Modélisation de la stéréochimie : une application à la chémoinformatique

Pierre-Anthony Grenier

UNIVERSITÉ CAEN NORMANDIE

Normandie Université

Université de Caen Normandie

École doctorale SIMEM

Thèse de doctorat

présentée et soutenue le : 26 Novembre 2015

par

Pierre-Anthony Grenier

pour obtenir le

Doctorat de l'Université de Caen Normandie
Spécialité : Informatique

Modélisation de la stéréochimie : une application à la chémoinformatique

Directeur de Thèse : Luc Brun
Co-directeur de Thèse : Didier Villemin

Jury

Rapporteurs

Amedeo Napoli	Directeur de Recherche, CNRS, Nancy
Christine Solnon	Professeur des Universités, INSA de Lyon

Examinateurs

Michel Petitjean	Chargé de Recherche HDR, Université Paris 7
Salvatore Antoine Tabbone	Professeur des Universités, Université de Lorraine
Luc Brun	Professeur des Universités, ENSICAEN
Didier Villemin	Professeur des Universités, ENSICAEN

Remerciements

Tout d'abord, je tiens à remercier mes directeurs de thèse. Je remercie Luc Brun pour son encadrement, sa patience ainsi que sa capacité à me comprendre et à m'aider à mettre mes idées au clair. Je remercie également Didier Villemin pour son encadrement et pour toutes les choses qu'il m'a appris.

Je remercie également mes rapporteurs Amedeo Napoli et Christine Solnon, pour avoir lu mon manuscrit et pour m'avoir permis d'en améliorer la forme et le contenu grâce à leurs remarques pertinentes. Je remercie aussi Salvatore Antoine Tabbone et Michel Petitjean pour avoir accepté de faire partie de mon jury.

Je remercie aussi l'ensemble de l'équipe image pour m'avoir accepté malgré le fait que ma thèse soit en chémoinformatique. Merci à Régis, Olivier, Jalal, David, Myriam, Stéphanie et Sébastien qui furent de bons enseignants avant de devenir de bons collègues. Merci aux autres permanents, Loïc, Sébastien, Julien, Abder et Amal pour leur bonne humeur. Et finalement merci à tous les autres doctorants et post-doc avec qui j'ai partagé ma thèse : Benoît, Pierre, Daniel, Jaume, Moncef, Xavier, Saleh, François, Youssef, Anas et les deux nouveaux docteurs Maxime et Matthieu.

Finalement je tiens à remercier ma mère Cécile, mon père Bernard et mes deux soeurs Sarah et Aurore pour leur soutien. Merci aussi à tous mes oncles, tantes, cousins, cousines et à mes deux grand mères. Et finalement merci à tous mes amis pour avoir toujours été là pour moi. Je remercie particulièrement Gwen, Félix, Hibou, Clément, Guillaume, Romain (même si bon ...), Thomas, Ingrid et Nils pour avoir relus mon manuscrit et/ou pour avoir assisté à ma soutenance.

Table des matières

Introduction générale

La chémoinformatique est une discipline scientifique qui consiste à utiliser l'informatique pour résoudre des problèmes de chimie. Les méthodes informatiques permettent de traiter de grands nombres de données issues de la chimie. Les principales applications de la chémoinformatique sont :

- La gestion de base de données de molécules et de réactions.
- La détermination de la structure des molécules.
- La fouille de données moléculaires et de bases de réactions [PN06].
- La prédiction de réactions chimiques.
- La prédiction de propriétés physiques, chimiques ou biologiques de molécules.

Cette dernière application permet de réduire le temps et le coût de la conception de nouvelles molécules. Elle est notamment utilisée afin de concevoir de nouveaux médicaments.

Afin de créer un nouveau médicament, les chercheurs commencent par définir quelles propriétés la molécule doit avoir afin d'être efficace. Puis ils définissent des ensembles de molécules candidates, qui sont synthétisées et testées jusqu'à trouver un ensemble vérifiant les propriétés désirées. Cette étape est appelée étape de criblage. De nombreuses molécules sont criblées, ce qui rend cette étape longue et coûteuse. Le nombre de molécules testées lors du criblage est estimé à environ dix mille, ce qui induit un coût estimé à environ un milliard de dollars [Ng15]. Après cette étape, de nombreuses phases de tests, sur des volontaires sains pour vérifier que le médicament n'a pas d'effets secondaires et sur des groupes de malades pour vérifier l'efficacité du médicament, sont alors effectuées afin de valider le nouveau médicament.

La prédiction de propriétés de molécules n'est cependant pas limitée à la conception de médicaments. Elle peut être utilisée pour n'importe quel

problème de conception de nouveau composé, comme la création de molécules captant le CO_2. Dans tous les cas, une étape de criblage sera nécessaire, et les systèmes de prédiction de propriétés permettent d'accélérer cette étape. On parle alors de criblage virtuel. L'usage de l'outil informatique permet de sélectionner quelles molécules auront le plus de chance d'avoir les propriétés recherchées. Ceci permet de ne synthétiser que les molécules les plus prometteuses, réduisant ainsi les coûts liés à cette étape.

Les sous domaines de la chémoinformatique dont le but est de prédire les propriétés des molécules sont appelés QSAR (*Quantitative Structure–Activity Relationship*) et QSPR (*Quantitative Structure–Property Relationship*). Dans ces domaines, on utilise un ensemble de molécules dont on connaît une ou plusieurs propriétés, afin de construire un système informatique permettant de prédire ces propriétés pour d'autres molécules. Afin de construire un tel système, on se base sur un principe de similarité :

« Des molécules similaires ont des propriétés similaires. »

Ainsi, selon ce principe, les propriétés d'une molécule inconnue sont déduites de sa similarité avec un ensemble de molécules aux propriétés connues. Il faut donc être capable de mesurer la similarité des molécules pour pouvoir prédire leurs propriétés.

Afin d'étudier les molécules avec des outils informatiques, il faut un modèle informatique pour les représenter. Différents modèles de représentation existent, chacun pouvant coder différents niveaux d'informations. On considère que deux molécules sont similaires si leur modèle sont similaires. Ainsi, la mesure de similarité entre molécules dépend du modèle choisi et il faut utiliser un modèle cohérent avec la propriété à prédire.

Le modèle le plus simple, permettant de représenter une molécule, est sa formule brute. Cependant, cette représentation ne code pas les liaisons entre les atomes. Ainsi, des molécules, appelées isomères, peuvent avoir une même formule brute mais être différentes. La plupart des propriétés chimiques de deux isomères sont différentes et comme leurs formule brute ne peuvent pas les différencier, ce modèle est peu adapté à la prédiction de propriétés moléculaires.

Afin de remédier à la limitation de la formule brute, une molécule peut être représentée par son graphe moléculaire. Un graphe est un composé d'un ensemble de sommets et d'un ensemble d'arêtes, reliant les sommets. Les graphes peuvent donc naturellement représenter des molécules qui sont composées d'atomes liés entre eux par des liaisons chimiques, et contrairement aux formules brutes, ils peuvent différencier les isomères.

Afin de définir une mesure de similarité entre molécules, de nombreuses méthodes de chémoinformatique utilisent une représentation vectorielle des

molécules. Un exemple de mesure de similarité entre deux vecteurs est leur produit scalaire. Ces vecteurs contiennent différentes caractéristiques de la molécule, par exemple le nombre d'atomes d'un certain type ou le nombre d'occurrences d'un certain motif dans la molécule, pouvant être extraites du graphe moléculaire. Les caractéristiques peuvent être choisies de manière à discerner les isomères. Le plus gros avantage de la représentation vectorielle est que de nombreuses méthodes d'apprentissage automatique peuvent être appliquées sur des vecteurs. Cependant, la taille des vecteurs étant fixée, il faut définir a priori quelles caractéristiques seront codées dans le vecteur. Ainsi, on se focalise davantage sur ce qui est encodé dans les vecteurs plutôt que sur la définition d'une mesure de similarité.

Les noyaux sur graphes peuvent être vus comme des mesures de similarité entre graphes. De plus, s'ils respectent certaines propriétés, les noyaux sur graphes correspondent à des produits scalaires. Grâce à cette propriété, on peut directement utiliser les noyaux sur graphes avec des méthodes classiques d'apprentissage automatique. La définition d'un noyau sur graphe permet de se concentrer d'avantage sur la conception d'une mesure de similarité, plutôt que sur le choix des caractéristiques utilisées pour construire un vecteur. Récemment, plusieurs noyaux sur graphes ont été définis et utilisés en chémoinformatique. Ces méthodes ont obtenu de bons résultats, prouvant la pertinence de cette approche pour la prédiction de propriétés moléculaires.

Dans un graphe moléculaire, à partir d'un noeud, nous n'avons accès qu'à la liste de ses voisins. Ainsi, les graphes moléculaires ont aussi une limitation : ils ne prennent pas en compte la position 3D des atomes au sein de la molécule. En effet, certaines molécules, appelées stéréoisomères, sont représentées par un même graphe moléculaire, mais ont des positionnements relatifs de leurs atomes différents dans l'espace. Les propriétés variant entre les stéréoisomères ne peuvent donc pas être prédites seulement grâce au graphe moléculaire. Ceci peut poser des problèmes car certains stéréoisomères peuvent avoir des propriétés biologiques très différentes. C'est notamment le cas de la thalidomide. Cette molécule fut commercialisée comme médicament pour les femmes enceintes. Cependant, un de ses stéréoisomères provoque des déformations du foetus. La différentiation des stéréoisomères est donc importante dans le cadre de la prédiction de propriétés moléculaires. Il faut un autre modèle afin de pouvoir représenter ces molécules.

L'objectif de cette thèse est de concevoir une méthode de prédiction de propriétés de molécules prenant en compte les stéréoisomères. Il faut en premier lieu construire un modèle qui, contrairement aux graphes, permet de discerner les stéréoisomères. Puis, le second objectif est de concevoir une mesure de similarité entre ces modèles, qui est pertinente pour la prédiction de propriétés

de molécules. Cette mesure de similarité est construite de façon à être un noyau, afin de pouvoir la combiner avec des méthodes d'apprentissage automatique.

Plan du manuscrit

Le chapitre 1 commence par présenter les différentes définitions concernant les graphes qui seront utilisées dans la suite du document, ainsi que le graphe moléculaire et d'autres modèles permettant de représenter les molécules. Puis, il décrit différentes méthodes de prédiction de propriétés moléculaires, en se focalisant sur les méthodes utilisant des noyaux sur graphes. La seconde partie de ce chapitre définit les stéréoisomères et explique en quoi le graphe moléculaire ne peut pas prendre en compte ces molécules. La fin de ce chapitre présente un état de l'art des modèles représentant les stéréoisomères et des méthodes de prédiction de propriétés prenant en compte ces molécules.

Le chapitre 2 définit un modèle permettant de représenter les stéréoisomères : le graphe moléculaire localement ordonné. Le positionnement relatif des atomes qui n'est pas pris en compte dans le graphe moléculaire, est pris en compte en ajoutant une notion d'ordre au voisinage de certains sommets. De plus, ce chapitre présente un isomorphisme entre les graphes localement ordonnés, construit de manière à ce que deux graphes localement ordonnés soient isomorphes si et seulement s'ils représentent un même stéréoisomère. Finalement, la notion de stéréo sommets, qui sont les sommets responsables de la stéréoisomérie, est définie à la fin de ce chapitre.

Les stéréo sommets, présentés dans le chapitre 2 sont définis globalement. Cependant, la totalité du graphe localement ordonné n'est pas forcément nécessaire à la caractérisation d'un stéréo sommet. Ainsi, le chapitre 3 présente la notion de stéréo sous-graphes minimaux, qui sont des sous-graphes des graphes localement ordonnés qui caractérisent localement les stéréo sommets. Un algorithme permettant de calculer ces sous-graphes, ainsi que la preuve de cet algorithme, sont donnés. Les stéréo sous-graphes minimaux sont alors utilisés afin de construire un noyau entre graphes localement ordonnés. L'utilisation d'un noyau permet de combiner cette mesure de similarité avec des méthodes classiques d'apprentissage automatique.

Le chapitre 4 propose trois extensions du noyau présenté dans le chapitre 3. La première permet de prendre en compte les relations entre les stéréo sous-graphes minimaux. La seconde extension ajoute une information sur le voisinage des stéréo sous-graphes minimaux dans le noyau. Finalement, la dernière extension permet de comparer deux stéréo sous-graphes minimaux différents.

En conclusion, nous évoquerons les différentes perspectives ouvertes par cette thèse.

Chapitre 1

Description du contexte applicatif

Sommaire

Comme présenté dans l'introduction, on suppose que des molécules ont des propriétés similaires si elles sont similaires. Il faut donc pouvoir mesurer une similarité entre les molécules pour prédire leurs propriétés. On considère que deux molécules sont similaires si leur modèle sont similaires. Les modèles permettant de coder les molécules sont présentés dans la sous-section 1.1.1. Ensuite, nous présenterons dans la sous-section 1.1.2 les descripteurs moléculaires, qui sont souvent utilisés en chémoinformatique. Puis dans la sous-section 1.1.3, nous présenterons les noyaux sur graphes, qui correspondent à une mesure de similarité entre graphes.

Finalement, nous présenterons dans la sous-section 1.1.4 des méthodes d'apprentissage automatique. Ces méthodes utilisent des mesures de similarité entre molécules afin de prédire leurs propriétés.

La section 1.2 présente le problème posé par les molécules appelées stéréoisomères. Afin de représenter ces molécules, il faut pouvoir prendre en compte le positionnement relatif des atomes, ce qui n'est pas le cas des modèles présentés dans la sous-section 1.1.1.

La sous-section 1.2.1 commencera par définir ce que sont les stéréoisomères. Puis, comme dans la section 1.1, la sous-section 1.2.2 présentera les modèles permettant de représenter les stéréoisomères. Enfin, la sous-section 1.2.3 concernera les méthodes de prédiction de propriétés moléculaires qui prennent en compte les stéréoisomères.

1.1 Prédiction des propriétés moléculaires

1.1.1 Modèles de représentation des molécules

Une molécule est un assemblage chimique d'atomes, liés entre eux par des liaisons covalentes. Si des atomes ont le même numéro atomique (c'est-à-dire un même nombre de protons), ils représentent un même élément chimique. Les liaisons covalentes ont un ordre, identifiant le nombre d'électrons impliqués dans la liaison. Selon cet ordre, les liaisons sont appelées liaisons simples, doubles, triples ou aromatiques.

Un graphe est un ensemble de sommets liés entre eux par des arêtes. Ces sommets et ces arêtes peuvent avoir des étiquettes. Ainsi, une molécule peut être naturellement représentée par un graphe. Dans ce cas, le graphe est appelé graphe moléculaire.

Définitions et notations

Afin de définir formellement un graphe moléculaire, nous allons dans un premier temps donner les définitions qui seront utilisées au cours de ce manuscrit.

Définition 1. Graphe orienté

Un graphe orienté $G = (V, E)$ est un couple composé de deux ensembles. Un premier ensemble V de sommets et un second ensemble $E \subset V \times V$ d'arcs entre ces sommets. Les arcs sont des couples ordonnés de sommets $(u, v) \in V \times V$ indiquant une liaison de u vers v.

Définition 2. Graphe non orienté

Un graphe non orienté $G = (V, E)$ est un couple composé de deux ensembles. Un premier ensemble V de sommets et un second ensemble $E \subset \mathcal{P}_2(V)$ d'arêtes entre ces sommets, où $\mathcal{P}_2(V)$ est l'ensemble des parties de taille 2 de V. Les arêtes sont des ensembles non ordonnés de deux sommets $\{u, v\} \in \mathcal{P}_2(V)$ indiquant une liaison entre u et v.

Dans tout ce manuscrit, un ensemble non ordonné sera entouré par des accolades $\{a_1, \ldots, a_n\}$ et un ensemble ordonné sera entouré par des parenthèses $(a_1, \ldots, a_n)$.

Les prochaines définitions sont données pour un graphe non orienté, mais sont aussi applicables aux graphes orientés. Le terme arête est réservé aux graphes non orientés et celui d'arc aux graphes orientés.

Définition 3. Arête incidente

Soit un graphe $G = (V, E)$. Une arête $e \in E$ est dite incidente à un sommet $v \in V$ si $v \in e$. On dit alors que v est une des extrémités e.

Définition 4. Adjacence de sommets

Soit un graphe $G = (V, E)$. Un sommet $v \in V$ est dit adjacent à un sommet $u \in V$, s'il existe une arête $e \in E$ telle que $e = \{u, v\}$.

Définition 5. Boucle

Soit un graphe $G = (V, E)$. Une boucle est une arête $e \in E$ reliant un sommet $v \in V$ à lui-même : $e = \{v, v\}$.

Définition 6. Arête multiple

Soit un graphe $G = (V, E)$. Une arête multiple est un ensemble d'arêtes $\{e_1, \ldots, e_n\} \subset E$ tel que chaque arête possède les mêmes extrémités $\{u, v\} \in V^2$: $e_1 = \{u, v\}, \ldots, e_n = \{u, v\}$.

Définition 7. Graphe fini

Un graphe $G = (V, E)$ est dit fini si son nombre de sommets $|V|$ est fini.

Définition 8. Graphe simple

Un graphe $G = (V, E)$ est dit simple si il ne possède ni boucle ni arête multiple.

Définition 9. Graphe étiqueté

Un graphe étiqueté $G = (V, E, \mu, \nu)$ est un graphe (V, E) avec deux fonctions d'étiquetage μ et ν. La fonction $\mu : V \to \mathcal{L}_V$ associe à chaque sommet v du graphe une étiquette $\mu(v)$ et la fonction $\nu : E \to \mathcal{L}_E$ associe à chaque arête e du graphe une étiquette $\nu(e)$.

Les ensembles d'étiquettes $\mathcal{L}_V$ et $\mathcal{L}_E$ peuvent être de n'importe quel type (chaîne de caractères, entier, vecteur ...).

Sauf mention contraire, dans la suite de ce manuscrit on désignera un graphe simple, fini, non orienté et étiqueté par le terme de graphe.

Définition 10. Voisinage

Soit un graphe $G = (V, E, \mu, \nu)$. Le voisinage d'un sommet $v \in V$, dénoté $N(v)$, est l'ensemble des sommets adjacents à v :

$$N(v) = \{u \in V | \exists e = \{u, v\} \in E\}$$

Étant donné un ensemble de sommets $S \subseteq V$, on note $N(S)$ l'ensemble des voisins des sommets de S :

$$N(S) = \bigcup_{u \in S} N(u)$$

Définition 11. Degré d'un sommet

Soit un graphe $G = (V, E, \mu, \nu)$. Le degré d_v d'un sommet $v \in V$ est la taille de son voisinage.

$$d_v = |N(v)|$$

Définition 12. Chemin

Soit un graphe $G = (V, E, \mu, \nu)$. Un chemin p est une séquence ordonnée de sommets $(v_1, \ldots, v_n)$ telle qu'il existe une arête entre deux sommets consécutifs :

$$\forall i \in \{1 \ldots, n-1\}, \exists e \in E \ t.q \ e = \{v_i, v_{i+1}\}$$

La longueur d'un chemin est définie par son nombre d'arêtes. Le chemin $p = (v_1, \ldots, v_n)$ est donc de longueur $n - 1$.

Définition 13. Distance entre sommets

Soit un graphe $G = (V, E, \mu, \nu)$. La distance $d(u, v)$ entre deux sommets $u \in V$ et $v \in V$ est la longueur du plus court chemin entre u et v.

Définition 14. Chemin simple

Soit un graphe $G = (V, E, \mu, \nu)$. Un chemin $p = (v_1, \ldots, v_n)$ est dit simple si toutes ses arêtes sont distinctes :

$$\forall (i, j) \in \{1, \ldots, n-1\}^2, i \neq j \iff e_i = \{v_i, v_{i+1}\} \neq e_j = \{v_j, v_{j+1}\}$$

Définition 15. Chemin élémentaire

Soit un graphe $G = (V, E, \mu, \nu)$. Un chemin $p = (v_1, \ldots, v_n)$ est dit élémentaire si tous ses sommets sont distincts :

$$\forall (i, j) \in \{1, \ldots, n\}^2, i \neq j \iff v_i \neq v_j$$

Tout chemin élémentaire est aussi un chemin simple.

Définition 16. Cycle

Soit un graphe $G = (V, E, \mu, \nu)$. Un cycle est un chemin simple $p = (v_1, \ldots, v_n)$ dont les deux extrémités sont identiques : $v_1 = v_n$.

Si le chemin est élémentaire (sauf pour les extrémités), alors le cycle est appelé cycle élémentaire.

Un graphe ne possédant aucun cycle est dit acyclique.

Définition 17. Sous-graphe

Soit un graphe $G = (V, E, \mu, \nu)$. Un graphe $H = (V_H, E_H, \mu_H, \nu_H)$ est un sous-graphe de G si H est contenu dans G, c'est-à-dire $V_H \subset V$, $E_H \subset E$, $\mu_H = \mu_{|V_H}$ et $\nu_H = \nu_{|E_H}$.

Dans tout ce manuscrit la notation V_H désignera l'ensemble des sommets du sous-graphe H.

Définition 18. Sous-graphe induit

Soit un graphe $G = (V, E, \mu, \nu)$ et $H = (V_H, E_H, \mu_H, \nu_H)$ un sous-graphe de G. On dit que H est un sous-graphe induit de G si pour tout couple de sommets de H une arête existe entre ces sommets dans H si et seulement si elle existe dans G :

$$E_H = E \cap \mathcal{P}_2(V_H)$$

L'ensemble des sommets V_H d'un sous-graphe induit H de G suffit à le décrire. On dit alors que H est le sous-graphe induit de G par V_H.

Soit H un sous-graphe de G (ou V' un sous-ensemble de V). La notation $G - H$ (ou $G - V'$) désigne le sous-graphe induit de G obtenu en supprimant l'ensemble des sommets de H (ou en supprimant les sommets de V').

Définition 19. Isomorphisme de graphes *Soit deux graphes $G_1 = (V_1, E_1, \mu_1, \nu_1)$ et $G_2 = (V_2, E_2, \mu_2, \nu_2)$. Une fonction f est un isomorphisme entre G_1 et G_2 si c'est une bijection entre les sommets qui préserve les arêtes et les étiquettes. Formellement, f est une fonction de V_1 vers V_2 respectant les conditions suivantes :*

1. $\forall u \in V_2, \exists! v \in V_1 \ t.q \ u = f(v)$
2. $\forall v \in V_1, \mu_1(v) = \mu_2(f(v))$
3. $\forall \{u, v\} \subseteq V_1, \ \{u, v\} \in E_1 \iff \{f(u), f(v)\} \in E_2$
4. $\forall \{u, v\} \in E_1, \nu_1(\{u, v\}) = \nu_2(\{f(u), f(v)\})$

S'il existe un isomorphisme f entre deux graphes G_1 et G_2, ces graphes sont dit isomorphes. On note $Isom(G_1, G_2)$ l'ensemble des isomorphismes entre G_1 et G_2. La relation "G_1 est isomorphe à G_2" est une relation d'équivalence. On note cette relation $G_1 \simeq G_2$.

Étant donné un ensemble de sommets $S \subseteq V_1$, et un isomorphisme f entre deux graphes G_1 et G_2, on note $f(S)$ l'ensemble des sommets associés à des sommets de S par f :

$$f(S) = \{f(u) | u \in S\}$$

Définition 20. Graphe connexe

Un graphe $G = (V, E, \mu, \nu)$ est dit connexe si pour tout couple de sommets $\{u, v\} \in \mathcal{P}_2(V)$ il existe un chemin entre u et v.

Définition 21. Arbre

Un arbre est un graphe acyclique et connexe.

Définition 22. Arbre enraciné

Un arbre enraciné est un arbre dont l'un des sommets est distingué et appelé racine de l'arbre.

Un arbre étant connexe et acyclique, il existe pour tout sommet v un unique chemin élémentaire $p = (v_1, \ldots, v_n)$ reliant $r = v_1$ à $v = v_n$. Si v est différent de la racine, le père de v est l'unique sommet v_{n-1} situé avant v dans le chemin le reliant à la racine (ce sommet peut être la racine elle-même). Le sommet v est appelé fils de v_{n-1}.

Définition 23. Permutation

Soit X un ensemble. Une permutation φ sur X est une bijection de X sur lui-même.

L'orbite $\mathcal{O}_\varphi(x)$ d'un élément x selon une permutation φ est l'ensemble de ses images successives obtenues par applications répétées de φ :

$$\mathcal{O}_\varphi(x) = \{\varphi^k(x) \mid k \in \mathbb{N}\}$$

Si l'ensemble X est fini, alors pour tout élément x de X l'orbite de x est fini et donc il existe k tel que $\varphi^k(x) = x$. De plus les orbites des éléments de X forment une partition $(X_1, X_2, \ldots, X_k)$ de X. Pour chaque sous-ensemble X_i de cette partition on note x_i un de ses éléments et l_i sa taille. La restriction de φ sur chaque X_i définit une permutation notée $\left(x_i, \varphi(x_i), \varphi^2(x_i), \ldots, \varphi^{l_i}(x_i)\right)$ appelée un cycle. La permutation φ peut donc être définie comme la composition de ses cycles sur chacun des X_i :

$$\varphi = \prod_{i=1}^{k} \left(x_i, \varphi(x_i), \varphi^2(x_i), \ldots, \varphi^{l_i}(x_i)\right)$$

Une permutation uniquement composée d'un cycle de taille 2 est appelée transposition.

Définition 24. Parité et signature d'une permutation

Soit φ une permutation de n éléments. La permutation φ est dite paire si elle peut être exprimée comme le produit d'un nombre pair de transpositions. De la même manière, φ est dite impaire si elle peut être exprimée comme le produit d'un nombre impair de transpositions. La signature $\epsilon(\varphi)$ d'une permutation paire est 1, et celle d'une permutation impaire est -1.

Définition 25. Groupe

Un groupe est un couple composé d'un ensemble G et d'une loi de composition $\bullet$, qui associe à deux éléments a et b de G un autre élément noté $a \bullet b$, respectant les quatre axiomes suivant :

- $\forall \{a, b\} \subset G$, $a \bullet b \in G$.
- $\forall \{a, b, c\} \subset G$, $(a \bullet b) \bullet c = a \bullet (b \bullet c)$.
- *Il existe un élément e dans G, appelé élément neutre, tel que pour tout a de G on ait : $e \bullet a = a \bullet e = a$.*
- *Soit e l'élément neutre de G, pour tout a de G il existe b dans G tel que : $a \bullet b = b \bullet a = e$.*

On dit que G est un groupe pour la loi de composition $\bullet$.

Graphe moléculaire

Le graphe moléculaire d'une molécule (page 8) est défini comme un graphe simple, non orienté et étiqueté où :

1. Chaque atome de la molécule est représenté par un sommet.
2. Chaque liaison chimique reliant une paire d'atomes est représentée par une arête incidente aux deux sommets représentant ces deux atomes.
3. Chaque sommet est étiqueté par une chaîne de caractères (par ex. “C”,“H”,“Br”, ...) représentant l'élément chimique de l'atome qu'il représente.
4. Chaque arête est étiquetée par un caractère représentant son ordre.

Pour représenter graphiquement un graphe moléculaire, plusieurs conventions sont utilisées (Figure 1.1).

1. L'étiquette de chaque noeud est représentée sauf pour les carbones.

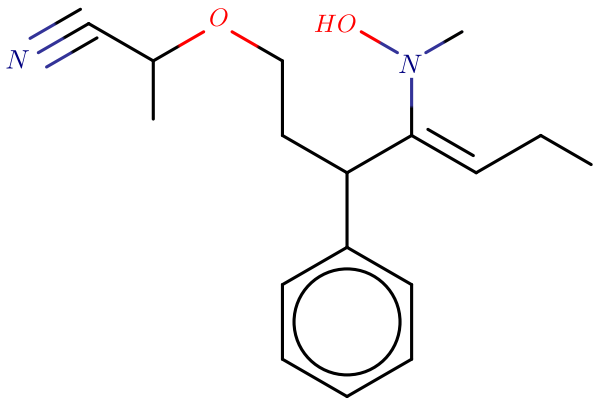


FIGURE 1.1 – Représentation graphique de la molécule : $C_{17}H_{24}N_2O_2$. Les extrémités des lignes brisées représentent les atomes de carbone. Les atomes d'hydrogène sont implicitement représentés par les liaisons manquantes. Les trois traits parallèle incident à l'atome d'azote (en haut à gauche) représentent une triple liaison. De même les deux traits parallèle à droite représentent une double liaison. Finalement, le cercle à l'intérieur de l'hexagone représente des liaisons aromatiques.

2. Les atomes d'hydrogène ne sont pas représentés lorsqu'ils sont liés à un carbone. Leur présence est déduite du nombre de liaisons du carbone. Si un hydrogène est lié à un autre atome qu'un carbone, la liaison entre l'hydrogène et cet atome n'est pas représentée (voir l'oxygène à droite de la Figure 1.1).

3. Les arêtes encodant des liaisons simples, doubles et triples sont respectivement représentées par un, deux ou trois traits.

4. Les liaisons aromatiques sont toujours présentes dans des cycles. Elles peuvent être représentées de deux façons. Soit par une alternance de trait simple et double, soit par un cercle à l'intérieur du cycle. Dans ce manuscrit nous utiliserons la seconde notation.

Autres représentations

D'autres modèles permettent de représenter les molécules. On peut par exemple représenter une molécule par sa formule brute. Ce modèle est constitué de la liste des atomes composant une molécule, accompagnée du nombre

d'occurrences de ceux-ci. Cette représentation est plus simple que le graphe moléculaire, mais code moins d'informations.

Au sein d'une molécule, les atomes peuvent subir des rotations autour des liaisons simples. Les différentes formes que peut prendre une molécule grâce à ces rotations sont appelées conformères. Parmi les conformères d'une molécule, il y en a des plus stables que d'autres. Les méthodes de modélisation moléculaire [Lea01] permettent de calculer les coordonnées cartésiennes des atomes d'une molécule pour une conformation donnée. On peut donc représenter une molécule par la liste de ses atomes, accompagnés de leurs coordonnées dans la conformation la plus stable. Cette représentation peut être couplée au graphe moléculaire, afin d'avoir la position des atomes ainsi que leurs relations d'adjacence.

Afin d'illustrer ces différents modèles de représentation, nous prenons l'exemple du dichlorométhane :

- Formule brute : CH_2Cl_2.
- Graphe moléculaire : $G = (V, E, \mu, \nu)$ avec
  - $V = \{v_1, v_2, v_3, v_4, v_5\}$
  - $E = \{e_1 = \{v_1, v_2\}, e_2 = \{v_1, v_3\}, e_3 = \{v_1, v_4\}, e_4 = \{v_1, v_5\}\}$
  - $\mu(v_1) = \mathrm{C}$, $\mu(v_2) = \mu(v_3) = \mathrm{Cl}$ et $\mu(v_4) = \mu(v_5) = \mathrm{H}$
  - $\nu(e_1) = \nu(e_2) = \nu(e_3) = \nu(e_4) = 1$
- Coordonnées des atomes :

C	0.00	0.00	0.00
H	0.00	0.66	0.89
H	0.00	0.66	-0.89
Cl	-1.46	-0.99	0.00
Cl	1.46	-0.99	0.00

1.1.2 Descripteurs moléculaires

Une méthode très utilisée en chémoinformatique afin d'avoir une mesure de similarité entre molécules est de représenter une molécule par un ensemble de valeurs, appelées descripteurs moléculaires. Si ces valeurs sont représentées dans un vecteur, alors un produit scalaire entre ces vecteurs donne une mesure de similarité entre les molécules.

La définition d'un descripteur moléculaire donnée par Todeschini et Consonni [TC00] permet de discerner deux grandes familles de descripteurs : "Un descripteur moléculaire est le résultat final d'une procédure logique et mathématique

qui transforme des informations chimiques codées par une représentation symbolique d'une molécule en un nombre utile, ou le résultat d'une expérience standardisée. “ Il y a donc des descripteurs issus de mesures expérimentales, et des descripteurs issus d'analyses sur différents modèles représentant une molécule.

Selon le modèle utilisé pour représenter une molécule, on pourra obtenir différents descripteurs moléculaires. Ainsi, on peut extraire de la formule brute d'une molécule son nombre d'atomes ou son nombre d'atomes de carbones. Les descripteurs moléculaires issus d'analyses sur le graphe moléculaire sont appelés descripteurs 2D. Des exemples de ces descripteurs sont les indices de Zagreb M_1 et M_2 [GT72] qui correspondent à la somme des carrés des degrés de tous les sommets du graphe moléculaire et la somme des produits des degrés de chaque paire de sommets adjacents :

$$M_1(G) = \sum_{v \in V} d_v^2 \tag{1.1}$$

$$M_2(G) = \sum_{\{u,v\} \in E} d_v d_u \tag{1.2}$$

Certains descripteurs moléculaires utilise la structure 3D d'une molécule. Un exemple de ces descripteurs sont ceux issus de la méthode CoMFA [CPB88]. Cette méthode consiste à aligner toutes les molécules d'une base dans une grille. Une molécule est alors décrite par la valeur du champ électrique qu'elle crée en chaque point de la grille.

1.1.3 Noyaux sur graphes

Définition d'un noyau

Soit $\mathcal{X}$ un ensemble d'objets (par exemple un ensemble de graphes). Un noyau est une fonction $k : \mathcal{X}^2 \to \mathbb{R}$ qui associe à deux objets x et x' une valeur réelle qui correspond à une mesure de similarité entre ces deux objets.

Définition 26. Noyau défini positif

Soit un noyau $k : \mathcal{X}^2 \to \mathbb{R}$. k est dit défini positif sur $\mathcal{X}$ s'il est symétrique :

$$\forall (x_1, x_2) \in \mathcal{X}^2, k(x_1, x_2) = k(x_2, x_1)$$

et s'il est semi-défini positif, c'est-à-dire :

$$\forall n \in \mathbb{N}, \begin{array}{c} \forall \{x_1, \ldots, x_n\} \subset \mathcal{X} \\ \forall \{a_1, \ldots, a_n\} \subset \mathbb{R} \end{array} \Bigg) \sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j k(x_i, x_j) \geq 0$$

Le théorème suivant, formulé et prouvé dans [Aro50], établit que, sous réserve qu'un noyau soit défini positif, il correspond à un produit scalaire dans un espace de Hilbert :

Théorème 1. *Soit un noyau $k : \mathcal{X}^2 \to \mathbb{R}$. Si k est défini positif alors il existe un espace de Hilbert $\mathcal{H}$, muni d'un produit scalaire $\langle ., . \rangle_{\mathcal{H}}$ et une application $\Phi : \mathcal{X} \to \mathcal{H}$ tel que :*

$$\forall (x, x') \in \mathcal{X}^2, k(x, x') = \langle \Phi(x), \Phi(x') \rangle_{\mathcal{H}}$$

L'intérêt de ce théorème est que l'on peut disposer d'un produit scalaire entre les plongement de deux objets x et x', dans un espace de Hilbert $\mathcal{H}$, sans avoir à calculer explicitement la fonction de plongement $\Phi : \mathcal{X} \to \mathcal{H}$ dans cet espace. De plus, s'il n'est pas possible de définir directement un produit scalaire entre deux objets (par exemple si ces objets sont des graphes), le plongement implicite fourni par les noyaux permet d'associer un produit scalaire à ces objets. Un algorithme d'apprentissage utilisant uniquement les produits scalaires entre les objets (par exemple les machines à vecteur de support) pourra donc être utilisé en remplaçant le produit scalaire par un noyau défini positif. Cette méthode est appelée astuce du noyau.

Définition 27. Matrice de Gram

Soit un noyau défini positif $k : \mathcal{X}^2 \to \mathbb{R}$ et $X = (x_1, \ldots, x_N) \in \mathcal{X}^N$. On appelle matrice de Gram de k sur l'ensemble X, la matrice K de taille $N \times N$ telle que :

$$\forall (i, j) \in \{1 \ldots, N\}^2, K_{ij} = k(x_i, x_j)$$

Une propriété des matrices de Gram est que pour tout ensemble d'objets $X = (x_1, \ldots, x_N) \in \mathcal{X}^N$, la matrice de Gram K associée à un noyau défini positif k est semi-définie positive. Si de plus pour tout ensemble d'objets $X = (x_1, \ldots, x_N) \in \mathcal{X}^N$, la matrice de Gram K associée à un noyau k est semi-définie positive, alors k est défini positif.

Afin de prouver qu'un noyau est semi-défini positif, on peut utiliser les propriétés de combinaison des noyaux semi-définis positifs [BCR84] :

Proposition 1. *Soit deux noyaux k_1 et k_2 définis positifs de $\mathcal{X}^2$ dans $\mathbb{R}$. On a alors :*

1. *Soit $a, b \in \mathbb{R}_+$, alors $k = ak_1 + bk_2$ est défini positif.*
2. *$k = k_1 k_2$ est défini positif.*
3. *Soit $k_X : \mathcal{X}^2 \to \mathbb{R}$ et $k_Y : \mathcal{Y}^2 \to \mathbb{R}$ deux noyaux définis positifs. Leur produit de tenseurs $k_X \otimes k_Y : (\mathcal{X}, \mathcal{Y})^2 \to \mathbb{R}$, défini par :*

$$k_X \otimes k_Y(x_1, y_1, x_2, y_2) = k_X(x_1, x_2) k_Y(y_1, y_2)$$

est défini positif.

Noyaux de convolution et noyaux d'appariement

Les noyaux de convolution et d'appariement [Hau99, SK08] sont des noyaux entre objets, fondés sur une décomposition de ces objets en parties.

Quand un objet $x \in \mathcal{X}$ est décomposable en parties $x_1, \ldots, x_D$, où x_d est dans un ensemble $\mathcal{X}_d$ pour tout $d \in \{1, \ldots, D\}$, le noyau de R-convolution, défini par [Hau99], est construit à partir de sous noyaux définis sur ses parties. Afin de définir ce noyau, on représente la relation "$x_1, \ldots, x_D$ sont des parties de x" par une relation R sur l'ensemble $\mathcal{X}_1 \times \ldots \times \mathcal{X}_D \times \mathcal{X}$, telle que $R(x_1, \ldots, x_D, x)$ est vraie si et seulement si $x_1, \ldots, x_D$ sont des parties de x. $R^{-1}(x) = \{x_1, \ldots, x_D | R(x_1, \ldots, x_D, x)\}$ représente alors l'ensemble des parties de x.

Définition 28. Noyau de R-convolution

Soit $x, y \in \mathcal{X}^2$ deux objets décomposables. Pour tout $d \in [1, \ldots, D]$ on suppose que l'on a un noyau défini positif k_d sur $\mathcal{X}_d \times \mathcal{X}_d$. Le noyau de R-convolution K entre x et y est alors défini par :

$$K(x, y) = \sum_{\substack{x_1, \ldots, x_D \in R^{-1}(x) \\ y_1, \ldots, y_D \in R^{-1}(y)}} \prod_{d=1}^{D} k_d(x_d, y_d)$$

Proposition 2. *Le noyau de R-convolution (Définition 28) est défini positif [Hau99].*

On considère maintenant le cas où D vaut 1. Soit x un objet de $\mathcal{X}$. On note $\mathcal{X}'_x$ l'ensemble de ses parties x' :

$$\mathcal{X}'_x = \{x' \in \mathcal{X}' | R(x', x)\}$$

Le noyau de R-convolution s'écrit alors :

$$K(x,y) = \sum_{(x',y') \in \mathcal{X}'_x \times \mathcal{X}'_y} k(x',y')$$

où k est un noyau sur $\mathcal{X}' \times \mathcal{X}'$.

Le noyau de R-convolution permet donc de comparer l'ensemble des paires (x', y') de parties de deux objets x et y. Cependant, dans certains cas, toutes les paires de parties ne sont pas pertinentes et produisent un bruit dans la mesure de similarité. Le noyau d'appariement, défini par [SK08], permet de ne comparer qu'un sous ensemble de ses paires. Ce sous ensemble est défini par un système d'appariement.

Définition 29. Système d'appariement transitif

Un système d'appariement $\mathcal{M}$ est un triplet $(\mathcal{X}, \{\mathcal{X}'_x | x \in \mathcal{X}\}, \{M_{x,y} \subseteqq \mathcal{X}'_x \times \mathcal{X}'_y | (x,y) \in \mathcal{X}^2\})$ tel que $|M_{x,y}| < \infty$ et $(x',y') \in M_{x,y}$ si $(y',x') \in M_{y,x}$.

Un système d'appariement $\mathcal{M}$ est dit transitif si et seulement si :

$$(x'_1, x'_2) \in M_{x_1,x_2} \wedge (x'_2, x'_3) \in M_{x_2,x_3} \implies (x'_1, x'_3) \in M_{x_1,x_3}$$

Proposition 3. *Soit k un noyau défini positif sur $\mathcal{X}' \times \mathcal{X}'$ et un système d'appariement $\mathcal{M}$.*

On appelle noyau d'appariement le noyau défini par :

$$k_m(x,y) = \sum_{(x',y') \in M_{x,y}} k(x',y') \tag{1.3}$$

Ce noyau est défini positif si et seulement si le système d'appariement $\mathcal{M}$ est transitif [SK08].

Exemples de noyaux sur graphes appliqués en chémoinformatique

Un exemple de noyau sur graphes utilisé en chémoinformatique est le noyau de motifs d'arbres défini par [RG03], puis généralisé par [MV09].

Un motif d'arbres est défini dans [MV09] de la manière suivante :

Définition 30. Motifs d'arbres

Soit un graphe $G = (V_G, E_G, \mu_G, \nu_G)$ et un arbre enraciné $t = (V_t, E_t, \mu_t, \nu_t)$. On note $V_t = (n_1, \ldots, n_{|t|})$ l'ensemble des sommets de t. Un $|t|$-uplet de sommets $(v_1, \ldots, v_{|t|}) \in V_G^{|t|}$ est un motif d'arbre de G selon t, ce qui est noté $(v_1, \ldots, v_{|t|}) = pattern(t)$, si :

$$\begin{cases} \forall i \in \{1, \ldots, |t|\} & \mu_G(v_i) = \mu_t(n_i) \\ \forall e_t = (n_i, n_j) \in E_t & e_G = \{v_i, v_j\} \in E_G \wedge \nu_G(e_G) = \nu_t(e_t) \\ \forall (n_i, n_j), (n_i, n_k) \in E_t & j \neq k \iff v_j \neq v_k \end{cases}$$

Définition 31. Fonction d'occurrence de motifs d'arbres

Soit un graphe $G = (V_G, E_G, \mu_G, \nu_G)$ et un arbre $t = (V_t, E_t, \mu_t, \nu_t)$. La fonction d'occurrence ψ_t du motif d'arbre t est définie par :

$$\psi_t(G) = |\{(\alpha_1, \ldots, \alpha_{|t|}) \in \{1, \ldots, |V_G|\}^{|t|} : (v_{\alpha_1}, \ldots, v_{\alpha_{|t|}}) = pattern(t)\}|$$

Les noyaux de motifs d'arbres sont alors définis par [MV09] de la manière suivante :

Définition 32. Noyaux de motifs d'arbres

Soit deux graphes G_1 et G_2, un noyau de motifs d'arbres k^h est défini par :

$$k^h(G_1, G_2) = \sum_{t \in \mathcal{T}_h} w(t)\psi_t(G_1)\psi_t(G_2)$$

où $\mathcal{T}_h$ est l'ensemble des motifs d'arbres de hauteur h, $w : \mathcal{T}_h \to \mathbb{R}$ est une fonction de poids et ψ_t est la fonction définie dans la Définition 31.

Selon la fonction de poids des arbres $w(t)$ choisie, on obtient différents noyaux qui pourront favoriser différents types de motifs. Afin de calculer un noyau de ce type, [RG03] a défini l'ensemble suivant :

Définition 33. Ensemble d'appariement de voisinage

Soit deux graphes $G_1 = (V_1, E_1, \mu_1, \nu_1)$, $G_2 = (V_2, E_2, \mu_2, \nu_2)$ et deux de leurs sommets $u \in V_1$ et $v \in V_2$. L'ensemble d'appariement de voisinage $\mathcal{M}(u, v)$ est défini par :

$$\mathcal{M}(u,v) = \Big\{ R \subseteq N(u) \times N(v) \;\Big|\; \Big(\forall (a,b),(c,d) \in R : a \neq c \wedge b \neq d\Big) \\ \wedge \Big(\forall (a,b) \in R : \mu_1(a) = \mu_2(b) \wedge \nu_1((u,a)) = \nu_2((v,b))\Big) \Big\} \quad (1.4)$$

Proposition 4. Calcul des noyaux de motifs d'arbres *Les noyaux de motifs d'arbres k^h entre deux graphes $G_1 = (V_1, E_1, \mu_1, \nu_1)$ et $G_2 = (V_2, E_2, \mu_2, \nu_2)$, définis dans la Définition 32, peuvent être calculés par :*

$$k^h(G_1, G_2) = w_1 \sum_{u \in V_1} \sum_{v \in V_2} k^h_{som}(u, v)$$

où k^n_{som} pour $n \in \{1, \ldots, h\}$ est défini récursivement par :

$$\begin{cases} k^1_{som}(u,v) = w_2 \mathbf{1}(\mu_1(u) = \mu_2(v)) \\ k^n_{som}(u,v) = w_3 \mathbf{1}(\mu_1(u) = \mu_2(v)) \sum\limits_{R \in \mathcal{M}(u,v)} w_4 \prod\limits_{(u',v') \in R} w_5 k^{n-1}_{som}(u',v'), n = 2, \ldots, h \end{cases}$$

La définition des poids w_1, w_2, w_3, w_4 et w_5 permet de définir la fonction de poids $w(t)$.

Dans [WWK08], il a été observé que l'utilisation de motifs ayant plus de cinq arêtes n'augmente pas significativement la précision des modèles de prédiction. En se basant sur cette observation, [Gaü13] a défini un noyau fondé sur l'ensemble des sous arbres ayant au plus six nœuds.

Ces sous arbres, appelés treelets, sont extraits de chaque graphe moléculaire par un algorithme linéaire en la taille du graphe. Chaque treelet est identifié par un code contenant un indice, encodant sa structure, et une chaîne de caractères, encodant l'ensemble des étiquettes de ses arêtes et ses nœuds. Ce code est construit de manière à ce que deux treelets isomorphes aient un même code et inversement.

Le noyau de treelets est alors défini de la manière suivante :

Définition 34. Noyau de treelets

Soit deux graphes G_1 et G_2, le noyau de treelets $k_{treelet}$ est défini par :

$$k_{treelet}(G_1, G_2) = \sum_{t \in \mathcal{T}(G_1) \bigcap \mathcal{T}(G_2)} k(f_t(G_1), f_t(G_2))$$

où $\mathcal{T}(G)$ désigne l'ensemble des treelets de G, $f_t(G)$ désigne le nombre d'occurrences du treelet t dans G et k est un noyau usuel entre réels.

1.1.4 Méthodes de classification/régression

Les problèmes de classification et de régression peuvent être définis formellement de la manière suivante : soit un ensemble d'objets $\mathcal{X}$ et un ensemble $\mathcal{Y}$ associé à ces objets. On dispose d'un ensemble d'apprentissage constitué de couples $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ pour $i \in \{1, \ldots, n\}$, tirés selon une loi inconnue $\mathcal{L}$ sur $\mathcal{X} \times \mathcal{Y}$. Le but d'un problème de classification ou de régression est de définir grâce à l'ensemble d'apprentissage une fonction de prédiction

$$f : \mathcal{X} \to \mathcal{Y}$$

qui approxime cette loi inconnue.

Pour un problème de classification binaire, l'ensemble $\mathcal{Y}$ sera égal à $\{-1, 1\}$. Pour un problème de régression, $\mathcal{Y}$ pourra prendre n'importe quelle valeur réelle : $\mathcal{Y} = \mathbb{R}$.

Machine à vecteurs de support

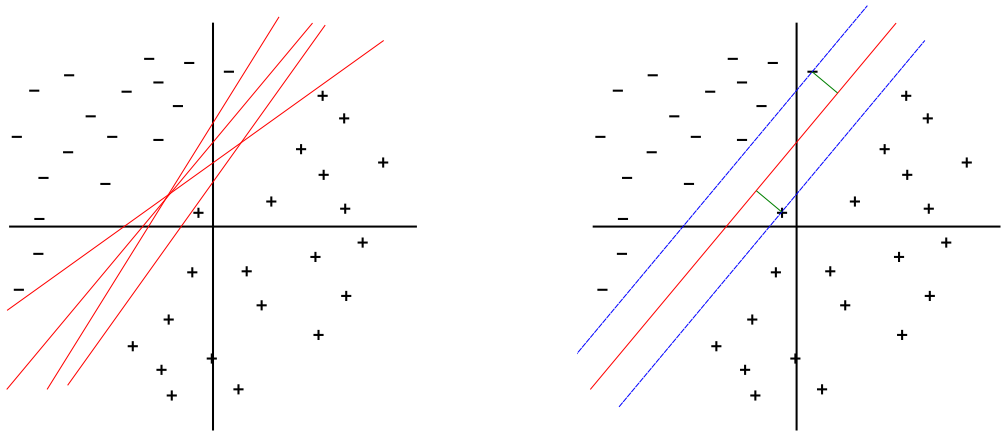
Les machines à vecteurs de support, décrites initialement dans [BGV92], sont des méthodes d'apprentissage automatique, utilisées pour résoudre des problèmes de classification.

Afin d'utiliser les méthodes à vecteurs de support pour résoudre un problème de classification, il faut que l'espace des objets $\mathcal{X}$ soit muni d'un produit scalaire, noté $\langle ., . \rangle$. Dans le papier [BGV92] $\mathcal{X}$ est un espace vectoriel de dimension n. On cherche alors un hyperplan de dimension $n-1$ capable de séparer nos données d'apprentissage $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$. Si un tel hyperplan existe, c'est-à-dire que nos données sont linéairement séparables, alors il en existe une infinité comme on peut le voir sur la Figure 1.2(a). On doit alors choisir un hyperplan optimal qui permettra d'approximer au mieux la loi inconnue $\mathcal{L}$. Pour cela, on va donc choisir l'hyperplan qui a la plus grande marge, où la marge d'un hyperplan est la distance entre cet hyperplan et les objets les plus proches de l'ensemble d'apprentissage pour chaque classe (voir Figure 1.2(b)). Si l'hyperplan est défini par l'équation $\langle \mathbf{w}, \mathbf{x} \rangle + b = 0$, alors maximiser la marge revient à résoudre le problème [BGV92] :

$$\begin{aligned} &\min_{\mathbf{w}} ||\mathbf{w}||^2 \\ &\text{sous les conditions :} \\ &y_i(\langle \mathbf{w}, \mathbf{x_i} \rangle + b) \geq 1, \forall i \in [1, \ldots, n] \end{aligned} \tag{1.5}$$

Avec $\mathbf{w}$ minimisant l'équation 1.5, la fonction de prédiction f est définie par :

$$f(\mathbf{x}) = \text{sign}(\langle \mathbf{w}, \mathbf{x} \rangle + b) \tag{1.6}$$



(a) (b)

FIGURE 1.2 – Cas de données linéairement séparables. Parmi les hyperplans séparant les données (a) on sélectionne l'hyperplan maximisant la marge (b).

où $\text{sign}(x) = 1$ si $x \geq 0$ et $\text{sign}(x) = -1$ sinon. Cette fonction de prédiction revient à voir si $\mathbf{x}$ est situé au-dessus ou en dessous de l'hyperplan d'équation $\langle \mathbf{w}, \mathbf{x} \rangle + b = 0$.

Il est possible que les données ne soient pas linéairement séparables. Dans ce cas, les auteurs de [CV95] proposent de trouver un hyperplan séparant les données en commettant le moins d'erreurs possible. Pour cela, ils ajoutent dans la formulation de [BGV92] des variables positives ξ_i représentant ces erreurs. La formulation du problème devient alors :

$$\begin{aligned} &\tfrac{1}{2} \min_{\mathbf{w}} ||\mathbf{w}||^2 + C \sum_{i=1}^{n} \xi_i \\ &\text{sous les conditions :} \\ &y_i(\langle \mathbf{w}, \mathbf{x_i} \rangle + b) \geq 1 - \xi_i, \forall i \in [1, \ldots, n] \\ &\xi_i \geq 0, \forall i \in [1, \ldots, n] \end{aligned} \tag{1.7}$$

où $C \in \mathbb{R}^+$ est un paramètre permettant de donner plus d'importance à la taille de la marge ou au nombre d'erreurs. Si cette valeur est élevée, on aura un système avec peu d'erreurs mais une petite marge.

Afin de résoudre l'équation 1.7 on utilise sa formulation duale. Pour cela on commence par expliciter le Lagrangien de l'équation 1.7 :

$$L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i(y_i(\langle \mathbf{w}, \mathbf{x_i}\rangle + b) - 1 + \xi_i) - \sum_{i=1}^{n}\beta_i\xi_i \tag{1.8}$$

Le problème de l'équation 1.7 est équivalent au problème de trouver le point selle de 1.8, correspondant au minimum de L selon $\mathbf{w}$, b et $\boldsymbol{\xi}$ et au maximum selon $\boldsymbol{\alpha}$ et $\boldsymbol{\beta}$. On sait qu'en ce point, les dérivées partielles sont nulles. On a donc les conditions suivantes :

$$\frac{\delta L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{\delta \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{n}\alpha_i y_i \mathbf{x_i} = 0 \tag{1.9}$$

$$\frac{\delta L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{\delta b} = \sum_{i=1}^{n}\alpha_i y_i = 0 \tag{1.10}$$

$$\frac{\delta L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})}{\delta \xi_i} = C - \alpha_i - \beta_i = 0 \tag{1.11}$$

En réinjectant les conditions 1.9, 1.10 et 1.11 dans l'expression du lagrangien 1.8, on obtient alors le problème dual :

$$\begin{aligned} &\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n}\alpha_i - \tfrac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \langle \mathbf{x_i}, \mathbf{x_j}\rangle \\ &\text{sous les conditions :} \\ &0 \geq \alpha_i \geq C, \forall i \in [1, \ldots, n] \\ &\sum_{i=1}^{n}\alpha_i y_i = 0 \end{aligned} \tag{1.12}$$

La condition 1.9 utilisée dans la définition de la fonction de décision 1.6 nous donne une fonction de décision dépendante de $\boldsymbol{\alpha}$:

$$f(\mathbf{x}) = \text{sign}(\sum_{i=1}^{n}\alpha_i y_i \langle \mathbf{x}, \mathbf{x_i}\rangle + b) \tag{1.13}$$

Dans la formulation duale 1.12 du problème 1.7, ainsi que dans la fonction de décision duale 1.13, on accède aux données d'apprentissage $\mathbf{x_i}$ uniquement par des produits scalaires. Le théorème 1 nous dit qu'un noyau défini positif est un produit scalaire. On peut donc remplacer dans 1.12 et 1.13 les produits scalaires par des noyaux.

Machine à vecteurs de support pour la régression

Les machines à vecteurs de support peuvent aussi être utilisées pour des problèmes de régression, grâce à l'adaptation proposée dans [DBK+97]. La formulation du problème pour la régression est :

$$\begin{aligned}
&\tfrac{1}{2}\min_{\mathbf{w}} ||\mathbf{w}||^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\
&\text{sous les conditions :} \\
&\forall i \in [1,\dots,n] \begin{cases} y_i - \langle \mathbf{w}, \mathbf{x_i}\rangle - b \geq \epsilon + \xi_i \\ \langle \mathbf{w}, \mathbf{x_i}\rangle + b - y_i \geq \epsilon + \xi_i^* \\ \xi_i \geq 0 \\ \xi_i^* \geq 0 \end{cases}
\end{aligned} \tag{1.14}$$

Comme dans la formulation des machines à vecteurs de support pour la classification, le paramètre C permet de régler l'importance qu'on accorde aux erreurs lors de l'apprentissage. Le paramètre ϵ permet de fixer un seuil de tolérance aux erreurs. Ainsi, une erreur de prédiction inférieure à ϵ ne sera pas considérée par le modèle.

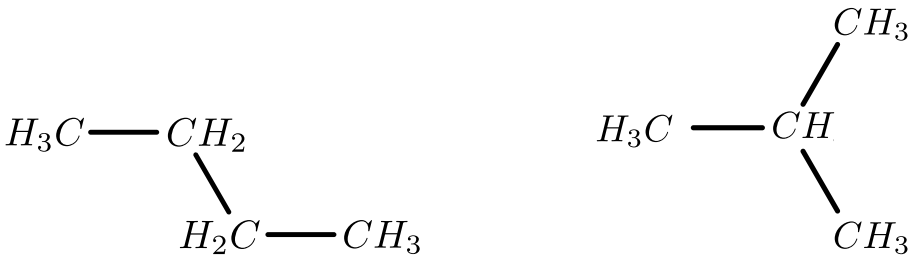
Le problème dual de 1.14 est :

$$\begin{aligned}
&\max_{\boldsymbol{\alpha},\boldsymbol{\alpha}^*} -\tfrac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\langle \mathbf{x_i}, \mathbf{x_j}\rangle - \epsilon\sum_{i=1}^{n}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{n} y_i(\alpha_i - \alpha_i^*) \\
&\text{sous les conditions :} \\
&0 \geq \alpha_i \geq C, \forall i \in \{1,\dots,n\} \\
&0 \geq \alpha_i^* \geq C, \forall i \in \{1,\dots,n\} \\
&\sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0
\end{aligned} \tag{1.15}$$

La fonction de prédiction est :

$$f(\mathbf{x}) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\langle \mathbf{x}, \mathbf{x_i}\rangle + b \tag{1.16}$$

Comme pour la classification, on remarque que le problème dual 1.15 ainsi que la fonction de prédiction 1.16 ne dépendent que de produits scalaires entre les données. On peut donc aussi remplacer ces produits scalaires par des noyaux.



(a) (b)

FIGURE 1.3 – Deux isomères de constitution, le butane (a) et le méthylpropane (b), ayant pour formule brute C_4H_{10}.

1.2 Stéréochimie

1.2.1 Description du problème

Différents modèles de représentation des molécules permettent d'obtenir différents niveaux d'information concernant ces molécules. Un modèle trop simple, comme la formule brute, peut ne pas être suffisant pour distinguer des molécules différentes. On appelle isomères les molécules ayant une même formule brute. Il existe deux types d'isomérie.

Le premier type d'isomérie, l'isomérie de constitution, apparaît lorsque des molécules ont des enchaînements d'atomes différents. Elles possèdent donc les mêmes atomes, mais les liaisons entre ces atomes ne sont pas les mêmes. Un exemple est le butane et le méthylpropane qui ont tous deux comme formule brute C_4H_{10} mais sont des molécules différentes (Figure 1.11). Le graphe moléculaire permet de différencier des isomères de ce type.

Le second type d'isomérie, la stéréoisomérie, apparaît lorsque des molécules ont une même constitution, mais n'ont pas une même organisation spatiale de leurs atomes. La stéréoisomérie se divise elle aussi en deux catégories : la stéréoisomérie de conformation et la stéréoisomérie de configuration. Deux isomères présentant une stéréoisomérie de conformation se différencient par des rotations autour de liaisons simples.

Ce manuscrit est focalisé sur la stéréoisomérie de configuration. Nous utiliserons donc le terme de stéréoisomérie pour parler de stéréoisomérie de configuration dans le reste de ce document. Cette isomérie a lieu lorsque deux molécules ont une même constitution, mais ne sont pas superposables. Formellement, on définit les stéréoisomères ainsi :

Définition 35. Stéréoisomères

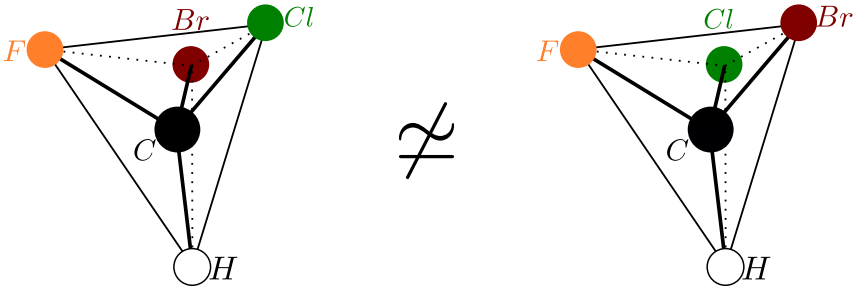


FIGURE 1.4 – Deux stéréoisomères de configuration, le (R)-bromochlorofluorométhane (à gauche) et le (S)-bromochlorofluorométhane (à droite).

Des molécules sont dites stéréoisomères si elles sont représentées par des graphes moléculaires isomorphes, mais ne sont pas superposables malgré des rotations autour de leurs liaisons simples.

Par exemple, le (R)-bromochlorofluorométhane et le (S)-bromochlorofluorométhane sont deux stéréoisomères de configuration (Figure 1.4).

Dans les deux cas, nous avons un carbone avec quatre voisins, chacun d'eux étant situé sur un des sommets d'un tétraèdre. L'échange de la position de deux voisins du carbone permet de passer d'une configuration spatiale à l'autre. Nous pouvons remarquer que la position précise de chaque atome n'est pas nécessaire à la caractérisation de la stéréochimie. En effet, seul le positionnement relatif des voisins du carbone est nécessaire pour différencier les deux molécules de la Figure 1.4. Afin de représenter graphiquement ce positionnement relatif autour d'un carbone, la notation classique en chimie, appelée représentation de Cram, utilise trois types de traits. Des traits simples signifient que les voisins du carbone et le carbone sont situés sur un même plan, un trait gras en forme de triangle indique que le voisin est situé à l'avant de ce plan et enfin un trait pointillé en forme de triangle signifie que le voisin est situé à l'arrière de ce plan. La Figure 1.5 représente les deux molécules de la Figure 1.4 en utilisant cette notation.

Le cas présenté dans la Figure 1.4 est un cas particulier de stéréoisomérie. En effet, dans cet exemple une molécule est l'image de l'autre molécule dans un miroir.

Définition 36. Énantiomères

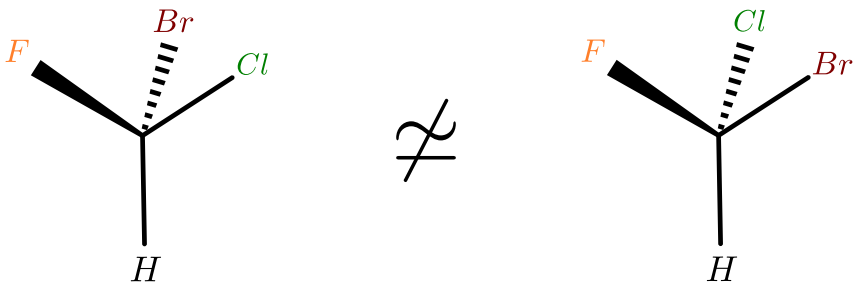


FIGURE 1.5 – Notation des configurations du (R)-bromochlorofluorométhane (à gauche) et du (S)-bromochlorofluorométhane (à droite).

Deux stéréoisomères sont dits énantiomères si l'un d'eux est l'image de l'autre par une réflexion.

Si deux molécules sont des stéréoisomères mais ne sont pas l'image l'une de l'autre dans un miroir, elles sont appelées diastéréoisomères.

D'un point de vue local, la stéréoisomérie est caractérisée par les centres stéréogènes :

Définition 37. Centres stéréogènes

Un atome est appelé centre stéréogène si la permutation de la position de deux de ses voisins crée un stéréoisomère différent. De la même manière, deux atomes liés par une liaison chimique n'autorisant pas de rotation (par exemple une liaison double), forment un centre stéréogène si une permutation (qui ne modifie pas la constitution de la molécule, c'est-à-dire ne modifie pas son graphe moléculaire) de deux atomes appartenant à l'union de leurs voisinages crée un stéréoisomère différent.

Un carbone lié à quatre voisins différents, appelé alors carbone asymétrique, est un exemple de centre stéréogène (Figure 1.4). Deux carbones liés par une double liaison (Figure 1.6) forment également un centre stéréogène si pour chaque carbone de la double liaison ses deux voisins sont différents.

La stéréoisomérie peut être due à la présence de centres stéréogènes ou à la présence d'un axe ou d'un plan chiral. Parmi les molécules actuellement utilisées en chimie, 98% des centres stéréogènes sont, soit des carbones asymétriques, soit des couples de deux carbones liés par une liaison double [JCW91]. Nous limitons notre étude à ces deux cas.

Il est important de remarquer que pour avoir un centre stéréogène, il faut que ses voisins soient différents. En effet, si au moins deux voisins d'un carbone sont identiques, alors une permutation de la position de ses voisins sera équivalente à une rotation de la molécule. Dans la Figure 1.7, permuter la

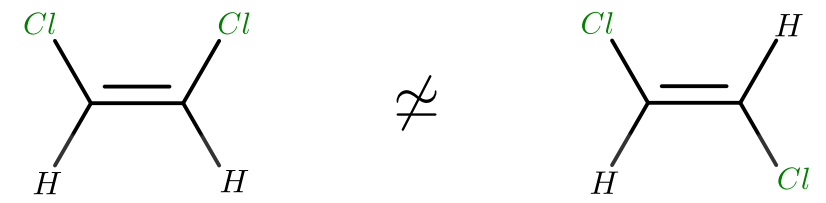


FIGURE 1.6 – Deux stéréoisomères, le (Z)-1,2-dichloroéthène (à gauche) et le (E)-1,2-dichloroéthène (à droite). Le couple de carbones liés par une liaison double est un centre stéréogène.

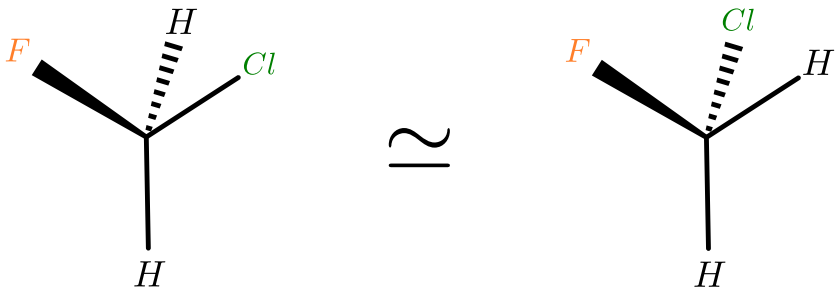


FIGURE 1.7 – Le chlorofluorométhane ne possède pas de stéréoisomères car son carbone n'est pas asymétrique.

position d'un hydrogène avec le chlore est équivalent à faire une rotation de la molécule autour de l'axe formé par le carbone et le fluor. De la même manière, si, dans une liaison double entre carbones, un des carbones a deux voisins identiques, alors le couple de carbones ne forme pas un centre stéréogène. Dans la Figure 1.8, il suffit de tourner toute la molécule autour de l'axe formé par la double liaison pour voir que les deux configurations sont identiques.

Cependant, la différence entre les voisins d'un centre stéréogène peut ne pas être située dans son voisinage direct. C'est le cas par exemple des stéréoisomères de l'acide lactique. On peut voir dans la Figure 1.9, que les

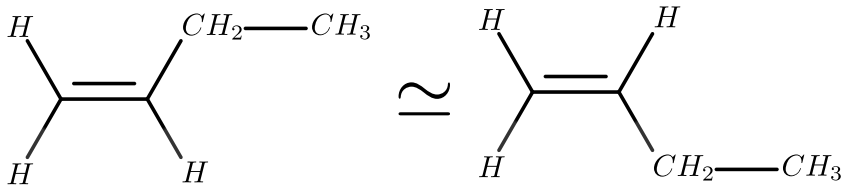


FIGURE 1.8 – Le but-1-ène ne possède pas de stéréoisomères car un des carbones de sa liaison double a deux voisins identiques.

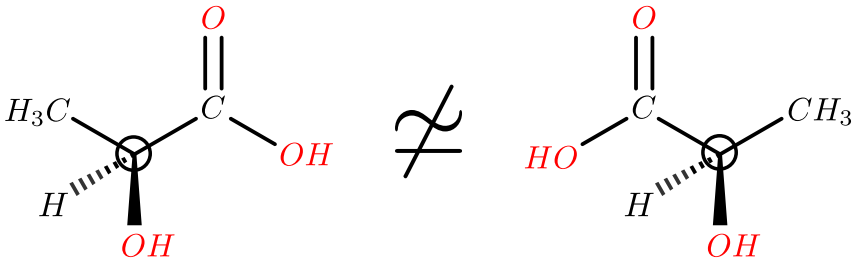


FIGURE 1.9 – Les deux stéréoisomères de l'acide lactique. Les carbones asymétriques sont soulignés par un cercle. On constate que ces carbones ont deux autres carbones comme voisins et ne sont donc pas stéréoisomères si on se fie uniquement au voisinage direct des carbones asymétriques. La stéréoisomérie vient de l'étude d'un voisinage plus large, en l'occurrence de la différence entre CH_3 et $COOH$.

carbones asymétriques ont pour voisins un oxygène, un hydrogène et deux carbones. Cependant, l'un de ces deux carbones est lié à trois hydrogènes, tandis que l'autre est lié à deux oxygènes. Cette différence fait que les deux molécules de la Figure 1.9 sont différentes.

Deux stéréoisomères sont deux molécules différentes. Elles peuvent donc avoir certaines propriétés différentes. Le pouvoir rotatoire d'une molécule, qui est sa capacité à faire tourner un faisceau lumineux polarisé la traversant, est par exemple opposé entre deux énantiomères. Les stéréoisomères peuvent également posséder des propriétés biologiques différentes. Par exemple, les deux énantiomères du limonène ont des odeurs différentes.

Le graphe moléculaire permet d'encoder les relations de voisinage entre les atomes, mais ne permet pas de représenter leur positionnement relatif. Ainsi, ce modèle est insuffisant pour différencier les stéréoisomères (de la même manière que la formule brute d'une molécule est insuffisante pour différencier les isomères de constitution). Afin de pouvoir prédire les propriétés des stéréoisomères, il faut donc une autre représentation que le graphe moléculaire.

1.2.2 Représentations de la stéréochimie

La représentation des molécules par les coordonnées cartésiennes de leurs atomes permet de différencier naturellement les stéréoisomères. Cependant, seul le positionnement relatif des atomes est nécessaire pour différencier les stéréoisomères. La représentation par les coordonnées des atomes induit donc plus de contraintes que celle liée à la propriété que l'on veut capturer. On

s'intéresse par conséquent à des représentations permettant de distinguer les stéréoisomères.

Nomenclature CIP

Afin de disposer d'une nomenclature permettant de distinguer dans tous les cas les stéréoisomères, Cahn, Ingold, et Prelog ont défini dans [CIP66] des règles permettant d'assigner une priorité à chaque voisin d'un centre stéréogène. Selon la priorité de ses voisins, on associe alors au centre stéréogène, une lettre qui permet d'identifier sa configuration. Les carbones asymétriques sont classés comme R ou S et les couples de carbones liés par une double liaison sont classés comme E ou Z.

Les règles de priorité sont les suivantes :

- Un atome de numéro atomique plus élevé est prioritaire sur un atome de numéro atomique plus faible.
- S'il y a des atomes identiques, on compare alors leurs voisins. Cette règle est répétée tant que l'on ne peut pas établir une différence de priorité. On parcourt donc la molécule de proche en proche jusqu'à trouver une différence.
- Lors du parcours de la molécule, les liaisons doubles ou triples sont considérées comme deux ou trois liaisons simples identiques. On considère donc le voisin, ainsi qu'une ou deux répliques. Ces répliques ne sont pas utilisées pour la suite du parcours de la molécule.
- Si on visite un atome déjà rencontré lors du parcours de la molécule (ce qui peut arriver si la molécule possède des cycles), alors cet atome est considéré comme une réplique.
- Un atome asymétrique R est prioritaire sur un S et une double liaison Z est prioritaire sur une double liaison E.

Pour un carbone asymétrique, une fois ses quatre voisins classés par ordre de priorité, on observe le plan contenant les trois voisins ayant les priorités les plus élevées. Ce plan est observé depuis le côté opposé au quatrième voisin. Si le parcours des trois voisins dans leur ordre de priorité décroissante ce fait dans le sens horaire, alors on assigne au centre asymétrique un R. Si au contraire ce parcours se fait dans le sens antihoraire, on assigne au centre asymétrique un S. La Figure 1.10 illustre comment est assigné le R du (R)-bromochlorofluoromethane de la Figure 1.4. Pour un couple de carbones liés

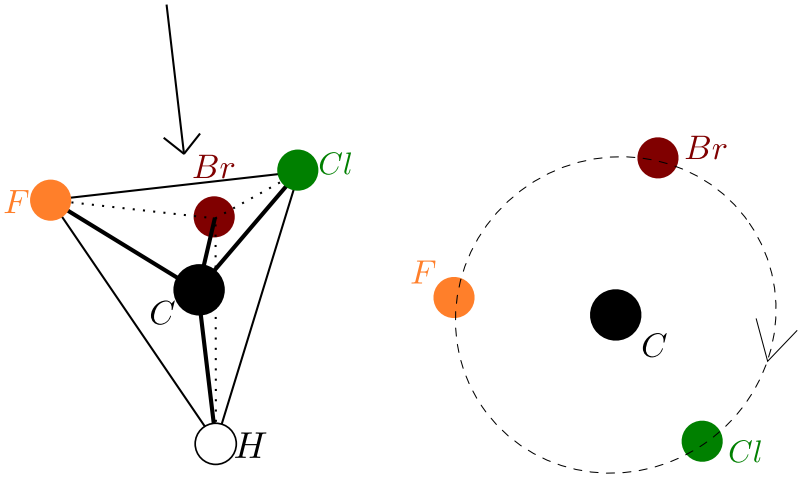


FIGURE 1.10 – Détermination de la nature R ou S du carbone asymétrique du (R)-bromochlorofluoromethane. On observe par la flèche, dans la direction du voisin de plus petite priorité, le H. Les sommets restants dans l'ordre de priorité décroissant, Br, Cl et F, sont rencontrés dans le sens horaire, on a donc une configuration R.

par une liaison double, la lettre Z est associée au couple si les deux voisins de plus haute priorité sont du même côté de la double liaison, et E s'ils sont sur des côtés opposés. La molécule située à gauche dans la Figure 1.6 a donc une double liaison Z, car les deux atomes de chlore sont du même côté, et celle située à droite a une double liaison E car ses atomes de chlore sont situés sur des côtés opposés.

Algorithme de dénomination des stéréoisomères

La méthode de Morgan [Mor65] permet d'assigner à chaque molécule une chaîne de caractères permettant de l'identifier. Cependant cette méthode ne prend pas en compte les stéréoisomères. Wipke et Dyott [WD74] ont proposé une adaptation de [Mor65] qui est capable de discerner les stéréoisomères. De plus, cet algorithme permet de détecter les centres stéréogènes dans une molécule.

Les deux méthodes commencent par construire la connectivité étendue d'un graphe, qui est une fonction ϵ associant à chaque sommet un entier $\epsilon : V \to \mathbb{N}$. Cette fonction est construite itérativement. Elle est initialisée par le degré de chaque sommet $\epsilon_0(v) = d_v$, comme on peut le voir dans la Figure 1.11(b). Puis, la nouvelle valeur associée à un sommet est la somme des valeurs associées à ses

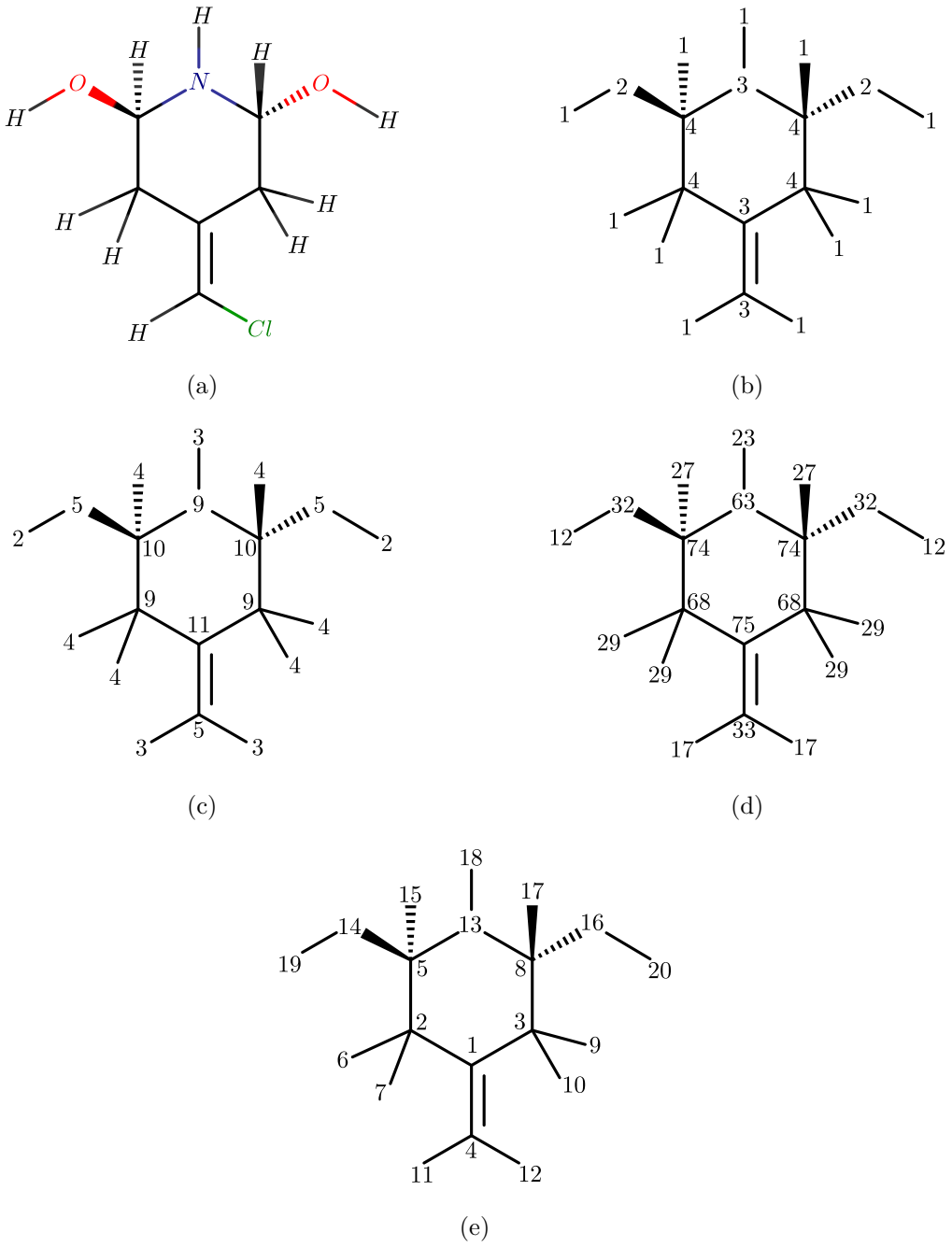


FIGURE 1.11 – Illustration de l'algorithme de dénomination des stéréoisomères sur (a). Initialisation du calcul de la connectivité étendue (b). Première itération du calcul de la connectivité étendue (c). Résultat du calcul de la connectivité étendue (d). Séquence de nombres associée à la molécule (e).

voisins $\epsilon_{i+1}(v) = \sum_{u \in N(v)} \epsilon_i(u)$. On calcule à chaque étape le nombre de valeurs différentes

$$N_i = |\{\epsilon_i(v) \mid v \in V\}|$$

que prend la fonction ϵ_i. L'opération est répétée tant que ce nombre augmente. La connectivité étendue est la fonction avec le plus de valeurs différentes : $\epsilon = \epsilon_k$ tel que $k = \min\{i \mid N_i \geq N_{i+1}\}$. Le résultat de cette procédure est illustré dans la Figure 1.11(d).

La connectivité étendue est alors utilisée pour assigner une numérotation aux sommets du graphe (Algorithme 1). Cette numérotation peut ne pas être unique, car plusieurs sommets peuvent avoir une même connectivité étendue. Un exemple de numérotation obtenue est montré dans la Figure 1.11(e).

Algorithme 1 : Numérotation des sommets

Données : Un graphe $G = (V, E, \mu, \nu)$ et sa connectivité étendue $\epsilon : V \to \mathbb{N}$
Résultat : Une numérotation de sommets de G $s : V \to \mathbb{N}$ et une liste $from : V \to \mathbb{N}$

$cur \leftarrow 1$;
$next \leftarrow 1$;
$v_{cur} \leftarrow \arg\max_{v \in V} \epsilon(v)$;
$s(v_{cur}) \leftarrow next$;
$from(v_{cur}) \leftarrow -1$;
$next \leftarrow next + 1$;
tant que $\exists u \in V$ *t.q* $s(u)$ *n'est pas déterminé* **faire**
  tant que $\exists w \in N(v_{cur})$ *t.q* $s(w)$ *n'est pas déterminé* **faire**
    $v_{tmp} \leftarrow \arg\max_{v \in N(v_{cur})} \epsilon(v)$;
    $s(v_{tmp}) \leftarrow next$;
    $from(v_{tmp}) \leftarrow cur$;
    $next \leftarrow next + 1$;
  $cur \leftarrow cur + 1$;
  $v_{cur} \leftarrow s^{-1}(cur)$;

Cette numérotation est alors utilisée pour générer un nom identifiant une molécule. Pour cela, quatre tableaux sont construits. Le premier contient la "From List" qui contient pour une case i, le numéro j du sommet ayant permis de numéroter i dans l'Algorithme 1 (ce tableau est obtenu dans l'Algorithme 1). Puis, le second tableau contient la "Ring Closure List" qui représente les arêtes

qui ne sont pas encodées par la "From List". Cette liste est vide si la molécule est acyclique. Chaque arête non parcourue lors de l'Algorithme 1 est alors représentée par le couple des numéros correspondant à ses sommets (i, j), avec $i < j$. Cette liste est triée dans l'ordre croissant. Puis vient le tableau contenant la "Atom Type List" qui contient la liste des étiquettes des sommets dans l'ordre de la numérotation. Et finalement le quatrième tableau contient la "Bond Type List". Cette liste associe à chaque arête son étiquette. Les arêtes sont rangées dans l'ordre par lequel on les a rencontrées dans l'Algorithme 1, puis dans leur ordre d'apparition dans la "Ring Closure List" si elle n'est pas vide.

Dans l'algorithme de Wipke et Dyott [WD74] deux tableaux sont ajoutés afin de prendre en compte la stéréochimie. Le premier contient la "Atom Configuration List" qui assigne à chaque sommet une valeur, permettant de définir la configuration de l'atome concerné si c'est un carbone asymétrique. Le second tableau ajouté contient la "Double Bond Configuration List" qui assigne à chaque arête une valeur, permettant de définir la configuration de la stéréochimie autour de la liaison représentée si c'est une liaison double entre carbones. Cette liste utilise le même arrangement des arêtes que la "Bond Type List".

La valeur assignée afin de déterminer la configuration d'un carbone asymétrique ou d'une double liaison, est obtenue de la même manière que pour assigner les R/S ou les E/Z. La seule différence est, qu'au lieu d'utiliser la priorité, la numérotation de l'Algorithme 1 est utilisée.

Comme il a été remarqué plus tôt, la numérotation peut ne pas être unique. En effet dans l'Algorithme 1 on prend parmi des sommets ceux de connectivité étendue maximale ($u \leftarrow \arg\max_v \epsilon(v)$). Or ce maximum peut ne pas être unique. Supposons que l'on soit dans ce cas, et que l'on ait un ensemble V_{id} de sommets ayant une même connectivité étendue. On choisit aléatoirement un des sommets $v \in V_{id}$ et on continue l'Algorithme 1. À la fin de l'algorithme on génère un nom n_v comme décrit précédemment. Puis on revient où le choix a été fait. On sélectionne alors un autre sommet $u \in V_{id} - \{v\}$. Puis on génère à la fin de l'algorithme un autre nom n_u. On compare alors les deux noms et on garde le minimum $n_{cur} = \min(n_v, n_u)$. On répète alors l'opération jusqu'à ce que V_{id} soit vide. Pour chaque *argmax* où au moins deux sommets ont une même connectivité étendue, on procède de cette manière. Les décisions prises après chaque génération de nom sont gardées en mémoire, ainsi si une égalité a déjà été traitée on prend la valeur déjà trouvée, afin de ne pas générer trop de noms.

L'article [WD74] ne parle pas de la complexité de cet algorithme. Cependant, comme des retours en arrière sont effectués lors des égalités, cet

algorithme a probablement une complexité exponentielle en le nombre de sommet.

Dans la sous-section 1.2.1, nous avons fait la remarque qu'un carbone ayant quatre voisins, ou deux carbones liés par une liaison double ne forment pas toujours un centre stéréogène. Supposons que l'on ait dans une molécule, un carbone v ayant quatre sommets, que nous connaissions le placement relatif de ses voisins, mais que ce carbone ne soit pas un centre stéréogène. Dans ce cas, lors de l'Algorithme 1, au moins deux noms seront générés, ayant pour seule différence, la valeur de la "Atom Configuration List" pour le carbone v. L'algorithme permet donc de détecter si un carbone possédant quatre voisins ou deux carbones liés par une liaison double sont des centres stéréogènes.

Définition de la chiralité

Comme dit dans la sous-section 1.2.1, une molécule possédant un carbone asymétrique est un cas particulier de stéréoisomérie, appelé chiralité. Plus généralement, un objet est appelé chiral, s'il n'est pas superposable avec son image dans un miroir.

Dans [Pet10], une définition formelle de la chiralité est donnée.

Soit E un ensemble d'éléments. On suppose que l'on possède une distance δ entre les éléments de E. Un objet est défini comme une fonction Y de E. La définition de la fonction n'est pas importante, la seule condition est que l'on doit être capable de décider si deux objets sont identiques. Soit G l'ensemble des bijections de E dans E. G est un groupe. On définit alors le sous groupe F de G qui contient les bijections qui préservent les distances :

$$F = \{U \in G \mid \forall (x, y) \in E^2, \delta(Ux, Uy) = \delta(x, y)\}$$

L'élément neutre de ce groupe est noté I_F.

Définition 38. Symétrie

Un objet Y est dit symétrique si :

$$\exists U \in F,\ U \neq I_F,\ t.q\ \forall x \in E,\ Y(Ux) = Y(x) \tag{1.17}$$

On note F^+ le sous groupe direct de F :

$$F^+ = \{U \in F \mid \exists k \in \mathbb{N}^*,\ \exists \{U_1, \ldots, U_k\} \subset F, U = \prod_{i=1}^{i=k} U_i^2\}$$

où $\prod$ représente le produit de composition.

Et on note F^- son complément : $F^- = F - F^+$.

Définition 39. Chiralité

Un objet Y est dit chiral si :

$$\nexists U \in F^- \ t.q \ \forall x \in E, Y(Ux) = Y(x) \tag{1.18}$$

En d'autres termes, un objet est chiral s'il n'existe pas de transformation indirecte qui le transforme en un objet identique.

Dans le cas où E est un espace euclidien et δ est la distance euclidienne classique, l'ensemble F contient les translations, les rotations, les réflexions et les compositions de ces transformations. Une translation du vecteur t est le produit de deux translations de vecteur $t/2$ et une rotation d'angle θ est le produit de deux rotations d'angle $\theta/2$. Les éléments de F^+ sont donc les compositions de translations et de rotations et les éléments de F^- sont les compositions d'un nombre impair de réflexions avec des translations et des rotations. Ainsi on retrouve bien la définition classique de la chiralité (Définition 36).

1.2.3 Méthodes de prédiction existantes

Descripteurs moléculaires

Les descripteurs moléculaires fondés sur un modèle 3D de la molécule prennent naturellement en compte la stéréochimie. Cependant, ces descripteurs ne sont pas toujours applicables, par exemple si l'on ne possède pas les coordonnées des atomes. De plus, une méthode telle que CoMFA [CPB88] demande d'aligner les molécules de la base, ce qui n'est pas toujours possible. Il a été montré [HCZ+99, ZT00] que les méthodes fondées sur des descripteurs moléculaires 2D et 3D ont des performances similaires, pour la prédiction de propriétés de molécules. Cependant les molécules utilisées pour ces tests ne sont pas des stéréoisomères, car les descripteurs moléculaires 2D ne peuvent pas prendre en compte la stéréochimie.

En se basant sur ces remarques, les auteurs de [GBT01] ont défini des descripteurs moléculaires ne demandant pas la connaissance des coordonnées de chaque atome, mais incorporant un moyen de discerner les stéréoisomères. De nombreux descripteurs moléculaires 2D utilisent les degrés des sommets dans le graphe, par exemple les indices de Zagreb. L'idée principale de [GBT01] est d'ajouter ou retirer une constante c aux degrés des sommets représentant un carbone asymétrique dans les calculs de ses descripteurs. Si le carbone est R selon la nomenclature CIP alors on ajoute à son degré c et s'il est S on lui

retranche c. La constante c peut être réelle ou imaginaire. Par exemple, l'indice de Zagreb défini dans l'équation 1.1 (page 16) devient :

$$M_1^{chir}(G) = M_1(G) + 2c(\sum_{j=1}^{n_R} d_{v_j} - \sum_{j=1}^{n_S} d_{u_j}) + \sum_{j=1}^{n_R+n_S} c^2 \tag{1.19}$$

où n_R et n_s sont respectivement le nombre de carbones asymétriques R et S dans la molécule, et $\{v_1, \ldots, v_{n_R}\}$ et $\{u_1, \ldots, u_{n_S}\}$ sont respectivement l'ensemble des sommets représentant les carbones asymétriques R et S.

Un inconvénient de cette méthode est que la nomenclature CIP a été créée afin d'avoir une notation unique, capable de distinguer les centres stéréogènes. Le fait qu'un carbone asymétrique soit R ou S n'apporte pas forcément d'information pertinente pour la prédiction de propriétés. Les auteurs de [ZAdS06] utilisent un principe similaire à la nomenclature CIP (assigner à chaque voisin une priorité), mais avec des règles différentes. Par exemple, la priorité d'un voisin est assignée selon le nombre d'atomes plus proches de ce voisin que d'un autre, selon la distance avec l'atome le plus éloigné ou encore selon des considérations électroniques. L'utilisation de priorités fondées non pas sur les numéros atomiques mais sur de nombreuses propriétés des voisins des carbones asymétriques, permet d'obtenir des informations ayant plus de sens. Un total de 21 différentes règles est introduit dans [ZAdS06], donnant un ensemble de 21 valeurs utilisées pour construire un descripteur moléculaire.

Polynômes chiraux

Dans la théorie de l'algèbre chirale [RS70], une molécule est considérée comme constituée d'un squelette et d'un ensemble de ligands. Par exemple, les molécules de la Figure 1.12 ont un même squelette (dans ce cas un carbone) et des ensembles de ligands différents ((F, Cl, H, Br), (F, H, H, H), (F, CO_2H, H, CH_3) et (Cl, OH, C_6H_5, H)). Étant donné un squelette, le but de la théorie de l'algèbre chirale est d'obtenir un polynôme dont les variables dépendent uniquement des ligands et qui permet d'évaluer une propriété des molécules concernées. Plusieurs conditions sont nécessaires afin de pouvoir appliquer cette théorie à un squelette : les squelettes ne doivent pas être chiraux, les ligands ne doivent pas être chiraux et il doit exister des ensembles de ligands permettant d'obtenir des molécules chirales. Dans l'exemple de la Figure 1.12, le squelette (un carbone) est achiral, chaque ligand est achiral, mais trois des ensembles de ligands (F, Cl, H, Br), (F, CO_2H, H, CH_3) et (Cl, OH, C_6H_5, H) permettent d'obtenir des molécules chirales.

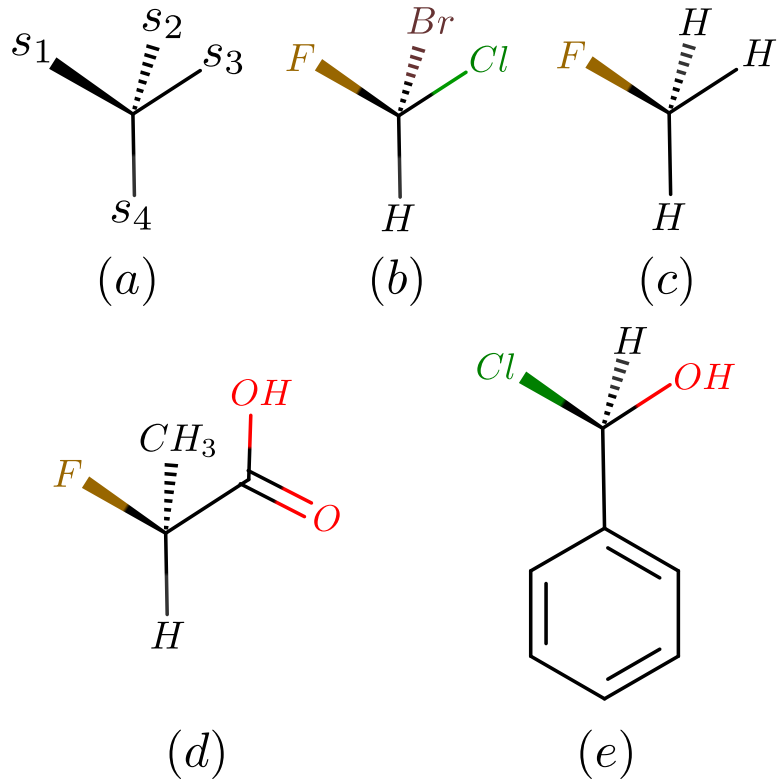


FIGURE 1.12 – Quatre molécules (b,c,d,e) ayant un même squelette (a) et des ensembles de ligands différents.

Les polynômes sont obtenus en étudiant les symétries du squelette. Par exemple, le polynôme décrivant le squelette composé uniquement d'un carbone asymétrique est $(s_4 - s_3)(s_4 - s_2)(s_4 - s_1)(s_3 - s_2)(s_3 - s_1)(s_2 - s_1)$, où s_i est un paramètre associé au ligand i du carbone. On peut remarquer qu'une transformation du squelette et de ses ligands menant à un énantiomère crée un polynôme de signe opposé. Par exemple, un miroir plan défini par le plan formé par le carbone et ses ligands 3 et 4 inverse la position de 1 et 2 et donne le polynôme suivant : $(s_4 - s_3)(s_4 - s_1)(s_4 - s_2)(s_3 - s_1)(s_3 - s_2)(s_1 - s_2) = -(s_4 - s_3)(s_4 - s_2)(s_4 - s_1)(s_3 - s_2)(s_3 - s_1)(s_2 - s_1)$.

Afin de décrire totalement un squelette et ses ligands, il est parfois nécessaire d'avoir plusieurs polynômes, avec pour chacun des variables différentes associées à chacun des ligands. Le nombre de polynômes nécessaires est donné par la formule $n!/|G|$ où n est le nombre de ligands du squelette et G le groupe des transformations laissant le squelette invariant. On peut remarquer que ce nombre a une dépendance factorielle en le nombre de ligands, ce qui empêche

d'appliquer cette méthode pour des squelettes avec un nombre important de ligands.

Noyau de motifs d'arbres adapté à la stéréochimie

Dans [BUT+10], une adaptation du noyau de motifs d'arbres [MV09] à la stéréochimie est proposée. L'idée de cette adaptation est d'ajouter une information spatiale aux motifs d'arbres. Ainsi deux motifs d'arbres $(v_1, \ldots, v_{|t|})$ et $(u_1, \ldots, u_{|t|})$ identiques au sens de la section 1.1.3 seront à présent considérés comme différents si leurs informations spatiales sont différentes.

Afin d'encoder la configuration autour d'un carbone asymétrique, [BUT+10] définit deux fonctions. Soit v un sommet ayant quatre voisins $N(v) = \{v_1, v_2, v_3, v_4\}$. La fonction $CH(v)$ associe à v la valeur 1 si l'on dispose d'une information concernant le positionnement relatif des voisins de v et 0 sinon. Ensuite, une seconde fonction *chiral* est définie telle que $chiral(v_1, v_2, v_3, v_4) = 1$ si v_2, v_3 et v_4 sont rencontrés dans le sens horaire en observant v depuis v_1, et $chiral(v_1, v_2, v_3, v_4) = -1$ s'ils sont rencontrés dans le sens antihoraire. Afin de prendre en compte les carbones asymétriques lors du calcul de noyaux de motifs d'arbres, l'ensemble d'appariement de voisinage (Définition 33 page 20) est modifié de la manière suivante :

$$\begin{aligned}\mathcal{M}(u,v) = \{R \subseteq N(u) \times N(v) | (\forall (a,b),(c,d) \in R : a \neq c \wedge b \neq d) \\ \wedge (\forall (a,b) \in R : \mu_1(a) = \mu_2(b) \wedge \nu_1((u,a)) = \nu_2((v,b))) \\ \wedge (|R| \neq 3 \vee CH(u) = 0 \vee CH(v) = 0 \vee chiral(u_0,a,c,e) = chiral(v_0,b,d,f))\}\end{aligned} \tag{1.20}$$

où u_0 et v_0 sont respectivement les parents de u et v. Par rapport à la formulation de la Définition 33 (page 20) on ajoute la condition $(|R| \neq 3 \vee CH(u) = 0 \vee CH(v) = 0 \vee chiral(u_0, a, c, e) = chiral(v_0, b, d, f))$. Cela signifie que pour apparier les voisins de deux carbones asymétriques, il faut qu'ils soient orientés de la même manière dans l'espace.

Pour prendre en compte le positionnement relatif autour d'une liaison double entre deux carbones (représentée par deux sommets v_0 et v_1) une fonction $CT : V \bigcup E \rightarrow \{-1, 0, 1, 2\}$ est définie. En notant $N(v_0) = \{v_1, v_{01}, v_{02}\}$ et $N(v_1) = \{v_0, v_{11}, v_{12}\}$, la fonction CT est alors définie par :

- Si $v \notin \{v_0, v_1\}$, alors $CT(v) = 0$.
- Si $e \in E$ n'est pas incidente à v_0 ou v_1, alors $CT(e) = 0$.

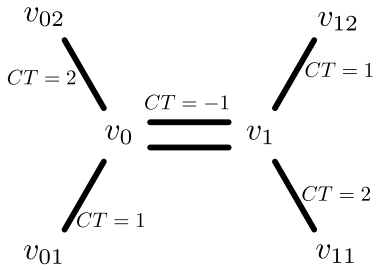


FIGURE 1.13 – Exemple de fonction CT encodant le positionnement des atomes autour d'une liaison double pour le noyau de motifs d'arbres.

- $CT(\{v_0, v_1\}) = -1$, $CT(v_0) = 1$ et $CT(v_1) = 1$.
- $CT(\{v_0, v_{01}\}) = 1$ et $CT(\{v_0, v_{02}\}) = 2$. Les arêtes $\{v_0, v_{01}\}$, $\{v_0, v_{02}\}$ et $\{v_0, v_1\}$ sont arrangées dans le sens horaire.
- Si ce placement fait que $\{v_1, v_{11}\}$, $\{v_1, v_{12}\}$ et $\{v_0, v_1\}$ sont dans le sens horaire alors $CT(\{v_1, v_{11}\}) = 1$ et $CT(\{v_1, v_{12}\}) = 2$, sinon $CT(\{v_1, v_{11}\}) = 2$ et $CT(\{v_1, v_{12}\}) = 1$ (Voir Figure 1.13).

Deux ensembles d'appariement de voisinage $\mathcal{M}^+(u, v)$ et $\mathcal{M}^-(u, v)$ sont alors considérés :

$$\begin{aligned}\mathcal{M}^+(u,v) = \{R \subseteq N(u) \times N(v) | (\forall (a,b),(c,d) \in R : a \neq c \wedge b \neq d) \\ \wedge (\forall (a,b) \in R : \mu_1(a) = \mu_2(b) \wedge \nu_1((u,a)) = \nu_2((v,b))) \\ \wedge (\forall (a,b) \in R, CT(\{u,a\}) = CT(\{v,b\}))\}\end{aligned} \quad (1.21)$$

$$\begin{aligned}\mathcal{M}^-(u,v) = \{R \subseteq N(u) \times N(v) | (\forall (a,b),(c,d) \in R : a \neq c \wedge b \neq d) \\ \wedge (\forall (a,b) \in R : \mu_1(a) = \mu_2(b) \wedge \nu_1((u,a)) = \nu_2((v,b))) \\ \wedge \left(\forall (a,b) \in R, \begin{cases} (CT(\{u,a\}) = CT(\{v,b\}) = 0) \\ \vee (CT(\{u,a\}) = CT(\{v,b\}) = -1) \\ \vee (CT(\{u,a\}) \neq CT(\{v,b\}) \end{cases} \right)\}\end{aligned} \quad (1.22)$$

Deux ensembles d'appariement de voisinages sont nécessaires car on peut apparier les liaisons doubles de deux manières différentes, comme on peut le voir dans la Figure 1.14. Considérons que l'on essaye d'apparier les motifs d'arbres des graphes G_1 et G_2 de la Figure 1.14. Sans perte de généralités, on considère que la racine du motif d'arbres pour G_1 est a_1.

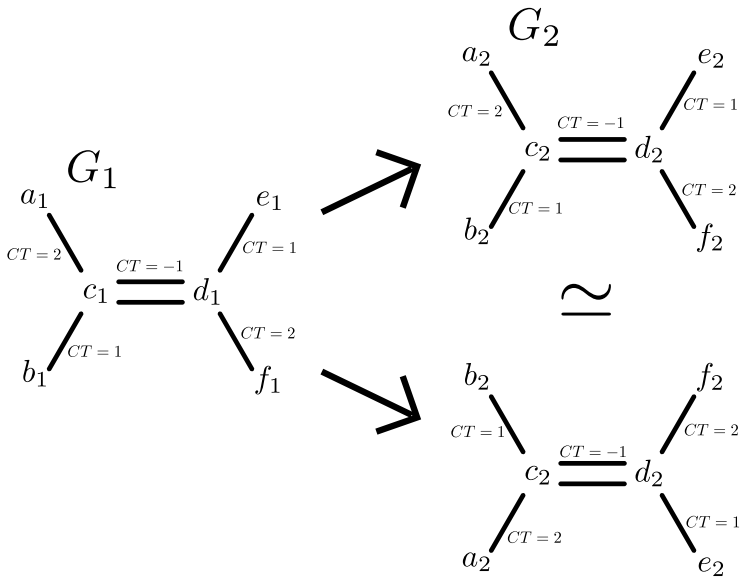


FIGURE 1.14 – Les motifs d'arbres $(a_1, b_1, c_1, d_1, e_1, f_1)$ et $(a_2, b_2, c_2, d_2, e_2, f_2)$, ainsi que $(a_1, b_1, c_1, d_1, e_1, f_1)$ et $(b_2, a_2, c_2, d_2, f_2, e_2)$, sont identiques.

Si a_1 est apparié à a_2, on aura $CT(\{a_1, c_1\}) = CT(\{a_2, c_2\}) = 2$. Dans ce cas, on devra suivre les conditions d'appariement correspondant au premier ensemble d'appariement de voisinage $\mathcal{M}^+(u, v)$:

$$\forall (a, b) \in R, CT(\{u, a\}) = CT(\{v, b\})$$

Ainsi, apparier a_1 à a_2 impose d'apparier e_1 à e_2 et f_1 à f_2. Cet appariement correspond à celui situé en haut dans la Figure 1.14.

Si a_1 est apparié à b_2, on aura $2 = CT(\{a_1, c_1\}) \neq CT(\{b_2, c_2\}) = 1$. Dans ce cas, on devra suivre les conditions d'appariement correspondant au second ensemble d'appariement de voisinage $\mathcal{M}^-(u, v)$:

$$\forall (a, b) \in R, \begin{cases} (CT(\{u, a\}) = CT(\{v, b\}) = 0) \\ \vee (CT(\{u, a\}) = CT(\{v, b\}) = -1) \\ \vee (CT(\{u, a\}) \neq CT(\{v, b\}) \end{cases}$$

Ainsi, apparier a_1 à b_2 impose d'apparier e_1 à f_2 et f_1 à e_2. Cet appariement correspond à celui situé en bas dans la Figure 1.14.

1.3 Conclusion

Les différents modèles présentés dans ce chapitre permettent de représenter les molécules avec différents niveaux de précision. Le graphe moléculaire permet de représenter une molécule en prenant en compte les liaisons entre ses atomes. Cependant, cette représentation a l'inconvénient de ne pas pouvoir représenter les stéréoisomères. Ces molécules, qui diffèrent par le positionnement relatif de leurs atomes, peuvent être représentées par des modèles incluant une notion d'ordonnancement des atomes. La nomenclature CIP, l'algorithme de dénomination des stéréoisomères ou les fonctions définies pour le noyau de motifs d'arbres adapté à la stéréochimie, utilisent tous une notion d'ordre afin de différencier les stéréoisomères.

Certains algorithmes d'apprentissage automatique, tels que les machines à vecteurs de support, demandent de pouvoir calculer un produit scalaire entre les modèles représentant des données afin de prédire les propriétés de ces données. Pour cela, deux approches sont utilisées. La première consiste à représenter directement les données par un vecteur. C'est le cas des méthodes fondées sur les descripteurs moléculaires. La seconde consiste à définir un noyau défini positif entre les modèles. Ce noyau correspond à un produit scalaire des modèles dans un espace de Hilbert. L'avantage des méthodes à noyaux est que cet espace n'a pas besoin d'être explicitement calculé. La seule méthode prenant en compte la stéréochimie des molécules et utilisant une méthode à noyaux est le noyau de motif d'arbres adapté à la stéréochimie défini par [BUT+10].

Chapitre 2

Graphes Localement Ordonnés

Sommaire

2.1 Introduction

Le graphe moléculaire est un objet qui représente naturellement les connexions entre les atomes d'une molécule. Cependant, le manque d'informations sur le positionnement relatif des atomes rend ce modèle incapable de discerner les stéréoisomères. Comme nous avons vu dans le chapitre précédent, de nombreuses méthodes utilisent une notion d'ordre afin de décrire ce positionnement. Nous

souhaitons donc ajouter une notion d'ordre au graphe moléculaire, afin de prendre en compte la stéréoisomérie.

Pour cela, on commence par introduire les structures localement ordonnées qui permettent d'associer des notions d'ordres et d'ordres équivalents à n'importe quel objet structuré. Puis dans la section 2.3, nous appliquerons cette notion aux graphes moléculaires. Nous obtiendrons alors des graphes moléculaires localement ordonnés, permettant de garder les avantages du graphe moléculaire tout en prenant en compte les stéréoisomères. Finalement, la section 2.4 montre que notre définition permet d'encoder correctement la chiralité, selon la définition donnée par Petitjean dans [Pet10].

2.2 Structures localement ordonnées

2.2.1 Définition des structures localement ordonnées

Définition 40. Objets structurés

On appelle objet structuré S un objet auquel on peut associer un unique, à un isomorphisme près, graphe $G(S) = (V, E, \mu, \nu)$.

Il existe une relation injective entre l'ensemble des isomorphismes entre objets structurés $Isom(S, S')$ et l'ensemble des isomorphismes entre leurs graphes associés $Isom(G(S), G(S'))$. L'ensemble des isomorphismes entre objets structurés $Isom(S, S')$ respecte les propriétés suivantes :

1. $\forall S_1, S_2,\ f \in Isom(S_1, S_2) \Leftrightarrow f^{-1} \in Isom(S_2, S_1)$
2. $\forall S_1, S_2, S_3,\ f \in Isom(S_1, S_2), g \in Isom(S_2, S_3) \Rightarrow g \circ f \in Isom(S_1, S_3)$
3. $\forall S,\ Isom(S, S)$ *est un groupe (Définition 25 page 13) pour la composition.*

Un objet structuré peut être par exemple un graphe (associé à lui-même) ou un arbre enraciné (associé à un graphe acyclique).

Les conditions sur l'ensemble des isomorphismes entre objets structurés imposent que cet ensemble ait les propriétés habituelles des isomorphismes. L'inverse d'un isomorphisme est un isomorphisme (condition 1), la composition de deux isomorphismes est un isomorphisme (condition 2) et enfin l'ensemble des automorphismes d'un objet structuré S est un groupe pour la composition (condition 3).

Par exemple, si les objets structurés sont des graphes, l'ensemble des isomorphismes pourra être égal à celui entre les graphes. Cependant, si les objets structurés sont des arbres enracinés, l'ensemble des isomorphismes

pourra être celui entre les arbres enracinés, qui est bien un sous-ensemble de l'ensemble des isomorphismes entre les graphes, et qui respecte les trois conditions de la Définition 40.

Comme nous avons vu dans la sous-section 1.2.2, il est usuel d'ordonner les voisins des sommets afin de représenter la stéréochimie. La nomenclature CIP [CIP66] utilise des règles de priorité afin de classer les voisins d'un centre stéréogène, tandis que l'algorithme de dénomination des stéréoisomères de [WD74] utilise une numérotation des sommets afin de définir la stéréochimie d'une molécule. Afin d'encoder le positionnement relatif des atomes dans une molécule, nous introduisons la notion d'ordre sur les objets structurés.

Définition 41. Structures localement ordonnées

Une structure localement ordonnée $S = (\hat{S}, ord)$ est un objet structuré $\hat{S}$ associé à un graphe $G(\hat{S}) = (V, E, \mu, \nu)$, avec une fonction d'ordre ord. Cette fonction associe à chaque sommet v d'un sous-ensemble V_{ord} de V, une liste ordonnée de son voisinage $N(v)$:

$$ord \begin{cases} V_{ord} & \rightarrow & V^* \\ v & \rightarrow & (v_1, \ldots, v_n) \end{cases}$$

où $N(v) = \{v_1, \ldots, v_n\}$ et V^* est l'ensemble des suites d'éléments de V.

La fonction ord induit donc une relation d'ordre total sur le voisinage des sommets de V_{ord}.

On note $\mathscr{S}$ l'ensemble des structures localement ordonnées.

Dans la suite de ce document, lorsque l'on aura une structure localement ordonnée S, la notation $\hat{S}$ désignera toujours l'objet structuré correspondant à cette structure localement ordonnée.

Définition 42. Isomorphismes entre structures localement ordonnées

Deux structures localement ordonnées $S = (\hat{S}, ord)$ et $S' = (\hat{S}', ord')$ sont isomorphes $S \underset{o}{\simeq} S'$ s'il existe un isomorphisme entre leurs objets structurés $\hat{S}$ et $\hat{S}'$ qui est compatible avec l'ordre défini pour chaque sommet de V_{ord} et de $V'_{ord'}$:

$$S \underset{o}{\simeq} S' \Leftrightarrow \exists f \in Isom(\hat{S}, \hat{S}') \ t.q.$$
$$\forall v \in V_{ord} \ avec \ ord(v) = (v_1, \ldots, v_n), \ ord'(f(v)) = (f(v_1), \ldots, f(v_n))$$

Dans ce cas, f est appelé un isomorphisme localement ordonné entre S et S', et on note $IsomOrd(S, S') \subset Isom(\hat{S}, \hat{S}')$ l'ensemble des isomorphismes localement ordonnés entre S et S'.

Afin de pouvoir définir les isomorphismes entre structures localement ordonnées, il faut que les ensembles de sommets possédant un ordre V_{ord} soient définis de la même manière. Formellement, cela veut dire que l'on impose que :

$$\forall (S, S') \in \mathscr{S}^2, \forall f \in Isom(\hat{S}, \hat{S}'), f(V_{ord}) = V'_{ord} \tag{2.1}$$

Proposition 5. *La relation d'isomorphisme localement ordonnée entre structures localement ordonnées est une relation d'équivalence.*

Démonstration. Cette preuve est donnée en annexe à la section 5.1.1 (page 113). □

2.2.2 Fonctions de réordonnancement

Les structures localement ordonnées vont être utilisées dans la section 2.3 afin de représenter les stéréoisomères. Cependant, un carbone asymétrique possède quatre voisins. Il y a donc $4! = 24$ façons de ranger ses voisins. Or il n'y a que deux configurations différentes possibles. Certains ordres différents devront donc représenter une même configuration, et il est donc nécessaire de pouvoir considérer que des ordres différents puissent être équivalents.

Pour cela nous introduisons la notion de fonction de réordonnancement, qui associe aux sommets d'une structure localement ordonnée une permutation de leurs voisins.

Définition 43. Fonctions de réordonnancement

Soit $S = (\hat{S}, ord)$ une structure localement ordonnée associée à un graphe $G(S) = (V, E, \mu, \nu)$. Une fonction de réordonnancement σ_S sur S associe à chaque sommet v appartenant à V_{ord} une permutation φ_v sur $\{1, \ldots, |N(v)|\}$.

$$\sigma_S \begin{cases} V_{ord} & \rightarrow & \mathcal{P} \\ v & \rightarrow & \varphi_v \in \Pi_{|ord(v)|} \end{cases}$$

où Π_n est le groupe des permutations de n éléments et $\mathcal{P}$ est l'union des Π_n pour tout entier naturel n.

L'application d'une fonction de réordonnancement à une structure localement ordonnée donne une nouvelle structure localement ordonnée définie ainsi :

Définition 44. Structures réordonnées

Soit $S = (\hat{S}, ord)$ une structure localement ordonnée et σ_S une fonction de réordonnancement. $\sigma_S(S) = (\hat{S}, ord_{\sigma_S})$ est alors une structure localement ordonnée obtenue après avoir appliqué la fonction de réordonnancement σ_S sur S :

$$\forall v \in V_{ord} \; t.q. \left(\begin{array}{l} ord(v) = (v_1, \ldots, v_n) \\ \quad et \\ \sigma_S(v) = \varphi_v, \end{array} \right) ord_{\sigma_S}(v) = (v_{\varphi_v(1)}, \ldots, v_{\varphi_v(n)})$$

Notez que les objets structurés associés à S et à $\sigma_S(S)$ sont identiques. Cela signifie que le réordonnancement d'une structure localement ordonnée ne modifie pas l'objet structuré qui lui est associé. Les opérations de réordonnancement étant définies comme des fonctions, on peut les combiner en utilisant la composition de fonctions :

Définition 45. Composition de fonctions de réordonnancement

Soit σ_S et σ'_S deux fonctions de réordonnancement d'une structure localement ordonnée $S = (\hat{S}, ord)$. La composition de σ_S et de σ'_S est une fonction de réordonnancement de S, notée $\sigma_S \circ \sigma'_S$ et définie par :

$$\sigma_S \circ \sigma'_S \left(\begin{array}{lcl} V_{ord} & \rightarrow & \mathcal{P} \\ v & \rightarrow & \sigma_S(v) \circ \sigma'_S(v) \in \Pi_{|ord(v)|} \end{array} \right.$$

où Π_n est le groupe des permutations de n éléments et $\mathcal{P}$ est l'union des Π_n pour tout entier naturel n.

L'identité pour la composition est la fonction de réordonnancement Id_S qui associe à chaque sommet v de l'ensemble V_{ord} la permutation identité Id_n sur Π_n.

L'inverse d'une fonction de réordonnancement σ_S est la fonction de réordonnancement σ_S^{-1} telle que $\sigma_S \circ \sigma_S^{-1}$ est égale à l'identité.

2.2.3 Structures localement ordonnées avec des ordres équivalents

Les fonctions de réordonnancement peuvent changer l'ordre d'une structure localement ordonnée en n'importe quel autre ordre, ce qui supprime tout l'intérêt à la notion d'ordre. Afin d'obtenir une notion utile de réordonnancement, nous devons choisir un sous ensemble de fonctions de réordonnancement.

Définition 46. Famille valide de fonctions de réordonnancement

Pour chaque structure localement ordonnée $S = (\hat{S}, ord)$, on note Σ_S un ensemble de fonctions de réordonnancement sur S. Une famille valide de fonctions de réordonnancement est un ensemble Σ défini par

$$\Sigma = \{\Sigma_S,\ S \in \mathscr{S}\}$$

et satisfaisant les deux propriétés suivantes :

- *Pour toute structure localement ordonnée S, Σ_S est un groupe pour la composition.*

- *Soit deux structures localement ordonnées $S = (\hat{S}, ord)$ et $S' = (\hat{S}', ord')$ telles qu'il existe un isomorphisme f entre leurs objets structurés $\hat{S}$ et $\hat{S}'$. Alors, toute fonction de réordonnancement $\sigma \in \Sigma_S$ est égale, à un isomorphisme près, à une fonction de réordonnancement de $\Sigma_{S'}$:*

$$\left.\begin{array}{l} \forall f \in Isom(\hat{S}, \hat{S}'), \\ \forall \sigma \in \Sigma_S \end{array}\right) \sigma \circ f^{-1} \in \Sigma_{S'}$$

La première contrainte de la Définition 46 signifie que les fonctions de réordonnancement d'une structure localement ordonnée peuvent être combinées en utilisant des opérations de composition. La seconde contrainte impose que deux structures localement ordonnées étant associées à un même objet structuré doivent avoir des ensembles équivalents de fonctions de réordonnancement.

Remarque 1. *La seconde contrainte de la Définition 46 peut être réécrite :*

$$\left.\begin{array}{l} \forall f \in Isom(\hat{S}, \hat{S}'), \\ \forall \sigma \in \Sigma_S \end{array}\right) \exists \sigma' \in \Sigma_{S'} \mid \sigma' \circ f = \sigma$$

Définition 47. Ordres équivalents

Soit deux structures localement ordonnées $S_a = (\hat{S}_a, ord_a)$ et $S_b = (\hat{S}_b, ord_b)$. Ces structures localement ordonnées sont dites d'ordres équivalents selon la famille valide de fonctions de réordonnancement Σ, ce que l'on note $S_a \underset{\Sigma}{\simeq} S_b$, si et seulement si :

$$\exists \sigma \in \Sigma,\ \sigma(S_a) \underset{o}{\simeq} S_b \tag{2.2}$$

Nous considérons donc que deux structures localement ordonnées sont équivalentes s'il existe, à une fonction de réordonnancement valide σ prés, un isomorphisme localement ordonné f entre elles. Dans ce cas, f est appelé isomorphisme d'équivalence d'ordres par σ entre S_a et S_b. On note $IsomEqOrd_\sigma(S_a, S_b)$ l'ensemble des isomorphismes d'équivalence d'ordres par σ entre S_a et S_b. De plus, on note $IsomEqOrd(S_a, S_b)$ l'union des $IsomEqOrd_\sigma(S_a, S_b)$:

$$IsomEqOrd(S_a, S_b) = \bigcup_{\sigma \in \Sigma_{S_a}} IsomEqOrd_\sigma(S_a, S_b)$$

Afin de pouvoir utiliser les structures localement ordonnées pour représenter des molécules, il faut s'assurer que la relation définie dans la Définition 47 est bien une relation d'équivalence, et donc qu'elle définit une relation cohérente d'égalité.1

Théorème 2. *Soit Σ une famille valide de fonctions de réordonnancement. La relation d'équivalence d'ordres sur les structures localement ordonnées, définie dans la Définition 47 et fondée sur la famille Σ est une relation d'équivalence.*

Démonstration. Cette preuve est donnée en annexe à la section 5.1.2 (page 115). □

2.3 Graphes localement ordonnés appliqués aux molécules

2.3.1 Graphes localement ordonnés

En appliquant la notion de structures localement ordonnées aux graphes, nous définissons les graphes localement ordonnés.

Définition 48. Ensemble de graphes localement ordonnés

Un graphe localement ordonné $S = (G = (V, E, \mu, \nu), ord)$ est une structure localement ordonnée avec $G(S) = G$ et une fonction ord qui associe à chaque sommet v de $V_{ord} \subset V$ une liste ordonnée de ses voisins :

$$ord \begin{cases} V_{ord} & \rightarrow & V^* \\ v & \rightarrow & (v_1, \ldots, v_n) \end{cases}$$

où $N(v) = \{v_1, \ldots, v_n\}$ est le voisinage de v.

On note $\mathcal{OG}$ l'ensemble des graphes localement ordonnés. L'ensemble d'isomorphismes entre les graphes non ordonnés (Définition 40) est l'ensemble classique des isomorphismes entre graphes.

Comme l'ensemble des graphes localement ordonnés est un ensemble de structures localement ordonnées, on peut définir pour les graphes localement ordonnés des fonctions de réordonnancement (Définition 43). Si les fonctions de réordonnancement définissent une famille valide (Définition 46), on peut alors définir une relation d'équivalence entre les graphes localement ordonnés par la Définition 47 (Théorème 2).

2.3.2 Graphes localement ordonnés pour représenter et différencier les stéréoisomères

Les définitions des ordres et des fonctions de réordonnancement pour des graphes localement ordonnés dépendent de l'application. Pour définir les graphes moléculaires localement ordonnés, qui vont permettre d'encoder les stéréoisomères, nous considérons d'abord les graphes moléculaires $G = (V, E, \mu, \nu)$ tels qu'ils sont définis dans la Section 1.1.

Nous avons vu dans la section 1.2.1 que nous voulons encoder le positionnement relatif autour des carbones asymétriques et des couples de carbones liés par une liaison double. Afin de distinguer ces configurations dans un graphe moléculaire nous définissons les deux sous-ensembles de sommets suivants :

Définition 49. Carbones potentiellement asymétriques

Soit $G = (V, E, \mu, \nu)$ un graphe moléculaire. On note V^{PAC} le sous-ensemble de V contenant tous les sommets représentant des carbones ayant quatre voisins :

$$V^{PAC} = \{v \in V \mid \mu(v) = \text{'C'} \text{ et } |N(v)| = 4\}$$

Définition 50. Ensemble de couples de carbones connectés par une liaison double

Soit $G = (V, E, \mu, \nu)$ un graphe moléculaire. On note V^{DB} le sous-ensemble de V contenant tous les sommets représentant des carbones qui sont liés par une liaison double à un autre carbone :

$$V^{DB} = \left\{ v \in V \mid \exists e = \{v, w\} \in E, \nu(e) = 2, \left(\begin{array}{lllll} |N(v)| & = & |N(w)| & = & 3 \\ et & & & & \\ \mu(v) & = & \mu(w) & = & \text{'C'} \end{array} \right) \right\}$$

Un sommet, représentant un atome de carbone, ayant deux arêtes incidentes représentant des liaisons doubles a un degré de deux. Donc, chaque sommet v appartenant à V^{DB} possède une seule arête incidente représentant une liaison double. On note alors $n_=(v)$ le carbone adjacent à v par cette liaison double. $n_=(v)$ appartient lui aussi à V^{DB}.

En utilisant ces deux ensembles, on peut maintenant définir la notion de graphe moléculaire localement ordonné :

Définition 51. Graphes moléculaires localement ordonnés

Un graphe moléculaire localement ordonné est un couple $G = (\hat{G}, ord)$ où $\hat{G}$ est un graphe moléculaire. La fonction ord est définie sur un ensemble V_{ord} défini par :

$$V_{ord} = V^{PAC} \cup V^{DB}$$

où V^{PAC} et V^{DB} sont respectivement l'ensemble des carbones potentiellement asymétriques (Définition 49) et l'ensemble des carbones des couples de carbones connectés par une liaison double (Définition 50).

La fonction ord est définie de la manière suivante :

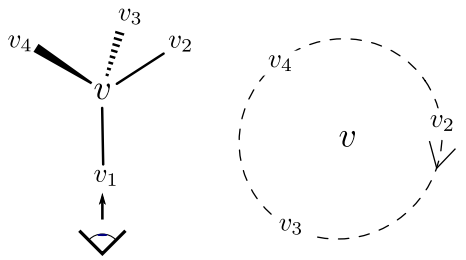
- *Si $v \in V^{PAC}$:*

  Soit $N(v) = \{v_1, v_2, v_3, v_4\}$ le voisinage de v. Pour définir un ordre des voisins, on commence par choisir arbitrairement un premier voisin (par exemple v_1). Les trois autres voisins de v sont ordonnés de manière à ce que si l'on regarde v depuis v_1, ils soient vus dans le sens horaire. Un des trois ordres satisfaisant cette condition est choisi arbitrairement (Figure 2.1).

- *Si $v \in V^{DB}$:*

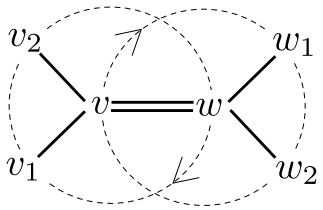
  Soit $w = n_=(v)$ le voisin de v par la double liaison. On note les voisinages de v et w, $N(v) = \{w, v_1, v_2\}$ et $N(w) = \{v, w_1, w_2\}$. Les ordres des voisinages de v et w sont définis par $ord(v) = (w, v_1, v_2)$ et $ord(w) = (v, w_1, w_2)$, tels que (w, v_1, v_2) et (v, w_1, w_2) soit traversé dans le sens horaire lorsque l'on tourne autour de v et de w (Figure 2.2).

On note $\mathcal{OM}$ l'ensemble des graphes moléculaires localement ordonnés.



$$ord(v) = (v_1, v_2, v_3, v_4)$$

FIGURE 2.1 – Exemple d'un sommet de V^{PAC} avec la liste ordonnée de ses voisins.



$$ord(v) = (w, v_1, v_2)\ ord(w) = (v, w_1, w_2)$$

FIGURE 2.2 – Exemple de sommets de V^{DB} avec les listes ordonnées de leurs voisins.

Afin de pouvoir définir une notion d'isomorphismes entre graphes moléculaires localement ordonnés, il faut que l'ensemble V_{ord} soit stable par isomorphisme (Définition 42).

Proposition 6. *Soit $G = (\widehat{G}, ord)$ et $G' = (\widehat{G'}, ord')$ deux graphes moléculaires localement ordonnés tel qu'il existe un isomorphisme f entre les graphes $\widehat{G}$ et $\widehat{G'}$.*

Alors

$$f(V_{ord}) = V'_{ord}$$

De plus nous avons :

- $f(V^{PAC}) = V'^{PAC}$
- $f(V^{DB}) = V'^{DB}$
- $\forall v \in V^{DB},\ f(n_=(v)) = n_=(f(v))$.

Démonstration. Cette preuve est donnée en annexe à la section 5.1.3 (page 120). □

L'ordre autour d'un sommet de V^{PAC} dépend de deux choix arbitraires. Ainsi une même molécule peut être définie par deux graphes localement ordonnés avec des ordres différents. Il nous faut donc définir une relation d'équivalence d'ordres entre graphes localement ordonnés moléculaires, et donc il nous faut définir un ensemble de fonctions de réordonnancement.

Autour d'un carbone asymétrique et de deux carbones liés par une double liaison deux configuration sont possibles (voir la Figure 1.4 à la page 27 et la Figure 1.6 à la page 29). L'échange de deux voisins d'un centre stéréogène (qui se traduit par une transposition en termes de permutation) crée le stéréoisomère opposé. Ainsi deux échanges de voisins d'un centre stéréogène donne l'opposé du stéréoisomère opposé et donc la molécule originale. Ces deux échanges correspondent à une permutation qui serait le produit de deux transpositions.

Intuitivement, nous pouvons donc associer le fait qu'une permutation change ou non la configuration autour d'un centre stéréogène au nombre de transpositions composant cette permutation. Ainsi, afin de définir les fonctions de réordonnancement des graphes localement ordonnés moléculaires, nous allons utiliser les notions de parité et de signature des permutations (Définition 24 page 12).

Définition 52. Ensemble de fonctions de réordonnancement pour les graphes moléculaires localement ordonnés

Soit G un graphe moléculaire localement ordonné. On note Σ_G^M l'ensemble de ses fonctions de réordonnancement. Σ_G^M contient toutes les fonctions de réordonnancement σ tel que :

- *Pour tout v dans V^{PAC}, la permutation $\sigma(v)$ est une permutation paire :*

$$\forall v \in V^{PAC},\ \epsilon(\sigma(v)) = 1.$$

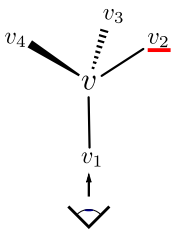
  où ϵ est la signature d'une permutation.

- *Pour tout v dans V^{DB}, les permutations $\sigma(v)$ et $\sigma(n_=(v))$ sont des permutations de même parité :*

$$\forall v \in V^{DB},\ \epsilon(\sigma(v)) = \epsilon(\sigma(w)) \text{ avec } w = n_=(v).$$

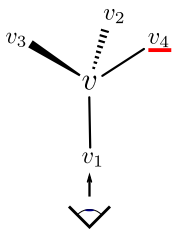
L'ensemble des fonctions de réordonnancement des graphes moléculaires localement ordonnés Σ^M est alors défini par :

$$\Sigma^M = \{\Sigma_G^M,\ G \in \mathcal{OM}\}$$



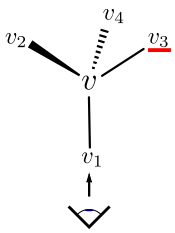
$ord(v) = (v_1, v_2, v_3, v_4)$

$\varphi = Id$



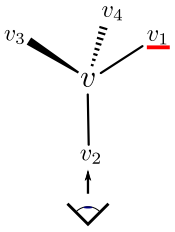
$ord_\varphi(v) = (v_1, v_4, v_2, v_3)$

$\varphi = (234) = (23)(24)$



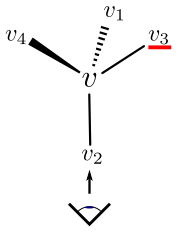
$ord_\varphi(v) = (v_1, v_3, v_4, v_2)$

$\varphi = (243) = (24)(23)$



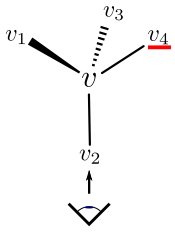
$ord_\varphi(v) = (v_2, v_1, v_4, v_3)$

$\varphi = (12)(34)$



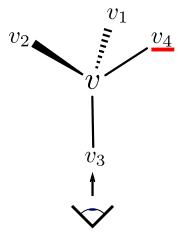
$ord_\varphi(v) = (v_2, v_3, v_1, v_4)$

$\varphi = (132) = (13)(12)$



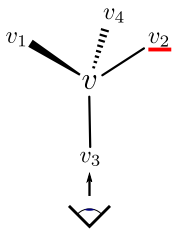
$ord_\varphi(v) = (v_2, v_4, v_3, v_1)$

$\varphi = (142) = (14)(12)$



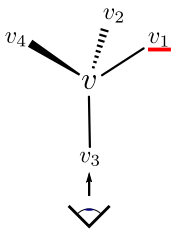
$ord_\varphi(v) = (v_3, v_4, v_1, v_2)$

$\varphi = (13)(24)$



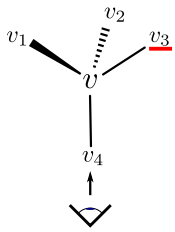
$ord_\varphi(v) = (v_3, v_2, v_4, v_1)$

$\varphi = (143) = (14)(13)$



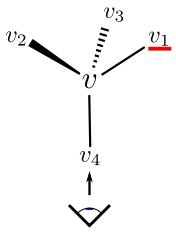
$ord_\varphi(v) = (v_3, v_1, v_2, v_4)$

$\varphi = (123) = (12)(13)$



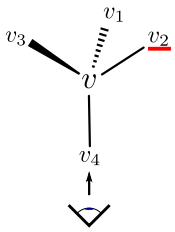
$ord_\varphi(v) = (v_4, v_3, v_2, v_1)$

$\varphi = (14)(23)$



$ord_\varphi(v) = (v_4, v_1, v_3, v_2)$

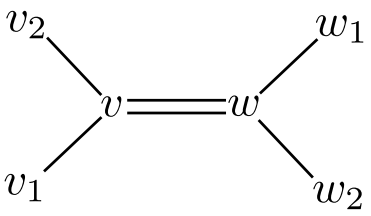
$\varphi = (124) = (12)(14)$

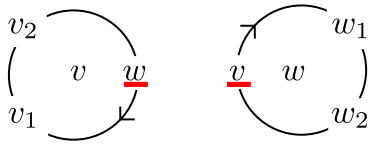


$ord_\varphi(v) = (v_4, v_2, v_1, v_3)$

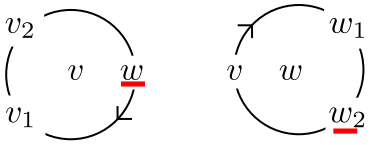
$\varphi = (134) = (13)(14)$

FIGURE 2.3 – Exemple d'un sommet de V^{PAC} avec la liste ordonnée de ses voisins et les ordres obtenus après l'application de l'ensemble des permutations autorisées.

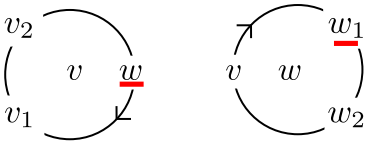




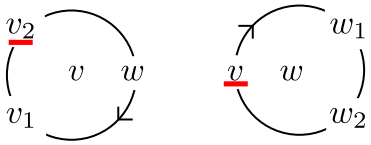
$ord(v) = (w, v_1, v_2)$, $ord(w) = (v, w_1, w_2)$
$\varphi_v = Id$, $\varphi_w = Id$



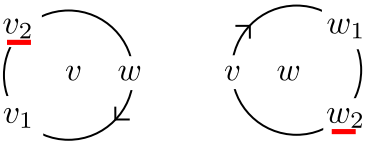
$ord_{\varphi_v}(v) = (w, v_1, v_2)$, $ord_{\varphi_w}(w) = (w_2, v, w_1)$
$\varphi_v = Id$, $\varphi_w = (123)$



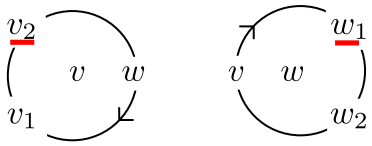
$ord_{\varphi_v}(v) = (w, v_1, v_2)$, $ord_{\varphi_w}(w) = (w_1, w_2, v)$
$\varphi_v = Id$, $\varphi_w = (132)$



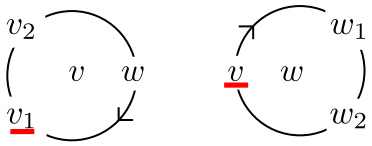
$ord_{\varphi_v}(v) = (v_2, w, v_1)$, $ord_{\varphi_w}(w) = (v, w_1, w_2)$
$\varphi_v = (123)$, $\varphi_w = Id$



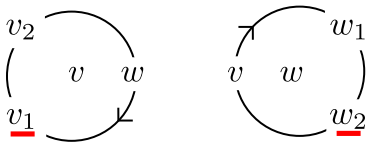
$ord_{\varphi_v}(v) = (v_2, w, v_1)$, $ord_{\varphi_w}(w) = (w_2, v, w_1)$
$\varphi_v = (123)$, $\varphi_w = (123)$



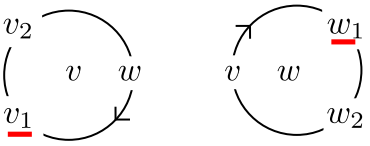
$ord_{\varphi_v}(v) = (v_2, w, v_1)$, $ord_{\varphi_w}(w) = (w_1, w_2, v)$
$\varphi_v = (123)$, $\varphi_w = (132)$



$ord_{\varphi_v}(v) = (v_1, v_2, w)$, $ord_{\varphi_w}(w) = (v, w_1, w_2)$
$\varphi_v = (132)$, $\varphi_w = Id$

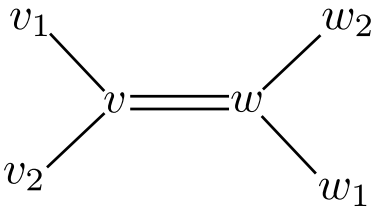


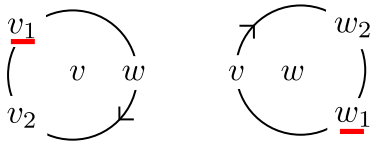
$ord_{\varphi_v}(v) = (v_1, v_2, w)$, $ord_{\varphi_w}(w) = (w_2, v, w_1)$
$\varphi_v = (132)$, $\varphi_w = (123)$



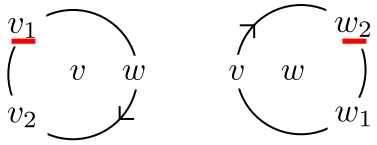
$ord_{\varphi_v}(v) = (v_1, v_2, w)$, $ord_{\varphi_w}(w) = (w_1, w_2, v)$
$\varphi_v = (132)$, $\varphi_w = (132)$

FIGURE 2.4 – Exemple de sommets de V^{DB} avec les listes ordonnées de leurs voisins et les ordres obtenus après l'application de l'ensemble des permutations paires autorisées.

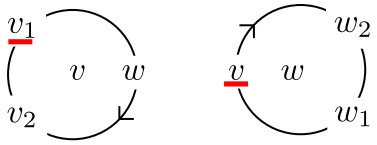




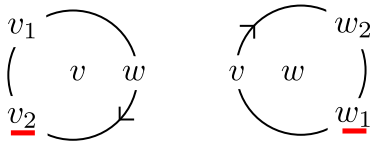
$ord_{\varphi_v}(v) = (v_1, w, v_2)$, $ord_{\varphi_w}(w) = (w_1, v, w_2)$
$\varphi_v = (12)$, $\varphi_w = (12)$



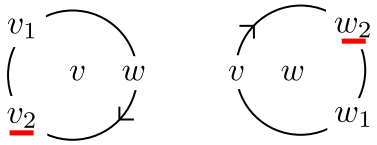
$ord_{\varphi_v}(v) = (v_1, w, v_2)$, $ord_{\varphi_w}(w) = (w_2, w_1, v)$
$\varphi_v = (12)$, $\varphi_w = (13)$



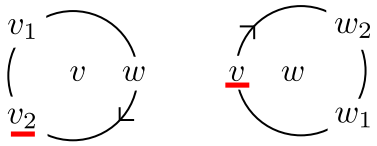
$ord_{\varphi_v}(v) = (v_1, w, v_2)$, $ord_{\varphi_w}(w) = (v, w_2, w_1)$
$\varphi_v = (12)$, $\varphi_w = (23)$



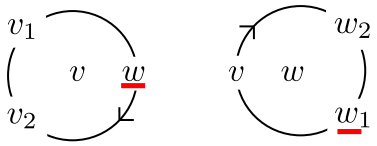
$ord_{\varphi_v}(v) = (v_2, v_1, w)$, $ord_{\varphi_w}(w) = (w_1, v, w_2)$
$\varphi_v = (13)$, $\varphi_w = (12)$



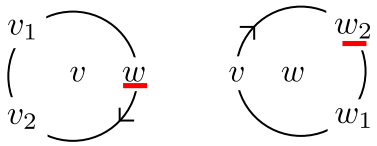
$ord_{\varphi_v}(v) = (v_2, v_1, w)$, $ord_{\varphi_w}(w) = (w_2, w_1, v)$
$\varphi_v = (13)$, $\varphi_w = (13)$



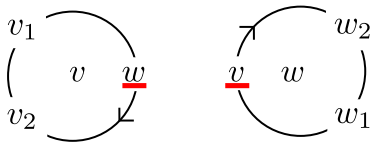
$ord_{\varphi_v}(v) = (v_2, v_1, w)$, $ord_{\varphi_w}(w) = (v, w_2, w_1)$
$\varphi_v = (13)$, $\varphi_w = (23)$



$ord_{\varphi_v}(v) = (w, v_2, v_1)$, $ord_{\varphi_w}(w) = (w_1, v, w_2)$
$\varphi_v = (23)$, $\varphi_w = (12)$



$ord_{\varphi_v}(v) = (w, v_2, v_1)$, $ord_{\varphi_w}(w) = (w_2, w_1, v)$
$\varphi_v = (23)$, $\varphi_w = (13)$



$ord_{\varphi_v}(v) = (w, v_2, v_1)$, $ord_{\varphi_w}(w) = (v, w_2, w_1)$
$\varphi_v = (23)$, $\varphi_w = (23)$

FIGURE 2.5 – Exemple de sommets de V^{DB} avec les listes ordonnées de leurs voisins et les ordres obtenus après l'application de l'ensemble des permutations impaires autorisées.

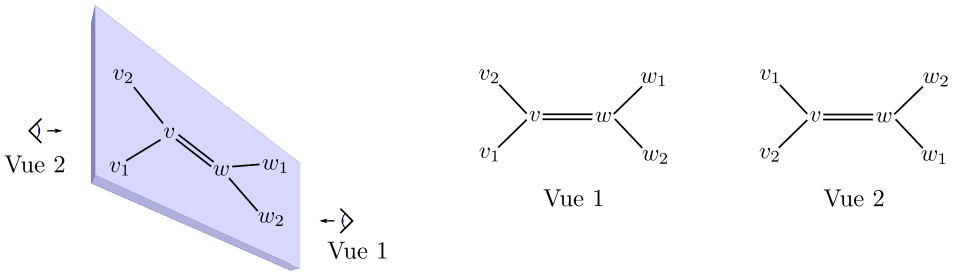


FIGURE 2.6 – Exemple de double liaison dans un plan et des deux vues obtenues en regardant ce plan d'un coté ou de l'autre. La vue 2 a subit une rotation de 180° afin que les sommets v et w soient dans le même ordre.

Remarque 2. *Les permutations autorisées autour des sommets de V^{PAC} sont les permutations paires. Comme les sommets de V^{PAC} ont quatre voisins, cela revient à autoriser les permutations circulaires sur trois voisins ainsi que les permutations formées de deux transpositions.*

La Figure 2.3 montre les différentes permutations autorisées pour les voisins d'un sommet de V^{PAC}, et les ordres obtenus en appliquant ces permutations. Lors de la définition de l'ordre des voisins d'un sommet de V^{PAC}, deux voisins sont sélectionnés arbitrairement (Définition 51). Comme les sommets de V^{PAC} possèdent quatre voisins, douze choix sont possibles. La Figure 2.3 illustre que ces douze choix correspondent aux douze ordres que l'on peut obtenir en appliquant les permutations autorisées par la Définition 52. Dans cette figure, le voisin par lequel on observe la molécule est situé en bas et le sommet par lequel on commence pour ordonner les trois restants est souligné en rouge.

Les Figures 2.4 et 2.5 montrent les différentes permutations autorisées pour les voisins d'un couple de sommets de V^{DB}, et les ordres obtenus en appliquant ces permutations. L'ordre des voisins d'un sommet v de V^{DB} est déterminé de manière à ce que ses voisins soient traversé dans le sens horaire lorsque l'on tourne autour v. Pour que cet ordre soit cohérent avec l'ordre des voisins de $w = n_{=}(v)$ on doit représenter v, w et leurs voisins dans un plan. Les ordres de la Figure 2.4 sont obtenus à partir de permutations paires sur les voisinages de v et w et correspondent donc à un même plongement dans le plan que la molécule initiale. Les ordres de la Figure 2.5 sont quant à eux obtenus par des permutations impaires et correspondent à une vision de la molécule sur la face du plan opposée à celui de la molécule initiale (voir Figure 2.6).

Théorème 3. *L'ensemble de fonction de réordonnancement $\Sigma^M = \{\Sigma^M_G,\ G \in \mathcal{OM}\}$ est une famille valide de fonctions de réordonnancement.*

Démonstration. Cette preuve est donnée en annexe à la section 5.1.4 (page 122). □

Comme l'ensemble défini dans la Définition 52 est une famille valide de fonctions de réordonnancement (Théorème 3), la relation d'équivalence d'ordres sur les graphes moléculaires localement ordonnés est une relation d'équivalence (Théorème 2).

Dans [JB99], la notion de graphes ordonnés est définie. Les principales différences entre ces graphes ordonnés et nos graphes localement ordonnés sont que l'ordre est déterminé pour tout les sommets (plutôt que pour un sous-ensemble de sommets) et que deux ordres sont équivalents s'il existe une permutation circulaire permettant de passer de l'un à l'autre (alors que nous définissons explicitement la famille valide de fonctions de réordonnancement). Notre définition est donc plus général que celle de [JB99].

2.3.3 Stéréo sommets

Dans la sous-section 1.2.2, nous avons fait la remarque qu'un carbone ayant quatre voisins, ou deux carbones liés par une liaison double ne forment pas nécessairement un centre stéréogène. Par définition, un centre stéréogène est un atome (ou un groupe d'atomes), tel que la permutation de la position de deux de ses voisins crée un stéréoisomère différent. Ainsi, si l'on considère un sommet de V^{PAC} ou V^{DB} qui ne correspond pas à un centre stéréogène, alors n'importe quelle permutation de leurs voisins va créer un graphe localement ordonné équivalent. On définit donc pour un sommet de V_{ord}, l'ensemble des isomorphismes entre le graphe moléculaire localement ordonné considéré et ce graphe ayant subi une permutation :

Définition 53. Ensemble des isomorphismes des centres stéréogènes potentiels

Soit $G = (\widehat{G} = (V, E, \mu, \nu), ord)$ *un graphe moléculaire localement ordonné. Soit* v *un sommet de* V_{ord}.

On note $\mathcal{F}_G^v$ l'ensemble des isomorphismes d'équivalence d'ordre f tel que :

$$\mathcal{F}_G^v = \bigcup_{\substack{(i,j)\in\{1,\dots,|N(v)|\}^2 \\ i\neq j}} \{f \mid f \in IsomEqOrd(G, \tau_{i,j}^v(G)) \text{ avec } f(v) = v\}$$

où $\tau_{i,j}^v$ est une fonction de réordonnancement égale à l'identité sur tous les sommets de V à part v pour lequel elle permute les sommets d'indices i et j dans $ord(v)$. Notez que $\tau_{i,j}^v$ n'appartient pas à Σ^M.

On définit alors un stéréo sommet (qui correspond à un centre stéréogène) comme un sommet pour lequel n'importe quelle permutation de deux de ses voisins produit un graphe localement ordonné non équivalent :

Définition 54. Stéréo sommets *Soit $G = (\widehat{G} = (V, E, \mu, \nu), ord)$ un graphe moléculaire localement ordonné. Un sommet v appartenant à V_{ord} est appelé stéréo sommet si :*

$$\mathcal{F}_G^v = \varnothing$$

On note $\mathcal{SV}(G)$ l'ensemble des stéréo sommets de G.

Proposition 7. *Un sommet v appartenant à V^{DB} est un stéréo sommet si et seulement si $n_=(v)$ est un stéréo sommet.*

Démonstration. Cette preuve est donnée en annexe à la section 5.1.5 (page 125). □

D'après la proposition précédente, deux sommets représentant des carbones liés par une liaison double sont soit tous les deux stéréo sommets, soit aucun des deux n'est un stéréo sommet (ce qui est cohérent avec le fait que chimiquement, uniquement le couple de carbones peut être un centre stéréogène, et pas les carbones pris séparément). On doit donc considérer leur propriété stéréo simultanément. On introduit les notations suivantes (illustrées dans la Figure 2.7) :

Définition 55. Ensemble de stéréo sommets liés

Soit s un stéréo sommet. On définit son ensemble de stéréo sommets liés $kernel(s)$ par :

$$kernel(s) = \begin{cases} \{s\} \text{ si } s \in V^{PAC} \\ \{s, n_=(s)\} \text{ si } s \in V^{DB} \end{cases}$$

Définition 56. Étoile de stéréo sommets

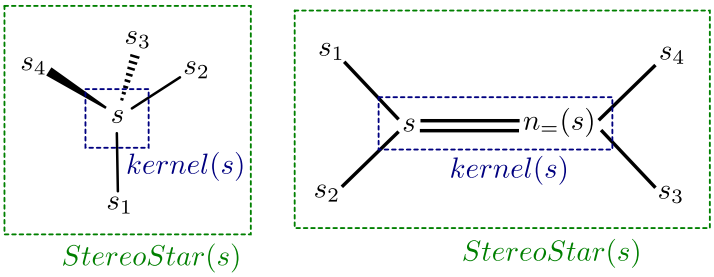


FIGURE 2.7 – Exemple d'ensembles et d'étoiles de stéréo sommets.

Pour s un stéréo sommet on définit l'ensemble $StereoStar(s)$ par :

$$StereoStar(s) = \begin{cases} N(s) \cup \{s\} \ si \ s \in V^{PAC} \\ N(s) \cup N(n_=(s)) \ si \ s \in V^{DB} \end{cases}$$

Définition 57. Ensemble des voisins des stéréo sommets

Soit s un stéréo sommet. Son ensemble de voisins $StereoStar^*(s)$ est défini par :

$$StereoStar^*(s) = \begin{cases} N(s) \ si \ s \in V^{PAC} \\ N(s) \cup N(n_=(s)) - \{s, n_=(s)\} \ si \ s \in V^{DB} \end{cases}$$

On peut remarquer que :

$$StereoStar^*(s) = StereoStar(s) - kernel(s)$$

2.4 Lien avec la définition de la chiralité

La chiralité est un cas particulier de stéréoisomérie (Sous-section 1.2.1). Nous montrons ici que notre définition des stéréo sommets est cohérente avec la définition de la chiralité donnée par Petitjean [Pet10] dans la Définition 39. L'exemple le plus simple de molécule chirale étant un carbone avec quatre voisins différents, nous nous focalisons sur cet exemple dans cette partie. Dans cette configuration, les quatre voisins sont situés sur les sommets d'un tétraèdre et le carbone est situé au centre de ce tétraèdre.

Selon la définition de Petitjean [Pet10], un objet est chiral s'il n'existe pas de transformation indirecte qui le transforme en un objet identique.

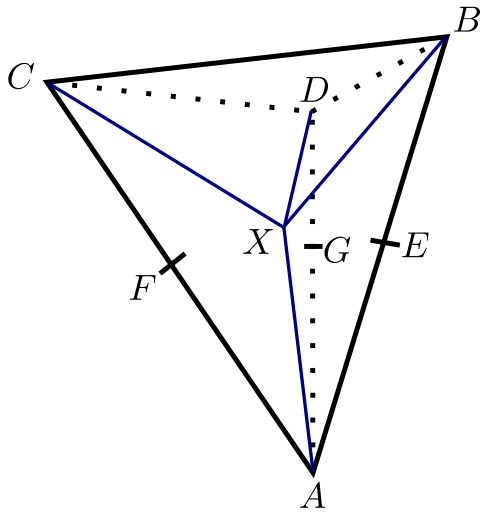


FIGURE 2.8 – Tétraèdre composé des sommets A, B, C et D et de centre X. Les sommets E, F et G sont respectivement les milieux des segments $[AB]$, $[AC]$ et $[AD]$.

On commence par considérer un tétraèdre dans un espace euclidien, avec tous ses sommets identiques (c'est-à-dire sans étiquette sur ses sommets). Cet objet possède vingt-quatre transformations le laissant invariant. En notant A, B, C et D les sommets du tétraèdre et X son centre (Figure 2.8), les transformations laissant le tétraèdre invariant sont :

1. L'identité.
2. Les rotations d'angles 120° et 240° autour des axes (AX), (BX), (CX) et (DX).
3. Les rotations d'angle 180° autour des axes (EX),(FX), et (GX) où E, F et G sont respectivement les milieux des segments $[AB]$, $[AC]$ et $[AD]$.
4. Les six réflexions par rapport aux plans formés par les points (A, B, X), (A, C, X), (A, D, X), (B, C, X), (B, D, X) et (C, D, X).
5. Les combinaisons d'une rotation d'angle 90° ou 270° autour d'une droite d et d'une réflexion par rapport au plan orthogonal à d et passant par X. Ces droites sont les axes (EX), (FX) et (GX).

On peut voir chaque transformation comme une permutation des sommets du tétraèdre. En associant respectivement aux sommets A, B, C et D les positions $1, 2, 3$ et 4, les transformations précédemment évoquées correspondent respectivement aux permutations :

1. L'identité.
2. Les permutations (234), (243), (134), (143), (124), (142), (123) et (132).
3. Les permutations (12)(34), (13)(24) et (14)(23).
4. Les permutations (34), (24), (23), (14), (13) et (12).
5. Les permutations (1324), (1423), (1234), (1432), (1342) et (1243).

Les transformations indirectes de l'espace euclidien sont celles impliquant un nombre impair de réflexions. Parmi les transformations laissant le tétraèdre invariant, les transformations indirectes sont celles des points 4 et 5. Ainsi, si les étiquettes des voisins du carbone interdisent ces transformations, la molécule sera chirale. Ces transformations correspondent aux permutations de la forme (ij) ou $(ijkl)$. On peut remarquer que ces permutations sont impaires, et que les permutations correspondant aux autres transformations du tétraèdre sont toutes paires. Ainsi, un tétraèdre est chiral selon la définition 39, s'il n'existe pas de permutation impaire de ses sommets, le transformant en lui-même.

Notre définition des stéréo sommets 54 et la Définition 53 impliquent qu'un sommet v représentant un carbone potentiellement asymétrique (Définition 49) est un stéréo sommet si :

$$\nexists (i,j) \in \{1,\dots,4\}^2 \ t.q \ \exists f \in \text{IsomEqOrd}(\tau^v_{i,j}(G), G) \text{ avec } f(v) = v$$

D'après la Définition 47 :

$$f \in \text{IsomEqOrd}(\tau^v_{i,j}(G), G) \Rightarrow \exists \sigma \in \Sigma_G, \ \sigma(\tau^v_{i,j}(G)) \underset{o}{\simeq} G$$

Enfin, comme $v \in V^{PAC}$ et d'après la Définition 52, si σ existe alors $\sigma(v)$ sera paire. Une transposition étant impaire et la composition d'une permutation paire avec une permutation impaire étant impaire, $\sigma(v) \circ \tau^v_{i,j}$ impaire. Ainsi le sommet v représentant un carbone potentiellement asymétrique est un stéréo sommet s'il n'existe pas de permutation impaire de son ordre, transformant le graphe moléculaire localement ordonné en lui-même.

Notre définition des stéréo sommets, appliquée aux molécules possédant un carbone asymétrique, est donc cohérente avec la définition de la chiralité donnée par Petitjean [Pet10].

2.5 Conclusion

Le graphe moléculaire ne permet pas de discerner les stéréoisomères. Cependant, les graphes moléculaires localement ordonnés présentés dans ce chapitre ajoutent aux graphes moléculaires une notion d'ordre local qui permet de différencier les stéréoisomères. De plus, la relation d'équivalence d'ordre entre graphes moléculaires localement ordonnés permet d'identifier sans ambiguïté deux graphes moléculaires localement ordonnés représentant un même stéréoisomère.

Finalement, le cadre théorique des graphes localement ordonnés permet de définir les stéréo sommets, qui encodent les centres stéréogènes.

Chapitre 3

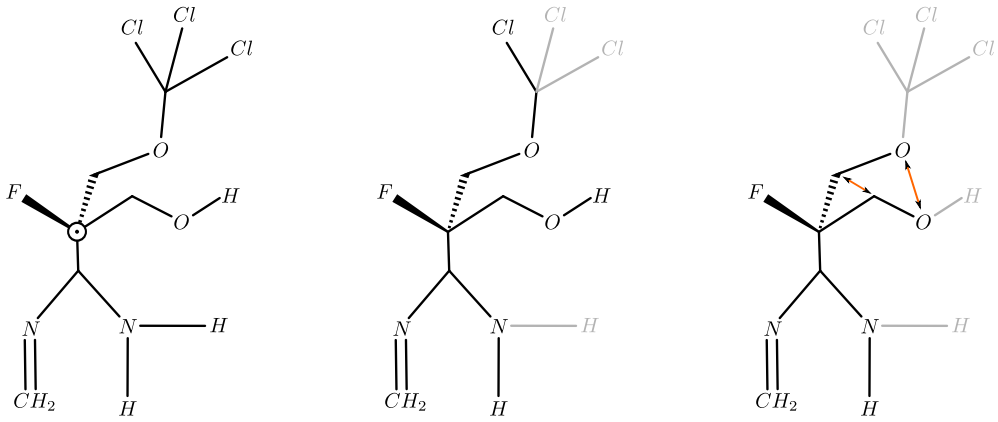
Caractérisation de la stéréoisomérie et mesure de similarité stéréo

Sommaire

3.1 Introduction

Les graphes localement ordonnés présentés dans le chapitre 2 permettent de représenter les stéréoisomères. Nous voulons maintenant construire une mesure de similarité entre ces graphes localement ordonnés.

D'un point de vue local, la stéréoisomérie est due à la présence de centres stéréogènes, qui sont des atomes (ou des groupements d'atomes) tels que la permutation de la position de leurs voisins forme un stéréoisomère différent.



(a) (b) (c)

FIGURE 3.1 – Un graphe localement ordonné (a) avec un stéréo sommet ⊙. La suppression de quelques sommets peut ne pas modifier le fait que le sommet soit un stéréo sommet (b), mais la suppression de trop de sommets peut faire perdre à un sommet sa propriété stéréo (c).

Afin d'avoir une mesure de similarité des stéréoisomères, nous allons chercher à comparer leurs centres stéréogènes. Ces atomes sont représentés par des stéréo sommets (Définition 54) lorsque les stéréoisomères sont encodés par des graphes localement ordonnés. On veut donc comparer les stéréo sommets.

Les centres stéréogènes caractérisent les stéréoisomères localement, cependant la Définition 54 utilise tout le graphe pour caractériser un stéréo sommet. Afin de caractériser les stéréo sommets de façon locale, la Section 3.2 introduit la notion de stéréo sous-graphe minimal. Puis, dans la Section 3.3, nous montrons que la définition des stéréo sous-graphes minimaux a bien les propriétés désirées.

La Section 3.4 parle de la complexité du calcul des stéréo sous-graphes minimaux. La Section 3.5 présente un noyau fondé sur les stéréo sous-graphes minimaux et finalement la Section 3.6 montre l'efficacité de ce noyau sur des problèmes de prédiction de propriétés moléculaires.

3.2 Stéréo sous-graphes minimaux

On considère un stéréo sommet s. Notons que dans certaines configurations, le retrait de certains sommets du graphe ne change pas le fait que s soit un stéréo sommet. Par exemple dans la Figure 3.1(b), le fait de retirer les sommets en

gris ne change pas le fait que le sommet du milieu soit un stéréo sommet. Mais si l'on retire trop de sommets, alors un isomorphisme d'ordres peut exister entre le graphe G et sa version permutée $\tau^v_{i,j}(G)$ (Figure 3.1(c)).

Afin de caractériser localement un stéréo sommet s, nous allons chercher un sous-graphe induit H de G, assez grand pour pouvoir caractériser s, mais assez petit pour ne garder que les informations nécessaires à la caractérisation de s. Ce sous-graphe est appelé le stéréo sous-graphe minimal de s.

Pour définir le stéréo sous-graphe minimal, nous commençons par définir formellement quelles propriétés doit avoir un sous-graphe de G afin de caractériser un stéréo sommet.

Définition 58. Sous-graphe caractérisant un stéréo sommet

Soit un graphe localement ordonné moléculaire $G = (\widehat{G} = (V, E, \mu, \nu), ord)$ et S un sous-graphe de G. Soit s un stéréo sommet de G. On dit que la propriété de stéréoisomérie de s est capturée par S si :

- $StereoStar(s) \subset V_S$.
- $\mathcal{F}^s_S = \varnothing$.

où $\mathcal{F}^s_S$ est l'ensemble des isomorphismes d'équivalence d'ordres défini dans la Définition 53 en remplaçant le graphe G par son sous-graphe S.

Avec les mêmes arguments que pour la preuve de la Proposition 7, nous pouvons montrer que si la propriété de stéréoisomérie d'un stéréo sommet s est capturée par un sous-graphe S et que s appartient à V^{DB} alors la propriété de stéréoisomérie de $n_=(s)$ est aussi capturée par S.

Un couple de carbones connectés par une liaison double a donc besoin d'un seul sous-graphe pour le caractériser.

Si un sous-graphe H, tel que $StereoStar(s)$ soit inclus dans son ensemble de sommets V_H, ne capture pas la propriété de stéréoisomérie d'un stéréo sommet s, c'est qu'il existe un isomorphisme entre H et $\tau^s(H)$. Nous définissons maintenant l'ensemble des sommets induisant cet isomorphisme :

Définition 59. Ensemble des sommets induisant un isomorphisme

Soit un graphe localement ordonné moléculaire $G = (\widehat{G} = (V, E, \mu, \nu), ord)$ et s un stéréo sommet de G. Soit H un sous-graphe de G qui ne capture pas *la propriété de stéréoisomérie de s et tel que $StereoStar(s)$ soit inclus dans les sommets de H.*

Soit f un isomorphisme d'équivalence d'ordres de $\mathcal{F}^s_H$.

On définit $\mathcal{E}_f^H$ *comme l'ensemble des sommets induisant l'isomorphisme* f *(Figure 3.2) :*

$$\mathcal{E}_f^H = \{v \in V_H \mid \exists p = (v_0, \ldots, v_q) \in H \text{ avec } v_0 \in kernel(s),\ q > 0,\ v_q = v \text{ et } f(v_1) \neq v_1\} \quad (3.1)$$

où V_H *est l'ensemble des sommets de* H *et* $(v_0, \ldots, v_q)$ *est un chemin élémentaire de* H*.*

De plus on note $\mathcal{E}^H$ *l'ensemble des sommets induisant les isomorphismes de* $\mathcal{F}_H^s$ *:*

$$\mathcal{E}^H = \bigcup_{f \in \mathcal{F}_H^s} \mathcal{E}_f^H$$

Notons que du fait de la Définition 58, $\mathcal{F}_H^s$ ne peut pas être vide.

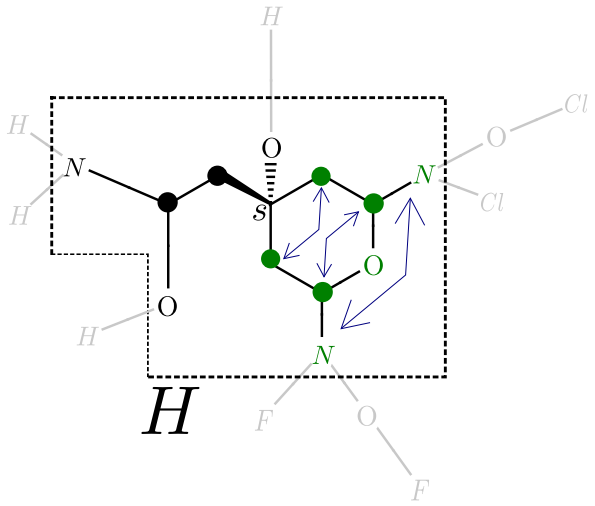


FIGURE 3.2 – L'ensemble des sommets de $\mathcal{E}_f^H$ est responsable de la présence de l'isomorphisme d'équivalence d'ordres f de $\mathcal{F}_H^s$.

Remarque 3. *Dans la Définition 59, on considère un isomorphisme d'équivalence d'ordres* f *de* $\mathcal{F}_H^s$*. Selon la définition de* $\mathcal{F}_H^s$*,* f *est un isomorphisme d'équivalence d'ordres entre* H *et* $\tau_{i,j}^s(H)$ *et* $f(s)$ *est égal à* s*. Comme* $IsomEqOrd(H, \tau_{i,j}^v(H))$ *est inclus dans* $Isom(\hat{H}, \hat{H})$*,* f *appartient à* $Isom(\hat{H}, \hat{H})$*.*

Selon la Proposition 6, pour tout couple de graphes localement ordonnés moléculaires G *et* G'*, tel qu'il existe un isomorphisme* f *entre* $\hat{G}$ *et* $\hat{G}'$*, on a :*

$$\forall v \in V^{DB},\ f(n_=(v)) = n_=(f(v))$$

Ainsi, si s appartient à V^{DB}, $f(n_=(s))$ est égal à $n_=(f(s))$, et donc il est aussi égal à $n_=(s)$, puisque $f(s)$ est égal à s. Ainsi, tout sommet v appartenant à $kernel(s)$ est associé à lui-même par f.

L'ensemble des sommets $\mathcal{E}_f^H$ induisant l'isomorphisme f peut donc être réécrit de la manière suivante :

$$\mathcal{E}_f^H = \{v \in V_H \mid \exists p = (v_1, \ldots, v_q) \in H \text{ avec } v_1 \in StereoStar^*(s),\ q \geq 1,\ v_q = v \text{ et } f(v_1) \neq v_1\} \quad (3.2)$$

Nous avons maintenant tous les outils nécessaires à la définition des stéréo sous-graphes minimaux.

Définition 60. Stéréo sous-graphes minimaux

Soit $G = (\widehat{G} = (V, E, \mu, \nu), ord)$ un graphe localement ordonné moléculaire et s un stéréo sommet de G. On considère la suite $(H_s^k)_{k\in\mathbb{N}}$ de sous-graphes induits de G définie par :

- $V_{H_s^0} = StereoStar(s)$
- $V_{H_s^{k+1}} = V_{H_s^k} \cup N(\mathcal{E}^{H_s^k})$.

On note alors S_s la limite de cette suite :

$$S_s = \lim_{k\to+\infty} H_s^k$$

Le sous-graphe induit S_s est appelé stéréo sous graphe minimal de s. On dit que s est le stéréo sommet de S_s (si s appartient à V^{DB}, le stéréo sommet de S_s est arbitrairement choisi entre s et $n_=(s)$). On note $\mathcal{H}(G)$ la collection des stéréo sous-graphes minimaux de G.

La Figure 3.3 montre un exemple de stéréo sous-graphe minimal.

3.3 Preuve d'algorithme

3.3.1 Convergence et caractérisation du stéréo sommet

Un stéréo sous-graphe minimal est défini comme la limite de la suite des $(H_s^k)_{k\in\mathbb{N}}$ (Définition 60). Afin que le stéréo sous-graphe minimal d'un stéréo sommet soit toujours défini, il faut que la suite des sous-graphes induits $(H_s^k)_{k\in\mathbb{N}}$ converge.

Proposition 8. *La suite $(H_s^k)_{k\in\mathbb{N}}$ converge.*

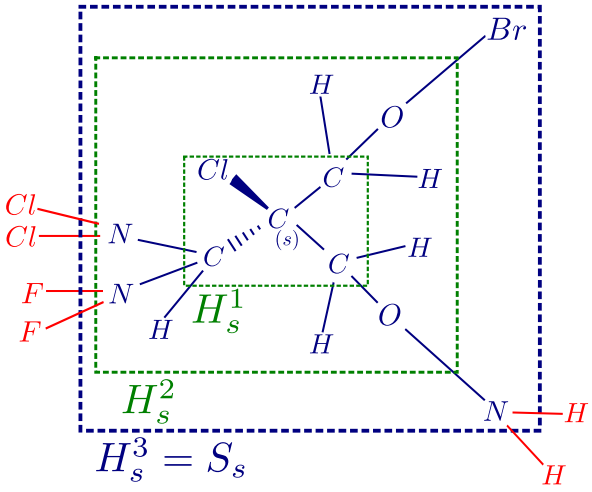


FIGURE 3.3 – Un carbone asymétrique et la suite de sous-graphes $(H_s^k)_{k\in\mathbb{N}}$ définie dans la Définition 60. Son stéréo sous-graphe minimal est $S_s = H_s^3$.

Démonstration. Comme $(H_s^k)_{k\in\mathbb{N}}$ est une suite de sous-graphes de G, on sait que :

$$\forall k \in \mathbb{N},\ V_{H_s^k} \subset V$$

La suite d'ensembles de sommets $(V_{H_s^k})_{k\in\mathbb{N}}$ est donc majorée par V.

Par Définition 60, on a

$$\forall k \in \mathbb{N},\ V_{H_s^{k+1}} = V_{H_s^k} \cup N(\mathcal{E}^{H_s^k})$$

Donc la suite d'ensembles de sommets $(V_{H_s^k})_{k\in\mathbb{N}}$ est croissante.

Comme $(V_{H_s^k})_{k\in\mathbb{N}}$ est majorée et croissante, $(V_{H_s^k})_{k\in\mathbb{N}}$ converge.

La suite de graphes $(H_s^k)_{k\in\mathbb{N}}$ est une suite de sous-graphes induits et la suite d'ensembles de sommets $(V_{H_s^k})_{k\in\mathbb{N}}$ correspondant à ces sous-graphes converge. Nous en déduisons que $(H_s^k)_{k\in\mathbb{N}}$ converge. □

Corollaire 1. *La suite d'ensembles de sommets $(V_{H_s^k})_{k\in\mathbb{N}}$ est une suite d'ensembles finis. Ainsi, la limite de cette suite est atteinte et donc la limite de la suite $(H_s^k)_{k\in\mathbb{N}}$ est aussi atteinte :*

$$\exists n \in \mathbb{N} \mid H_s^n = S_s$$

Dans la Définition 58, nous avons défini formellement quels critères doivent remplir un sous-graphe afin de caractériser un stéréo sommet. Le stéréo sous-graphe minimal est présenté comme le plus petit sous graphe respectant ces critères. Afin de prouver cette affirmation, nous allons montrer qu'un sous-graphe de la suite, différent de la limite, n'est pas suffisant pour caractériser

un stéréo sommet. Puis nous prouverons que le stéréo sous-graphe minimal caractérise son stéréo sommet.

Proposition 9. *Pour tout entier naturel k tel que $V_{H_s^k}$ soit différent de V_{S_s}, la propriété de stéréoisomérie de s n'est pas capturée par H_s^k.*

Démonstration. Soit un entier naturel k tel que $V_{H_s^k}$ soit différent de V_{S_s}.

Comme H_s^{k+1} est construit seulement en utilisant H_s^k, on a :

$$V_{H_s^k} = V_{H_s^{k+1}} \Rightarrow V_{H_s^k} = V_{S_s}$$

Comme $V_{H_s^k}$ est différent de V_{S_s}, on a par contraposée que $V_{H_s^k}$ est différent de $V_{H_s^{k+1}}$.

On en déduit que

$$N(\mathcal{E}^{H_s^k}) = \bigcup_{f \in \mathcal{F}^s_{H_s^k}} N(\mathcal{E}_f^k) \neq \varnothing$$

Il existe donc au moins un isomorphisme d'équivalence d'ordres f appartenant à $\mathcal{F}^s_{H_s^k}$ tel que $\mathcal{E}_f^k$ ne soit pas vide. $\mathcal{F}^s_{H_s^k}$ n'est donc pas vide.

Donc par Définition 58 la propriété de stéréoisomérie de s n'est pas capturée par H_s^k. □

Afin de prouver que la propriété de stéréoisomérie d'un stéréo sommet est capturée par son stéréo sous-graphe minimal, nous avons besoin de trois lemmes. Ces lemmes, ainsi que leurs preuves sont donnés dans la section 5.2 en annexes.

Théorème 4. *La propriété de stéréoisomérie de s est capturée par S_s.*

Démonstration. On note n un entier naturel tel que $H_s^n = S_s$. Cet entier existe d'après le corollaire 1.

La suite $(H_s^k)_{k\in\mathbb{N}}$ commence par $V_{H_s^0} = StereoStar(s)$. Comme cette suite est croissante, $V_{H_s^0}$ est inclus dans $V_{H_s^n}$, et donc la première condition de la Définition 58 est vérifiée.

On doit maintenant montrer que $\mathcal{F}^s_{H_s^n}$ est vide.

Supposons que $\mathcal{F}^s_{H_s^n}$ ne soit pas vide.

Soit f appartenant à $\mathcal{F}^s_{H_s^n}$.

Par le Lemme 5 on a

$$N(\mathcal{E}_f^{H_s^n}) \not\subset V_{H_s^n}$$

Donc

$$V_{H_s^{n+1}} = V_{H_s^n} \cup \bigcup_{f \in \mathcal{F}^s_{H_s^k}} N(\mathcal{E}_f^{H_s^k}) \neq V_{H_s^n}$$

Ceci est en contradiction avec le fait que H_s^n soit égal à la limite de la suite $(H_s^k)_{k\in\mathbb{N}}$.

L'ensemble $\mathcal{F}^s_{H_s^n}$ est donc vide.

En conclusion, la propriété de stéréoisomérie de s est capturée par $H_s^n = S_s$.

□

3.3.2 Minimalité du stéréo sous-graphe "minimal"

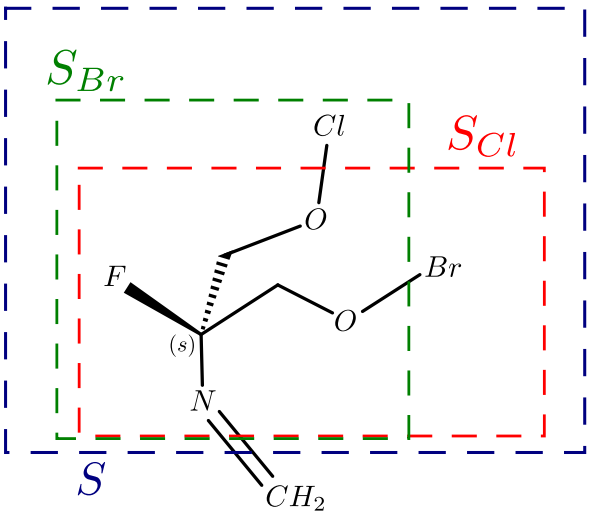


FIGURE 3.4 – Le stéréo sous graphe minimal du stéréo sommet noté s est S. Cependant les sous-graphes S_{Cl} et S_{Br} capturent aussi la stéréo propriété de s.

On considère le graphe moléculaire localement ordonné G de la Figure 3.4. Il possède un stéréo sommet noté s. Le stéréo sous-graphe minimal de ce stéréo sommet est S.

Remarquons que l'on peut trouver des sous-graphes de G plus petits que S capturant la propriété de stéréoisomérie de s. Par exemple, le sous-graphe S_{Br}, obtenu en enlevant l'atome de brome Br de S, capture la propriété de stéréoisomérie de s. En effet, $StereoStar(s)$ est inclus dans S_{Br} et il n'existe pas d'isomorphisme d'équivalence d'ordres entre S_{Br} et $\tau^s_{i,j}(S_{Br})$, pour i et j

différent et compris entre 1 et 4 (Définition 58). De la même manière, le sous-graphe S_{Cl}, obtenu en enlevant l'atome de chlore Cl de S, capture lui aussi la propriété de stéréoisomérie de s.

Cependant, ces graphes capturent la propriété de stéréoisomérie de s uniquement grâce à l'absence de Cl pour S_{Br} et l'absence de Br dans S_{Cl}, ce qui empêche d'associer l'un des chemins CO à l'autre. Ces graphes encodent donc la propriété de stéréoisomérie implicitement, grâce à un manque d'information.

De plus, il n'y a aucune raison de privilégier l'un de ces deux graphes pour caractériser le stéréo sommet s. Ce dernier point peut induire un biais si l'on souhaite comparer deux graphes grâce à leurs collections de stéréo sous-graphes minimaux $\mathcal{H}(G)$.

Le stéréo sous-graphe minimal introduit dans la Définition 60 n'est pas minimal dans le sens qu'il contient le moins de sommets possibles. Cependant, contrairement aux sous-graphes S_{Br} et S_{Cl}, il encode le fait que le sommet s est un stéréo sommet à cause de la différence d'étiquettes entre l'atome de chlore et l'atome de brome et non grâce à l'absence de l'un de ces deux atomes.

3.4 Complexité

Les stéréo sous-graphes minimaux sont définis comme la limite d'une suite de sous-graphes induits (Définition 60). Un algorithme calculant un stéréo sous-graphe minimal peut donc être directement déduit de sa définition (Algorithme 2).

Algorithme 2 : Construction d'un stéréo sous-graphe minimal

Données : Un graphe moléculaire localement ordonné $G = (\widehat{G} = (V, E, \mu, \nu), ord) \in \mathcal{OM}$ et un de ces stéréo sommet $s \in \mathcal{SV}(G)$

Résultat : Le stéréo sous-graphe minimal S_s de s

$H_s^0 \leftarrow StereoStar(s)$;

$(\mathcal{F}^s_{H_s^0}, \mathcal{E}^{H_s^0}) \leftarrow getIsomorphism(s, H_s^0)$;

$k \leftarrow 0$;

tant que $\mathcal{F}^s_{H_s^k} \neq \varnothing$ **faire**

  $k \leftarrow k + 1$;

  $V_{H_s^k} \leftarrow V_{H_s^{k-1}} \cup N(\mathcal{E}^{H_s^{k-1}})$;

  $(\mathcal{F}^s_{H_s^k}, \mathcal{E}^{H_s^k}) \leftarrow getIsomorphism(s, H_s^k, \mathcal{F}_{H_s^{k-1}})$;

$S_s \leftarrow H_s^k$;

Le nombre d'itération de cet algorithme est borné par le diamètre du stéréo sous-graphe minimal. Cependant, à chaque itérations, la fonction $getIsomorphism(s, H)$ calcule les isomorphismes d'équivalence d'ordres $\mathcal{F}_H^s$ et l'ensemble de sommets $\mathcal{E}^H$ induisant ces isomorphismes. La recherche d'isomorphismes entre graphes est un problème NP [GJ79]. En revanche, la complexité exacte de ce problème reste un problème ouvert : si personne n'a trouvé d'algorithme polynomial pour résoudre ce problème, personne n'a réussi à prouver qu'il est NP-complet.

Cependant, dans le calcul du stéréo sous-graphe minimal, nous pouvons initialiser la recherche d'isomorphismes entre H_s^k et $\tau_{i,j}^s(H_s^k)$ par les isomorphismes entre H_s^{k-1} et $\tau_{i,j}^s(H_s^{k-1})$.

L'algorithme présenté par [BGP+13] est l'un des algorithmes obtenant les meilleurs temps de calcul. De plus, il peut être facilement adapté afin de prendre en compte l'ordre des graphes localement ordonnés, et peut être initialisé en utilisant des isomorphismes entre les sous-graphes des graphes considérés. Nous avons donc utilisé cet algorithme dans la construction de stéréo sous-graphes minimaux.

3.5 Définition du noyau

Afin de comparer deux graphes localement ordonnés G et G', on compare leurs collections de stéréo sous-graphes minimaux $\mathcal{H}(G)$ et $\mathcal{H}(G')$.

Pour cela, on souhaite pouvoir comparer rapidement deux stéréo sous-graphes minimaux. On associe donc à chaque stéréo sous-graphe minimal S un code c_S, obtenu en utilisant l'algorithme de dénomination [WD74], présenté dans la section 1.2.2, sur S.

On peut alors représenter la collection de stéréo sous-graphes minimaux $\mathcal{H}(G)$ par un vecteur $t(G)$ représentant le nombre d'occurrences d'un stéréo sous-graphe minimal $S \in \mathcal{H}(G)$:

$$t(G) = (t_S(G))_{S \in \mathcal{H}(G)} \text{ avec } t_S(G) = |\{H \in \mathcal{H}(G) | H \underset{\Sigma}{\simeq} S\}| \tag{3.3}$$

La comparaison de ces vecteurs pour deux graphes localement ordonnés G et G' est alors utilisée afin de définir un noyau entre G et G' :

$$k_{stereo}(G, G') = \sum_{S \in \mathcal{H}(G) \cap \mathcal{H}(G')} k(t_S(G), t_S(G')) \tag{3.4}$$

où k est un noyau entre les nombres d'occurrences d'un stéréo sous-graphe minimal S dans G et G'. Le noyau k correspond à un noyau entre valeurs réelles

TABLE 3.1 – Propriétés du premier jeu de données

Nombre de molécules	35
Taille moyenne des molécules	21
Taille maximale des molécules	32
Taille minimale des molécules	14
Taille moyenne des stéréo sous-graphes minimaux	10
Taille maximale des stéréo sous-graphes minimaux	13
Taille minimale des stéréo sous-graphes minimaux	6

pouvant être choisi pour chaque application. Le noyau k_{stereo} est appelé noyau stéréo.

Proposition 10. *Le noyau stéréo (3.4) est défini positif.*

Démonstration. Cette preuve est donnée en annexe à la section 5.3.1 (page 137). □

3.6 Expérimentations

Afin de tester les performances de notre noyau stéréo, nous l'avons testé sur des problèmes de régression. Pour les deux jeux de données, l'algorithme de machine à vecteurs de support pour la régression, présenté dans la section 1.1.4, est utilisé.

Le premier jeu de données est issu de [ZRPJ07]. Ce jeu de données contient 90 molécules, mais nous n'utilisons que 35 d'entre elles. La majorité des molécules de ce jeu de données ne possèdent qu'un seul centre stéréogène, et pour 55 d'entre elles le stéréo sous-graphe minimal associé à ce centre stéréogène est unique. Notre noyau ne peut donc pas différencier ces molécules et donc nous ne sélectionnons que les 35 molécules restantes.

La propriété à prédire dans ce jeu de données est le pouvoir rotatoire des molécules. Le pouvoir rotatoire d'une molécule est une propriété physique qui mesure l'angle de déviation d'une lumière polarisée passant à travers une solution de cette molécule. Dans ce jeu de données, les pouvoirs rotatoires varient de $-89°$ à $78°$ avec un écart type de 38. Le tableau 3.1 montre diverses statistiques de ce jeu de données, telles que la taille moyenne des molécules et la taille moyenne des stéréo sous-graphes minimaux.

Certaines molécules ont une similarité non nulle avec un nombre restreint de molécules. Une division en ensembles de test, de validation et d'apprentissage

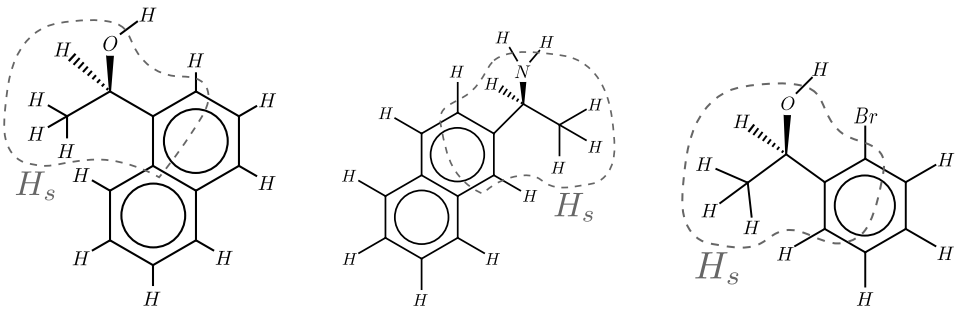
TABLE 3.2 – Résultats de la prédiction pour le premier jeu de données

	Méthodes	Erreur Moyenne	RMSE
1 -	Noyau de motifs d'arbres [MV09]	23.6	34.1
2 -	Noyau de treelets [Gaü13]	17.6	26.2
3 -	Noyau de motifs d'arbres stéréo [BUT+10]	*17.2*	*24.2*
4 -	Noyau stéréo	**10.7**	**14.8**

n'est pas possible, car des molécules pourraient avoir une similarité nulle avec toutes les molécules de l'ensemble d'apprentissage. Chaque molécule est donc prédite en utilisant le reste du jeu de données comme ensemble d'apprentissage. Les différents paramètres sont choisis grâce à une grille de recherche. Les types de sous noyaux (le k dans 3.4) testés sont les noyaux linéaires, binaires, intersection et gaussien. Les différentes valeurs testées pour les paramètres de l'algorithme de machine à vecteurs de support pour la régression sont $0.01, 0.02, 0.05, 0.1, 0.2, 0.5, \ldots, 1000$ pour le C et $\sigma/5, \sigma/10, \sigma/20$ pour le ϵ, où σ correspond à l'écart type des valeurs à prédire. Comme il n'y a pas d'ensemble de validation, les paramètres sélectionnés pour chaque méthode sont ceux obtenant la plus petite erreur quadratique.

Le tableau 3.2 montre les erreurs moyennes et les racines carrés des erreurs quadratiques (RMSE) obtenues par notre noyau, ainsi que celles obtenues par trois autres noyaux [MV09, BUT+10, Gaü13]. Les deux premières lignes montrent les résultats obtenus par des méthodes ne prenant pas en compte la stéréoisomérie des molécules [MV09, Gaü13]. Le pouvoir rotatoire est une propriété liée à la stéréoisomérie des molécules, deux énantiomères ont des pouvoirs rotatoires opposés. Ainsi, les méthodes ne différenciant pas les stéréoisomères obtiennent de mauvais résultats. Pour le noyau de motifs d'arbres, l'erreur quadratique moyenne est proche de l'écart type de la propriété. L'adaptation du noyau de motifs d'arbres à la stéréoisomérie [BUT+10] n'obtient des résultats que très légèrement supérieurs à ceux du noyau de treelets. Ceci peut être expliqué par le fait que cette méthode combine sans pondération les motifs possédant une information sur la stéréoisomérie et les motifs n'en possédant pas. Le pouvoir rotatoire étant uniquement lié à la stéréoisomérie, les motifs ne possédant pas d'informations sur la stéréoisomérie agissent comme du bruit et dégradent la prédiction. A l'inverse, le noyau stéréo n'utilise que des informations concernant la stéréoisomérie. Le noyau stéréo obtient donc les meilleurs résultats.

La Figure 3.5 montre trois molécules de ce jeu de données. La molécule la plus semblable à celle de la Figure 3.5(a) est celle de la Figure 3.5(b)



(a) Pouvoir rotatoire : 78 (b) Pouvoir rotatoire : −57 (c) Pouvoir rotatoire : 54

FIGURE 3.5 – Exemples de molécules avec leurs pouvoir rotatoire. Les stéréo sous-graphe minimaux H_s sont entourés en gris.

selon le noyau de treelets. En effet, les différences entre ces molécules sont uniquement la configuration autour du carbone asymétrique et l'oxygène de la première molécule qui est remplacé par un azote. Ainsi, tous les treelets, à part ceux incluant l'oxygène ou l'azote, sont identiques. La troisième molécule (Figure 3.5(c)), est la plus similaire à la première selon le noyau stéréo. Comme le noyau de treelet considère comme similaire des molécules ayant des pouvoirs rotatoires très différents, il ne peut pas correctement prédire cette propriété. De la même manière, de nombreux motifs d'arbres sont communs entre les deux premières molécules (notamment à cause des deux cycles aromatiques). Ainsi, le noyau de motifs d'arbres adapté à la stéréoisomérie considère aussi ces molécules comme proches.

Le second jeu de données est composé de 69 dérivés synthétiques de la vitamine D. Ce jeu de données est issu de [BUT+10]. La propriété à prédire est l'activité biologique des molécules. Il y a en moyenne 9 centres stéréogènes par molécule. Après normalisation, l'écart type de l'activité biologique est de 0.258. Comme pour le premier jeu de données, le tableau 3.3 contient divers statistiques sur ce jeu de données.

Contrairement au premier jeu de données, chaque molécule possède une similarité non nulle avec toutes les autres molécules pour le noyau stéréo. Nous pouvons donc utiliser un ensemble de validation afin de sélectionner les paramètres. Nous utilisons deux validations croisées imbriquées afin de sélectionner les paramètres et d'estimer les performances de chaque noyau. La validation croisée externe est une procédure de "leave-one-out", servant à calculer une erreur pour chaque molécule du jeu de données. Nous utilisons une autre procédure de "leave-one-out" sur les molécules restantes, afin de calculer une erreur de validation. Les paramètres obtenant la plus petite erreur quadratique sur l'ensemble de validation sont sélectionnés. Les différentes

TABLE 3.3 – Propriétés du jeu de données des dérivés synthétiques de la vitamine D

Nombre de molécules	69
Taille moyenne des molécules	77
Taille maximale des molécules	88
Taille minimale des molécules	68
Taille moyenne des stéréo sous-graphes minimaux	14
Taille maximale des stéréo sous-graphes minimaux	24
Taille minimale des stéréo sous-graphes minimaux	10

TABLE 3.4 – Prédiction de l'activité biologique des dérivés synthétiques de la vitamine D.

	Méthodes	Erreur Moyenne	RMSE
1 -	Noyau de motifs d'arbres [MV09]	0.193	0.251
2 -	Noyau de treelets [Gaü13]	0.208	0.271
3 -	Noyau de motifs d'arbres stéréo [BUT+10]	*0.138*	**0.184**
4 -	Noyau stéréo	**0.134**	*0.194*

valeurs testées pour le paramètre C de l'algorithme de machine à vecteurs de support pour la régression sont $0.01, 0.1, \ldots, 1000, 10000$. Pour les autres paramètres, on utilise les même valeurs que pour le premier jeu de données.

Les erreurs moyennes et les racines carrées des erreurs quadratiques pour ce jeu de données sont reportées dans le tableau 3.4. Les méthodes utilisant le graphe moléculaire [MV09, Gaü13] obtiennent à nouveau les moins bons résultats. Contrairement au jeu de données précédent, l'adaptation du noyau de motifs d'arbres à la stéréoisomérie [BUT+10] obtient des résultats très proches de ceux que l'on obtient avec le noyau stéréo. L'activité biologique n'est pas uniquement liée à la stéréoisomérie, ainsi l'inclusion de motifs d'arbres n'encodant pas d'information sur la stéréoisomérie dans [BUT+10] ne dégrade pas ses résultats pour ce jeu de données.

Pour ce jeu de données, il faut quatre minutes afin de calculer les matrices de gram pour le noyau de motifs d'arbres adapté à la stéréoisomérie [BUT+10] et une seconde pour le noyau stéréo. Nous utilisons pour le calcul de ce noyau un algorithme d'isomorphisme, cependant nous pouvons voir dans le tableau 3.3, que les graphes sur lesquels cet algorithme est utilisé, sont petits. Ceci permet d'obtenir un temps de calcul raisonnable. De plus, notre noyau est basé sur une

énumération explicite de motifs. L'ensemble des stéréo sous-graphes minimaux caractérisant une molécule n'est donc calculé qu'une seule fois par molécule.

3.7 Conclusion

Les stéréo sous-graphes minimaux, présentés dans ce chapitre, permettent de caractériser les stéréo sommets. Ces sous-graphes sont les plus petits sous-graphes contenant la partie du graphe qui confère à un stéréo sommet sa propriété de stéréoisomérie. On compare alors les sacs de stéréo sous-graphes minimaux de deux graphes localement ordonnés afin d'obtenir une mesure de similarité entre molécules qui prend en compte la stéréoisomérie. Le noyau stéréo a l'avantage de n'encoder que la stéréoisomérie des molécules.

Chapitre 4

Extensions

Sommaire

4.1 Introduction

Le noyau stéréo présenté dans le chapitre 3 est fondé sur l'énumération des stéréo sous-graphes minimaux d'un graphe moléculaire localement ordonné. Deux différences majeures distinguent ce noyau du noyau de motifs d'arbres adapté à la stéréoisomérie de [BUT+10]. Premièrement, le noyau de motifs d'arbres adapté à la stéréoisomérie mélange des motifs d'arbres contenant des informations concernant la stéréoisomérie et d'autres n'en contenant pas. De plus, les motifs d'arbres ont une taille limitée par un paramètre, alors que la taille d'un stéréo sous-graphe minimal est induite par la configuration autour

de son stéréo sommet. En conséquence, le noyau de motif d'arbres ne capture généralement pas l'ensemble des informations définissant la stéréoisomérie. Toutefois, si un stéréo sous-graphe minimal est plus petit que la taille d'un motif d'arbre, ce motif pourra encoder le stéréo sous-graphe minimal et son entourage.

On peut supposer que deux stéréo sous-graphes minimaux identiques, mais possédant des voisinages différents peuvent ne pas avoir exactement la même influence sur une propriété. Les extensions présentées dans les sections 4.2 et 4.3 se basent sur cette supposition.

En effet, selon [BKW41], deux centres stéréogènes n'auront pas une même influence s'ils sont proches ou éloignés dans une molécule. Cependant la notion de proximité entre centres stéréogènes ne peut être déterminée a priori. La section 4.2 présente une méthode qui vise à prendre en compte les recouvrements entre les stéréo sous-graphes minimaux.

L'influence d'un stéréo sous-graphe minimal sur une propriété peut également être déterminée par son environnement immédiat. L'extension du noyau stéréo présentée dans la section 4.3 vise donc à prendre en compte le voisinage d'un stéréo sous-graphe minimal.

Une famille de noyaux sur graphes appliqués en chémoinformatique est fondée sur la décomposition du graphe en ensembles de sous-structures. La similarité entre les graphes est alors définie comme la similarité entre les ensembles de sous-structures. Les noyaux présentés dans le chapitre 1 font tous partie de cette famille. La similarité entre les sous-structures est souvent déterminée en comparant le nombre d'occurrences de sous-structures isomorphes. Le noyau stéréo fait partie de ces méthodes car nous considérons les stéréo sous-graphes minimaux isomorphes.

Cependant, plusieurs méthodes [She12, Gaü13] visent à établir une notion de similarité non binaire entre les sous-structures. Ainsi, dans la section 4.4, nous présentons une extension du noyau stéréo permettant de comparer des stéréo sous-graphes minimaux non isomorphes.

4.2 Recouvrements

Le noyau stéréo présenté dans le chapitre 3 considère chaque stéréo sous-graphe minimal de façon isolé. Le but de cette partie est de construire un noyau qui va permettre de prendre en compte les recouvrements entre les stéréo sous-graphes minimaux.

4.2.1 Graphes de recouvrements

Afin d'encoder les recouvrements entre les stéréo sous-graphes minimaux, on définit des fonctions de recouvrements entre ces sous-graphes. Plusieurs fonctions sont considérées car on ne peut pas définir a priori à quel point deux centres stéréogènes doivent être "proches" pour se recouvrir [BKW41]. En effet, la définition de la quantité de recouvrement entre deux stéréo sous-graphes minimaux, pouvant influer sur une propriété, est un problème ouvert.

Les fonctions de recouvrements sont définies à l'aide d'un ensemble de conditions $(c_0, \ldots, c_n)$. Ces conditions sont de plus en plus contraignantes :

- $\forall i \in \{1, \ldots, n-1\}\, c_{i+1} \Rightarrow c_i$
- $c_0 = \neg c_1$

La condition c_0 correspond à la négation de la condition la moins contraignante c_1. Elle décrit donc l'absence de recouvrement.

Définition 61. Fonctions de recouvrements

Soit $G = (\hat{G} = (V, E, \mu, \nu), ord)$ un graphe moléculaire localement ordonné. Soit S_1 et S_2 deux stéréo sous-graphes minimaux de G ayant respectivement pour stéréo sommets s_1 et s_2. La valeur $F(S_1, S_2)$ de la fonction de recouvrements F entre S_1 et S_2 est obtenue en prenant l'indice maximal j tel que la condition $c_j(S_1, S_2)$ soit vraie :

$$F(S_1, S_2) = \max\{j \in \{0, \ldots, n\} \mid c_j(S_1, S_2)\}$$

On considère trois ensembles de conditions, qui définissent trois fonctions de recouvrements F_i différentes :

- *F_1 est définie en utilisant* $\begin{cases} c_1(S_1, S_2) : S_1 \bigcap S_2 \neq \emptyset \\ c_2(S_1, S_2) : kernel(s_1) \subset S_2 \\ c_3(S_1, S_2) : StereoStar(s_1) \subset S_2 \\ c_4(S_1, S_2) : S_1 \subset S_2 \end{cases}$
- *F_2 est définie en utilisant* $\begin{cases} c_1(S_1, S_2) : kernel(s_1) \subset S_2 \\ c_2(S_1, S_2) : StereoStar(s_1) \subset S_2 \\ c_3(S_1, S_2) : S_1 \subset S_2 \end{cases}$
- *F_3 est définie en utilisant* $\begin{cases} c_1(S_1, S_2) : StereoStar(s_1) \subset S_2 \\ c_2(S_1, S_2) : S_1 \subset S_2 \end{cases}$

On peut remarquer que les fonctions de recouvrements ne sont pas symétriques.

On utilise les fonctions de recouvrements pour construire des graphes de recouvrements orientés. Dans ces graphes, chaque sommet représente un stéréo sous-graphe minimal, et les arcs représentent les recouvrements entre les stéréo sous-graphes.

Définition 62. Graphe de recouvrements orienté

Soit $G = (\hat{G} = (V, E, \mu, \nu), ord)$ *un graphe moléculaire localement ordonné. Un graphe de recouvrements orienté* $G_i = (V_i, A_i, \mu_i, \nu_i)$ *de* G *est un graphe orienté tel que :*

- *Il existe une bijection* S *qui associe à chaque sommet* u *de* V_i *un unique stéréo sous-graphe minimal* $S(u) \in \mathcal{H}(G)$.
- $\forall u \in V_i,\ \mu_i(u) = c_{S(u)}$.
- $A_i = \{(u_1, u_2) \in V_i \times V_i | F_i(S(u_1), S(u_2)) \neq 0\}$.
- $\forall a = (u_1, u_2) \in A_i,\ \nu_i(a) = F_i(S(u_1), S(u_2))$.

où c_S *est le code obtenu par l'algorithme de dénomination [WD74] utilisé dans la section 3.5 afin d'identifier le stéréo sous-graphe minimal* S. F_i *est une des fonctions de recouvrements définies dans la Définition 61.*

En pratique, un graphe moléculaire localement ordonné possède peu de stéréo sous-graphes minimaux identiques. Ainsi, peu de sommets ont des étiquettes identiques au sein d'un graphe de recouvrements orienté. Le sens des arcs dans ces graphes n'apporte donc pas beaucoup d'informations. Nous définissons donc des graphes de recouvrements non orientés qui ne perdent pas beaucoup d'informations par rapport aux graphes de recouvrements orientés, mais qui peuvent être en revanche comparés par de nombreux noyaux sur graphes.

Définition 63. Graphe de recouvrements non orienté

Soit $G = (\hat{G} = (V, E, \mu, \nu), ord)$ *un graphe moléculaire localement ordonné. Un graphe de recouvrements non orienté* $G_i = (V_i, E_i, \mu_i, \nu_i)$ *de* G *est un graphe non orienté tel que :*

- *Il existe une bijection* S *qui associe à chaque sommet* u *de* V_i *un unique stéréo sous-graphe minimal* $S(u) \in \mathcal{H}(G)$.
- $\forall u \in V_i,\ \mu_i(u) = c_{S(u)}$.
- $E_i = \{\{u_1, u_2\} \in \mathcal{P}_2(V_i) | F_i(S(u_1), S(u_2)) \neq 0 \vee F_i(S(u_2), S(u_1)) \neq 0\}$.
- $\forall e = (u_1, u_2) \in E_i$,
  $\nu_i(e) = \min(F_i(S_1, S_2), F_i(S_2, S_1)) \odot \max(F_i(S_1, S_2), F_i(S_2, S_1))$.
  Avec $S_1 = S(u_1)$ *et* $S_2 = S(u_2)$.

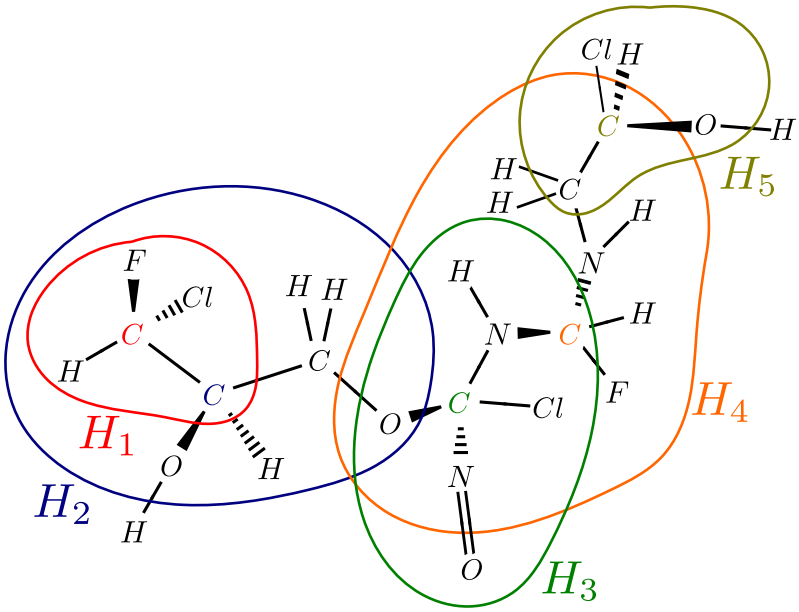
où c_S *est le code obtenu par l'algorithme de dénomination [WD74] et utilisé dans la section 3.5 afin d'identifier le stéréo sous-graphe minimal* S. $\odot$ *est la concaténation et* F_i *est une des fonctions de recouvrements définies dans la Définition 61.*

La Figure 4.1 montre les différents graphes de recouvrements obtenus en prenant les différentes fonctions de recouvrements de la Définition 61.

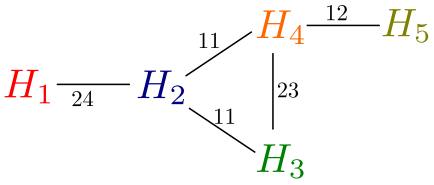
Pour le premier graphe de recouvrements G_1 on considère que quatre niveaux de recouvrements sont possibles. Pour ce graphe, la condition minimale pour que deux stéréo sous-graphes minimaux interagissent est que leur intersection ne soit pas vide. Le recouvrement d'un stéréo sous-graphe minimal S_1 par un autre stéréo sous-graphe minimal S_2 est considérée comme plus fort si le stéréo sommet de S_2 est inclus dans S_1. Le troisième niveau de recouvrement a lieu si le stéréo sommet de S_2 et son voisinage sont inclus dans S_1. Finalement, le niveau de recouvrement est considéré comme maximal lorsque S_2 est inclus dans S_1.

Cependant, certains de ces niveaux de recouvrements peuvent être considérés comme pas assez pertinents. On peut supposer que le fait d'avoir une intersection non vide n'est pas suffisant pour dire que deux stéréo sous-graphes minimaux interagissent. Le graphe G_2 est donc construit en considérant que la condition minimale pour que deux stéréo sous-graphes minimaux interagissent est que le stéréo sommet de l'un soit inclus dans le stéréo sous-graphe minimal de l'autre.

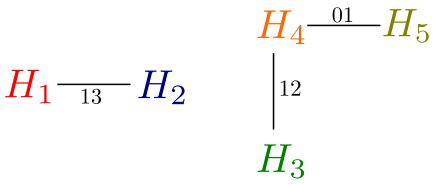
Un atome peut être un centre stéréogène à cause du positionnement relatif de ses voisins. Si un stéréo sommet est inclus dans un stéréo sous-graphe minimal ($kernel(s_2) \in S_1$), mais pas son voisinage ($StereoStar(s_2) \notin S_1$), on



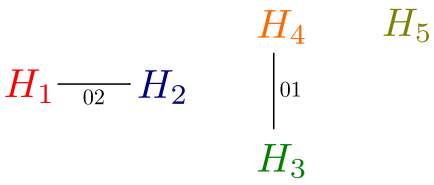
(a) Un graphe moléculaire localement ordonné et ses stéréo sous-graphes minimaux



(b) G_1



(c) G_2



(d) G_3

FIGURE 4.1 – Exemple de graphe moléculaire localement ordonné et des trois graphes de recouvrements non orientés obtenus en utilisant les trois fonctions de recouvrements.

peut alors supposer que ce stéréo sommet a la même influence qu'un sommet qui n'est pas stéréo. Ainsi le graphe de recouvrements G_3 est construit sans considérer que la condition "le stéréo sommet de S_2 est inclus dans S_1" est différente de la condition "l'intersection de S_1 et S_2 est non vide". Ainsi pour ce graphe, la condition minimale de recouvrement entre deux stéréo sommets est que le premier stéréo sommet et son voisinage soient inclus dans le stéréo sous-graphe minimal du second.

Après avoir calculé les stéréo sous-graphes minimaux, vérifier si un des sommets appartient à un de ces sous-graphes est fait en temps constant. Ainsi, les complexités des temps de calcul des quatre conditions de $F_1(S_1, S_2)$ sont respectivement $\mathcal{O}(max(|S_1|, |S_2|))$, $\mathcal{O}(|kernel(s_1)|)$, $\mathcal{O}(|StereoStar(s_1)|)$ et $\mathcal{O}(|S_1|)$. La complexité en temps de calcul de la construction des graphes de recouvrement est donc égale à :

$$\mathcal{O}(|\mathcal{H}(G)|^2 \max_{H \in \mathcal{H}(G)} |H|)$$

En pratique, cette valeur est petite. Par exemple, pour le jeu de données des dérivés synthétiques de la vitamine D présenté dans la section 3.6, nous avons dans le pire des cas :

- $|\mathcal{H}(G)| = 9$
- $\max_{H \in \mathcal{H}(G)} |H| = 24$

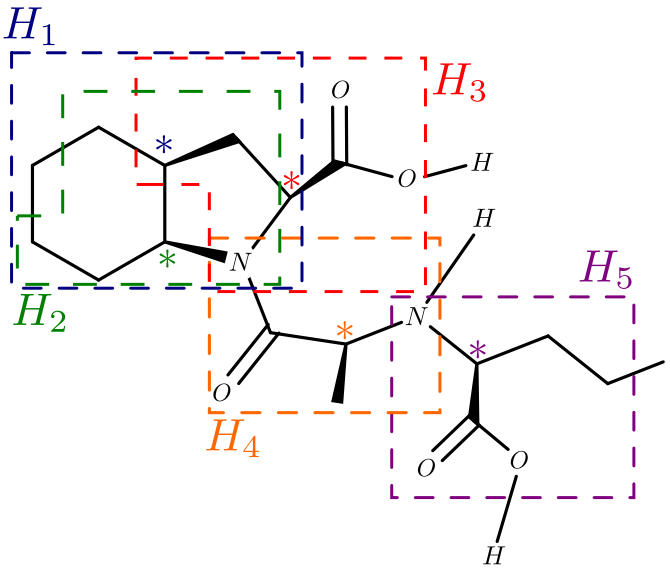
Les graphes de recouvrements ne sont pas des graphes localement ordonnés, on peut donc utiliser n'importe quel noyau usuel sur graphes (par exemple [MV09, SSVL+11, Gaü13]) afin de mesurer leurs similarités. En considérant le noyau de treelets, les treelets de taille 1 correspondent aux sommets des graphes de recouvrements. Ils encodent donc exactement la même information que le noyau stéréo présenté dans la section 3.5.

4.2.2 Expérimentations

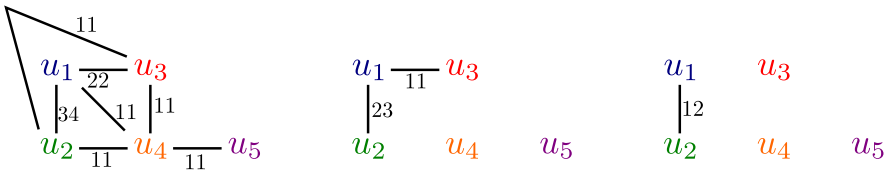
Afin de prouver l'intérêt des graphes de recouvrements, nous les avons utilisés pour deux problèmes, un de classification et un de régression.

Le premier problème est fondé sur un jeu de données contenant tous les stéréoisomères de la périndoprilate et issu de [CGMPTR07]. Comme cette molécule contient 5 centres stéréogènes, le jeu de données est composé de $2^5 = 32$ molécules. Le jeu de données est divisé en deux classes :

- 9 molécules inhibant l'enzyme de conversion de l'angiotensine.



(a) Une molécule du jeu de données



(b) Les trois graphes de recouvrements

FIGURE 4.2 – Exemple d'une molécule du jeu de données des stéréoisomères de la périndoprilate et de ses trois graphes de recouvrements non orientés. Les stéréo sous-graphes minimaux de la molécule sont entourés en pointillés. Pour chacun d'eux, le stéréo sommet correspondant est représenté par une étoile de la même couleur.

- 23 molécules n'inhibant pas l'enzyme de conversion de l'angiotensine.

Comme ce jeu de données est assez petit, nous utilisons le même protocole que pour le jeu de données des dérivés de la vitamine D de la section 3.6. On utilise donc deux procédures "leave-one-out" imbriquées, l'une servant à sélectionner les paramètres et l'autre à évaluer les performances des différentes méthodes.

La Figure 4.2(a) montre le graphe d'une molécule de ce jeu de données avec ses stéréo sous-graphes minimaux et la Figure 4.2(b) montre les trois graphes de recouvrements obtenus à partir de ce graphe. Toutes les molécules de ce jeu de données sont des stéréoisomères. Ainsi, si l'on considère une autre molécule

TABLE 4.1 – Moyennes du nombre d'arêtes ($\overline{|E|}$), nombre d'étiquettes différentes ($\overline{|\mathcal{L}_V|}$,$\overline{|\mathcal{L}_E|}$), des degrés moyens ($\overline{d}$) des graphes de recouvrements pour le jeu de données des stéréoisomères de la périndoprilate. Tous les graphes de ce jeu de données ont cinq sommets. Pour la moitié de ces graphes tous les sommets ont des étiquettes différentes et pour l'autre il y a quatre étiquettes différentes, ce qui explique que l'on ait en moyenne $4,5$ étiquettes différentes.

	$\overline{\lvert E\rvert}$	$\overline{\lvert\mathcal{L}_V\rvert}$	$\overline{\lvert\mathcal{L}_E\rvert}$	$\overline{d}$
Graphe de recouvrements 1	7	4.5	3	2.8
Graphe de recouvrements 2	2	4.5	2	0.8
Graphe de recouvrements 3	1	4.5	1	0.4

de ce jeu de données, ses graphes de recouvrements seront identiques à ceux représentés dans la figure 4.2(b), à l'exception des labels des sommets.

Le tableau 4.1 contient plusieurs informations concernant les graphes de recouvrements obtenus avec ce jeu de données. On peut notamment remarquer que le degré moyen ($\overline{d}$) des sommets varie beaucoup selon les fonctions de recouvrements utilisées pour construire le graphe de recouvrements.

Le tableau 4.2 montre le pourcentage de molécules correctement classées par le noyau stéréo (chapitre 3), le noyau de motifs d'arbres adapté à la stéréoisomérie [BUT+10] et différents noyaux [MV09, SSVL+11, Gaü13] utilisés sur les graphes de recouvrements. On ne donne pas les résultats pour des méthodes fondées sur le graphe moléculaire car les molécules de ce jeu de données sont toutes stéréoisomères l'une de l'autre et ont donc toutes le même graphe moléculaire.

Pour ces molécules, deux stéréo sommets ont des stéréo sous-graphes minimaux identiques ou opposés (ces deux stéréo sous-graphes minimaux sont notés H_3 et H_5 dans la Figure 4.2). Les stéréo sous-graphes minimaux associés à ces deux stéréo sommets ont des voisinages différents. Le noyau de motifs d'arbre adapté à la stéréoisomérie [BUT+10] (ligne 1) est capable de distinguer ces deux stéréo sommets et obtient donc une meilleure précision que le noyau stéréo présenté dans le chapitre 3 (ligne 2). Comme toutes les molécules sont stéréoisomères l'une de l'autre, les motifs d'arbres ne contenant pas d'information sur la stéréochimie sont les mêmes pour chaque molécule. Ils ajoutent donc une même valeur à chaque noyau, et n'ont donc pas de mauvaise influence sur la prédiction.

On peut voir dans la figure 4.2(b) que les sommets u_3 et u_5 sont isolés dans le graphe de recouvrements G_3. Ainsi, comme le noyau stéréo (ligne 2), le graphe de recouvrements G_3 n'est pas capable de discerner les deux stéréo sous-graphes

TABLE 4.2 – Classification des stéréoisomères de la périndoprilate en fonction de leur action sur l'enzyme de conversion de l'angiotensine.

	Méthode	Précision
1 -	Noyau de motifs d'arbres stéréo [BUT+10]	*96.875*
2 -	Noyau stéréo	87.5
	Graphes de recouvrements avec [Gaü13]	
3 -	Graphes de recouvrements 1	93.75
4 -	Graphes de recouvrements 2	93.75
5 -	Graphes de recouvrements 3	84.375
	Graphes de recouvrements avec [MV09]	
6 -	Graphes de recouvrements 1	93.75
7 -	Graphes de recouvrements 2	62.5
8 -	Graphes de recouvrements 3	62.5
	Graphes de recouvrements avec [SSVL+11]	
9 -	Graphes de recouvrements 1	93.75
10 -	Graphes de recouvrements 2	62.5
11 -	Graphes de recouvrements 3	62.5
	Graphes de recouvrements avec [Gaü13] et MKL [VB09]	
12 -	Graphes de recouvrements 1	**100**
13 -	Graphes de recouvrements 2	87.5
14 -	Graphes de recouvrements 3	90.625

minimaux H_3 et H_5. On obtient donc une mauvaise précision en utilisant ce graphe de recouvrements (lignes 5, 8, 11 et 14). En appliquant le noyau de treelets aux deux autres graphes de recouvrements G_1 et G_2 (lignes 3 et 4), nous obtenons de bons résultats, mais cependant moins bons que les résultats obtenus par [BUT+10]. Cela peut être expliqué par le fait que les treelets de taille 1 donnent la même information que le noyau stéréo. En utilisant un algorithme d'apprentissage à noyaux multiples [VB09], on peut calculer un poids optimal pour chaque treelet lors de l'apprentissage. Ceci nous permet d'enlever les treelets de taille 1 et nous permet d'obtenir une précision parfaite avec le graphe de recouvrements G_1 (ligne 12).

Lorsqu'on utilise d'autre noyaux [MV09, SSVL+11] avec le second graphe de recouvrements, on obtient de mauvais résultats (lignes 7 et 10). On peut voir dans le Tableau 4.1 et la Figure 4.2, que les second graphes de recouvrements

TABLE 4.3 – Moyennes du nombre de sommets ($\overline{|V|}$), nombre d'arêtes ($\overline{|E|}$), nombre d'étiquettes différentes ($\overline{|\mathcal{L}_V|}$,$\overline{|\mathcal{L}_E|}$), des degrés moyens ($\overline{d}$) des graphes de recouvrements pour le jeu de données des dérivés synthétiques de la vitamine D.

	$\overline{\lvert V\rvert}$	$\overline{\lvert E\rvert}$	$\overline{\lvert\mathcal{L}_V\rvert}$	$\overline{\lvert\mathcal{L}_E\rvert}$	$\overline{d}$
Graphe de recouvrements 1	8.55	17.4	8.38	5.71	4.07
Graphe de recouvrements 2	8.55	11.3	8.38	4.71	2.62
Graphe de recouvrements 3	8.55	6.14	8.38	2.71	1.43

ne possèdent que deux arêtes pour cinq sommets. On voit notamment que le sommet noté u_4 dans la Figure 4.2, a quatre voisins dans le premier graphe de recouvrements et aucun dans le second. Ce sommet est donc présent dans de nombreux motifs extraits par les noyaux [MV09, SSVL+11, Gaü13] pour le premier graphe de recouvrements mais dans un seul motif pour le second graphe. Implicitement, ce sommet, et donc le stéréo sous-graphe minimal qui le représente, a donc moins d'influence dans les noyaux construits à partir du second graphe de recouvrements que dans ceux construits à partir du premier. Or il joue un rôle important pour la prédiction de la propriété. Le noyau de treelets est moins affecté que les deux autres noyaux car il extrait moins de motifs. En effet, on extrait des seconds graphes de recouvrements huit treelets mais onze motifs d'arbres. Ainsi l'influence du stéréo sous-graphe minimal associé au sommet s_4 est moins réduite pour le noyau de treelets que pour les deux autres, ce qui explique qu'avec le second graphe de recouvrement ces noyaux obtiennent de moins bons résultats (lignes 7 et 10), que le noyau de treelets (ligne 4).

Le second jeu de données utilisé pour valider l'approche utilisant les graphes de recouvrements est le jeu de données des dérivés synthétiques de la vitamine D présenté dans la section 3.6. Comme pour le premier jeu de données, nous montrons dans le Tableau 4.3 diverses statistiques sur les graphes de recouvrements.

Pour ce jeu de données, l'utilisation des graphes de recouvrements G_2 et G_3 (lignes 6, 7, 9, 10, 12 et 13), permet d'obtenir de meilleurs résultats que le noyau stéréo (ligne 4) ou l'adaptation du noyau de motifs d'arbres à la stéréochimie (ligne 3). Contrairement au premier jeu de données, on peut voir que les résultats obtenus en utilisant le graphe de recouvrements G_1 (lignes 5, 8 et 11), sont moins bons que ceux obtenus avec les deux autres graphes de recouvrements. Dans le Tableau 4.3, on peut voir que les graphes de recouvrements G_1 ont un degré moyen de 4 pour 8.55 sommets en moyenne.

TABLE 4.4 – Prédiction de l'activité biologique des dérivés synthétiques de la vitamine D.

	Méthodes	Erreur Moyenne	RMSE
1 -	Noyau de motifs d'arbres [MV09]	0.193	0.251
2 -	Noyau de treelets [Gaü13]	0.208	0.271
3 -	Noyau de motifs d'arbres stéréo [BUT+10]	0.138	0.184
4 -	Noyau stéréo	0.134	0.194
	Graphes de recouvrements avec [Gaü13]		
5 -	Graphes de recouvrements 1	0.127	0.177
6 -	Graphes de recouvrements 2	0.113	0.169
7 -	Graphes de recouvrements 3	0.116	0.171
	Graphes de recouvrements avec [MV09]		
8 -	Graphes de recouvrements 1	0.127	0.185
9 -	Graphes de recouvrements 2	0.109	*0.162*
10 -	Graphes de recouvrements 3	**0.107**	**0.161**
	Graphes de recouvrements avec [SSVL+11]		
11 -	Graphes de recouvrements 1	0.148	0.201
12 -	Graphes de recouvrements 2	0.111	0.166
13 -	Graphes de recouvrements 3	*0.108*	*0.162*

Ainsi les sommets de ces graphes de recouvrements sont liés en moyenne à la moitié du graphe. De plus il y a en moyenne 8.38 labels différents pour les 8.55 sommets. Ainsi quasiment tous les sommets ont des labels différents. Ceci crée de nombreux motifs uniques pour chaque molécule, ce qui explique pourquoi les résultats sont moins bons en utilisant ce graphe.

4.3 Voisinage des stéréo sous-graphes minimaux

Dans cette partie, nous présentons un noyau qui prend en compte le voisinage des stéréo sous-graphes minimaux. Contrairement aux graphes de recouvrements, qui permettent de représenter la proximité entre les stéréo sous-graphes minimaux, le noyau présenté dans cette partie permet de représenter le voisinage direct d'un stéréo sous-graphe minimal.

L'idée générale est d'associer à chaque sommet, situé à la frontière du stéréo sous-graphe minimal, son voisinage. La similarité entre deux stéréo sous-graphes minimaux identiques sera alors pondérée par la similarité de leurs voisinages.

4.3.1 Construction des voisinages

Afin de comparer les voisinages de deux stéréo sous-graphes minimaux on commence par considérer les sommets situés à la frontière de ces sous-graphes :

Définition 64. Frontière d'un stéréo sous-graphe minimal

Soit $G = (\hat{G} = (V, E, \mu, \nu), ord)$ un graphe moléculaire localement ordonné. Soit s un stéréo sommet de G et S son stéréo sous-graphe minimal. L'ensemble $\delta_{in}(S)$ des sommets situés à la frontière de S est défini par :

$$\delta_{in}(S) = \{v \in V_S \mid N(v) \not\subset S\}$$

Les sommets situés à la frontière d'un stéréo sous-graphe minimal connectent ce sous-graphe au reste du graphe. Nous définissons maintenant le k-voisinage de ces sommets :

Définition 65. k-voisinage d'un sommet situé à la frontière d'un stéréo sous-graphe minimal

Soit $G = (\hat{G} = (V, E, \mu, \nu), ord)$ un graphe moléculaire localement ordonné. Soit s un stéréo sommet de G et S son stéréo sous-graphe minimal. Soit v un sommet de $\delta_{in}(S)$. Le k-voisinage S_v^k de v est le sous-graphe induit de G tel que :

$$V_{S_v^k} = \left\{ u \in V_{G-S} \;\middle|\; \begin{array}{l} d(u,v) \leq k \\ \forall v' \in \delta_{in}(S),\ d(u,v) \leq d(u,v') \end{array} \right\}$$

Autrement dit, le k-voisinage d'un sommet v est constitué des sommets n'appartenant pas à S et situés à une distance inférieure ou égale à k de v. De plus, un sommet u n'appartient au k-voisinage d'un sommet v que si v est le sommet de $\delta_{in}(S)$ le plus proche de u.

La Figure 4.3 montre des exemples de k-voisinages associés à des sommets de la frontière d'un stéréo sous-graphe minimal. On peut remarquer qu'un k-voisinage n'est pas forcément connexe (par exemple celui associé à v_3) et que certains sommets peuvent appartenir à plusieurs k-voisinages (l'atome de chlore appartient aux k-voisinages de v_4 et v_5).

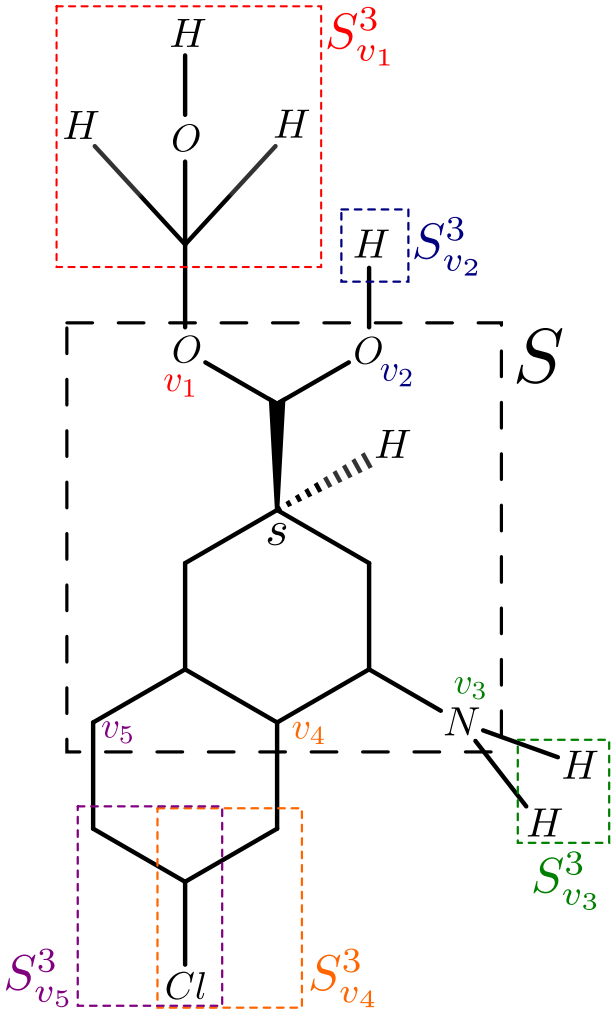


FIGURE 4.3 – Exemple d'un stéréo sous-graphe minimal S avec les sommets à sa frontière $\{v_1, v_2, v_3, v_4, v_5\}$ et leurs 3-voisinages.

Les k-voisinages des stéréo sous-graphes minimaux sont finalement comparés afin de donner une mesure de similarité entre stéréo sous-graphes minimaux. Cependant, on ne compare pas toutes les paires de k-voisinages entre deux stéréo sous-graphes minimaux mais seulement les k-voisinages liés à des sommets v et u tels qu'il y ait un isomorphisme f avec $u = f(v)$. De plus, afin de prendre en compte la stéréoisomérie, cet isomorphisme doit être un isomorphisme d'équivalence d'ordres.

Comme on ne compare qu'un sous-ensemble de paires, on va définir le noyau comme un noyau d'appariement (Sous-section 1.1.3).

Définition 66. Noyau d'influence

Soit S un stéréo sous-graphe minimal. On note $(v_1, \ldots, v_n)$ une séquence ordonnée des n sommets de l'ensemble $\delta_{in}(S)$. De plus, on note $seq(S)$ la séquence de k-voisinages associés aux sommets de la séquence $(v_1, \ldots, v_n)$:

$$seq(S) = (S^k_{v_1}, \ldots, S^k_{v_n})$$

On définit l'appariement entre deux stéréo sous-graphes minimaux S et S' suivant :

$$M_{S,S'} = \{(seq(S), seq(S')) | \exists f \in IsomEqOrd(S, S') \; t.q \; \forall i \in [1, \ldots, n], f(v_i) = v'_i\}$$

où $seq(S) = (S^k_{v_1}, \ldots, S^k_{v_n})$ et $seq(S') = (S'^k_{v'_1}, \ldots, S'^k_{v'_n})$.

Le noyau $k_{influence}(S, S')$ entre deux stéréo sous-graphes minimaux S et S' est défini par :

$$k_{influence}(S, S') = \delta(S, S') \frac{1}{\sqrt{g(S)!}} \frac{1}{\sqrt{g(S')!}} \sum_{(seq(S), seq(S')) \in M_{S,S'}} \prod_{i=1}^{n} k_t(S^k_{v_i}, S'^k_{v'_i}) \tag{4.1}$$

où $\delta(S, S')$ vaut 1 si S et S' sont d'ordres équivalents (et 0 sinon), k_t est un noyau entre graphes et g est une fonction qui associe à un stéréo sous-graphe minimal S la taille $|\delta_{in}(S)|$ de l'ensemble des sommets situés à sa frontière.

La taille k des voisinages considérés est un paramètre de la méthode. On le sélectionne donc par validation croisée.

Proposition 11. *Le noyau $k_{influence}(S, S')$ entre deux stéréo sous-graphes minimaux S et S' défini dans la définition 66 est défini positif.*

Démonstration. Cette preuve est donnée en annexe à la section 5.3.2 (page 138). □

Finalement, le noyau d'influence entre deux graphes moléculaires localement ordonnés est calculé en comparant deux à deux leurs stéréo sous-graphes minimaux :

$$k_{influenceG}(G, G') = \sum_{S \in \mathcal{H}(G)} \sum_{S' \in \mathcal{H}(G')} k_{influence}(S, S') \tag{4.2}$$

TABLE 4.5 – Prédiction de l'activité biologique des dérivés synthétiques de la vitamine D.

	Méthodes	Erreur Moyenne	RMSE
1 -	Noyau de motifs d'arbres [MV09]	0.193	0.251
2 -	Noyau de treelets [Gaü13]	0.208	0.271
3 -	Noyau de motifs d'arbres stéréo [BUT+10]	0.138	0.184
4 -	Noyau stéréo	0.134	0.194
5 -	Graphes de recouvrements 3 avec [MV09]	**0.107**	**0.161**
6 -	Noyau d'influence avec sous-noyaux [Gaü13]	0.136	0.184
7 -	Noyau d'influence avec sous-noyaux (4.3)	*0.131*	*0.177*

Le lemme 1 de [Hau99] dit que si l'on considère un noyau K défini positif sur $U \times U$ alors pour tout couple d'ensembles A et B inclus dans U, le noyau $\sum_{\substack{x \in A \\ y \in B}} K(x, y)$ est défini positif. Par ce lemme et comme $k_{influence}(S, S')$ est défini positif, $k_{influenceG}(G, G')$ est défini positif.

4.3.2 Expérimentations

Ce noyau a été testé sur le jeu de données des dérivés synthétiques de la vitamine D présenté dans la section 3.6.

Les valeurs testées pour la taille k des voisinages considérés sont $1, 2, 3, 4$ et 5. Pour le noyau k_t entre graphes, utilisé dans l'équation (4.1), nous avons testé le noyau de treelets [Gaü13] et un noyau de différence de poids entre les graphes moléculaires défini par :

$$k_t(G, G') = e^{\frac{-(p-p')^2}{d}} \tag{4.3}$$

où d est un paramètre et p le poids d'un graphe moléculaire (défini comme la somme du poids des atomes encodés par les sommets du graphe).

Les résultats sont montrés dans le tableau 4.5. La prise en compte du voisinage des stéréo sous-graphes minimaux, notamment avec le sous-noyau défini dans l'équation (4.3), permet d'obtenir de meilleurs résultats que ceux obtenus avec le noyau stéréo. Ces résultats sont néanmoins moins bon que ceux obtenus grâce aux graphes de recouvrements de la section 4.2. Ceci est dû au fait que seule une petite partie du jeu de données profite de l'ajout d'informations de ce noyau. Pour une majorité des molécules, les voisinages de stéréo sous-graphes minimaux identiques se ressemblent, et donc

le noyau d'influence n'ajoute que peu d'informations. Cependant, pour les six molécules de la Figure 4.4, le noyau d'influence permet d'ajouter des informations importantes. Les molécules des figures 4.4(a), 4.4(b) et 4.4(c) ont les mêmes ensembles de stéréo sous-graphes minimaux. Les molécules des figures 4.4(d), 4.4(e) et 4.4(f) ont également les mêmes ensembles de stéréo sous-graphes minimaux, la seule différence avec les molécules précédentes étant les stéréo sous-graphes minimaux associés aux sommets notés s'. Le stéréo sous-graphe minimal S est identique dans les six molécules. La prise en compte du voisinage permet d'augmenter la similarité entre les molécules des figures 4.4(a) et 4.4(d), entre celles des figures 4.4(b) et 4.4(e), et finalement entre celles des figures 4.4(c) et 4.4(f). Au vu des valeurs à prédire, on comprend que sur ces molécules le noyau d'influence permet d'obtenir de meilleurs résultats. La molécule de la figure 4.4(f) est notamment bien mieux prédite par le noyau d'influence que par n'importe quelle autre méthode.

4.4 Noyau entre différents stéréo sous-graphes

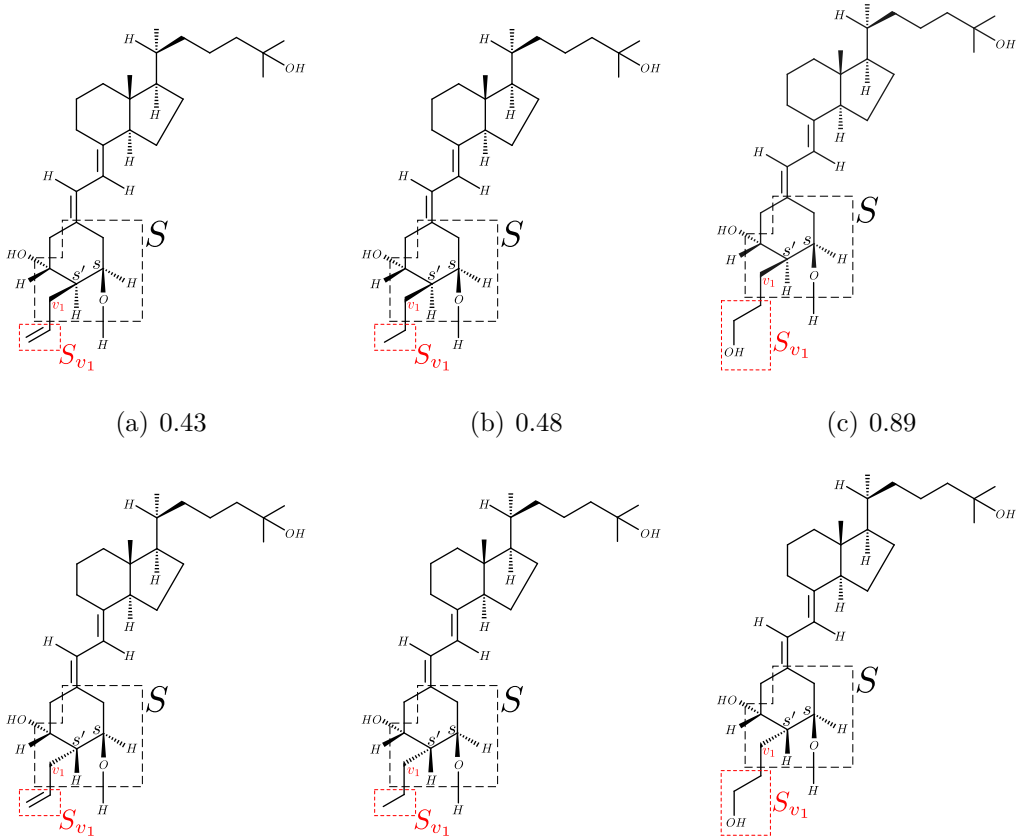
Le noyau stéréo présenté dans le chapitre 3 est fondé sur une mesure binaire de la similarité des stéréo sous-graphes minimaux. S'ils sont parfaitement identiques, leur similarité vaut un, sinon elle est nulle. Cependant, dans certains jeux de données (par exemple le jeu de données contenant des molécules avec leur pouvoir rotatoire, utilisé dans la section 3.6) des stéréo sous-graphes minimaux peuvent être uniques. On ne peut donc pas les utiliser pour mesurer la similarité entre les molécules. Dans cette partie, nous construisons une mesure de similarité entre deux stéréo sous-graphes minimaux différents.

4.4.1 Comparaison de stéréo sous-graphes minimaux

Les stéréo sous-graphes minimaux sont des graphes localement ordonnés. On ne peut donc pas utiliser une mesure classique de similarité entre graphes, il faut prendre en compte l'ordre des voisins des stéréo sommets considérés.

Afin de caractériser un stéréo sous-graphe minimal S d'un stéréo sommet s, on commence par construire des sous-graphes induits de S, un pour chaque sommet de $StereoStar^*(s)$. Ces sous-graphes permettent de caractériser le sommet de $StereoStar^*(s)$ auquel ils sont associés.

Définition 67. Sous-graphe associé à un voisin d'un stéréo sommet



(a) 0.43 (b) 0.48 (c) 0.89

(d) 0.33 (e) 0.38 (f) 0.71

FIGURE 4.4 – Six molécules du jeu de données avec la valeur de leurs activités biologique.

Soit $G = (\hat{G} = (V, E, \mu, \nu), ord)$ un graphe moléculaire localement ordonné. Soit S un stéréo sous-graphe minimal de G et s un de ses stéréo sommets associés. Soit v_q un sommet de $StereoStar^*(s)$. Le sous-graphe H_q associé au sommet v_q est le sous-graphe induit de S composé des sommets V_{H_q} tel que :

$$V_{H_q} = \{v \in S - kernel(s) \mid d(v, v_q) = \min_{u \in StereoStar^*(s)} d(v, u)\}$$

On note $sub(S)$ l'ensemble de ces sous-graphes.

La Figure 4.5 montre un exemple de sous-graphes associés aux voisins d'un couple de stéréo sommets.

Plusieurs sommets de $StereoStar^*(s)$ peuvent être à une même distance d'un sommet v. Ce sommet sera alors présent dans différents sous-graphes de

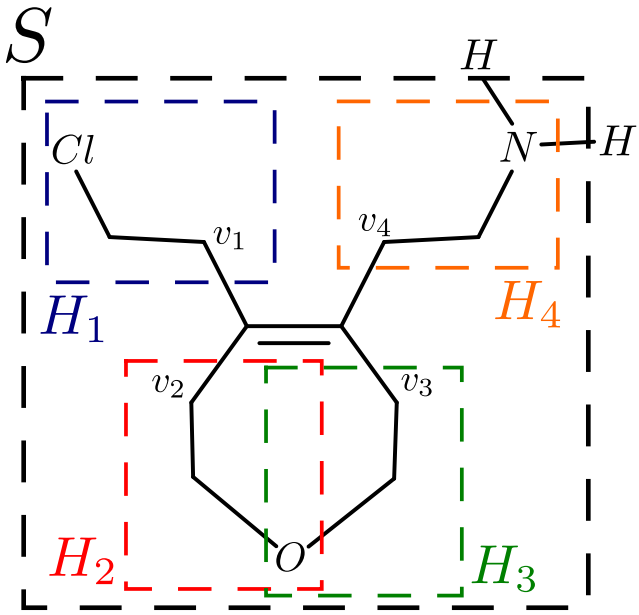


FIGURE 4.5 – Exemple d'un stéréo sous-graphe minimal S divisé en sous-graphes associés aux sommets de $StereoStar^*(s)$.

S. Ces sous-graphes peuvent donc avoir une intersection non vide. Par exemple, dans la Figure 4.5 l'atome d'oxygène est présent dans les sous graphes H_2 et H_3.

Pour comparer deux stéréo sommets, on va comparer les sous-graphes associés à leurs voisins. Ces voisins peuvent être ordonnés de différentes façons équivalentes. Afin de construire une mesure de similarité qui prend en compte les ordres équivalents, nous utilisons l'ensemble de fonctions de réordonnancement suivant :

Définition 68. Réordonnancement d'un stéréo sous-graphe minimal

Soit $G = (\hat{G} = (V, E, \mu, \nu), ord)$ un graphe moléculaire localement ordonné. Soit S un stéréo sous-graphe minimal de G et s un de ses stéréo sommets associés. On note Σ^S l'ensemble des fonctions de réordonnancement valides qui ne modifient que l'ordre autour des sommets de $kernel(s)$:

$$\Sigma^S = \{\sigma \in \Sigma_S^M \mid \forall v \in S - kernel(s), \sigma(v) = Id_{|N(v)|}\}$$

où Id_n est la permutation identité sur Π_n.

On ne compare que les stéréo sous-graphes minimaux associés à des stéréo sommets de même type.

Définition 69. Noyau inter stéréo sous-graphes minimaux

Soit S et S' deux stéréo sous-graphes minimaux. On note s le stéréo sommet associé à S et s' celui associé à S'. On considère que s et s' sont des stéréo sommets de même type (c'est-à-dire s et s' représentent soit tous les deux des carbones asymétriques, soit tous les deux l'un des carbones d'une double liaison).

Si s et s' appartiennent à V^{PAC}, on note $ord(s) = (s_1, \ldots, s_4)$ et $ord(s') = (s'_1, \ldots, s'_4)$. S'ils appartiennent à V^{DB}, on note $ord(s) = (n_=(s), s_1, s_2)$, $ord(n_=(s)) = (s, s_3, s_4)$, $ord(s') = (n_=(s'), s'_1, s'_2)$ et $ord(n_=(s')) = (s', s'_3, s'_4)$ (Figure 4.6).

Pour tout i compris entre 1 et 4, on note H_i le sous-graphe de $sub(S)$ associé au sommet s_i et H'_i le sous-graphe de $sub(S')$ associé au sommet s'_i (Définition 67).

Pour toute fonction de réordonnancement σ appartenant à Σ^S (Définition 68), on note φ_σ la permutation que σ effectue sur les sommets de $StereoStar^*(s)$. On peut remarquer que si s appartient à V^{PAC}, φ_σ est égale à $\sigma(s)$.

Le noyau inter stéréo sous-graphes minimaux est défini par :

$$k(S, S') = \sum_{\substack{\sigma \in \Sigma^S \\ \sigma' \in \Sigma^{S'}}} \prod_{i=1}^{|StereoStar^*(s)|} k_t(H_{\varphi_\sigma(i)}, H'_{\varphi_{\sigma'}(i)}) \tag{4.4}$$

où k_t est un noyau classique entre graphes, par exemple le noyau de treelets [Gaü13].

Proposition 12. *Le noyau de la définition 69 est défini positif.*

Démonstration. Cette preuve est donnée en annexe à la section 5.3.3 (page 139). □

Un inconvénient du noyau défini par l'équation (5.3.3) est que dans certains cas, deux stéréo sous-graphes minimaux qui ne diffèrent que par l'ordre autour de leurs stéréo sommets peuvent être considérés comme similaire. Considérons la molécule de la Figure 4.7(a), où les sous-graphes H_1 et H_3 sont similaires. Nous comparons cette molécule à son opposée (Figure 4.7(b)). Comme $H_1 = H'_3$ et $H_3 = H'_1$, H_1 est similaire à H'_1 et H_3 à H'_3. Ainsi, le produit $\prod_{i=1}^{4} k_t(H_i, H'_i)$ a une valeur élevée. Ceci peut poser des problèmes pour prédire correctement une propriété. Par exemple, les pouvoirs rotatoires de deux énantiomères sont opposés. Ainsi, tenter de prédire cette propriété à l'aide de ce noyau serait inefficace pour des molécules comme celle de la Figure 4.7.

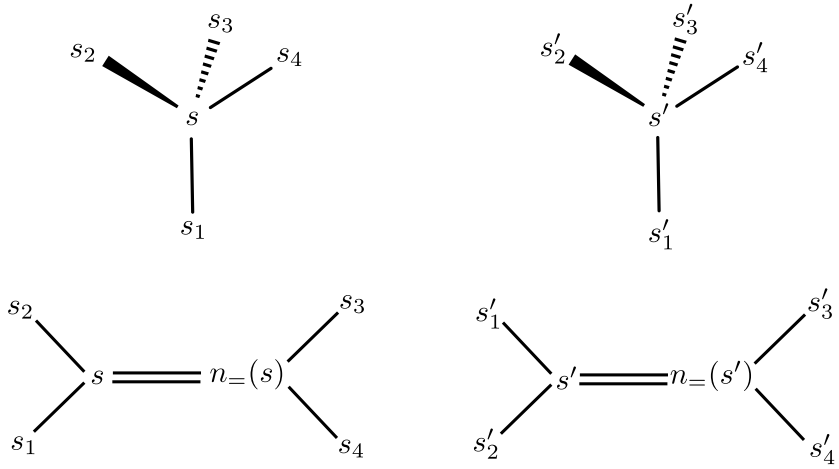


FIGURE 4.6 – Notation des voisins des stéréo sommets de la Définition 69.

Afin de diminuer la similarité entre deux stéréo sommets lorsque leurs stéréo sous-graphes minimaux ont des orientations opposées, on utilise le noyau suivant :

$$k_{inter}(S, S') = \frac{k(S, S')}{\sqrt{k(S, S) \times k(S', S')}} + \delta(S, S') - \delta(S, \tau(S')) \qquad (4.5)$$

où k est le noyau défini dans l'équation (5.3.3), $\delta(S_1, S_2)$ vaut 1 si S_1 et S_2 sont d'ordres équivalents (Définition 47) et τ est une fonction de réordonnancement qui permute deux voisins de s'.

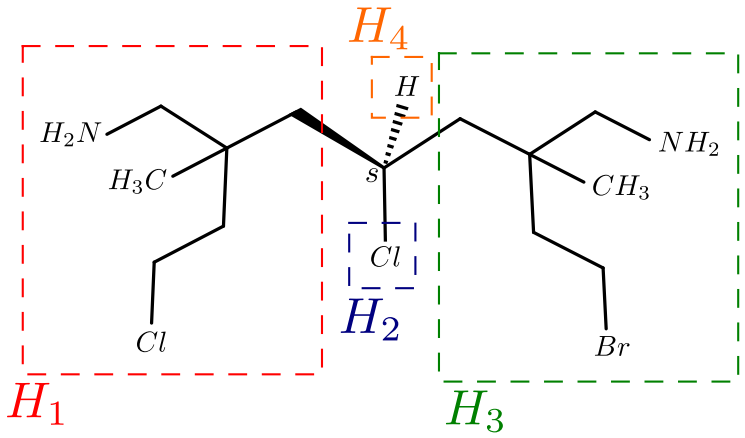
Proposition 13. *Le noyau de l'équation (4.5) est défini positif.*

Démonstration. Cette preuve est donnée en annexe à la section 5.3.4 (page 140). □

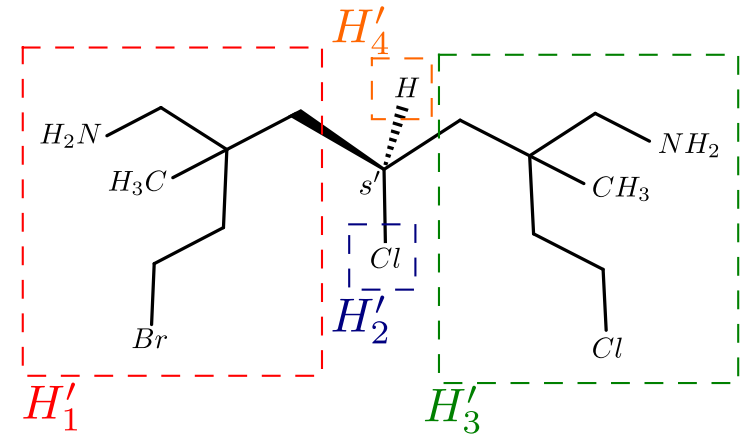
Si deux stéréo sous-graphes minimaux ont des orientations opposées, comme ceux de la Figure 4.7, on aura $\delta(S, \tau(S'))$ égale à 1 et $\delta(S, S')$ à 0. Ainsi, bien que H_1, H_3, H'_1 et H'_3 soient similaires, la valeur du noyau sera réduite de 1, S et S' ne seront donc pas considérés comme similaires.

Si nous avons deux stéréo sous-graphes minimaux S et S' tels que ni S' ni $\tau(S')$ ne soient égales à S, alors la notion d'orientations opposées n'a plus de sens. Dans ce cas, le terme $(\delta(S, S') - \delta(S, \tau(S'))$ est nul.

Comme pour le noyau défini dans la sous-section 4.3.1, la similarité entre deux graphes moléculaires localement ordonnés est calculée en comparant deux à deux l'ensemble de leurs stéréo sous-graphes minimaux :



(a) S



(b) S'

FIGURE 4.7 – Exemple de deux stéréo sous graphes minimaux ayant des orientations opposées. Ils sont divisés en sous graphes associés aux sommets de $StereoStar^*(s)$ et $StereoStar^*(s')$.

$$k_{interG}(G, G') = \sum_{S \in \mathcal{H}(G)} \sum_{S' \in \mathcal{H}(G')} k_{inter}(S, S') \tag{4.6}$$

4.4.2 Expérimentations

Le noyau inter stéréo, décrit dans l'équation (4.6), permet de comparer deux stéréo sous-graphes minimaux différents. Nous montrons dans cette section

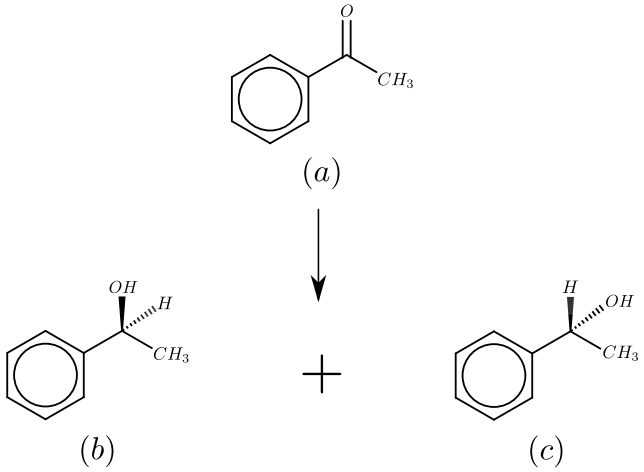


FIGURE 4.8 – Exemple de réduction créant deux stéréoisomères.

deux jeux de données sur lesquels ce noyau permet d'obtenir de meilleurs résultats que le noyau stéréo présenté dans le chapitre 3.

Ces jeux de données sont issus de [SZL+13]. Dans les deux cas, il est question d'une molécule transformée par une réduction en deux stéréoisomères. Par exemple, dans la Figure 4.8 la molécule (a) est transformée en (b) et (c). Cependant, dans toutes ces réductions un des deux stéréoisomères est présent en plus grande quantité que le second. On assigne alors une classe A au stéréoisomère le plus présent et une classe B à l'autre. On a donc des problèmes de classification où chaque molécule est forcément dans la classe opposée à celle de son stéréoisomère.

Comme pour les jeux de données précédents, on utilise deux validations croisées imbriquées afin de sélectionner les paramètres et d'estimer les performances de chaque méthode. Cependant, au lieu d'utiliser une procédure "leave-one-out", on utilise une procédure "leave-one-pair-out". À chaque étape des validations croisées, on enlève une paire de stéréoisomères. Les deux stéréoisomères appartenant nécessairement à deux classes différents, si ils obtiennent la même classe, on considère que la paire n'est pas classée.

Le premier jeu de données contient 50 paires de stéréoisomères. Chaque molécule ne contient qu'un seul centre stéréogène.

Les résultats de la classification pour ce jeu de données sont montrés dans le Tableau 4.6. Dans ce jeu de données, 15 paires de stéréoisomères contiennent des stéréo sous-graphes minimaux uniques. Ainsi, pour chacune de ces 15 paires, le noyau stéréo assigne aux deux stéréoisomères une même classe, et obtient donc une précision bien plus faible que celles obtenues par le noyau de motifs d'arbres adapté à la stéréoisomérie [BUT+10] ou le noyau inter stéréo présenté

TABLE 4.6 – Résultats de la classification pour le premier jeu de données.

Méthode	Précision	Taux de paires non classées
Noyau de motifs d'arbres stéréo [BUT+10]	90%	0%
Noyau stéréo	66%	30%
Noyau inter stéréo	**92**%	0%

TABLE 4.7 – Résultats de la classification pour le second jeu de données.

	Méthode	Précision	Taux de paires non classées
1	Noyau de motifs d'arbres stéréo [BUT+10]	80.6%	4.5%
2	Noyau stéréo	58.2%	38.8%
3	Noyau inter stéréo	76.1%	16.4%
4	Noyau inter stéréo avec sélection	**88.1**%	0%

dans cette partie. La précision de ce noyau est légèrement plus élevée que celle obtenue par le noyau [BUT+10], une paire de plus est correctement prédite.

Le second jeu de données est composé de 67 paires de stéréoisomères. Contrairement au précédent, certaines molécules possèdent plus d'un centre stéréogène. Dans ce cas, seul un stéréo sous-graphe minimal varie entre les deux stéréoisomères. Ainsi, pour quelques paires de stéréoisomères, les deux molécules possèdent des stéréo sous-graphes minimaux en commun.

La précision et le taux de paires non classées pour ce jeu de données sont montrés dans le Tableau 4.7. Comme pour le jeu de données précédent, le noyau stéréo (ligne 2) assigne une même classe aux deux molécules pour de nombreuses paires. Cependant, contrairement au premier jeu de données, les autres noyaux (ligne 1 et 3) obtiennent aussi des paires non classées. Ceci est dû à la présence de stéréo sous-graphes minimaux identiques entre les deux molécules d'une paire. Le noyau de motifs d'arbres adapté à la stéréoisomérie [BUT+10] est moins affecté que le noyau inter stéréo et obtient donc une meilleure précision.

Cependant, pour le noyau inter stéréo nous pouvons sélectionner les stéréo sous-graphes minimaux que l'on considère. On peut choisir de ne comparer que les stéréo sous-graphes minimaux changeant entre les deux molécules d'une paire. En effet, si les molécules d'une paire possèdent des stéréo sous-graphes minimaux identiques, alors ces stéréo sous-graphes ne peuvent pas permettre de

distinguer ces molécules, or le but de la classification est justement d'assigner à chaque molécule de la paire une classe différente.

Les résultats obtenus en faisant une telle sélection sont montrés dans la ligne 4 du Tableau 4.7. Grâce à cette sélection, le taux de paires non classées devient nul. Cette méthode obtient la meilleure précision. On peut remarquer que la somme du taux de paires non classées et de la précision obtenues par le noyau de motifs d'arbres adapté à la stéréoisomérie [BUT+10] est plus faible que la précision obtenue par le noyau stéréo avec sélection. On peut donc supposer que s'il existait un moyen de sélectionner certains motifs d'arbres dans la méthode de [BUT+10], on obtiendrait une précision plus faible que celle obtenue avec le noyau inter stéréo.

4.5 Conclusion

Les trois extensions du noyau stéréo présentées dans ce chapitre améliorent les performances de ce noyau en remédiant à ses défauts principaux. Le noyau stéréo, fondé sur l'extraction et la comparaison des stéréo sous-graphes minimaux, considère chaque stéréo sous-graphe minimal de façon isolée. De plus la mesure de similarité entre les stéréo sous-graphes est binaire : elle vaut un si les deux sous-graphes sont identiques et zéro sinon.

L'extension proposée dans la section 4.2 permet de prendre en compte les recouvrements entre les stéréo sous-graphes minimaux. Au lieu de comparer des ensembles de stéréo sous-graphes minimaux, on compare des graphes de stéréo sous-graphes minimaux. Ainsi, un stéréo sous-graphe minimal n'est plus considéré indépendamment de sa position par rapport aux autres stéréo sous-graphes minimaux.

La seconde extension, présentée dans la section 4.3, ajoute aux stéréo sous-graphes minimaux des informations sur leurs voisinages. Comme pour l'extension précédente, cela permet de considérer les stéréo sous-graphes minimaux avec leurs voisinages, mais de manière locale. La mesure de similarité des voisinages est utilisée pour pondérer la similarité entre deux stéréo sous-graphes minimaux identiques. On a donc une mesure de similarité qui n'est plus binaire. Cependant si un stéréo sous-graphe minimal est différent d'un autre, leur similarité sera toujours nulle.

La dernière extension remédie à ce problème en proposant une mesure de similarité entre deux stéréo sous-graphes minimaux différents. Pour cela, les stéréo sous-graphes minimaux sont divisés en quatre parties, associées aux voisins des stéréo sommets. Ces sous-graphes sont ensuite comparés en prenant en compte l'ordre autour des stéréo sommets.

Conclusion et Perspectives

Cette thèse a pour cadre la chémoinformatique et plus particulièrement les méthodes de prédiction de propriétés moléculaires. Ce domaine est basé sur l'hypothèse que des molécules similaires ont des propriétés similaires. Les méthodes de ce domaine demandent donc de construire une mesure de similarité entre les modèles représentant les molécules. L'un des modèles les plus utilisés pour représenter les molécules est le graphe moléculaire. Cependant, cette représentation ne peut pas distinguer les stéréoisomères qui sont des molécules qui n'ont pas le même positionnement relatif de leurs atomes. D'un point de vue local, les stéréoisomères sont dus à la présence d'atomes ou de groupement d'atomes appelés centres stéréogènes.

En s'inspirant des méthodes permettant de représenter les stéréoisomères, nous avons ajouté au graphe moléculaire une notion d'ordre dans le chapitre 2. Cette notion d'ordre permet de distinguer des stéréoisomères différents. Cependant, un même stéréoisomère peut être représenté par deux graphes localement ordonnés différents. Nous avons donc défini les notions de fonctions de réordonnancement puis d'isomorphisme d'ordres équivalents tels que deux graphes localement ordonnés ont des ordres équivalents si et seulement s'ils représentent un même stéréoisomère. Finalement, nous avons défini les stéréo sommets, qui représentent les centres stéréogènes lorsque les graphes localement ordonnés représentent les molécules.

La définition des stéréo sommets utilise tout le graphe localement ordonné. Or, certains sommets peuvent ne pas avoir d'influence sur la propriété stéréo d'un sommet. Nous avons donc défini dans le chapitre 3, le stéréo sous-graphe minimal qui est le plus petit sous-graphe permettant de caractériser un stéréo sommet. Nous avons aussi donné un algorithme permettant de calculer ces sous-graphes et une preuve de cet algorithme. L'algorithme commence par considérer un sous-graphe formé du stéréo sommet et de son voisinage, et agrandit itérativement ce sous-graphe jusqu'à ce qu'il soit suffisamment grand pour caractériser le stéréo sommet.

La similarité entre les graphes localement ordonnés est alors définie par le noyau stéréo, construit en comparant le nombre d'occurrences de chaque

stéréo sous-graphe minimal commun aux deux graphes localement ordonnés comparés. Le fait que la mesure de similarité entre graphes localement ordonnés soit un noyau nous permet d'utiliser les méthodes classiques d'apprentissage automatique. Comme le noyau stéréo compare des ensembles de stéréo sous-graphes minimaux, il permet de n'encoder que la stéréoisomérie. Cependant, ce noyau a deux inconvénients : chaque stéréo sous-graphe minimal est considéré de manière indépendante et deux stéréo sous-graphes minimaux qui ne sont pas strictement identiques ont une similarité nulle.

Le chapitre 4 propose de résoudre ces problèmes avec trois extensions.

La première vise à considérer les recouvrements entre les stéréo sous-graphes minimaux en construisant des graphes de recouvrements. Ces graphes sont construits en associant à chaque stéréo sous-graphe minimal un sommet. Une arête relie deux sommets si leurs stéréo sous-graphe minimaux associés interagissent. Le type de recouvrement entre deux stéréo sous-graphes minimaux est codé par le label de l'arête reliant les sommets. Ainsi, les graphes de recouvrements sont des modèles permettant d'encoder les relations entre les stéréo sous-graphes minimaux. On utilise alors des noyaux sur graphes classiques entre ces graphes de recouvrements afin d'obtenir une mesure de similarité qui prend en compte simultanément les caractéristiques des stéréo sous-graphes minimaux et leurs recouvrements.

La seconde extension présentée dans le chapitre 4 permet aussi de ne pas considérer les stéréo sous-graphes minimaux de manière isolée. Pour cela, on prend en compte le voisinage des stéréo sous-graphe minimaux. Afin de mesurer la similarité entre deux stéréo sous-graphes minimaux identiques, on utilise un noyau entre les k-voisinages des sommets situés à leurs frontières. Contrairement aux graphes de recouvrements, qui caractérisent globalement les relations entre les stéréo sous-graphes minimaux, cette extension résout le premier inconvénient du noyau stéréo grâce à une approche plus locale.

Finalement, la dernière extension du chapitre 4 permet de mesurer la similarité entre deux stéréo sous-graphes minimaux différents. Pour cela, on commence par associer à chaque voisin d'un stéréo sommet un sous-graphe. Ce sous-graphe est composé des sommets les plus proches du voisin considéré dans le stéréo sous-graphe minimal. La mesure de similarité entre stéréo sous-graphes est basée sur la comparaison de ces sous-graphes. L'inconvénient de cette mesure de similarité est que si deux des sous-graphes sont similaires, alors le stéréo sous-graphe minimal qui les contient sera considéré comme similaire à son opposé. On ajoute donc un terme baissant la similarité de deux stéréo sous-graphes minimaux opposés.

Perspectives

Les trois extensions permettent de résoudre l'un ou l'autre des points faibles du noyau stéréo, cependant aucun d'entre eux ne résout tous les problèmes. Il pourrait donc être intéressant de combiner ces approches, afin d'avoir une mesure de similarité entre graphes localement ordonnés qui ne considère pas les stéréo sous-graphes minimaux de manière indépendante et qui ne donne pas une similarité nulle à deux stéréo sous-graphes minimaux semblables mais différents. Avoir une similarité non nulle entre deux stéréo sous-graphes minimaux différents implique que l'on ne peut plus compter les motifs dans les graphes de recouvrements. La combinaison de ces deux approches n'est donc pas évidente.

Cette thèse ce concentre sur la stéréoisomérie de configuration, mais nous avons vu dans la section 1.2 qu'il existe un second type de stéréoisomérie : la stéréoisomérie de conformation. Comme pour la stéréoisomérie de configuration, cette stéréoisomérie a lieu lorsque deux molécules ont un même graphe moléculaire mais n'ont pas la même organisation spatiale de leurs atomes. La différence est que dans le cas de la stéréoisomérie de conformation, on peut passer d'un stéréoisomère (appelé dans ce cas conformère) à l'autre grâce à des rotations autour des liaisons simples. Le positionnement relatif des atomes, encodé dans les graphes moléculaires localement ordonnés, n'est donc pas suffisant pour différencier les conformères car il est insensible aux rotations autour des liaisons simples. Ainsi, notre modèle et donc nos noyaux ne peuvent pas être appliqués aux conformères. Une extension de cette thèse pourrait donc être de déterminer un nouveau modèle qui serait capable de différencier les conformères, par exemple en incluant les coordonnées des atomes, et de construire une mesure de similarité entre ces modèles.

Chapitre 5

Annexes

5.1 Preuves des théorèmes et propositions du chapitre 2

5.1.1 Preuve de la Proposition 5

Rappel de la proposition : *La relation d'isomorphisme localement ordonnée entre structures localement ordonnées est une relation d'équivalence.*

Démonstration. Il faut montrer que la relation d'isomorphisme localement ordonnée entre structures localement ordonnées est réflexive, symétrique et transitive :

1. L'isomorphisme entre structures localement ordonnées est réflexif.

   Soit $S = (\hat{S}, ord)$ une structure localement ordonnée, associée à un graphe $G(S) = (V, E, \mu, \nu)$. Soit f l'identité sur V :

   $$\forall v \in V, f(v) = v$$

   Alors f est un isomorphisme entre $\hat{S}$ et $\hat{S}$ (Définition 40, condition 3) et :

   $$\forall v \in V_{ord} \subset V, (f(v_1), \ldots, f(v_n)) = ord(f(v)) = ord(v) = (v_1, \ldots, v_n)$$

   où $N(v) = \{v_1, \ldots, v_n\}$.

   Nous avons donc :

   $$S \underset{o}{\simeq} S$$

2. L'isomorphisme entre structures localement ordonnées est symétrique.

   Soit $S_a = (\hat{S}_a, ord_a)$ et $S_b = (\hat{S}_b, ord_b)$ deux structures localement ordonnées, associées respectivement aux graphes $G(S_a) = (V_a, E_a, \mu_a, \nu_a)$ et $G(S_b) = (V_b, E_b, \mu_b, \nu_b)$. On suppose que

   $$S_a \underset{o}{\simeq} S_b$$

   On note f un isomorphisme localement ordonné entre S_a et S_b.

   Par définition f est un isomorphisme entre $\hat{S}_a$ et $\hat{S}_b$, donc il existe un isomorphisme f^{-1} entre $\hat{S}_b$ et $\hat{S}_a$ par la condition 1 de la Définition 40. Soit v_b un sommet de V_{ord_b} et $v_a = f^{-1}(v_b)$. Selon l'équation (2.1), v_a appartient à V_{ord_a}. Comme f est un isomorphisme localement ordonné entre S_a et S_b, nous avons :

   $$\begin{cases} ord_a(v_a) &= (v_{a1}, \ldots, v_{an}) \\ ord_b(v_b) &= ord_b(f(v_a)) = (v_{b1}, \ldots, v_{bn}) \ t.q \ \forall i \in \{1, \ldots, n\}, v_{bi} = f(v_{ai}) \end{cases}$$

   Donc :

   $$\begin{cases} ord_b(v_b) &= (v_{b1}, \ldots, v_{bn}) \\ ord_a(v_a) &= ord_a(f^{-1}(v_b)) = (v_{a1}, \ldots, v_{an}) = (f^{-1}(v_{b1}), \ldots, f^{-1}(v_{bn})) \end{cases}$$

   Par définition des isomorphismes localement ordonnés (Définition 42), on peut déduire que f^{-1} est un isomorphisme localement ordonné entre S_b et S_a et donc que

   $$S_a \underset{o}{\simeq} S_b \Rightarrow S_b \underset{o}{\simeq} S_a$$

3. L'isomorphisme entre structures localement ordonnées est transitif.

   Soit $S_a = (\hat{S}_a, ord_a)$, $S_b = (\hat{S}_b, ord_b)$ et $S_c = (\hat{S}_c, ord_c)$ trois structures localement ordonnées, associées respectivement aux graphes $G(S_a) = (V_a, E_a, \mu_a, \nu_a)$, $G(S_b) = (V_b, E_b, \mu_b, \nu_b)$ et $G(S_c) = (V_c, E_c, \mu_c, \nu_c)$. On suppose que

   - $S_a \underset{o}{\simeq} S_b$
   - $S_b \underset{o}{\simeq} S_c$

   On note f un isomorphisme localement ordonné entre S_a et S_b, et g un isomorphisme localement ordonné entre S_b et S_c.

   Comme l'isomorphisme entre les objets structurés est transitif, $g \circ f$ est un isomorphisme entre $\hat{S}_a$ et $\hat{S}_c$ (condition 2 de la Définition 40).

Soit v_a un sommet de V_{ord_a}. On note $v_b = f(v_a)$ et $v_c = g(v_b)$. Selon l'équation (2.1), le fait que v_a appartient à V_{ord_a} implique que v_b appartient à V_{ord_b}. En appliquant cette équation une fois de plus, on obtient que v_c appartient à V_{ord_c}. Comme f est un isomorphisme localement ordonné entre S_a et S_b, et g est un isomorphisme localement ordonné entre S_b et S_c, on a :

$$\begin{cases} ord_a(v_a) & = & (v_{a1}, \ldots, v_{an}) \\ ord_b(f(v_a)) & = & ord_b(v_b) = (f(v_{a1}), \ldots, f(v_{an})) \underset{not.}{=} (v_{b1}, \ldots, v_{bn}) \\ ord_c(g(v_b)) & = & (g(v_{b1}), \ldots, g(v_{bn})) \end{cases}$$

On peut en déduire que :

$$ord_c(g(v_b)) = ord_c(g \circ f(v_a)) = (g \circ f(v_{a1}), \ldots, g \circ f(v_{an}))$$

Par définition des isomorphismes localement ordonnés (Définition 42), on peut déduire que $g \circ f$ est un isomorphisme localement ordonné entre S_a et S_c et donc que :

$$S_a \underset{o}{\simeq} S_b \wedge S_b \underset{o}{\simeq} S_c \Rightarrow S_a \underset{o}{\simeq} S_c$$

En conclusion, la relation d'isomorphisme localement ordonnée entre structures localement ordonnées est réflexive, symétrique et transitive. C'est donc une relation d'équivalence. □

5.1.2 Preuve du Théorème 2

Rappel du théorème : *Soit Σ une famille valide de fonctions de réordonnancement. La relation d'équivalence d'ordres sur les structures localement ordonnées, définie dans la Définition 47 et fondée sur la famille Σ est une relation d'équivalence.*

Pour prouver ce théorème, nous avons besoin des deux lemmes suivants :

Lemme 1. *Soit Σ une famille valide de fonctions de réordonnancement. Pour toute structure localement ordonnée S et toute fonction de réordonnancement $\sigma \in \Sigma_S$, le groupe des fonctions de réordonnancement $\Sigma_{\sigma(S)}$ de $\sigma(S)$ est égal au groupe Σ_S des fonctions de réordonnancement de S :*

$$\left.\begin{array}{l} \forall S \in \mathscr{S}, \\ \forall \sigma \in \Sigma_S, \end{array}\right) \Sigma_{\sigma(S)} = \Sigma_S$$

Démonstration. Soit σ une fonction de réordonnancement appartenant à Σ_S. Comme S et $\sigma(S)$ ne diffèrent que par l'ordre autour de leurs sommets, l'identité Id est un isomorphisme entre leurs objets structurés associés $\widehat{\sigma(S)}$ et $\hat{S}$. Alors, en utilisant la deuxième condition de la Définition 46, pour toute fonction de réordonnancement $\sigma' \in \Sigma_{\sigma(S)}$, il existe une fonction de réordonnancement $\sigma'' \in \Sigma_S$ telle que :

$$\forall v \in V_{ord},\ \sigma''(v) = \sigma'(Id^{-1}(v)) = \sigma'(v)$$

σ' et σ'' sont donc égales, et donc σ' appartient à Σ_S. $\Sigma_{\sigma(S)}$ est donc inclus dans Σ_S. L'inclusion inverse est montrée de la même manière. □

Lemme 2. *Soit Σ une famille valide de fonctions de réordonnancement. Soit deux structures localement ordonnées $S_a = (\hat{S}_a, ord_a)$ et $S_b = (\hat{S}_b, ord_b)$ telles que :*

$$S_a \underset{o}{\simeq} S_b$$

Soit σ_a et σ_b deux fonctions de réordonnancement appartenant respectivement à Σ_{S_a} et Σ_{S_b}. On suppose que :

$$\forall v \in V_{ord_a},\ \sigma_a(v) = \sigma_b(f(v)),$$

où f est un isomorphisme localement ordonné entre S_a et S_b.

Alors :

$$\sigma_a(S_a) \underset{o}{\simeq} \sigma_b(S_b)$$

Démonstration. Soit f un isomorphisme localement ordonné entre S_a et S_b, et un sommet v appartenant à V_{ord_a}. On note $u = f(v)$. Par la Définition 42 on a :

$$\begin{cases} ord_a(v) &= (v_1, \ldots, v_n) \\ ord_b(u) &= (u_1, \ldots, u_n),\ t.q\ \forall i \in \{1, \ldots, n\}, u_i = f(v_i) \end{cases}$$

Notons φ_v la permutation définie sur les sommets v et $f(v)$ par σ_a et σ_b :

$$\varphi_v = \sigma_a(v) = \sigma_b(f(v))$$

En considérant les structures réordonnées $\sigma_a(S_a) = (\hat{S}_a, ord_{\sigma_a})$ et $\sigma_b(S_b) = (\hat{S}_b, ord_{\sigma_b})$, nous avons par la Définition 44 :

$$\begin{cases} ord_{\sigma_a}(v) &= (v_{\varphi_v(1)}, \ldots, v_{\varphi_v(n)}) \\ ord_{\sigma_b}(u) &= (u_{\varphi_v(1)}, \ldots, u_{\varphi_v(n)}) \end{cases}$$

Par définition, pour tout i compris entre 1 et n, u_i est égal à $f(v_i)$. De plus, la même permutation φ_v est appliquée à $v_1 \dots v_n$ et à $u_1 \dots u_n$. On a donc :

$$\forall i \in \{1, \dots, n\}\ u_{\varphi_v(i)} = f(v_{\varphi_v(i)})$$

L'isomorphisme f associe donc l'ordre de $\sigma_a(S_a)$ à l'ordre de $\sigma_b(S_b)$.

De plus, comme f est un isomorphisme entre structures localement ordonnées, il correspond aussi à un isomorphisme entre les objets structurés associés à ces structures. On a donc par Définition 42 :

$$\sigma_a(S_a) \underset{o}{\simeq} \sigma_b(S_b)$$

□

Nous donnons maintenant la preuve du Théorème 2.

Démonstration. Soit Σ une famille valide de fonctions de réordonnancement. Il faut montrer que la relation d'équivalence d'ordres entre structures localement ordonnées est réflexive, symétrique et transitive :

1. La relation d'équivalence d'ordres entre structures localement ordonnées est réflexive.

   Soit $S = (\hat{S}, ord)$ une structure localement ordonnée. Par Définition 46, Σ_S est un groupe et donc Id_S appartient à Σ_S.

   On a par définition de Id_S (Définition 45) :

   $$\forall v \in V_{ord},\ ord_{Id_S}(v) = ord(v)$$

   Ainsi on a par la Définition 42 :

   $$Id_S(S) \underset{o}{\simeq} S$$

   Donc

   $$S \underset{\Sigma}{\simeq} S$$

2. La relation d'équivalence d'ordres entre structures localement ordonnées est symétrique.

   Soit $S_a = (\hat{S}_a, ord_a)$ et $S_b = (\hat{S}_b, ord_b)$ deux structures localement ordonnées d'ordres équivalents :

   $$S_a \underset{\Sigma}{\simeq} S_b$$

Par la Définition 47 on a :

$$\exists \sigma \in \Sigma_{S_a} \; t.q. \; \sigma(S_a) \underset{o}{\simeq} S_b$$

Comme Σ_{S_a} est un groupe (Définition 46), il existe une fonction de réordonnancement σ^{-1} appartenant à Σ_{S_a} telle que :

$$\sigma^{-1}(\sigma(S_a)) = S_a$$

De plus, d'après le Lemme 1, les ensembles Σ_{S_a} et $\Sigma_{\sigma(S_a)}$ sont identiques. Donc σ^{-1} appartient à $\Sigma_{\sigma(S_a)}$.

Soit f un isomorphisme localement ordonné entre $\sigma(S_a)$ et S_b. f est aussi un isomorphisme entre $\widehat{\sigma(S_a)}$ et $\hat{S}_b$, et σ^{-1} appartient à Σ_{S_a}. Ainsi, d'après la remarque 1, il existe une fonction de réordonnancement $(\sigma^{-1})'$ appartenant à Σ_{S_b} et équivalente à σ^{-1}. En d'autres termes :

$$\exists (\sigma^{-1})' \in \Sigma_{S_b} \; t.q \; \forall v \in V_{ord_a}, \sigma^{-1}(v) = (\sigma^{-1})'(f(v))$$

Les structures localement ordonnées $\sigma(S_a)$ et S_b sont isomorphes, et il existe deux fonctions de réordonnancement σ^{-1} et $(\sigma^{-1})'$ appartenant respectivement à $\Sigma_{\sigma(S_a)}$ et à Σ_{S_b} telles que :

$$\forall v \in V_{ord_a}, \sigma^{-1}(v) = (\sigma^{-1})'(f(v))$$

On a donc par le Lemme 2 :

$$\sigma^{-1}(\sigma(S_a)) \underset{o}{\simeq} (\sigma^{-1})'(S_b)$$

Ainsi

$$S_a \underset{o}{\simeq} (\sigma^{-1})'(S_b)$$

Par symétrie de l'isomorphisme localement ordonné (Proposition 5)

$$(\sigma^{-1})'(S_b) \underset{o}{\simeq} S_a$$

Finalement par la Définition 47 on a

$$S_b \underset{\Sigma}{\simeq} S_a$$

En conclusion :

$$S_a \underset{\Sigma}{\simeq} S_b \Rightarrow S_b \underset{\Sigma}{\simeq} S_a$$

Les deux cas étant symétriques l'implication inverse est montrée de la même manière.

3. La relation d'équivalence d'ordres entre structures localement ordonnées est transitive.

   Soit $S_a = (\hat{S}_a, ord_a)$, $S_b = (\hat{S}_b, ord_b)$ et $S_c = (\hat{S}_c, ord_c)$ trois structures localement ordonnées telles

   - $S_a \underset{\Sigma}{\simeq} S_b$
   - $S_b \underset{\Sigma}{\simeq} S_c$

   Selon la Définition 47, il existe deux fonctions de réordonnancement σ_a et σ'_b appartenant respectivement à Σ_{S_a} et à Σ_{S_b}, telles que

   - $\sigma_a(S_a) \underset{o}{\simeq} S_b$
   - $\sigma'_b(S_b) \underset{o}{\simeq} S_c$

   Par symétrie de l'isomorphisme localement ordonné (Proposition 5) nous avons :

   $$S_b \underset{o}{\simeq} \sigma_a(S_a)$$

   On note f_{ba} un isomorphisme localement ordonné entre S_b et $\sigma_a(S_a)$. Par définition des isomorphismes localement ordonnés, f_{ba} est aussi un isomorphisme entre les objets structurés $\hat{S}_b$ et $\widehat{\sigma_a(S_a)}$. Comme σ'_b appartient à Σ_{S_b}, il existe (Remarque 1) une fonction de réordonnancement σ'_a appartenant à $\Sigma_{\sigma_a(S_a)}$ telle que :

   $$\forall v \in V_{ord_b},\ \sigma'_b(v) = \sigma'_a(f_{ba}(v))$$

   Les structures localement ordonnées S_b et $\sigma_a(S_a)$ sont isomorphes, et il existe deux fonctions de réordonnancement σ'_b et σ'_a appartenant respectivement à Σ_{S_b} et à $\Sigma_{\sigma_a(S_a)}$ telles que :

   $$\forall v \in V_{ord_b}, \sigma'_b(v) = \sigma'_a(f_{ba}(v))$$

   On a donc par le Lemme 2 :

   $$\sigma'_b(S_b) \underset{o}{\simeq} \sigma'_a(\sigma_a(S_a))$$

   Par transitivité et symétrie de l'isomorphisme localement ordonné (Proposition 5) et comme $\sigma'_b(S_b)$ et S_c sont des structures localement ordonnées isomorphes, nous avons :

   $$\sigma'_a(\sigma_a(S_a)) \underset{o}{\simeq} S_c$$

D'après le Lemme 1, $\Sigma_{\sigma_a(S_a)}$ et Σ_{S_a} sont égales, donc σ'_a appartient à Σ_{S_a}. σ_a appartient aussi à Σ_{S_a}. Comme Σ_{S_a} est un groupe (Définition 46) leur composition $\sigma'_a \circ \sigma_a$ appartient à Σ_{S_a}.

Finalement par Définition 47, nous avons :

$$S_a \underset{\Sigma}{\simeq} S_c$$

En conclusion, la relation d'équivalence d'ordres entre structures localement ordonnées est réflexive, symétrique et transitive. C'est donc une relation d'équivalence. □

5.1.3 Preuve de la Proposition 6

Rappel de la proposition : *Soit* $G = (\widehat{G}, ord)$ *et* $G' = (\widehat{G'}, ord')$ *deux graphes moléculaires localement ordonnés tel qu'il existe un isomorphisme* f *entre les graphes* $\widehat{G}$ *et* $\widehat{G'}$.

Alors $f(V_{ord}) = V'_{ord}$

De plus nous avons :

- $f(V^{PAC}) = V'^{PAC}$
- $f(V^{DB}) = V'^{DB}$
- $\forall v \in V^{DB}$, $f(n_=(v)) = n_=(f(v))$.

Démonstration. Soit v un sommet appartenant à V_{ord}.

- Si v appartient à V^{PAC} :

  Par la Définition 49,

  $$v \in V^{PAC} \Leftrightarrow \begin{cases} \mu(v) = \text{'C'} \\ |N(v)| = 4 \end{cases}$$

  Comme f est un isomorphisme entre $\widehat{G}$ et $\widehat{G'}$, nous avons :

  $$\begin{cases} \mu(f(v)) & = & \mu(v) & = & \text{'C'} \\ |N(f(v))| & = & |N(v)| & = & 4 \end{cases}$$

  Donc par Définition 49, $f(v)$ appartient à V'^{PAC}.

$f(V^{PAC})$ est donc inclus dans V'^{PAC}. Par symétrie de la relation d'isomorphisme et en considérant f^{-1}, on peut démontrer de la même manière que V'^{PAC} est inclus dans $f(V^{PAC})$.

Nous avons donc :

$$f(V^{PAC}) = V'^{PAC}$$

- Si v appartient à V^{DB} :

  On note $w = n_{=}(v)$. Par la Définition 50, nous avons :

  $$v \in V^{DB} \Leftrightarrow \begin{cases} \mu(v) = \mu(w) = \text{'C'} \\ |N(v)| = |N(w)| = 3 \end{cases}$$

  Comme f est un isomorphisme entre $\widehat{G}$ et $\widehat{G'}$, nous avons :

  $$\begin{cases} \mu(f(v)) & = & \mu(v) & = & \text{'C'} \\ \mu(f(w)) & = & \mu(w) & = & \text{'C'} \\ |N(f(v))| & = & |N(v)| & = & 3 \\ |N(f(w))| & = & |N(w)| & = & 3 \end{cases}$$

  Comme v appartient à V^{DB}, il existe une arête $e = \{v, w\}$ entre v et w ayant pour étiquette 2. Cette arête est préservée par l'isomorphisme f entre $\widehat{G}$ et $\widehat{G'}$, donc il existe une arête $e' = \{f(v), f(w)\}$ telle que son étiquette soit égale à 2.

  Nous avons donc par Définition 50 :

  - $f(v) \in V'^{DB}$
  - $f(w) \in V'^{DB}$
  - $f(n_{=}(v)) = f(w) = n_{=}(f(v))$

  $f(V^{DB})$ est donc inclus dans V'^{DB}. Par symétrie de la relation d'isomorphisme et en considérant f^{-1}, on peut démontrer de la même manière que V'^{DB} est inclus dans $f(V^{DB})$.

  Nous avons donc :

  $$f(V^{DB}) = V'^{DB}$$

Par Définition 51 :

$$V_{ord} = V^{PAC} \cup V^{DB}$$

Nous pouvons en conclure que :

$$f(V_{ord}) = V'_{ord}$$

□

5.1.4 Preuve du Théorème 3

Rappel du théorème : *L'ensemble de fonction de réordonnancement* $\Sigma^M = \{\Sigma^M_G,\ G \in \mathcal{OM}\}$ *est une famille valide de fonctions de réordonnancement.*

Démonstration. Afin de prouver que Σ^M est une famille valide de fonctions de réordonnancement, nous commençons par prouver que pour tout graphe moléculaire localement ordonné G, Σ^M_G est un groupe pour la composition.

Pour cela, il faut montrer que la fonction de réordonnancement identité fait partie de Σ^M_G, que Σ^M_G est fermé pour la composition et que pour chaque fonction de réordonnancement de Σ^M_G, son inverse est aussi incluse dans Σ^M_G.

1. La fonction de réordonnancement identité Id_G fait partie de Σ^M_G.

   Nous considérons la fonction de réordonnancement identité Id_G :

   $$\forall v \in V_{ord}, Id_G(v) = Id_{|N(v)|}$$

   où Id_n est la permutation identité sur Π_n.

   Comme l'identité est une permutation paire, nous avons :

   $$\begin{cases} \forall v \in V^{PAC} & \epsilon(Id_G(v)) = 1 \\ \forall v \in V^{DB} & \epsilon(Id_G(v)) = \epsilon(Id_G(n_=(v)) = 1 \end{cases}$$

   Donc par Définition 52 nous avons :

   $$Id_G \in \Sigma^M_G$$

2. Σ^M_G est fermé pour la composition.

   Soit σ et σ' deux fonctions de réordonnancement de Σ^M_G. Nous considérons un sommet v appartenant à V_{ord}.

   - Si v appartient à V^{PAC} :
     Comme ϵ est un morphisme entre $\Pi_{|N(v)|}$ et $(\{-1, 1\}, \times)$ nous avons :

     $$\epsilon(\sigma(v) \circ \sigma'(v)) = \epsilon(\sigma(v))\epsilon(\sigma'(v)) = 1 \times 1 = 1$$

   - Si v appartient à V^{DB} :
     On note $w = n_=(v)$. Comme $\sigma(v)$ et $\sigma(w)$ ont une même signature et $\sigma'(v)$ et $\sigma'(w)$ également, nous avons :

     $$\begin{aligned} \epsilon(\sigma(v) \circ \sigma'(v)) &= \epsilon(\sigma(v))\epsilon(\sigma'(v)) \\ &= \epsilon(\sigma(w))\epsilon(\sigma'(w)) \\ &= \epsilon(\sigma(w) \circ \sigma'(w)) \end{aligned}$$

Les permutations $\sigma(v) \circ \sigma'(v)$ et $\sigma(w) \circ \sigma'(w)$ ont donc une même parité.

Donc par Définition 52 nous avons :

$$\sigma \in \Sigma_G^M \wedge \sigma' \in \Sigma_G^M \Rightarrow \sigma \circ \sigma' \in \Sigma_G^M$$

3. L'inverse de chaque fonction de réordonnancement de Σ_G^M est aussi incluse dans Σ_G^M.

   Nous considérons la fonction de réordonnancement σ^{-1} telle que :

   $$\forall v \in V_{ord},\ \sigma^{-1}(v) = (\sigma(v))^{-1}.$$

   Nous avons par Définition 45 :

   $$\forall v \in V_{ord},\ \sigma \circ \sigma^{-1}(v) = \sigma(v) \circ (\sigma(v))^{-1} = Id_{|N(v)|}$$

   Ainsi

   $$\sigma \circ \sigma^{-1} = Id_G$$

   Nous devons montrer que, pour toute fonction de réordonnancement σ appartenant à Σ_G^M, σ^{-1} appartient aussi à Σ_G^M. On considère un sommet v appartenant à V_{ord}.

   - Si v appartient à V^{PAC} :

     $$\epsilon(\sigma^{-1}(v)) = \epsilon(\sigma(v)^{-1}) = \epsilon(\sigma(v)) = 1$$

     Donc $\sigma^{-1}(v)$ est paire.

   - Si v appartient à V^{DB} :
     On note $w = n_=(v)$. Comme $\sigma(v)$ et $\sigma(w)$ ont la même parité et deux permutations inverses ont la même parité on a :

     $$\epsilon(\sigma^{-1}(v)) = \epsilon(\sigma(v)) = \epsilon(\sigma(w)) = \epsilon(\sigma^{-1}(w))$$

     $\sigma^{-1}(v)$ et $\sigma^{-1}(w)$ ont donc la même parité.

   Donc par Définition 52, la fonction de réordonnancement σ^{-1} appartient à Σ_G^M.

La composition de fonctions est associative, la fonction de réordonnancement identité fait partie de Σ_G^M, Σ_G^M est fermé pour la composition et pour chaque fonction de réordonnancement de Σ_G^M, son inverse est aussi incluse dans Σ_G^M, donc Σ_G^M est un groupe pour la composition.

Afin de prouver que Σ^M est une famille valide de fonctions de réordonnancement, il nous faut maintenant montrer que :

$$\left.\begin{array}{l}\forall f \in \mathrm{Isom}(\hat{G}, \hat{G}') \\ \forall \sigma \in \Sigma_G^M\end{array}\right) \sigma' = \sigma \circ f^{-1} \in \Sigma_{G'}^M$$

On considère deux graphes moléculaires localement ordonnés G et G' tels que leurs objets structurés associés, les graphes moléculaires $\hat{G}$ et $\widehat{G'}$, soient isomorphes par une fonction f. On considère une fonction de réordonnancement σ appartenant à Σ_G^M et on définit la fonction de réordonnancement σ' par :

$$\sigma' = \sigma \circ f^{-1}$$

On veut prouver que σ' appartient à $\Sigma_{G'}^M$. On considère un sommet v' appartenant à V'_{ord} et on note v son image par la fonction f^{-1}.

- Si v' appartient à V'^{PAC} :

  Nous avons par définition de σ' :

  $$\sigma'(v') = \sigma(v)$$

  f^{-1} étant un isomorphisme entre $\widehat{G'}$ et $\hat{G}$, nous avons par la Proposition 6 que v appartient à V^{PAC}.

  Comme σ appartient à Σ_G^M et v à V^{PAC}, $\sigma(v)$ est paire. Ainsi $\sigma'(v')$ est également paire.

- Si v' appartient à V'^{DB} :

  On note $w' = n_=(v')$. Nous avons par définition de σ' :

  $$\left\{\begin{array}{lcl}\sigma'(v') & = & \sigma(v) \\ \sigma'(w') & = & \sigma(w) \quad \text{avec} \quad w = f^{-1}(w')\end{array}\right.$$

  f^{-1} étant un isomorphisme entre $\widehat{G'}$ et $\hat{G}$, nous avons par la Proposition 6 que v et w appartiennent à V^{DB} et $w = n_=(v)$.

  Comme σ appartient à Σ_G^M, nous avons par définition de Σ_G^M :

  $$\epsilon(\sigma(v)) = \epsilon(\sigma(w))$$

Donc par définition de σ' :

$$\epsilon(\sigma'(v')) = \epsilon(\sigma(v)) = \epsilon(\sigma(w)) = \epsilon(\sigma'(w'))$$

Les permutations $\sigma'(v')$ et $\sigma'(w')$ sont donc de même parité.

Nous avons donc :

$$\sigma' = \sigma \circ f^{-1} \in \Sigma^M_{G'}$$

Σ^M possède donc bien les deux propriétés des familles valides de fonctions de réordonnancement (Définition 46), c'est donc une famille valide de fonctions de réordonnancement.

□

5.1.5 Preuve de la Proposition 7

Rappel de la proposition : *Un sommet v appartenant à V^{DB} est un stéréo sommet si et seulement si $n_=(v)$ est un stéréo sommet.*

Démonstration. Soit $G = (\widehat{G} = (V, E, \mu, \nu), ord)$ un graphe moléculaire localement ordonné et un sommet v appartenant à V^{DB}. On note $w = n_=(v)$.

On suppose que v n'est pas un stéréo sommet.

Alors l'ensemble d'isomorphisme $\mathcal{F}^v_G$ n'est pas vide. On considère donc un isomorphisme d'équivalence d'ordre f appartenant à $\mathcal{F}^v_G$. D'après la Définition 53, nous avons

- $f \in \text{IsomEqOrd}(G, \tau^v_{i,j}(G))$
- $f(v) = v$

où i et j sont deux entiers différents compris entre 1 et 3 et $\tau^v_{i,j}$ est une fonction de réordonnancement égale à l'identité sur tous les sommets de V à part v pour lequel elle permute les sommets d'indices i et j dans $ord(v)$.

Comme f appartient à $\text{IsomEqOrd}(G, \tau^v_{i,j}(G))$ qui est un sous-ensemble de $\text{Isom}(\widehat{G}, \widehat{G})$, la Proposition 6 donne :

$$f(w) = w$$

On définit σ comme la composition de $\tau^w_{i',j'}$ avec $\tau^v_{i,j}$ où i' et j' sont deux entiers différents compris entre 1 et 3 et $\tau^w_{i',j'}$ est une fonction de réordonnancement égale à l'identité sur tous les sommets de V à part w pour lequel elle permute les sommets d'indices i' et j' dans $ord(w)$.

$\tau^v_{i,j}(v)$ est une transposition et $\tau^v_{i,j}(w)$ est l'identité. $\tau^w_{i',j'}(v)$ est égale à l'identité et $\tau^w_{i',j'}(w)$ est une transposition.

Nous avons donc :

- $\epsilon(\sigma(v)) = \epsilon(\tau^w_{i',j'}(v)) \times \epsilon(\tau^v_{i,j}(v)) = 1 \times -1 = -1$
- $\epsilon(\sigma(w)) = \epsilon(\tau^w_{i',j'}(w)) \times \epsilon(\tau^v_{i,j}(w)) = -1 \times 1 = -1$

De plus, pour tout sommet u différent de v et w, la permutation $\sigma(u)$ est l'identité. Selon la Définition 52, σ appartient à Σ^M_G.

Comme f est un isomorphisme d'équivalence d'ordres entre G et $\tau^v_{i,j}(G)$, la Définition 47 implique qu'il existe une fonction de réordonnancement σ' appartenant à Σ^M_G, telle que f soit un isomorphisme localement ordonné entre $\sigma'(G)$ et $\tau^v_{i,j}(G)$. Donc par le Lemme 2, f est un isomorphisme localement ordonné entre $\sigma \circ \sigma'(G)$ et $\sigma \circ \tau^v_{i,j}(G)$. σ étant définie comme la composition de $\tau^v_{i,j}$ et $\tau^w_{i',j'}$, sa composition avec $\tau^v_{i,j}$ est égale à $\tau^w_{i',j'}$. Ainsi nous avons :

$$f \in \text{IsomOrd}(\sigma \circ \sigma'(G), \tau^w_{i',j'}(G))$$

Comme Σ^M est une famille valide de fonctions de réordonnancement (Théorème 3), Σ^M_G est un groupe. $\sigma \circ \sigma'$ appartient donc à Σ^M_G, et :

- $f \in \text{IsomEqOrd}(G, \tau^w_{i',j'}(G))$
- $f(w) = w$

Ainsi, f appartient à $\mathcal{F}^w_G$, ce qui prouve que cet ensemble n'est pas vide et que w n'est pas un stéréo sommet. Nous avons donc :

$$v \notin \mathcal{SV}(G) \Rightarrow n_=(v) \notin \mathcal{SV}(G) \tag{5.1}$$

La contraposée de (5.1) nous donne :

$$n_=(v) \in \mathcal{SV}(G) \Rightarrow v \in \mathcal{SV}(G) \tag{5.2}$$

Pour tout sommet u de V^{DB} nous avons :

$$n_=(n_=(u)) = u \tag{5.3}$$

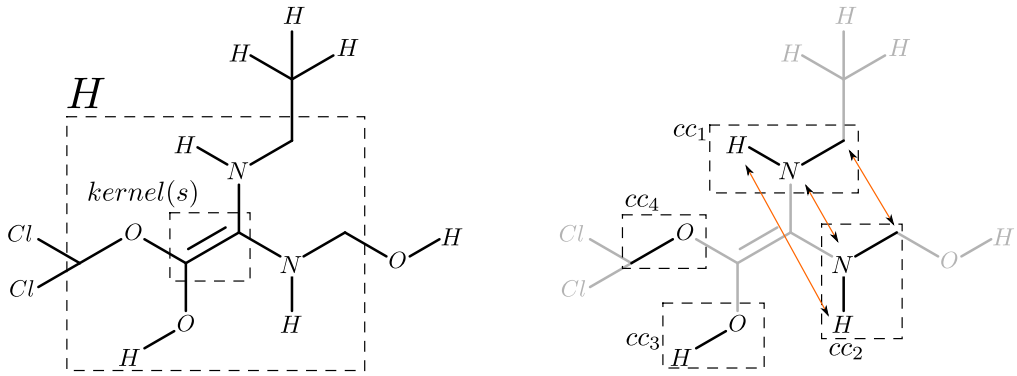
La relation (5.3) appliquée à (5.2) nous donne :

$$v \in \mathcal{SV}(G) \Rightarrow n_=(v) \in \mathcal{SV}(G) \tag{5.4}$$

Nous avons donc par (5.2) et (5.4) :

$$v \in \mathcal{SV}(G) \Leftrightarrow n_=(v) \in \mathcal{SV}(G)$$

□



(a) (b)

FIGURE 5.1 – On considère un sous-graphe H (a) d'un graphe localement ordonné. Les composantes connexes $\{cc_1, cc_2, cc_3, cc_4\}$ (b) de H sont séparées en deux groupes. Dans ce cas on a $\mathcal{I} = \{1, 2\}$ et $\mathcal{J} = \{3, 4\}$.

5.2 Lemmes utilisés pour prouver le théorème 4 du chapitre 3

Les lemmes donnés dans cette section servent à prouver que la propriété de stéréoisomérie d'un stéréo sommet est capturée par son stéréo sous-graphe minimal.

Dans ces lemmes, un sous-graphe H sera divisé en plusieurs parties. On considérera les composantes connexes de $H - kernel(s)$. La Figure 5.1 montre un exemple de cette division de H.

Lemme 3. *Soit $G = (\widehat{G} = (V, E, \mu, \nu), ord)$ un graphe moléculaire localement ordonné et s un stéréo sommet de G. Soit H un sous-graphe induit de G. On suppose que H est connexe, qu'il ne capture pas la propriété de stéréoisomérie de s, et que $StereoStar(s)$ est inclus dans H. Soit f un isomorphisme d'équivalence d'ordres appartenant à $\mathcal{F}_H^s$.*

On note $\{cc_1 \dots cc_n\}$ les composantes connexes obtenues en enlevant $kernel(s)$ de H. Comme H contient au moins $StereoStar(s)$, $H - kernel(s)$ contient au moins $StereoStar^*(s)$. Ainsi, les composantes connexes de $H - kernel(s)$ existent et ne sont pas vides.

On définit les ensemble d'indices (voir Figure 5.1 pour un exemple) :

- $\mathcal{I} = \{i \mid cc_i \cap \mathcal{E}_f^H \neq \varnothing\}$.
- $\mathcal{J} = \{j \mid cc_j \cap \mathcal{E}_f^H = \varnothing\}$.

Finalement, on note

- $cc_{\mathcal{I}} = \bigcup_{i \in \mathcal{I}} cc_i$.
- $cc_{\mathcal{J}} = \bigcup_{j \in \mathcal{J}} cc_j$.

On a alors :

$$cc_{\mathcal{I}} \subset \mathcal{E}_f^H. \tag{5.5}$$

Si $\mathcal{J}$ n'est pas vide on a :

$$\forall j \in \mathcal{J},\ cc_j \cap StereoStar^*(s) \neq \varnothing \tag{5.6}$$
$$\forall v \in cc_j \cap StereoStar^*(s),\ f(v) = v. \tag{5.7}$$

Démonstration. Dans un premier temps, nous montrons que :

$$cc_{\mathcal{I}} \subset \mathcal{E}_f^H.$$

Soit i appartenant à $\mathcal{I}$. Par définition de $\mathcal{I}$, l'intersection de cc_i avec $\mathcal{E}_f^H$ n'est pas vide. On note donc u un sommet appartenant à la fois à cc_i et à $\mathcal{E}_f^H$. Par définition de $\mathcal{E}_f^H$ (et la remarque 3), il existe un chemin $p_u = (u_1, \ldots, u_q)$ appartenant à H tel que $u_q = u$, u_1 appartienne à $StereoStar^*(s)$ et que u_1 soit différent de son image par f. Comme cc_i est une composante connexe de $H - kernel(s)$, p_u est inclus dans cc_i. Ainsi u_1 appartient à cc_i.

Comme cc_i est une composante connexe, nous avons :

$$\forall v \in cc_i,\ \exists p = (u_1, v_1, \ldots, v) \in cc_i$$

où p est un chemin.

Comme u_1 appartient à $StereoStar^*(s)$ et u_1 est différent de $f(u_1)$, on peut en déduire que pour tout v de cc_i, v appartient aussi à $\mathcal{E}_f^H$.

Ainsi $cc_i \subset \mathcal{E}_f^H$.

Nous montrons maintenant que :

$$\forall j \in \mathcal{J},\ cc_j \cap StereoStar^*(s) \neq \varnothing$$
$$\forall v \in cc_j \cap StereoStar^*(s),\ f(v) = v.$$

On suppose que $\mathcal{J}$ n'est pas vide. Soit j appartenant à $\mathcal{J}$. Soit u un sommet de cc_j. Comme les sommets de cc_j sont inclus dans H, u appartient à H.

On considère un chemin $p = (u_0, u_1 \ldots, u_q)$ reliant $u = u_q$ à u_0 un sommet de $kernel(s)$. Ce chemin existe car H est connexe. Sur ce chemin on considère le sommet u_r d'indice r tel que :

$$r = \arg\min_k \{\forall l \geqslant k \mid u_l \notin kernel(s)\}$$

r est compris entre 1 et $q-1$ car u_0 appartient à $kernel(s)$ et u_q n'appartient pas à $kernel(s)$.

Par la définition de l'indice r, le chemin $p' = (u_r \ldots, u_q)$ est contenu dans cc_j. On en déduit que u_r appartient à cc_j. De plus, u_r est voisin de u_{r-1} qui est un sommet appartenant à $kernel(s)$. u_r n'appartient pas à $kernel(s)$, donc u_r appartient à $StereoStar^*(s)$.

En conclusion, u_r appartient à $StereoStar^*(s)$ et cc_j, donc l'intersection de ces ensembles n'est pas vide :

$$\forall j \in \mathcal{J},\ cc_j \cap StereoStar^*(s) \neq \varnothing$$

Soit j appartenant à $\mathcal{J}$. On considère v appartenant à l'intersection de cc_j et de $StereoStar^*(s)$. Si l'on suppose que v est différent de son image $f(v)$ alors v appartient à $\mathcal{E}_f^H$. Ainsi, l'intersection de cc_j et de $\mathcal{E}_f^H$ n'est pas vide. Ceci est contradictoire avec la définition de cc_j, et donc v est égal à $f(v)$.

□

Lemme 4. *Avec les mêmes hypothèses et notations que le Lemme 3, nous avons :*

$$f(cc_{\mathcal{I}}) = cc_{\mathcal{I}}$$

Démonstration. Soit i appartenant à $\mathcal{I}$. Soit v un sommet de cc_i.

Selon le Lemme 3, les sommets de $cc_{\mathcal{I}}$ sont inclus dans $\mathcal{E}_f^H$, donc v appartient à $\mathcal{E}_f^H$. Selon la Définition 59 il existe donc un chemin $p = (v_1, \ldots, v_q)$ inclus dans H avec $v_q = v$, et tel que v_1 soit un sommet de $StereoStar^*(s)$ avec $f(v_1) \neq v_1$. Comme f est un isomorphisme appartenant à $\mathcal{F}_H^s$, l'image $f(u)$ de tout sommet u appartenant à H, appartient elle aussi à H. Le chemin $(f(v_1), \ldots, f(v_q))$ est donc inclus dans H.

On note $\tilde{v_1} = f(v_1)$.

Dans la remarque 3, on a vu que tout sommet appartenant à $kernel(s)$ est associé à lui-même par f. Comme v_1 appartient à $StereoStar^*(s)$, son image $\tilde{v_1}$ appartient aussi à $StereoStar^*(s)$. Comme $\tilde{v_1}$ est différent de v_1 et f est bijective, $f(\tilde{v_1})$ est aussi différent de $f(v_1) = \tilde{v_1}$.

En conclusion, il existe un chemin $p' = (\tilde{v_1}, \ldots, f(v))$ appartenant à H tel que $\tilde{v_1}$ appartienne à $StereoStar^*(s)$ et $\tilde{v_1}$ soit différent de son image $f(\tilde{v_1})$. Ainsi $f(v)$ appartient à $\mathcal{E}_f^H$.

Comme v n'appartient pas à $kernel(s)$, mais appartient à H, il appartient à une des composantes connexes $\{cc_1 \dots cc_n\}$. Comme il appartient aussi à $\mathcal{E}_f^H$, il appartient à une des composantes connexes $cc_{i'}$, telle que i' appartient à $\mathcal{I}$.

$f(cc_\mathcal{I})$ est donc inclus dans $cc_\mathcal{I}$.

Comme f est bijective on a $|f(cc_\mathcal{I})| = |cc_\mathcal{I}|$. On peut en conclure que

$$f(cc_\mathcal{I}) = cc_\mathcal{I}$$

□

Lemme 5. *Avec les mêmes hypothèses et notations que le Lemme 3, pour tout f appartenant à $\mathcal{F}_H^s$, l'ensemble $N(\mathcal{E}_f^H)$ n'est pas inclus dans V_H.*

Démonstration. Supposons que

$$N(\mathcal{E}_f^H) \subset V_H$$

Pour obtenir une contradiction, nous allons construire un isomorphisme d'équivalence d'ordres entre G et $\tau_{i,j}^s(G)$ ce qui contredira le fait que s soit un stéréo sommet. Afin de prouver que la fonction que nous allons construire est bien un isomorphisme d'équivalence d'ordres entre G et $\tau_{i,j}^s(G)$, nous avons besoin d'une propriété sur les cc_i et les cc_j découlant de notre hypothèse « $N(\mathcal{E}_f^H)$ est inclus dans V_H ».

Conséquence de l'hypothèse « $N(\mathcal{E}_f^H)$ est inclus dans V_H »

Nous devons montrer que l'inclusion de $N(\mathcal{E}_f^H)$ dans V_H implique que :

$$\left.\begin{array}{l} \forall v \in V_{G-H} \\ \forall p = (v, \dots, v_q) \end{array}\right\} v_q \in H \wedge kernel(s) \cap p = \varnothing \Rightarrow v_q \in cc_\mathcal{J} \tag{5.8}$$

Cela signifie que si un chemin ne passant pas par $kernel(s)$ relie un sommet v de G n'appartenant pas à H à un sommet v_q de H, alors le sommet v_q appartiendra à une composante connexe cc_j telle que j appartienne à $\mathcal{J}$.

Soit v un sommet appartenant à V_{G-H}, et $p = (v_0, \dots, v_q)$ un chemin reliant $v = v_0$ à un sommet v_q appartenant à H tel qu'aucun des sommets de $kernel(s)$ n'appartienne au chemin p. Comme v_q appartient à H et n'appartient pas à $kernel(s)$ il appartient à une composante connexe cc_i. On suppose que i appartient à $\mathcal{I}$.

On note r un entier compris entre 0 et $q-1$ tel que le sommet v_r n'appartienne pas à V_H et que le sommet v_{r+1} appartienne à cc_i. Un tel indice existe car v_0 n'est pas inclus dans V_H et v_q est inclus dans cc_i.

Comme v_{r+1} appartient à cc_i, il appartient aussi à $\mathcal{E}_f^H$ d'après l'équation (5.5) du Lemme 3. v_r étant un voisin de v_{r+1} il appartient donc à $N(\mathcal{E}_f^H)$.

En conclusion v_r appartient à $N(\mathcal{E}_f^H)$, mais pas à V_H, or nous avons supposé que $N(\mathcal{E}_f^H)$ était inclus dans V_H. Nous avons une contradiction, et donc i n'appartient pas à $\mathcal{I}$. Ainsi v_q appartient à $cc_{\mathcal{J}}$.

On peut déduire de (5.8) que si un sommet de G n'appartenant pas à H a un voisin dans H, ce voisin appartiendra à $cc_{\mathcal{J}}$. Les $cc_{\mathcal{I}}$ et $cc_{\mathcal{J}}$ sont des composantes connexes de $H - kernel(s)$, on a donc :

$$\forall v \in V_{G-H},\ N(v) \subset V_{G-H} \cup cc_{\mathcal{J}} \tag{5.9}$$

$$\forall v \in cc_{\mathcal{J}},\ N(v) \subset V_{G-H} \cup cc_{\mathcal{J}} \cup kernel(s) \tag{5.10}$$

$$\forall v \in cc_{\mathcal{I}},\ N(v) \subset cc_{\mathcal{I}} \cup kernel(s) \tag{5.11}$$

Définition d'un isomorphisme d'équivalence d'ordres g entre G et $\tau_{i,j}^s(G)$

On définit une fonction g telle que :

$$\forall v \in V, g(v) = \begin{cases} f(v) \text{ si } v \in cc_{\mathcal{I}} \\ v \text{ si } v \in cc_{\mathcal{J}} \cup V_{G-H} \\ v = f(v) \text{ si } v \in kernel(s) \end{cases}$$

On veut montrer que $g \in \text{IsomEqOrd}(G, \tau_{i,j}^s(G))$.

La fonction g est bijective

Premièrement, nous devons montrer que g est bijective.

D'après le Lemme 4, les ensembles $cc_{\mathcal{I}}$ et $f(cc_{\mathcal{I}})$ sont égaux. Sachant que $cc_{\mathcal{I}}$ et son image par f sont identiques et que f est bijective, la restriction de f à $cc_{\mathcal{I}}$ est bijective.

De plus, comme pour tout sommet v de $cc_{\mathcal{I}}$, la fonction g est égale à la fonction f, la restriction de g à $cc_{\mathcal{I}}$ est aussi bijective et nous avons :

$$g(cc_{\mathcal{I}}) = f(cc_{\mathcal{I}}) = cc_{\mathcal{I}}$$

La restriction de g à $V - cc_{\mathcal{I}}$ est égale à l'identité. Elle est donc bijective et nous avons :

$$g(V - cc_{\mathcal{I}}) = V - cc_{\mathcal{I}}$$

On peut donc en conclure que la fonction g est bijective.

La fonction g est un automorphisme de $\widehat{G}$

On montre maintenant que g est un automorphisme de $\widehat{G}$. Comme $\widehat{G}$ est égal à $\widehat{\tau^s_{i,j}(G)}$, cela équivaut à montrer que g appartient à $\text{Isom}(\widehat{G}, \widehat{\tau^s_{i,j}(G)})$.

Soit $e = (u, v)$ une arête de G :

- Si u appartient à $cc_{\mathcal{I}}$.

  g est identique à f sur $cc_{\mathcal{I}}$, donc $g(u)$ est égal à $f(u)$. Comme u appartient à $cc_{\mathcal{I}}$, selon (5.11) v appartient soit à $cc_{\mathcal{I}}$, soit à $kernel(s)$. Dans les deux cas, on a $g(v)$ est égal à $f(v)$. Comme f est un isomorphisme d'équivalence d'ordres entre H et $\tau^s_{i,j}(H)$, c'est aussi un automorphisme de $\hat{H}$. On a donc :

  $$(u, v) \in E \Rightarrow (f(u), f(v)) \in E$$

  On peut en conclure que $(g(u), g(v)) = (f(u), f(v))$ appartient à E.

- Si u appartient à $cc_{\mathcal{J}}$ ou à V_{G-H}.

  Par définition de g, $g(u)$ est égal à u. Comme u appartient à $cc_{\mathcal{J}}$ ou à V_{G-H}, selon (5.9) et (5.10) v appartient à $v \in V_{G-H}$, à $cc_{\mathcal{J}}$, ou à $kernel(s)$.

  Dans les trois cas, $g(v)$ est égal à v. Ainsi, $(g(u), g(v)) = (u, v)$ appartient à E.

- Si u appartient à $kernel(s)$.

  Par définition de g, $g(u)$ est égal à $f(u)$. Comme v est un voisin de u, v appartient à $StereoStar^*(s)$ ou à $kernel(s)$.

  - Si v appartient à $cc_{\mathcal{I}}$, alors $g(v)$ est égal à $f(v)$.
  - Si v appartient à $cc_{\mathcal{J}}$, alors $g(v)$ est égal à v. Mais dans ce cas l'équation (5.7) du Lemme 3 donne que v est égal à $f(v)$, et donc $g(v)$ est égal à $f(v)$.
  - Si v appartient à $kernel(s)$, alors $g(v)$ est égal à $f(v)$.

  Dans tous les cas, $g(v)$ est égal à $f(v)$. Comme f est un isomorphisme d'équivalence d'ordres entre H et $\tau^s_{i,j}(H)$, c'est aussi un automorphisme de $\hat{H}$. On a donc :

  $$(u, v) \in E \Rightarrow (f(u), f(v)) \in E$$

  On peut en conclure que $(g(u), g(v)) = (f(u), f(v))$ appartient à E.

Dans tous les cas on a, $(g(u), g(v))$ est égal soit à $(f(u), f(v))$ soit à (u, v). On peut donc en déduire que :

$$e = (u, v) \in E \Rightarrow e' = (g(u), g(v)) \in E \\ \text{Avec } \nu(e) = \nu(e'),\ \mu(g(u)) = \mu(u) \text{ et } \mu(g(v)) = \mu(v) \quad (5.12)$$

On définit $\tilde{g}$ tel que :

$$\tilde{g} \begin{cases} V \times V & \rightarrow & V \times V \\ (u, v) & \rightarrow & (g(u), g(v)) \end{cases}$$

Comme g est bijective, $\tilde{g}$ est bijective.

Par (5.12), $\tilde{g}(E)$ est inclus dans E. Comme $\tilde{g}$ est bijective, $\tilde{g}(E)$ est égal à E.

Ainsi, pour tout couple de sommets (u, v) de V^2, tel que $\tilde{g}(u, v) = (g(u), g(v))$ appartienne à E, (u, v) appartient aussi à E. Grâce à l'équation (5.12), on peut conclure que :

$$e = (g(u), g(v)) \in E \Rightarrow e' = (u, v) \in E \\ \text{Avec } \nu(e) = \nu(e'),\ \mu(g(u)) = \mu(u) \text{ et } \mu(g(v)) = \mu(v) \quad (5.13)$$

Par (5.12) et (5.13), g est un automorphisme de $\widehat{G}$.

Définition d'une fonction de réordonnancement σ'

Finalement, on montre qu'il existe une fonction de réordonnancement σ' appartenant à Σ^M telle que g soit un isomorphisme localement ordonné entre $\sigma'(G)$ et $\tau^s_{i,j}(G)$.

f est un isomorphisme d'équivalence d'ordres entre H et $\tau^s_{i,j}(H)$ donc selon la Définition 47, il existe une fonction de réordonnancement σ appartenant à Σ^M tel que f soit un isomorphisme d'ordres entre $\sigma(H)$ et $\tau^s_{i,j}(H)$.

On note σ' la fonction de réordonnancement telle que :

$$\forall v \in V_{ord}, \sigma'(v) = \begin{cases} \sigma(v) \text{ si } v \in cc_{\mathcal{I}} \cup kernel(s) \\ Id_n \text{ si } v \in cc_{\mathcal{J}} \cup V_{G-H} \end{cases}$$

où Id_n est la permutation identité de n éléments et $n = d_v$ est le degré de v.

Soit v un sommet appartenant à V_{ord} et à $cc_{\mathcal{I}}$ ou à $kernel(s)$. Comme σ est une fonction de réordonnancement sur H, $\sigma(v)$ est une permutation des indices

des voisins de v dans H. σ' est une fonction de réordonnancement sur G, ainsi on ne peut définir que $\sigma'(v)$ soit égale à $\sigma(v)$ que si le voisinage de v est inclus dans H.

H contient au moins $StereoStar(s)$, donc si v appartient à $kernel(s)$, H contient son voisinage.

D'après l'équation (5.5) du Lemme 3 $cc_{\mathcal{I}}$ est inclus dans $\mathcal{E}_f^H$. Ainsi :

$$N(cc_{\mathcal{I}}) \subset N(\mathcal{E}_f^H)$$

On a supposé que $N(\mathcal{E}_f^H)$ est inclus dans V_H. On a donc :

$$N(cc_{\mathcal{I}}) \subset N(\mathcal{E}_f^H) \subset V_H$$

Ainsi, si v appartient à $cc_{\mathcal{I}}$, son voisinage est inclus dans H.

La fonction de réordonnancement σ' est donc bien définie sur $cc_{\mathcal{I}}$ et $kernel(s)$.

La fonction de réordonnancement σ' est une fonction de réordonnancement valide

On montre que σ' appartient à Σ^M.

Soit v appartenant à V^{PAC}.

Si v appartient à $kernel(s)$ ou à $cc_{\mathcal{I}}$ alors $\sigma'(v)$ est égale à $\sigma(v)$. Sachant que σ appartient à Σ^M et qu'une fonction de réordonnancement de Σ^M associe à un sommet de V^{PAC} une permutation paire (Définition 52), la permutation $\sigma(v)$ est paire et donc $\sigma'(v)$ l'est également.

Si v appartient à V_{G-H} ou à $cc_{\mathcal{J}}$ alors la permutation $\sigma'(v)$ est égale à la permutation identité. L'identité est paire et donc $\sigma'(v)$ l'est également.

Ainsi, pour tout sommet v appartenant à V^{PAC}, $\sigma'(v)$ est paire.

Soit v appartenant à V^{DB}.

- Selon la Définition 55, v appartient à $kernel(s)$ si et seulement si $n_=(v)$ appartient lui aussi à $kernel(s)$.

  Donc, si v appartient à $kernel(s)$, alors la permutation $\sigma'(v)$ est égale à $\sigma(v)$ et la permutation $\sigma'(n_=(v))$ est égale à $\sigma(n_=(v))$.

  σ appartient à Σ^M donc selon la Définition 52, $\sigma(v)$ et $\sigma(n_=(v))$ ont la même parité.

  Ainsi $\sigma'(v)$ et $\sigma'(n_=(v))$ ont la même parité.

- Si v appartient à $cc_{\mathcal{I}}$ alors par (5.11) $n_{=}(v)$ appartient à $cc_{\mathcal{I}}$ ou à $kernel(s)$.

  Comme $n_{=}(v)$ appartient à $kernel(s)$ si et seulement si v appartient lui aussi à $kernel(s)$ (Définition 55) et que v n'appartient pas à $kernel(s)$ alors $n_{=}(v)$ n'appartient pas à $kernel(s)$. Ainsi $n_{=}(v)$ appartient à $cc_{\mathcal{I}}$.

  Les permutations $\sigma'(v)$ et $\sigma(v)$ sont donc égales, ainsi que les permutations $\sigma'(n_{=}(v))$ et $\sigma(n_{=}(v))$. σ appartient à Σ^M donc selon la Définition 52 $\sigma(v)$ et $\sigma(n_{=}(v))$ ont la même parité.

  Ainsi $\sigma'(v)$ et $\sigma'(n_{=}(v))$ ont la même parité.

- Finalement, si v appartient à V_{G-H} ou à $cc_{\mathcal{J}}$ alors par (5.9) et (5.10), $n_{=}(v)$ appartient à V_{G-H}, à $cc_{\mathcal{J}}$ ou à $kernel(s)$.

  Comme $n_{=}(v)$ appartient à $kernel(s)$ si et seulement si v appartient à $kernel(s)$ (Definition 55) et v appartient à V_{G-H} ou à $cc_{\mathcal{J}}$, $n_{=}(v)$ n'appartient pas à $kernel(s)$. Donc $n_{=}(v)$ appartient soit à V_{G-H}, soit à $cc_{\mathcal{J}}$. Dans les deux cas la permutation $\sigma'(n_{=}(v))$ est égale à la permutation identité.

  $\sigma'(v)$ étant aussi égale à la permutation identité, $\sigma'(v)$ et $\sigma'(n_{=}(v))$ ont une même parité.

En conclusion, σ' appartient à Σ^M d'après la Définition 52.

La fonction g est un isomorphisme d'équivalence d'ordres entre G et $\tau^s_{i,j}(G)$

Nous montrons maintenant que g est un isomorphisme localement ordonné entre $\sigma'(G)$ et $\tau^s_{i,j}(G)$.

On note ord' l'ordre de $\tau^s_{i,j}(G)$. Comme $\tau^s_{i,j}$ ne permute que l'ordre de s, pour tout sommet v appartenant à $V_{ord} - \{s\}$, les ordres $ord'(v)$ et $ord(v)$ sont identiques.

Soit v appartenant à V_{ord} avec $ord_{\sigma'}(v) = (v_1, \ldots, v_n)$:

- On suppose que v appartient à $cc_{\mathcal{J}}$ ou à V_{G-H}.

  Par définition de σ', le fait que v appartienne à $cc_{\mathcal{J}}$ ou à V_{G-H} implique que $\sigma'(v)$ est égale à la permutation identité. On a donc :

  $$ord(v) = ord_{\sigma'}(v) = (v_1, \ldots, v_n)$$

  Comme v appartient à $V_{ord} - \{s\}$ on a :

  $$ord'(v) = ord(v) = (v_1, \ldots, v_n)$$

Par définition de g, $g(v)$ est égal à v. Ainsi :

$$ord'(g(v)) = ord'(v) = (v_1, \ldots, v_n)$$

Par (5.10) et (5.11) on sait que pour tout k compris entre 1 et n, le sommet v_k appartient à V_{G-H}, à $cc_{\mathcal{J}}$ ou à $kernel(s)$. Ainsi $g(v_k)$ est égal à v_k et donc :

$$ord'(g(v)) = (g(v_1), \ldots, g(v_n))$$

- On suppose que v appartient à $cc_{\mathcal{I}}$ ou à $kernel(s)$.

  Par définition de σ', le fait que v appartienne à $cc_{\mathcal{I}}$ ou à $kernel(s)$ implique que $\sigma'(v)$ est égale à $\sigma(v)$. On a donc :

  $$ord_{\sigma}(v) = ord_{\sigma'}(v) = (v_1, \ldots, v_n)$$

  Sachant que f est un isomorphisme localement ordonné entre $\sigma(H)$ et $\tau^s_{i,j}(H)$, et que v appartient à H, on a :

  $$ord_{\sigma}(v) = (v_1, \ldots, v_n) \Rightarrow ord'(f(v)) = (f(v_1), \ldots, f(v_n))$$

  Par définition de g, $g(v)$ est égal à $f(v)$. Ainsi :

  $$ord'(g(v)) = ord'(f(v)) = (f(v_1), \ldots, f(v_n))$$

  – Si v appartient à $cc_{\mathcal{I}}$.

    Par (5.11) on sait que pour tout k compris entre 1 et n, le sommet v_k appartient à $cc_{\mathcal{I}}$ ou à $kernel(s)$. Dans les deux cas on a $f(v_k)$ est égal à $g(v_k)$ et donc :

    $$ord'(g(v)) = (g(v_1), \ldots, g(v_n))$$

  – Si v appartient à $kernel(s)$.

    Soit k un indice compris entre 1 et n.

    Alors v_k appartient à $StereoStar^*(s)$ ou à $kernel(s)$.

    * Si v_k appartient à $cc_{\mathcal{I}}$, alors $g(v_k)$ est égal à $f(v_k)$.
    * Si v_k appartient à $cc_{\mathcal{J}}$, alors $g(v_k)$ est égal à v_k. Mais dans ce cas l'équation (5.7) du Lemme 3 donne que v_k est égal à $f(v_k)$, et donc $g(v_k)$ est égal à $f(v_k)$.
    * Si v_k appartient à $kernel(s)$, alors $g(v_k)$ est égal à $f(v_k)$.

Dans tous les cas, $g(v_k)$ est égal à $f(v_k)$, ce qui implique que :

$$ord'(g(v)) = (g(v_1), \ldots, g(v_n))$$

En résumé, pour tout sommet v appartenant à V_{ord} on a :

$$ord_{\sigma'}(v) = (v_1, \ldots, v_n) \Rightarrow ord'(g(v)) = (g(v_1), \ldots, g(v_n))$$

La fonction g préserve bien l'ordre autour de chaque sommet v de $\sigma'(G)$. Ainsi, g est un isomorphisme d'ordres entre $\sigma'(G)$ et $\tau^s_{i,j}(G)$, où σ' est une fonction de réordonnancement de Σ^M.

Conclusion

On peut donc en conclure que g est un isomorphisme d'équivalence d'ordres entre G et $\tau^s_{i,j}(G)$, avec $g(s) = s$. Ceci est en contradiction avec le fait que s soit un stéréo sommet de G.

Finalement, nous avons :

$$N(\mathcal{E}^H_f) \not\subset V_H$$

□

5.3 Noyaux définis positifs

Cette section contient les preuves que les noyaux définis dans cette thèse sont définis positifs.

5.3.1 Preuve de la Proposition 10

Rappel de la proposition : *Le noyau stéréo (3.4) est défini positif.*

Démonstration. Afin de prouver que le noyau stéréo est défini positif, nous allons montrer que c'est un noyau de R-convolution.

On considère une relation R sur l'ensemble $\mathcal{OM} \times \mathbb{R} \times \mathcal{OM}$ telle que $R(S, t_S, G)$ est vraie si le stéréo sous-graphe minimal S est présent t_S fois dans la collection $\mathcal{H}(G)$ des stéréo sous-graphes minimaux de G.

On considère un noyau quelconque k sur deux réels et le noyau δ sur des graphes localement ordonnées. δ est défini tel que :

$$\delta(G, G') = \begin{cases} 1 & \text{si } G \underset{\Sigma}{\simeq} G' \\ 0 & \text{sinon} \end{cases}$$

Le noyau de R-convolution fondé sur ces deux noyaux est donc :

$$K(G,G') = \sum_{\substack{\{S,t_S\}\in R^{-1}(G)\\ \{S',t'_{S'}\}\in R^{-1}(G')}} \delta(S,S') \times k(t_S(G), t_{S'}(G'))$$

Comme $\delta(S,S')$ vaut 1 si S et S' sont isomorphes et 0 sinon ce noyau peut se réécrire :

$$K(G,G') = \sum_{\substack{\{S,t_S\}\in R^{-1}(G)\\ \{S,t'_S\}\in R^{-1}(G')}} k(t_S(G), t_S(G'))$$

Finalement, comme $R(S,t_S,G)$ est vraie si le stéréo sous-graphe minimal S est présent t_S fois dans la collection $\mathcal{H}(G)$ des stéréo sous-graphes minimaux de G, le noyau de R-convolution s'écrit :

$$K(G,G') = \sum_{S\in\mathcal{H}(G)\cap\mathcal{H}(G')} k(t_S(G), t_S(G')) = k_{stereo}(G,G')$$

Le noyau stéréo k_{stereo} est donc un noyau de R-convolution, il est donc d'après la proposition 2 (page 18) défini positif. □

5.3.2 Preuve de la Proposition 11

Rappel de la proposition : *Le noyau $k_{influence}(S,S')$ entre deux stéréo sous-graphes minimaux S et S' défini dans la définition 66 est défini positif.*

Démonstration. Afin de prouver que $k_{influence}(S,S')$ est un noyau défini positif, on commence par montrer que $\sum_{(seq(S),seq(S'))\in M_{S,S'}} \prod_{i=1}^{n} k_t(S^k_{v_i}, S'^k_{v'_i})$ est bien un noyau d'appariement. On note ce noyau $k_m(S,S')$.

Le produit $\prod_{v\in\delta_{in}(S)} k_t(S^k_v, S'^k_{f(v)})$ apparaissant dans le noyau d'influence est un produit de noyaux définis positifs et est donc défini positif (Proposition 1).

On montre maintenant que $M_{S,S'}$ est symétrique.

Soit $(seq(S), seq(S'))$ appartenant à $M_{S,S'}$. Soit f l'isomorphisme d'équivalence d'ordres entre S et S' tel que, pour tout entier i compris entre 1 et n, v'_i soit l'image de v_i par f. Comme la relation d'équivalence d'ordres sur les graphes moléculaires localement ordonnés est une relation d'équivalence, il existe un isomorphisme d'équivalence d'ordres f^{-1} entre S' et S tel que, pour tout entier i compris entre 1 et n, v_i soit l'image de v'_i par f^{-1}. $(seq(S'), seq(S))$ appartient donc à $M_{S',S}$.

Le noyau $k_m(S, S')$ est donc bien un noyau d'appariement. Il faut maintenant montrer que $M_{S,S'}$ est transitif, afin de prouver que $k_m(S, S')$ est défini positif.

Soit $(seq(S), seq(S'))$ appartenant à $M_{S,S'}$ et $(seq(S'), seq(S''))$ appartenant à $M_{S',S''}$. Soit f et g les isomorphismes d'équivalence d'ordres entre S et S' et entre S' et S'' tels que, pour tout entier i compris entre 1 et n, v'_i soit l'image de v_i par f et v''_i soit l'image de v'_i par g. Comme la relation d'équivalence d'ordres sur les graphes moléculaires localement ordonnés est une relation d'équivalence, $f \circ g$ est un isomorphisme d'équivalence d'ordres entre S et S''. De plus, pour tout entier i compris entre 1 et n, v_i est l'image de v''_i par $f \circ g$. Ainsi $(seq(S), seq(S''))$ appartient à $M_{S,S''}$ et donc le système d'appariement $M_{S,S'}$ est bien transitif.

D'après la proposition 3, le noyau $k_m(S, S')$ est donc bien défini positif.

Le noyau d'influence est le produit de $k_m(S, S')$ par $\delta(S, S')\frac{1}{\sqrt{g(S)!}}\frac{1}{\sqrt{g(S')!}}$.

$\delta(S, S')$ est défini positif. En effet, $\delta(S, S')$ correspond à un produit scalaire dans un espace vectoriel où les vecteurs sont des vecteurs de taille infinie dont chaque composante correspond à un graphe localement ordonné.

Le terme $\frac{1}{\sqrt{g(S)!}}$ est une fonction de S et le terme $\frac{1}{\sqrt{g(S')!}}$ une fonction de S', leur produit est donc un noyau défini positif.

Finalement le produit de noyaux définis positifs est défini positif (Proposition 1) donc $k_{influence}(S, S')$ est défini positif.

□

5.3.3 Preuve de la Proposition 12

Rappel de la proposition : *Le noyau (Définition 69) :*

$$k(S, S') = \sum_{\substack{\sigma \in \Sigma^S \\ \sigma' \in \Sigma^{S'}}} \prod_{i=1}^{|StereoStar^*(s)|} k_t(H_{\varphi_\sigma(i)}, H'_{\varphi_{\sigma'}(i)})$$

est défini positif.

Démonstration. On reprend les notations de la Définition 69 (page 101).

On considère la relation $R(H_1, H_2, H_3, H_4, S)$ vraie s'il existe une fonction de réordonnancement σ appartenant à Σ^S telle que pour tout i compris entre 1 et 4, H_i soit le sous-graphe de l'ensemble $sub(S)$ associé au sommet $s_{\varphi_\sigma(i)}$. Le noyau défini dans l'équation (5.3.3) est le noyau de R-convolution entre S et S' pour cette relation.

D'après la proposition 2, ce noyau est défini positif.

□

5.3.4 Preuve de la Proposition 13

Rappel de la proposition : *Le noyau (équation (4.5)) :*

$$k_{inter}(S, S') = \frac{k(S, S')}{\sqrt{k(S, S) \times k(S', S')}} + \delta(S, S') - \delta(S, \tau(S'))$$

est défini positif.

Démonstration. Le terme $\delta(S, S') - \delta(S, \tau(S'))$ est défini positif. En effet, ce terme correspond à un produit scalaire que nous allons expliciter.

On considère une fonction de plongement Ψ qui associe à chaque stéréo sous-graphe minimal S un vecteur de taille infinie $\Psi(S)$. Chaque composante de ce vecteur, notée $\Psi(S)_c$ correspond à un code c obtenu grâce à l'algorithme de dénomination [WD74] utilisé dans la section 3.5 afin d'identifier les stéréo sous-graphes minimaux. Les valeurs des composantes du vecteur sont définies de la manière suivante :

$$\Psi(S)_c = \begin{cases} \frac{1}{\sqrt{2}} & \text{si } c = c_S \\ -\frac{1}{\sqrt{2}} & \text{si } c = c_{\tau(S)} \\ 0 & \text{sinon} \end{cases} \tag{5.14}$$

où c_S est le code obtenu grâce à l'algorithme de dénomination [WD74] sur S et $c_{\tau(s)}$ celui obtenu pour le stéréo sous-graphe minimal obtenu en permutant deux voisins du stéréo sommet de S.

Soit S et S' deux stéréo sous-graphes minimaux. Leur produit scalaire dans l'espace vectoriel défini par la fonction de plongement Ψ est :

$$\begin{aligned} < \Psi(S), \Psi(S') >= \frac{1}{2} \times \Big(& \mathbf{1}(c_S = c_{S'}) - \mathbf{1}(c_S = c_{\tau(S')}) \\ & - \mathbf{1}(c_{\tau(S)} = c_{S'}) + \mathbf{1}(c_{\tau(S)} = c_{\tau(S')}) \Big) \end{aligned} \tag{5.15}$$

où $\mathbf{1}(C)$ vaut 1 si la condition C est vraie, et 0 sinon.

Par la définition des fonctions de réordonnancement des graphes localement ordonnés moléculaires (Définition 52), nous pouvons facilement voir que :

$$c_S = c_{S'} \iff c_{\tau(S)} = c_{\tau(S')} \tag{5.16}$$

et :

$$c_S = c_{\tau(S')} \iff c_{\tau(S)} = c_{S'} \tag{5.17}$$

Ainsi, d'après les équations 5.15, 5.16 et 5.17 on a :

$$< \Psi(S), \Psi(S') >= \delta(S, S') - \delta(S, \tau(S'))$$

Le terme $\delta(S, S') - \delta(S, \tau(S'))$ correspond donc bien à un produit scalaire.

La somme de deux noyaux définis positifs étant définie positive (Proposition 1), $k_{inter}(S, S')$ est défini positif.

□

Liste des publications

[GBV13a] Pierre-Anthony Grenier, Luc Brun et Didier Villemin : Chiral kernel : Taking into account stereoisomerism. *In 6èmes journées de la Société Française de Chémoinformatique (SFCi)*, 2013.

[GBV13b] Pierre-Anthony Grenier, Luc Brun et Didier Villemin : Incorporating stereo information within the graph kernel framework. Rapport technique, CNRS UMR 6072 GREYC, 2013. http ://hal.archives-ouvertes.fr/hal-00809066.

[GBV13c] Pierre-Anthony Grenier, Luc Brun et Didier Villemin : Treelet kernel incorporating chiral information. *In Graph-Based Representations in Pattern Recognition*, pages 132–141. Springer, 2013.

[GBV14a] Pierre-Anthony Grenier, Luc Brun et Didier Villemin : A graph kernel incorporating molecule's stereisomerism information. *Proceedings of 22nd International Conference on Pattern Recognition (ICPR)*, pages 631–636, 2014.

[GBV14b] Pierre-Anthony Grenier, Luc Brun et Didier Villemin : Incorporating molecule's stereisomerism within the machine learning framework. *In Structural, Syntactic, and Statistical Pattern Recognition*, pages 12–21. Springer, 2014.

[GBV14c] Pierre-Anthony Grenier, Luc Brun et Didier Villemin : Un noyau sur graphe prenant en compte la stéréoisomérie des molécules. *In Reconnaissance de Formes et Intelligence Artificielle (RFIA) 2014*, 2014.

[GBV14d] Pierre-Anthony Grenier, Luc Brun et Didier Villemin : Taking into account interaction between stereocenters in a graph kernel framework. Rapport technique, CNRS UMR 6072 GREYC, 2014. https ://hal.archives-ouvertes.fr/hal-01103318.

[GBV15] Pierre-Anthony Grenier, Luc Brun et Didier Villemin : From bags to graphs of stereo subgraphs in order to predict molecule's properties. *In Graph-Based Representations in Pattern Recognition*, pages 305–314. Springer, 2015.

[GGBV15] Benoit Gaüzère, Pierre-Anthony Grenier, Luc Brun et Didier Villemin : Treelet kernel incorporating cyclic, stereo and inter pattern information in chemoinformatics. *Pattern Recognition*, 48(2):356–367, 2015.

References

[Aro50] Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68 :337–404, 1950.

[BCR84] Christian Berg, Jens Peter Reus Christensen, and Paul Ressel. Harmonic analysis on semigroups. 1984.

[BGP+13] Vincenzo Bonnici, Rosalba Giugno, Alfredo Pulvirenti, Dennis Shasha, and Alfredo Ferro. A subgraph isomorphism algorithm and its application to biochemical data. *BMC Bioinformatics*, 14(Suppl 7) :S13, 2013.

[BGV92] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM, 1992.

[BKW41] Seymour Bernstein, Walter J Kauzmann, and Everett S Wallis. The relationship between optical rotatory power and constitution of the sterols. *The Journal of Organic Chemistry*, 6(2) :319–330, 1941.

[BUT+10] JB Brown, Takashi Urata, Takeyuki Tamura, Midori A Arai, Takeo Kawabata, and Tatsuya Akutsu. Compound analysis via graph kernels incorporating chirality. *Journal of Bioinformatics and Computational Biology*, 8(1) :63–81, 2010.

[CGMPTR07] Juan A Castillo-Garit, Yovani Marrero-Ponce, Francisco Torrens, and Richard Rotondo. Atom-based stochastic and non-stochastic 3d-chiral bilinear indices and their applications to central chirality codification. *Journal of Molecular Graphics and Modelling*, 26(1) :32–47, 2007.

[CIP66] R. S. Cahn, C. Ingold, and V. Prelog. Spezifikation der molekularen chiralität. *Angewandte Chemie*, 78(8) :413–447, 1966.

[CPB88] Richard D Cramer, David E Patterson, and Jeffrey D Bunce. Comparative molecular field analysis (comfa). 1. effect of shape on binding of steroids to carrier proteins. *Journal of the American Chemical Society*, 110(18) :5959–5967, 1988.

[CV95] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3) :273–297, 1995.

[DBK+97] Harris Drucker, Christopher J.C. Burges, Linda Kaufman, Alex J. Smola, and Vladimir Vapnik. Support vector regression machines. In *Advances in Neural Information Processing Systems*, pages 155–161, 1997.

[Gaü13] Benoit Gaüzère. *Application des méthodes à noyaux sur graphes pour la prédiction des propriétés des molécules.* PhD thesis, Université de Caen, 2013.

[GBT01] Alexander Golbraikh, Danail Bonchev, and Alexander Tropsha. Novel chirality descriptors derived from molecular topology. *Journal of Chemical Information and Computer Sciences*, 41(1) :147–158, 2001.

[GJ79] Michael R Garey and David S Johnson. Computers and intractability : a guide to the theory of np-completeness. 1979. *San Francisco, LA : Freeman*, 1979.

[GT72] Ivan Gutman and N Trinajstić. Graph theory and molecular orbitals. total φ-electron energy of alternant hydrocarbons. *Chemical Physics Letters*, 17(4) :535–538, 1972.

[Hau99] David Haussler. Convolution kernels on discrete structures. Technical report, Technical report, Department of Computer Science, University of California at Santa Cruz, 1999.

[HCZ+99] Brian Hoffman, Sung Jin Cho, Weifan Zheng, Steven Wyrick, David E Nichols, Richard B Mailman, and Alexander Tropsha. Quantitative structure-activity relationship modeling of dopamine d1 antagonists using comparative molecular field analysis, genetic algorithms-partial least-squares, and k nearest neighbor

methods. *Journal of Medicinal Chemistry*, 42(17) :3217–3226, 1999.

[JB99] Xiaoyi Jiang and Horst Bunke. Optimal quadratic-time isomorphism of ordered graphs. *Pattern Recognition*, 32(7) :1273–1283, 1999.

[JCW91] Jean Jacques, André Collet, and Samuel H. Wilen. *Enantiomers, racemates, and resolutions*. Wiley, 1991.

[Lea01] Andrew R. Leach. *Molecular modelling : principles and applications*. Pearson Education, 2001.

[Mor65] HL Morgan. The generation of a unique machine description for chemical structures - a technique developed at chemical abstracts service. *Journal of Chemical Documentation*, 5(2) :107–113, 1965.

[MV09] Pierre Mahé and Jean-Philippe Vert. Graph kernels based on tree patterns for molecules. *Machine Learning*, 75(1) :3–35, October 2009.

[Ng15] Rick Ng. *Drugs : from discovery to approval*. John Wiley & Sons, 2015.

[Pet10] Michel Petitjean. Chirality in metric spaces. *Symmetry, Culture and Science*, 21 :27–36, 2010.

[PN06] Frédéric Pennerath and Amedeo Napoli. La fouille de graphes dans les bases de données réactionnelles au service de la synthese en chimie organique. In *6èmes Journées Francophones" Extraction et gestion des connaissances"-EGC 2006*, volume 2, pages 517–528. Cépaduès-éditions, 2006.

[RG03] Jan Ramon and Thomas Gärtner. Expressivity versus efficiency of graph kernels. In *First International Workshop on Mining Graphs, Trees and Sequences*, pages 65–74, 2003.

[RS70] Ernst Ruch and Alfred Schönhofer. Theorie der chiralitätsfunktionen. *Theoretica Chimica Acta*, 19(3) :225–287, 1970.

[She12] Nino Shervashidze. *Scalable Graph Kernels*. PhD thesis, Universität Tübingen, 2012.

[SK08] Kilho Shin and Tetsuji Kuboyama. A generalization of Haussler's convolution kernel : mapping kernel. In *Proceedings of the 25th International Conference on Machine learning*, pages 944–951. ACM, 2008.

[SSVL+11] Nino Shervashidze, Pascal Schweitzer, Erik Jan Van Leeuwen, Kurt Mehlhorn, and Karsten M Borgwardt. Weisfeiler-lehman graph kernels. *The Journal of Machine Learning Research*, 12 :2539–2561, 2011.

[SZL+13] Jing-Jie Suo, Qing-You Zhang, Jing-Ya Li, Yan-Mei Zhou, and Lu Xu. The derivation of a chiral substituent code for secondary alcohols and its application to the prediction of enantioselectivity. *Journal of Molecular Graphics and Modelling*, 43 :11–20, 2013.

[TC00] Roberto Todeschini and Viviana Consonni. *Handbook of molecular descriptors*, volume 11. Wiley-vch, 2000.

[VB09] Manik Varma and Bodla Rakesh Babu. More generality in efficient multiple kernel learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1065–1072. ACM, 2009.

[WD74] W. Todd Wipke and Thomas M. Dyott. Stereochemically unique naming algorithm. *Journal of the American Chemical Society*, 96(15) :4834–4842, 1974.

[WWK08] Nikil Wale, Ian A Watson, and George Karypis. Comparison of descriptor spaces for chemical compound retrieval and classification. *Knowledge and Information Systems*, 14(3) :347–375, 2008.

[ZAdS06] Qing-You Zhang and João Aires-de Sousa. Physicochemical stereodescriptors of atomic chiral centers. *Journal of Chemical Information and Modeling*, 46(6) :2278–2287, 2006.

[ZRPJ07] Hua-Jie Zhu, Jie Ren, and Charles U Pittman Jr. Matrix model to predict specific optical rotations of acyclic chiral molecules. *Tetrahedron*, 63(10) :2292–2314, 2007.

[ZT00] Weifan Zheng and Alexander Tropsha. Novel variable selection quantitative structure-property relationship approach based

on the k-nearest-neighbor principle. *Journal of Chemical Information and Computer Sciences*, 40(1) :185–194, 2000.

Table des figures

Liste des Algorithmes

Index

V

Résumé

Dans le domaine de la prédiction de propriétés moléculaires, les noyaux sur graphes permettent de combiner la représentation naturelle des molécules par des graphes avec des méthodes classiques d'apprentissage automatique. Malheureusement, le positionnement relatif des atomes dans l'espace peut être différent pour des molécules représentées par un même graphe. Ces molécules, appelées stéréoisomères, peuvent avoir des propriétés différentes. L'objectif de cette thèse est la prise en compte des stéréoisomères dans les méthodes à noyaux pour la prédiction de propriétés moléculaires. Nous commençons par présenter les méthodes de prédiction de propriétés moléculaires, en particulier les méthodes à noyaux. Puis, après une présentation de la stéréochimie, nous présentons un état de l'art des méthodes la prenant en compte. En se basant sur cet état de l'art, nous proposons d'utiliser des graphes ordonnés afin de représenter les stéréoisomères. Nous définissons ensuite les stéréo sous-graphes minimaux qui sont des sous-graphes caractérisant localement la stéréochimie. Ces sous-graphes servent alors à définir un noyau permettant de prendre en compte la stéréochimie. Finalement nous présentons trois extensions à ce noyau. Ces extensions permettent de considérer les voisinages des stéréo sous-graphes minimaux et de comparer des stéréo sous-graphes minimaux différents.

Indexation RAMEAU : Noyaux (analyse fonctionnelle), Chimie – Informatique, Reconnaissance des formes (informatique), Stéréochimie, Apprentissage automatique

---

Title : Encoding of stereochimistry applied to cheminformatics.

Abstract

In the framework of prediction of molecule's properties, graph kernels allow to combine a natural encoding of a molecule by a graph with classical statistical tools. Unfortunately some molecules encoded by a same graph and differing only by the three dimensional orientations of their atoms in space have different properties. Such molecules are called stereoisomers. This work aims to take into account stereoisomerism into graph kernels method. In this document we first present the main methods of prediction of molecule's properties, and we focus on methods based on graph kernels. Based on this state of the art, we present stereoisomers and a state of the art of methods which take into account those molecules. Then we propose to encode stereoisomers by ordered graphs. We define minimal stereo subgraphs, which are subgraphs that locally characterizes the stereochemistry. Those subgraphs are used to define a kernel that take into account the stereochemistry. Finally we propose three extensions for this kernel. Those extensions allow to consider the neighbourhoods of minimal stereo subgraphs and to compare different minimal stereo subgraphs.

RAMEAU Index : Kernel functions, Cheminformatics, Pattern recognition systems, Stereochemistry, Machine learning

---

Discipline : Informatique et applications.

Université de Caen Basse-Normandie, ENSICAEN, CNRS
GREYC - équipe image
6 Boulevard Maréchal Juin, 14050 Caen cedex, France