquente, et donc plus difficile à optimiser. L'avantage de cette structure par rapport à la précédente est que l'algorithme a la liberté de modifier n'importe quel composé de n'importe quelle plaque, ce qui devrait théoriquement permettre d'arriver à une diversité sinon idéale, tout au moins meilleure. Par contre avec les algorithmes génétiques plaque par plaque, il est possible de prendre les 5 premières plaques parmi les 6 sélectionnées en gardant une très bonne diversité (la sixième plaque contient les composés qui complètent au mieux la diversité des 5 premières). Cela n'est pas possible avec les algorithmes génétiques globaux.

Nous avons utilisé comme paramètres 2000 générations, 20 chromosomes, un taux de mutation de 0,01, et les deux meilleurs chromosomes survivent dans la génération suivante.

Lors de l'analyse de la Figure 39, nous constatons que les algorithmes génétiques plaque par plaques donnent de meilleurs résultats que ceux optimisant directement la totalité des composés. La taille des chromosomes joue donc bien un rôle important dans l'obtention de bons résultats par ces méthodes.

Pour avoir une idée du travail effectué par les algorithmes génétiques, pour 12 plaques, le nombre de solutions évaluées est de 480 000 (12 plaques * 2000 générations * 20 chromosomes).

C. Mise en plaques concrète de la chimiothèque réelle ICOA

D'après les tests précédents nous avons choisi l'algorithme Maxmin, utilisant une diversité à la fois par frameworks et par fingerprints, pour la mise en plaques de la chimiothèque ICOA. Cet algorithme a été choisi pour ses résultats et sa rapidité. Nous utiliserons des plaques à 96 puits, avec 80 composés par plaques.

Sur les 1671 composés que nous considérons pour notre étude, nous en mettrons 480 en plaques. Nous sélectionnerons 80 molécules en plus afin de pallier à des problèmes de masses disponibles.

La première étape de la mise en plaque consiste en la création de plaques grand-mères comportant dans chaque puits 1 mL de solution à 10 mM. La masse molaire moyenne des composés étant de 324 $g.mol^{-1}$, on mettra 3,2 g de produit en solution dans 1 mL de DMSO.

Les deux prochaines étapes ont été réalisées à l'aide d'un robot par la société GreenPharma. Les plaques grand-mère seront tout d'abord réparties en 10 plaques mères contenant chacune 100 µL à 10 mM. On additionnera 10 µL de ces plaques mères à 90 µL de DMSO pour créer les plaques filles, qui auront donc une concentration de 1 mM.

Ce processus de mise en plaque permet donc la génération de 100 plaques filles. Ces plaques seront dans un premier temps testées lors de collaborations universitaires. La chimiothèque ICOA mise en plaque sera également utilisée dans le cadre du projet BioPhenics, porté par l'Institut Curie, qui utilise un système de criblage basé sur l'imagerie cellulaire afin d'identifier de nouveaux principes actifs, notamment dans le domaine des anticancéreux. Toujours dans le cadre de cette collaboration, nous utiliserons nos algorithmes pour réaliser une filtration et une proposition de mise en plaque de la chimiothèque de l'Institut Curie.

IV. Conclusion

Nous avons présenté plusieurs travaux de sélections de composés pour des tests de criblages. La sélection d'un ensemble de composés pour un criblage par docking, la sélection de plaques, et la mise en plaques de composés ont été traitées.

Il apparait clairement dans cette partie que chaque projet nécessite une sélection bien adaptée. Les problèmes rencontrés sont en effet souvent très différents. Il faut évidemment dans chaque cas adapter les filtres utilisés. De plus les problèmes liés à la diversité nécessitent souvent d'adapter nos programmes, ou d'en créer de nouveaux.

On notera également que, s'il y a une profusion de publications concernant la sélection de composés par diversité, certains cas particuliers sont très peu abordés par la communauté chemoinformatique. Nous avons vu que cela est par exemple le cas de la sélection de plaques.

1. Wunberg, T; Hendrix, M; Hillisch, A; Lobell, M; Meier, H; Schmeck, C. Wild, H; Hinzen, B. Improving the hit-to-lead process: data-driven assessment of drug-like and lead-like screening hits. *Drug Discov. Today* **2006**, *11*, 175-180.
2. Hopkins, A.L.; Groom, C.R.; Alex, A. Ligand efficiency: a useful metric for lead selection. *Drug Discov. Today* **2004**, *9*, 430-431.
3. Kuntz, I.D.; Chen, K.; Sharp, K.A.; Kollman, P.A. The maximal affinity of ligands. *Proc Natl Acad Sci U S A.* **1999**, *96*, 9997-10002.
4. MacCoss, M.; Baillie, T.A. Organic Chemistry in Drug Discovery. *Science* **2004**, *303*, 1810-1813.
5 SVL Exchange, Chemical Computing Group, http://svl.chemcomp.com/
6. Kazius, J.; McGuire, R.; Bursi, R. Derivation and Validation of Toxicophores for Mutagenicity Prediction. *J. Med. Chem.* **2005**, *48*, 312-320.
7. Wolpert, D.H., Macready, W.G. No Free Lunch Theorems for Search, *Technical Report SFI-TR-95-02-010 (Santa Fe Institute)*, **1995**.
8. Wolpert, D.H., Macready, W.G. (1997), *No Free Lunch Theorems for Optimization*, *IEEE Transactions on Evolutionary Computation 1* **1997**, *67*.
9. Dembski, W.A.. No Free Lunch. Why Specified Complexity Cannot be Purchased Without Intelligence. (Lanham, Maryland: Rowman and Littlefield Publishers) **2002**.
10. Dembski, W.A. c. Evolution's Logic of Credulity: An Unfettered Response to Allen Orr. **2002** http://www.designinference.com/documents/2002.12.Unfettered_Resp_to_Orr.htm
11. Erakh, M. The No Free Lunch Theorems and their Applications to Evolutionary Algorithms, **2003**. http://www.talkreason.org/articles/orr.cfm
12. Holland, J. Les algorithmes génétiques. *Pour la Science* **1992**, *179*, 44-51.
13. Haupt, R.L.; Haupt, S.E. John Wiley & Sons, Pratical Genetic Algorithms, second edition, **2004**, Hoboken, New Jersey.
14. Devillers, J. Genetic Algorithms in Computer-Aided Molecular Design, Genetic Algorithms in Molecular Modeling, Academic Press, San Diego, **1996**.
15. Kirkpatrick, S.; Gelatt Jr., C.D.; Vecchi, M.P. Optimization by simulated annealing. *Science* **1983**, *220*, 671-680.
16. Kennedy, J.; Eberhart, R.C. Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks, Piscataway*, NJ. *1995*, 1942-1948.
17. Heppner, F.; Grenander, U. A stochastic nonlinear model for coordinated bird flocks. In S. Krasner, Ed., *The Ubiquity of Chaos*. AAAS Publications, Washington, DC, **1990**.
18. Reynolds, C.W. Flocks, herds and schools: a distributed behavioral model. *Computer Graphics*, **1987**, *21*, 25-34.
19. Wilson, E.O. Sociobiology: The new synthesis. Cambridge, MA: Belknap Press, **1975**.
20. Russell C.E.; Shi, Y.; Kennedy, J. Swarm Intelligence (The Morgan Kaufmann Series in Artificial Intelligence), Kaufmann, M. **2001**.
21. Optimisation par essaims particulaires, http://en.wikipedia.org
22. Dorigo, M., *Optimization, Learning and Natural Algorithms*, PhD thesis, Politecnico di Milano, Italie, **1992**.
23. Java Genetics Algorithms Package, http://jgap.sourceforge.net/
24. JGAP, http://jgap.sourceforge.net/
25. Cantú-Paz, E. Efficient and Accurate Parallel Genetic Algorithms. Boston, MA: Kluwer Academic Publishers. **2000**.
26. Agrafiotis, D.K.; Rassokhin, D.N. Design and Prioritization of Plates for High-Throughput Screening. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 798-805.
27. M.S. Lajiness, in QSAR : Rational Aproaches to the Design of Bioactive Compounds, C. Silipo, A. Vittoria (Eds.), Elsevier, Amsterdam, **1991**, 201-204.

CONCLUSION

Ce travail de thèse a porté sur l'utilisation de la chemoinformatique pour la gestion de chimiothèques destinées aux tests de criblages virtuels ou à haut débits. On peut dégager trois grands résultats de ce travail :

- la conception du logiciel de gestion de chimiothèque *ScreeningAssistant*. C'est un outil gratuit et qui permet ainsi aux laboratoires universitaires et aux sociétés de biotechnologies de concevoir et d'exploiter des chimiothèques destinées au criblage. C'est à notre connaissance le seul logiciel pré à l'emploi qui offre ces fonctionnalités. Il a été développé dans son intégralité pendant cette thèse. Des fonctionnalités de filtrage et de génération de sous-ensembles par diversité sont disponibles. Le logiciel est open source et disponible sur le site *SourceForge*, permettant une continuité du projet au-delà de cette thèse. Il a été téléchargé 180 fois en 5 mois.

- l'analyse des librairies, principalement commerciales, de molécules destinées aux criblages. Cette analyse a porté sur les propriétés « drug-like », « lead-like », les structures privilégiées, les doublons et les molécules originales. Elle a également porté sur la diversité en se basant sur des fingerprints sous-structurales, des fingerprints pharmacophoriques, les frameworks, les scaffolds et les chaînes latérales, et les fragments rétrosynthétiques. Un score de diversité globale, prenant en compte tous ces types de diversités, a été proposé. Cette étude permet d'avoir une vision globale des propriétés des chimiothèques disponibles.

- L'utilisation des techniques et résultats de la thèse pour la réalisation de travaux sur les chimiothèques. La chimiothèque de l'ICOA a ainsi été filtrée et un sous-ensemble de composés divers sélectionnés afin de créer des plaques de diversité croissante. Cette étude de mise en plaques a également été réalisée pour la chimiothèque de l'institut Curie, et celle de la société Hybrigenics. Nous avons également conçu des chimiothèques de criblages pour les sociétés Hybrigenics (100 000 produits), Athelas et Trophos.

Outre le logiciel, une publication sur ce travail est parue à ce jour :

Monge, A.; Arrault, A.; Marot, C.; Morin-Allory, L. Managing, Profiling and Analyzing a Library of 2.6 Million Compounds Gathered from 32 Chemical Providers. *Mol Divers*. **2006**, DOI : 10.1007/s11030-006-9033-5.

Il a été présenté lors de trois communications orales,

Monge, A.; Arrault, Al.; Marot, C.; Morin-Allory, L. De la chimiothèque au criblage virtuel. *4ème journée du PPF "CASCIMODOT",* **06-2006** - Tours.

Monge, A.; Arrault, Al.; Marot, C.; Morin-Allory, L. La chimiothèque : un élément clé de la découverte de nouveaux médicaments. *Sciences en Sologne 2005*, **06-2005** - Orléans.

Monge, A. ; Arrault, Al. ; Marot, C. ; Morin-Allory, L. Analyse de l'espace chimique de plus de 3 millions de molécules destinées au criblage virtuel ou au criblage à haut débit. *XIVème Colloque du Groupe de Graphisme et Modélisation Moléculaire (GGMM'2005),* **05 - 2005** - îles des Embiez.

ainsi que dans onze posters dans des congrès nationaux et internationaux.

Enfin, nous avons avec Alban Arrault, également doctorant dans le laboratoire de modélisation moléculaire et de chimiométrie de l'ICOA, participé au concours de création d'entreprise 2004/2005 organisé par Orléans Technopole. Nous avons proposé un projet de société de services basé sur les compétences développées lors de nos travaux de thèse, à savoir la gestion et l'utilisation de chimiothèques et le criblage virtuel. Les objectifs étaient la réalisation d'une étude de marché, et d'un business plan. Le projet a été récompensé par un prix pour le business plan, et le prix du meilleur prototype pour *ScreeningAssistant*.

En conclusion, les travaux de thèse ont permis à la fois le développement de nouvelles méthodes et de nouveaux outils, tout en restant proches des besoins du monde industriel. Le projet *ScreeningAssistant* est à l'heure actuelle utilisé pour gérer la chimiothèque virtuelle de l'ICOA comportant 5 millions de références. De nombreuses fonctionnalités pourraient être apportées au logiciel. Il serait notamment possible d'intégrer un module de QSAR basé sur des techniques maîtrisées par le laboratoire telles que les algorithmes génétiques et le recuit simulé pour la sélection de descripteurs, et les machines à support de vecteurs et les réseaux de neurones pour la corrélation entre les descripteurs et l'activité biologique. Il serait également possible d'implémenter un algorithme de conception *de novo*, lui aussi basé sur une technique d'optimisation ainsi que sur la méthode de fragmentation RECAP. L'utilisateur aurait le choix entre différents scores qui pourraient être combinés (docking, pharmacophores, similarité de surface en utilisant le logiciel ROCS, similarité électrostatique en utilisant le logiciel EON, score « drug-like »…).

Enfin, la version distribuée est utilisée dans plusieurs laboratoires, ce qui correspond à l'objectif initial de ce travail.

ANNEXES

I. Définition des fingerprints SSKey-3DS

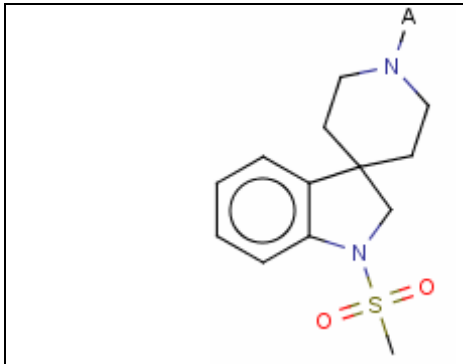
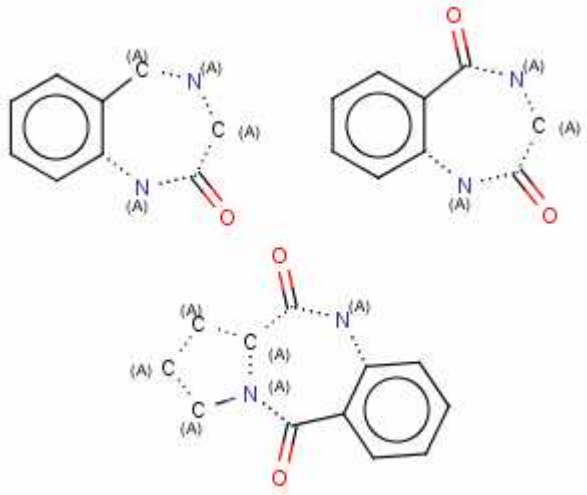
Nous avons corrigé les fingerprints SSKey-3DS implémentées dans JOELib à partir de la publication originale.

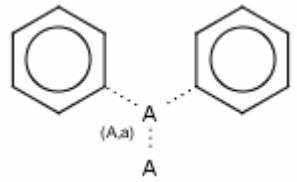
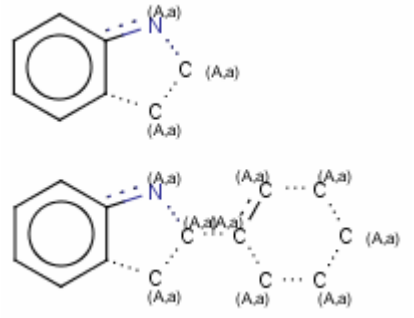
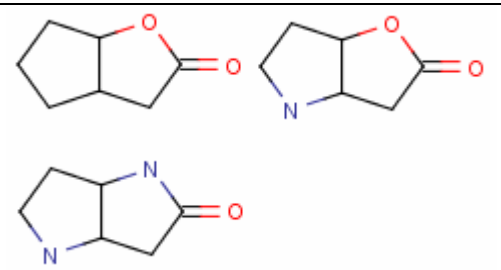
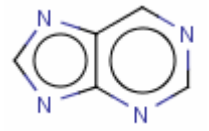
Numéro de bit	Définition	Code SMARTS (pour les définitions le nécessitant)
1	FRB 0 to 0.1	
2	FRB 0.1 to 0.2	
3	FRB 0.2 to 0.3	
4	FRB 0.3 to 0.4	
5	FRB >0.4	
6	ARB 2 to 7	
7	ARB 8 to 15	
8	ARB 16 to 19	
9	ARB 20 to 25	
10	ARB 26 to 31	
11	ARB 32 to 37	
12	ARB >38	
13	Heterocycle	
14	Aromatic OH	[$([OX2H1]-a)]
15	Aliphatic OH	[$([OX2H1]-C)]
16	Aliphatic secondary amine	[$(N(C)C);!$(N(C)(C)C)]
17	Aliphatic tertiary amine	[$(N(C)(C)C)]
18	Phenyl ring	[$(c1ccccc1)]
19	Nitrogen-containing aromatic ring	
20	=-SO2	[$(S(=O)(=O)-*)]
21	=-SO	[$(S(=O)-*);!$(S(=O)=O)]
22	Ester	[$(C(=[OX1])-OC)]
23	Amide	[$(C(=[OX1])-N)]
24	5-membered non-aromatic ring	
25	5-membered aromatic ring	

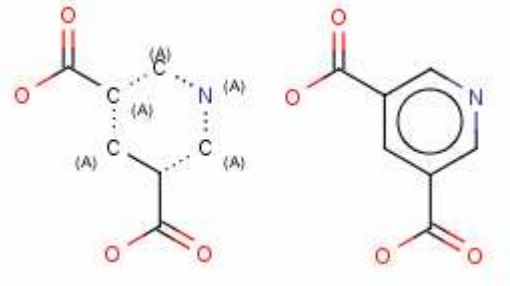
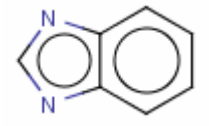
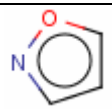
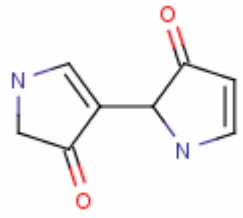
26	9-membered or larger (fused) ring	
27	Fused ring system	[*R][*R]([*R])[*R]
28	Fused aromatic ring system	[aR][aR]([aR])[aR]
29	=-OSO	[$(OSO)]
30	Halogen atom	
31	Nitrogen attached to a-carbon of aromatic system	[$(cCN)]
32	=-NO2	[$(N(=[OX1])[OX1])]
33	Rings separated by 2-3 non-ring atoms	[$([*;R][*;!R][*;!R][*;R]),$([*;R][*;!R][*;!R][*;!R][*;R])]
34	Rings separated by 4-5 non-ring atoms	[$([*;R][*;!R][*;!R][*;!R][*;!R][*;R]),$([*;R][*;!R][*;!R][*;!R][*;!R][*;!R][*;R])]
35	NN	[$(NN)]
36	C attached to 3 carbons and a hetero atom	C([#6])([#6])[#6][*;!C;!H]
37	Oxygens separated by 2 atoms	[#8]**[#8]
38	Methyl attached to hetero atom	[CH3][*;!#6;!#1]
39	Double bond	
40	Non-H atom linked to 3 heteroatoms	[*]([*;!#6;!#1])([*;!#6;!#1])[*;!#6;!#1]
41	Quaternary atom	[*][*X4]([*])([*])[*]
42	2 methylenes separated by 2 atoms	*[CH2]**[CH2]*
43	Non-ring oxygen attached to aromatic system	[O;R0]~a
44	2 non-C,H atoms separated by 2 atoms	[!#6;!#1]~*~*~[!#6;!#1]
45	HBA=1	
46	HBA=2	
47	HBA=3	

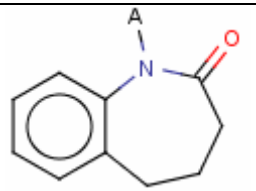
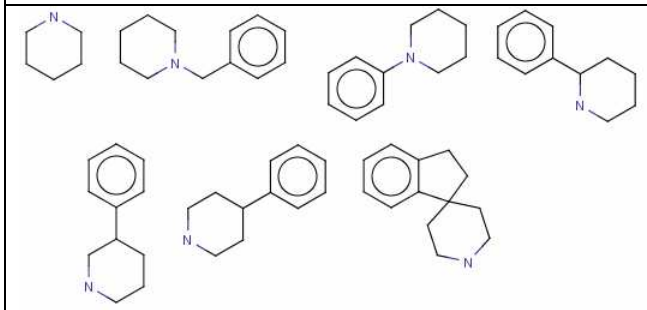
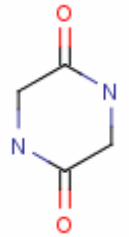
48	HBA=4	
49	HBA=5	
50	HBA=6	
51	HBA=7	
52	HBA=8	
53	HBA=9	
54	HBA >10	

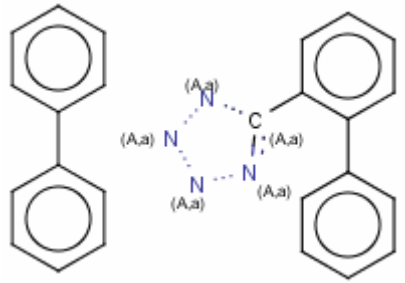
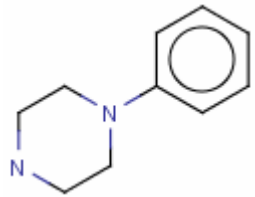
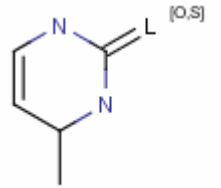
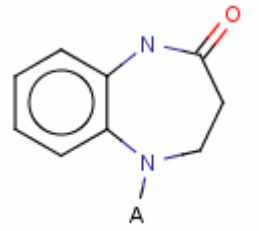
II. Définitions des structures privilégiées et correspondance avec le numéro utilisé par *ScreeningAssistant*.

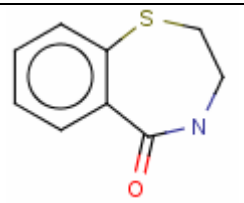
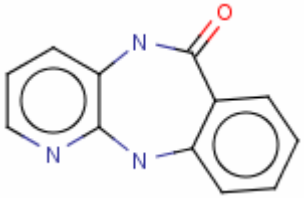
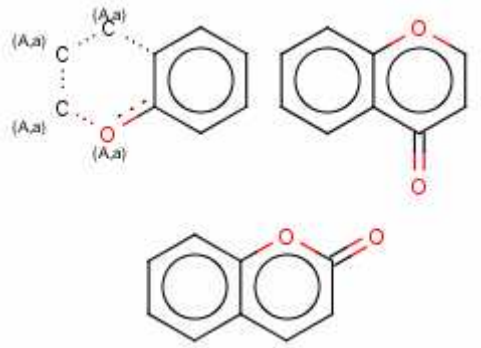
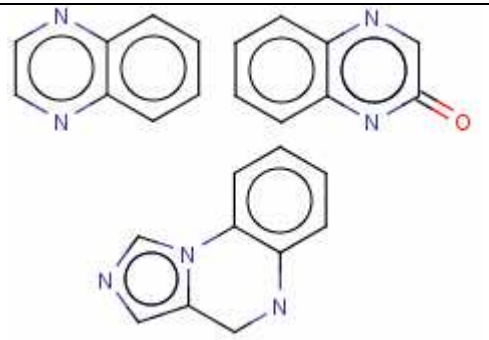
	0 : spiroindoline sulfonamide	c12c(cccc1)N(CC23CCN(CC3)*)S(=O)(=O)C
	100 : 1,4-Benzodiazepin-2-ones 101 : 1,4-Benzodiazepin-2,5-diones 102 : Pyrolo-[2,1-c][1,4]benzodiazepin-5,11-diones	c1cccc2c1~N~C(=O)~C~N~C2 c1cccc2c1~N~C(=O)~C~N~C2(=O) c1cccc2c1~N~C(=O)~C(~C~C~C3)~N3~C2(=O)

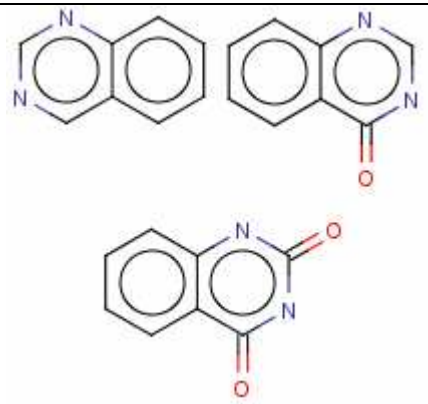
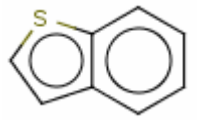
	200 : Di-PhenylMethane	c1ccccc1~*(~*)~c2ccccc2
	300 : Indole 301 : Tryptophan	c1cccc2c1~[#6]~[#6]~[#7]2 c1cccc2c1~[#6]~[#6](~[#6]3~[#6]~[#6]~[#6]~[#6]~[#6]3)~[#7]2
	400 : 5,5-trans lactone	C1CCC2C1CC(=O)O2 N1CCC2C1CC(=O)O2 N1CCC2C1CC(=O)N2
	500 : Purine	c1ncc2ncnc2n1

	600 : 1,4-dihydropyridine (DHP) 502 :	[#8]C(=O)C1~C~N~C~C(~C1)C([#8])=O [#8]C(=O)c1cncc(c1)C([#8])=O
	700 : Benzimidazole	c1ccc2ncnc2c1
	800 : Benzofuran	c1ccc2~~[#8]~[#6]~[#6]~[#6]2c1
	900 : Isoxazole	c1ccon1
	1000 : 3,5-linked pyrolin-4ones	N1C=CC(=O)C1C2=CNCC2(=O)

	1100 : benzazepinone	c12c(cccc1)CCCC(N2*)=O
	1200 : 4-substitued piperidine 1201 : Benzyl piperidine 1202 : Arylpiperidine 1203 : Arylpiperidine alpha 1204 : Arylpiperidine beta 1205 : Arylpiperidine gamma 1206 : Spiro indoline piperidine	C1CCNCC1 c1ccccc1CN2CCCCC2 C1CCCCN1c2ccccc2 C1CCCC(c2ccccc2)N1 C1CCC(c2ccccc2)CN1 C1CC(c2ccccc2)CCN1 C1CC2(CCN1)CCc3ccccc23
	1300 : cyclic dipeptides: diketopiperazines or piperazin-2,5-diones	O=C1CNC(=O)CN1

	1400 : biphenyl 1500 : bi-phenyl tetrazole	c1ccc(cc1)-c2ccccc2 c1ccc(cc1)-c2ccccc2-[#6]3~[#7]~[#7]~[#7]~[#7]3
	1500 : arylpiperazines	C1CN(CCN1)c2ccccc2
	1600 : Dihydropyrimidones	[#6]C1NC(=[#8,#16])NC=C1
	1700 : 1,5-Benzodiazepin-2-ones	*N1CCC(=O)Nc2ccccc12

	1800 : 1,4-Benzothiazepin-5-ones	O=C1NCCSc2ccccc12
	1900 : 5,11-Dihydro-benzo[e]pyrido[3,2-b][1,4]-diazepin-6-ones	O=C1Nc2cccnc2Nc3ccccc13
	2000 : Benzopyrans 2001 : Chromones 2002 : Coumarins (& Pyranocoumarins)	c1(~[#6]~[#6]~[#6]~[#8]2)c2cccc1 O=c1ccoc2ccccc12 O=c1oc2ccccc2cc1
	2100 : Quinoxalines 2101 : Quinoxalinones 2103 : fused imidazole ring	c1ccc2nccnc2c1 n1c(=O)cnc2ccccc12 C1Nc2ccccc2-n3cncc13

	2200 : Quinazolines 2201 : Quinazolinones 2202 : quinazolindiones	c1ccc2ncncc2c1 O=c1ncnc2ccccc12 O=c1nc2ccccc2c(=O)n1
	2300 : Benzothiophenes	c1ccc2sccc2c1

III. Tableaux contenant les résultats de l'analyse de la chimiothèque virtuelle de l'ICOA.

Pour CFMS ≤ 1, PDL ≤ 1, PLL ≤ 1 les valeurs données sont le nombre de composés répondant à ces critères. La colonne Structure Privilégiée correspond aux nombre de composés possédant une structure privilégiée. Pour SSKey3DS, MACCS, TGD, TGT, les valeurs données sont le nombre de clusters obtenus avec ces différentes fingerprints. Le second tableau liste les nombres de frameworks, scaffolds, chaînes latérales et fragments RECAP obtenus dans chacune des bases, ainsi que les scores de diversité globale et de diversité relative.

	CFMS ≤ 1	**PDL ≤ 1**	**PLL ≤ 1**	**Structures Privilégiées**	**SSKey3DS**	**MACCS**	**TGD**	**TGT**
ACB Blocks	54320	58665	28122	34688	484	1373	297	98
AnalytiCon Discovery	5903	5942	1540	3987	265	513	64	16
Asinex	299796	306071	143935	84774	5597	28774	3970	696
Aurora Fine Chemicals	28925	30069	18204	6452	1918	5631	1570	442
BioFocus	21219	21334	8291	8116	306	1115	134	27
CB R&D	134	149	146	0	41	73	37	21
Cerep	16243	16579	4889	7643	1164	2980	398	74
ChemBridge	395303	405740	212958	96612	6422	35021	4452	731
ChemDiv	475242	485393	185183	184847	6654	39105	4589	720
ChemStar	50289	52100	26364	14110	3285	11718	2332	482
ChemT&I	399938	412100	189363	68219	4490	20121	2378	460
Chim. Nat.	21120	23443	17396	7575	2788	9043	2509	590
Combi-Blocks	386	423	413	121	163	328	145	59
CombiPure	876	876	221	164	36	54	7	3
EMC Microcollections	19613	20116	7628	4725	261	784	112	34
Enamine	363774	380997	163538	108623	6867	41867	4748	788
Florida Center for Heterocyclic Compounds	23748	26175	18684	6174	2760	9488	2710	678
Frontier Scientific	175	262	259	53	149	287	146	84
ICOA	2369	2658	1957	692	675	1323	499	174
InFarmatik	508	537	499	130	111	218	106	54

InterBioScreen	370940	380592	151776	148102	6121	32895	4732	823
KeyOrganics	174370	180704	73893	38849	2861	13067	2354	473
LaboTest	2512	2883	2533	620	850	1822	879	327
Matrix Scientific	11656	14428	13402	2070	1249	3848	1323	438
MayBridge	60414	65253	46355	12999	4093	19273	4023	820
MDPI	7685	8948	6630	1902	1741	4190	1565	511
MedChemLab	146092	148736	48555	35890	2733	11200	1614	316
NCI	173232	202054	153123	52176	10647	62724	11325	1971
Otava	68647	70843	36016	18627	2570	9342	1700	389
Prestwick	889	948	673	369	490	724	366	141
Specs	185222	191208	94068	43259	6079	32006	4876	833
Spectrum Info	620	824	633	308	200	276	128	66
SynChem	409	588	543	221	124	215	105	60
SynphaBase	92	116	94	28	82	100	76	49
TimTec	560351	576751	285915	129495	7946	48154	6174	1076
TOSLab	17902	18587	8276	6062	1429	4422	934	206
Tripos	56713	58630	23865	23795	1460	4236	429	87
VitasM Laboratory	198226	201899	98260	50721	4268	19342	2714	511

	Frameworks	**Scaffolds**	**Chaînes latérales**	**RECAP**	**Diversité Globale**	**Diversité Relative**
ACB Blocks	830	3589	125	7545	3.7810	6.174E-05
AnalytiCon Discovery	981	3980	188	886	1.4142	1.634E-04
Asinex	17489	78088	4971	47699	43.8819	1.269E-04
Aurora Fine Chemicals	1674	6968	1349	6484	11.2228	3.561E-04
BioFocus	2039	6431	181	1167	2.1998	9.277E-05
CB R&D	10	39	35	97	0.2920	1.659E-03
Cerep	4505	11017	516	2435	5.6380	2.808E-04
ChemBridge	15927	89921	5562	53186	48.6846	1.143E-04
ChemDiv	27118	119920	5779	70460	58.0721	1.050E-04
ChemStar	5497	19677	2244	16309	19.5807	3.261E-04
ChemT&I	12496	59863	3446	35202	31.4484	6.486E-05
Chim. Nat.	2666	10743	2472	11545	17.3702	6.597E-04
Combi-Blocks	30	157	127	288	0.9784	9.274E-04

CombiPure	109	312	35	193	0.1955	2.149E-04
EMC Microcollections	1343	3808	245	2504	2.0814	8.696E-05
Enamine	30930	148325	6032	56913	59.7260	1.395E-04
Florida Center for Heterocyclic Compounds	2651	10786	3043	11021	18.6212	6.309E-04
Frontier Scientific	40	196	86	166	1.0577	1.731E-03
ICOA	603	1630	545	1984	3.8402	1.195E-03
InFarmatik	78	289	73	242	0.7832	1.448E-03
InterBioScreen	19754	86924	6120	56117	50.7324	1.192E-04
KeyOrganics	5711	26980	2336	21606	20.8891	1.117E-04
LaboTest	295	1202	528	1441	5.3080	1.714E-03
Matrix Scientific	437	3118	893	2948	8.1409	5.441E-04
MayBridge	4520	25620	3081	20461	27.6731	4.003E-04
MDPI	1349	4649	1055	4158	10.2281	1.003E-03
MedChemLab	7687	28788	1842	22330	18.8797	1.053E-04
NCI	16428	81368	18382	72993	89.7036	3.670E-04
Otava	5119	18895	1644	13261	15.7044	2.044E-04
Prestwick	340	771	348	1009	2.6623	2.383E-03
Specs	15909	64567	4645	53730	46.6398	2.125E-04
Spectrum Info	217	462	107	535	1.1340	9.618E-04
SynChem	46	243	52	304	0.8165	1.384E-03
SynphaBase	36	84	72	154	0.6202	4.219E-03
TimTec	24173	118531	7906	86653	69.5710	1.057E-04
TOSLab	3317	8487	768	8426	8.5838	3.694E-04
Tripos	4942	20964	1067	5142	8.0493	1.233E-04
VitasM Laboratory	10752	46581	3482	32066	29.9402	1.323E-04

IV. Manuel d'utilisation de *ScreeningAssistant*

ScreeningAssistant Manual

More scientific details about the software can be found **here**.

NEEDED SOFTWARES

Before you can use ScreeningAssistant, 3 softwares must be installed on your computer:

- **Java 5 JRE** (ScreeningAssistant won't work with previous Java virtual machine)
- **Java3D** (Java3D must be installed AFTER Java 5 JRE)
- A MySQL server to manage databases. It is recommended to run MySQL on a separated Linux computer. However, for testing purpose, you can use **EasyPHP**. EasyPHP is a Windows software which automatically installs a MySQL server on your computer.

- Optionally, you can use **Marvin** as molecular viewer by adding MarvinBeans.jar to your classpath. If you don't have Marvin, ScreeningAssistant will use its internal molecular viewer (JOELib molecular viewer).

FOR USERS OF A PREVIOUS VERSION OF SCREENINGASSISTANT

This new version introduces changes in the structure of the database used by ScreeningAssistant. In consequence you can't use this new version directly with a database created with a previous version of ScreeningAssistant. Your old databases need to be updated. It could be done using a little java program. Here is an example for updating a database named MyDataBase:

```
C:\Program Files\ScreeningAssistant>java sa/SA01DBToSA02DB 127.0.0.1 MyDataBase
Toto MyPassword
```

In this example the MySQL server is installed on this computer, the user name is Toto and his password is MyPassword.

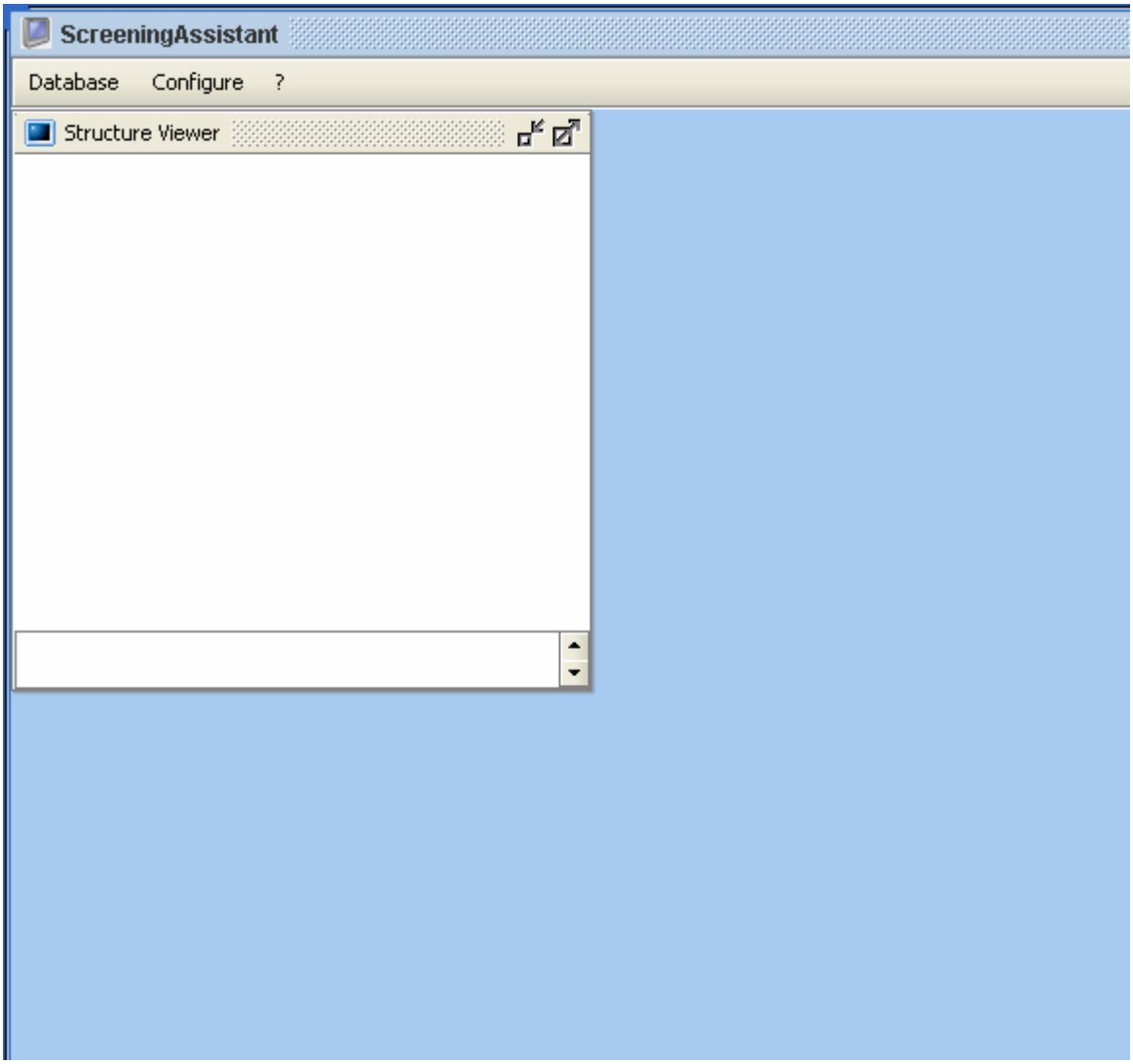
USER GUIDE

The ScreeningAssistant desktop environment: create your first database

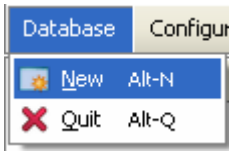
Launch your MySQL server (e.g. easyPHP) and double-click on the ScreeningAssistant icon on your desktop.

(You must launch EasyPHP each time you want to use ScreeningAssistant. EasyPHP launch two softwares: Apache (a web server) and MySQL. Apache don't need to be started, only MySQL wil be used by ScreeningAssistant.)

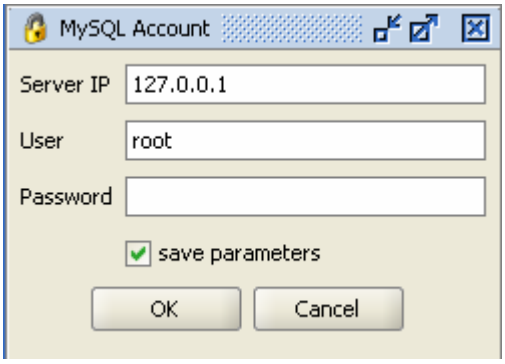
The following windows will appear on the screen :



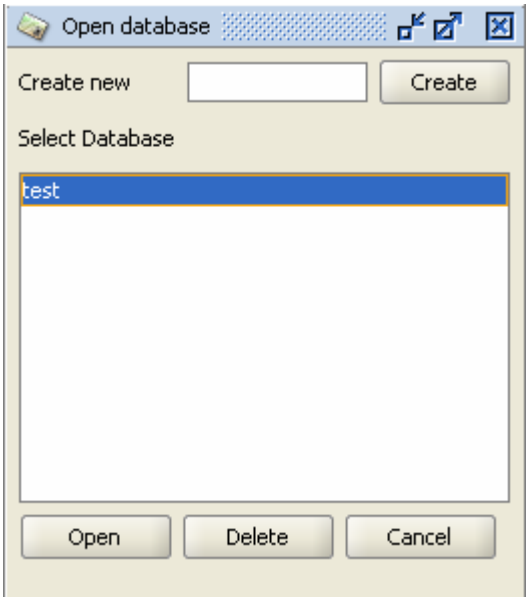
You can see that only an internal window is open, the molecule viewer, which is empty. ScreeningAssistant GUI is built as a virtual desktop, and it means that you can minimize, resize and move the opened windows. First thing to do is to open an existing database or to create a new one. To do this open the Database menu and click on New.



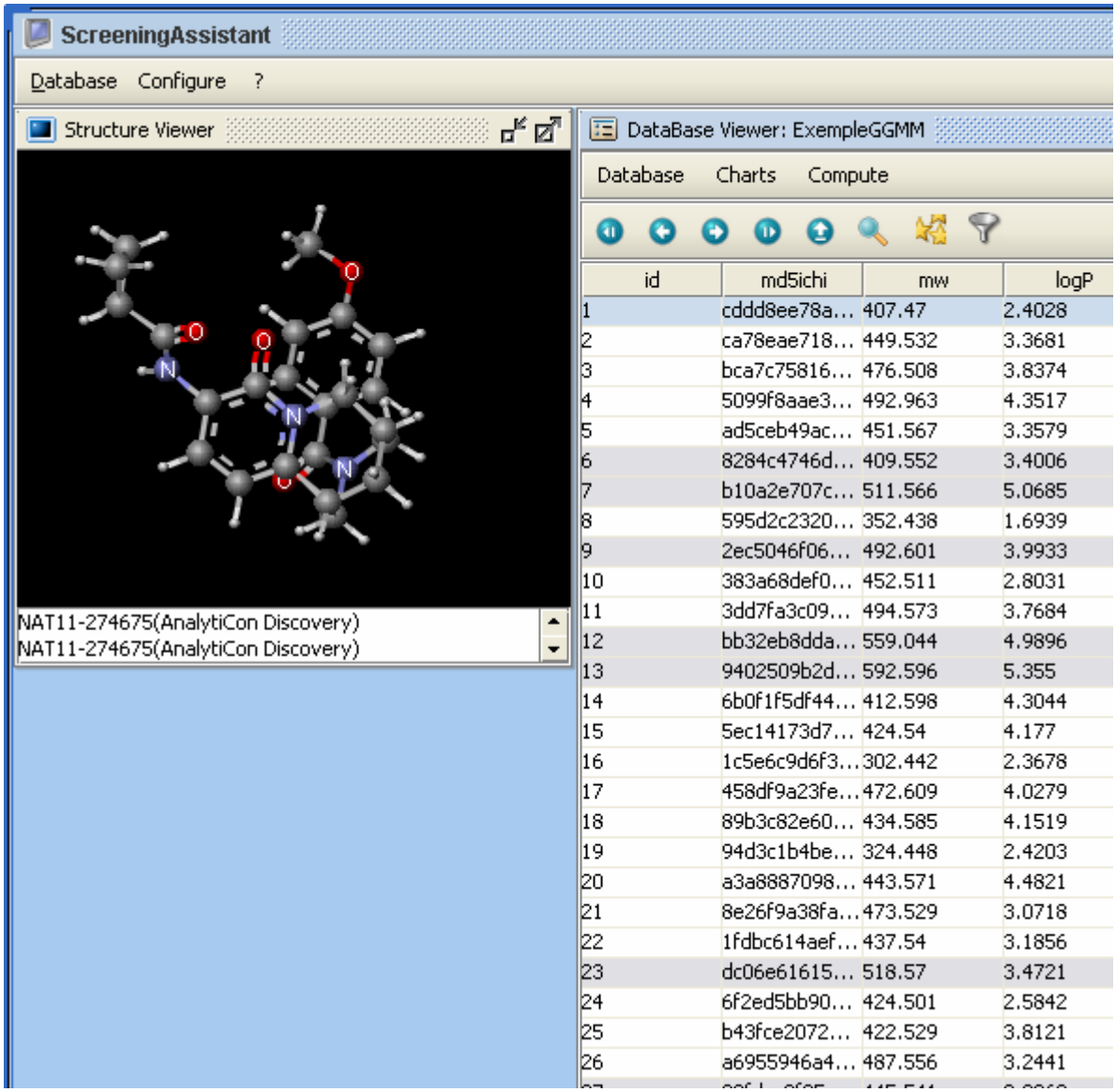
In the login window, enter the server's IP address (127.0.0.1 if you use EasyPHP on your computer), a user name of a MySQL administrator account and the corresponding password (by default MySQL admin name is "root" with no password).



Clicking OK in the login window will let you create a new database or open an existing one. To create a new database, enter the name in the text box and click create. To open an existing database select it and click Open. You can also delete a selected database.



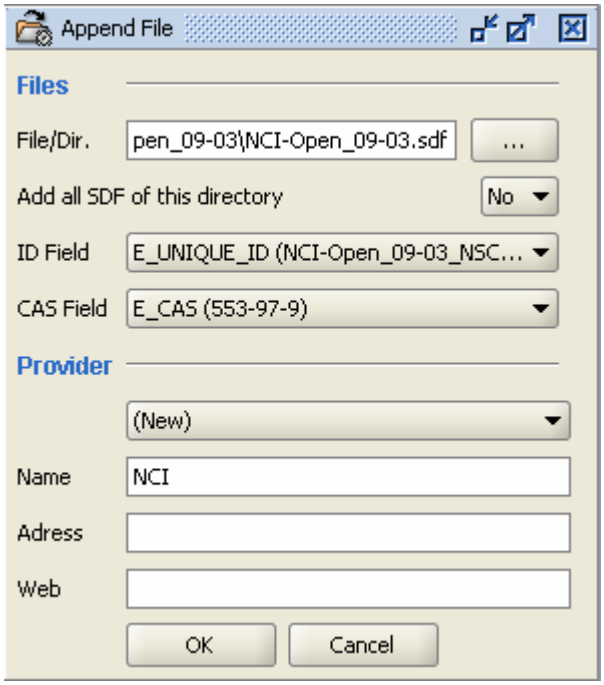
When a database is opened, you can see its compounds in the Database Viewer window. Grey rows means that corresponding molecules are estimated as not drug-like. Clicking in any row of the database viewer will display the molecule in the structure viewer.



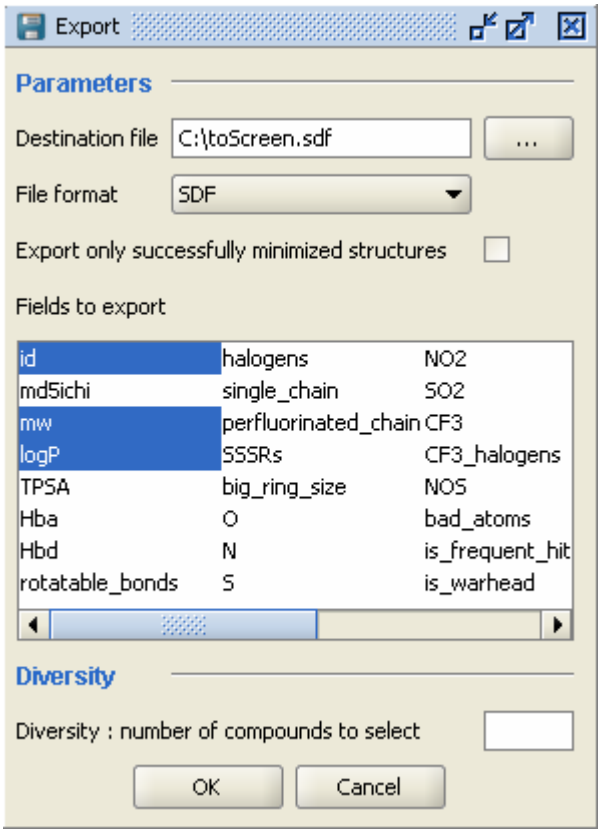
At your first use you won't have any row in your Database Viewer. The next step will be to import a SDF.

The menu bar: import/export compounds and analyse your database

To append compounds in the database go to the Database menu of the Database Viewer window, and click Append.

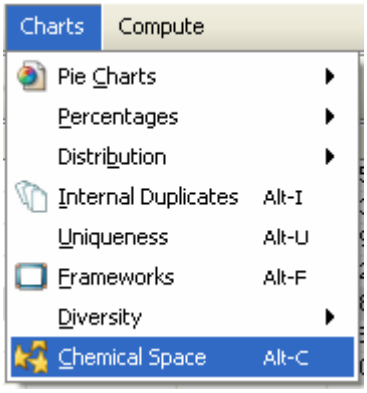


In the Append window, firstly you must choose the file to import. If "Add all SDF of this directory" is set to Yes, you must be sure that all the SDF of the directory have the same ID and CAS field names. Then, you must choose which field names corresponds to ID field and CAS field. CAS field is optional, but you must define ID field. In the provider part, you can choose the provider of the compounds, either an existing one or a new one (you must then define at least the Name field). Click OK to begin the importation process.



To export the selection of the Database Viewer (we will explain how to change the selection in the tool bar part of this manual), click on Export in the Database menu. In the export window, you can define name of the exported file and its type. If you have minimized structures with Corina, you can choose to export only correctly minimized structures.

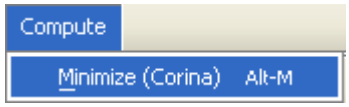
If a number is entered in the field "Diversity: number of compounds to select", then a diversity algorithm will be used to select a number of compounds close to the number entered (the number of compounds exported by diversity can't be specified exactly). Diversity algorithm is quick for a small number of compounds, but can be very long for million structures !



Many charts are available in ScreeningAssistant :

- Pie Charts : compounds by providers
  - Compounds Repartition
  - Drug-like
  - Lead-like
- Percentages
  - Drug-like
  - Lead-like
- Distribution
  - Molecular Weight
  - Number of H bond acceptors
  - Number of H bond donors
  - LogP
  - Number of rotatable bonds
  - TPSA
- Internal Duplicates
- Unicity: compound not present in the databases of the other providers
- Frameworks: number of different frameworks by providers
- Diversity: estimated with the number of clusters created by the SCA algorithm with 0.8 cut-off
  - Global
  - Drug-like
  - Lead-like
- Chemical Space: display selected compounds in a chemical space. The axis are defined by the PCA1 and PCA2 axis equations obtained from the PCA analysis of 18 000 compounds selected by diversity among 2.6 millions.

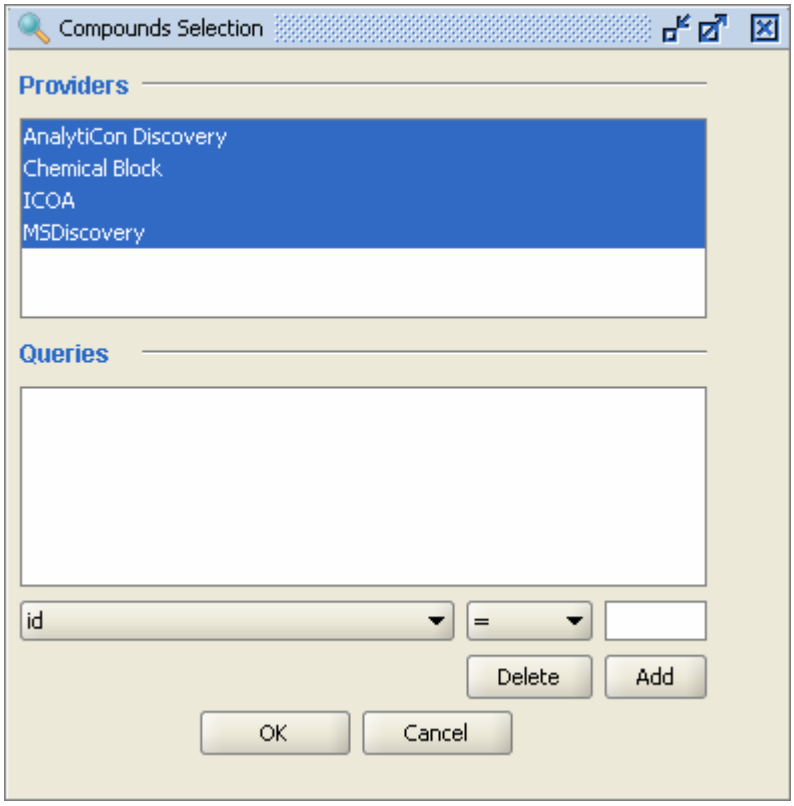
At the present time, only the minimization of the compounds by Corina is available. Of course you need the corina.exe file in your computer to use this option. Once corina.exe put in the ScreeningAssistant\external\corina directory, you can minimize compounds of your databases.



The tool bar: navigate in your database and select compounds

The database viewer displays 1000 molecules by page. You can change the pages using the five blue icons. The  icon allows to directly go to a page entering is number.

You can select desired compounds using the  icon.



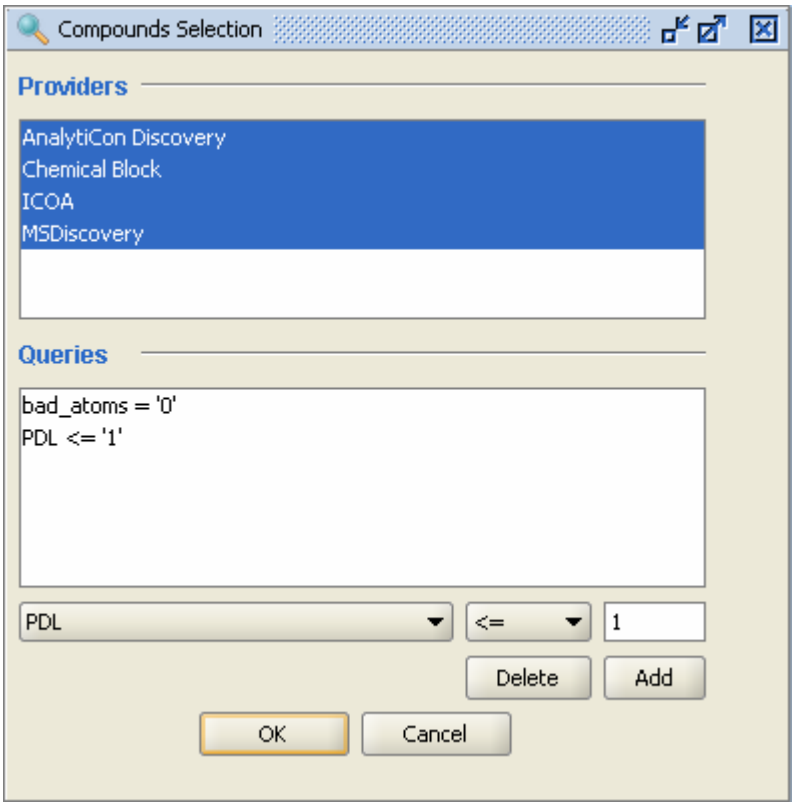
You can select the providers. By default all providers are selected. In the query part of the screen you can define filters for the selection. You must choose the parameter, the equality sign, the value and then click Add to define a new parameter. There is no limit in the number of filters you can define.

Parameters:

- id: internal ID of the compound
- md5ichi: hascode of the unique identifier
- mw: molecular weight of the compounds, without counter-ion(s) and at pH=7
- logP
- TPSA: Topological Polar Surface Area
- Hba: number of H bond acceptors
- Hbd: number of H bond donors
- rotatable bonds
- halogens
- single_chains: single chains longer than ≤ $-(CH_2)_6CH_3$ (0 for no, 1 for yes)
- perfluorinated_chain: presence of $-CF_2CF_2CF_3$ (0 for no, 1 for yes)
- SSSRs: number of Smallest Sets of Smallest Rings
- big_ring_size: size of the bigger ring found by SSSRs
- O: number of O atoms
- N: number of N atoms
- S: number of S atoms
- NO2: number of NO_2
- SO2: number of SO2
- CF3: number of CF_3
- CF3_halogens: number of CF_3 and other halogen atoms
- NOS: number of N, O and S atoms
- bad_atoms: number of atoms other than C, O, N, S, P, F, Cl, Br, I, Na, K, Mg, Ca, or Li
- is_reactive: presence of a reactive function (0 for no, 1 for yes)
- is_warhead: presence of a warhead type substructure (0 for no, 1 for yes)

- is_promiscuous: the molecule is known to be a promiscuous inhibitor (0 for no, 1 for yes)
- is_privileged: if > -1, then a privileged structure is detected in the structure of the molecule
- absorption: not used yet
- caco2: not used yet
- solubility: not used yet
- BBB: not used yet
- drug-like_failures: number of non-fitted drug-like criteria
- lead-like_failures: number of non-fitted lead-like criteria
- PDL_score: our internal Progressive 'Drug-Like' score. ≤ 1 means the compounds is estimated drug-like, more than 2 means the compounds is absolutely not drug-like
- PLL_score: our internal Progressive 'Lead-Like' score. ≤ 1 means the compounds is estimated lead-like, more than 2 means the compounds is absolutely not lead-like
- CFMS: Cleaning For My Screening. By default it is based on PDL and add penalties for the presence of reactive functions, warheads, primiscuous aggregating inhibitors, single chains, perfluorinated chains, and for the abscence of N or O
- scaffold: scaffold id (scaffold used is rings + linkers)
- framework: id of the general 2D shape of the molecule
- ClogP: not used yet.
- entrie_date

Example of a drug-like selection:

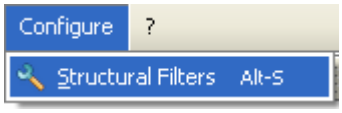


Example of a selection for screening:

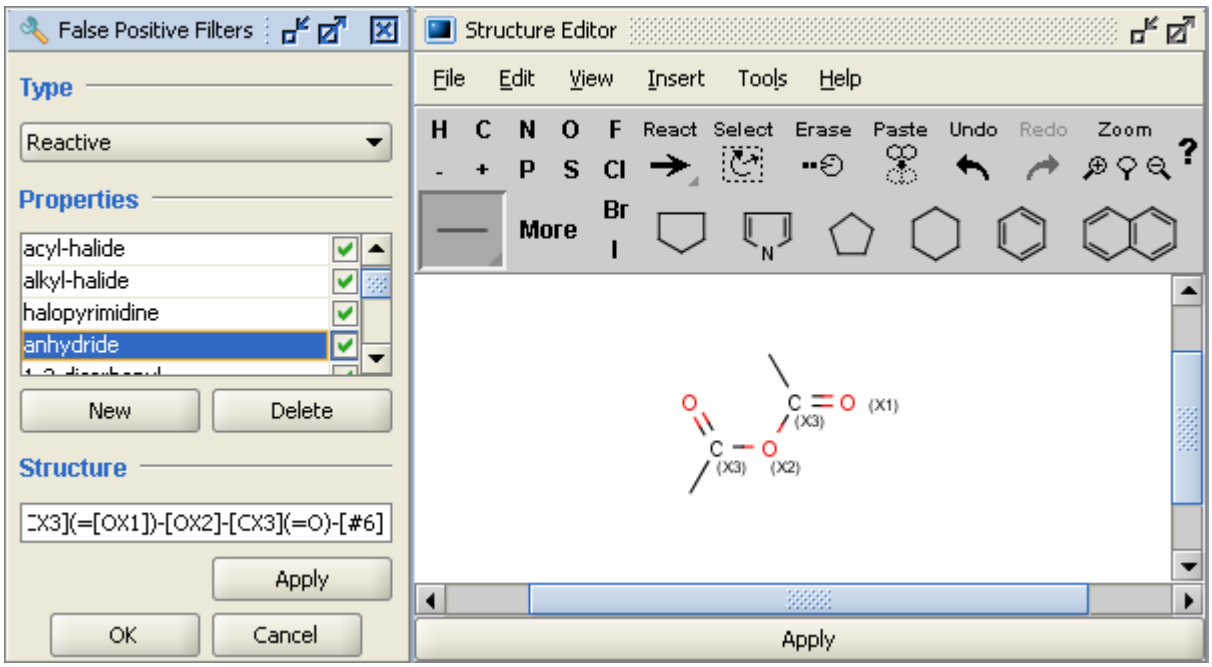
A screening selection can be done in the same way that a drug-like selection, but using the criteria bad_atoms=0 and CFMS=0. By default, CFMS is the same thing that PDL, but it takes into account additional features such as reactive functions.

CFMS can be personalized using the  icon. Then, you can choose the base of CFMS ('drug-like' or 'lead-like' *i.e.* PDL or PLL) the additional penalties can be chosen. Clicking Ok after user's choices will recompute the new CFMS value for all the compounds of the current database.

The reactive functions and warheads penalties can also be personalized through Configure in the main menu of ScreeningAssistant :

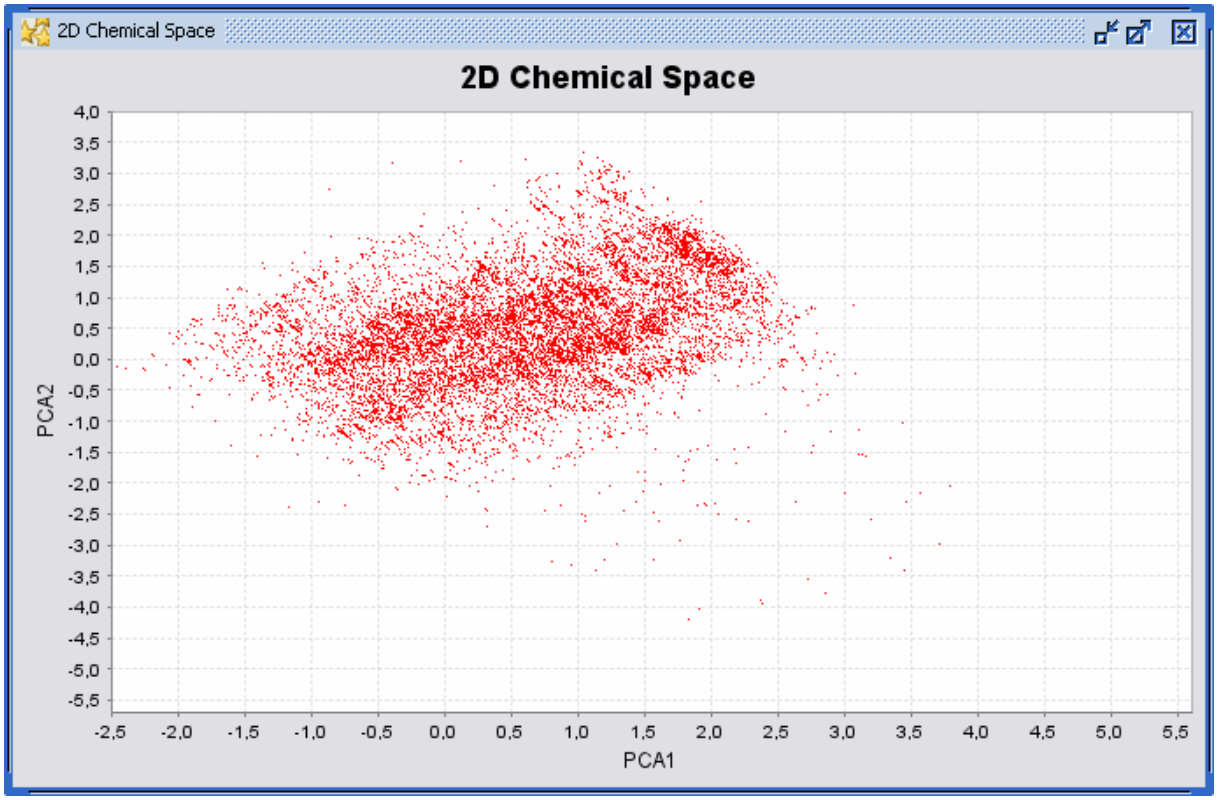


The False Positive Filters window is now open:



This window will allow you to modify definitions of reactive functions and warheads. You can create and delete sub-structures filters using New and Delete buttons. The name of a property can be changed double clicking on it. The Structure field let you to edit the **SMARTS** code of the filter. If you have MarvinBeans.jar in your classpath, the sub-structure can be edited graphically.
The modification of the filters will apply when new compounds are added.

2D chemical space covered by the selected compounds can be viewed using the  icon. Here is an example of chemical space:



This chemical space use the PCA1 and PCA2 axis equation computed on 18 000 diverse molecule selected from 2.6 millions with the descriptors used are SSKey3DS, MW and logP.