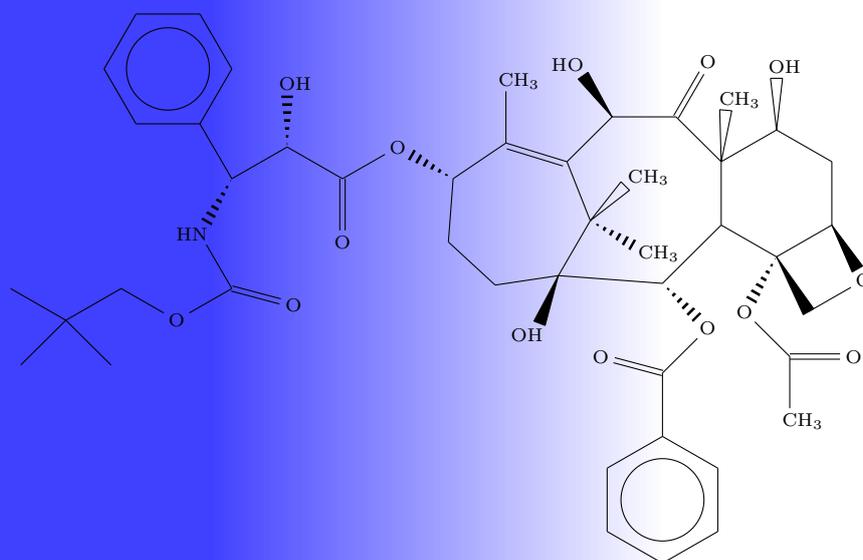


ChemFig

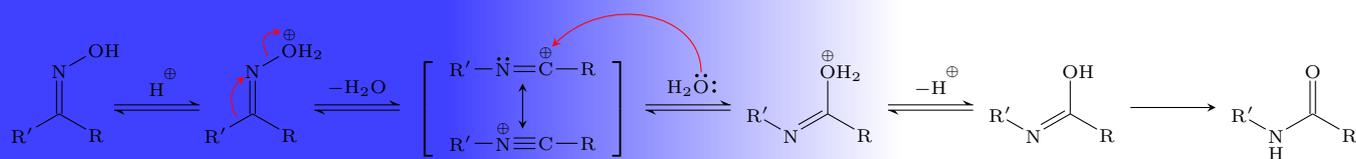
v0.3

21 novembre 2010
Christian Tellechea

Une extension L^AT_EX pour dessiner des molécules



Le taxotère



Réarrangement de Beckmann

Table des matières

I	Introduction	3
1	Nouveautés de la version 0.3	3
2	Avant propos	3
3	Présentation	4
II	ChemFig pour l' impatient	5
1	Syntaxe	5
2	Les différents types de liaisons	5
3	Les différents types de formules	6
3.1	Formule développée	6
3.2	Formule semi-développée	6
3.3	Représentation de CRAM	7
3.4	Formule topologique	7
3.5	Formule de Lewis	7
4	Molécules ramifiées	7
5	Cycles	8
6	Ions	9
7	Équations chimiques	9
III	Fonctionnement de ChemFig	10
1	Groupes d'atomes	10
2	Différents types de liaisons	10
3	Angle d'une liaison	12
3.1	Angle prédéfini	12
3.2	Angle absolu	12
3.3	Angle relatif	12
4	Longueur d'une liaison	13
5	Atome de départ et d'arrivée	14
6	Personnalisation des liaisons	14
7	Valeurs par défaut	15
8	Ramifications	15
8.1	Principe	15
8.2	Imbrication	16
8.3	Méthode	17
9	Lier des atomes éloignés	17

10 Cycles	19
10.1 Syntaxe	19
10.2 Position angulaire	20
10.2.1 Au départ	20
10.2.2 Après une liaison	20
10.3 Ramifications partant d'un cycle	21
10.4 Cycles emboîtés	22
10.5 Cycles et groupes d'atomes	23
11 Représentation des déplacements d'électrons	23
11.1 Effets mésomères	24
11.2 Mécanismes réactionnels	25
12 Écrire un nom sous une molécule	26
IV Utilisation avancée	28
1 Découpage des atomes	28
2 Affichage des atomes	28
3 Arguments optionnels	29
4 Alignement vertical	30
5 Liaisons doubles déportées	31
6 Liaisons doubles délocalisées	32
7 Sauvegarde d'une sous molécule	33
8 Décorations	34
8.1 Formule de Lewis	34
8.2 Empilement de caractères	35
8.3 Réactions chimiques	35
9 Utilisation de \chemfig dans l'environnement tikzpicture	36
10 Au delà de la chimie	36
11 Exemples commentés	38
11.1 L'éthanal	38
11.2 L'acide 2-amino-4-oxohexanoïque	39
11.2.1 Angles absolus	39
11.2.2 Angles relatifs	39
11.2.3 Cycle	40
11.2.4 Cycles imbriqués	40
11.3 Glucose	41
11.3.1 Formule topologique	41
11.3.2 Projection de Fisher	41
11.3.3 Représentation « chaise »	42
11.3.4 Projection de Haworth	43
11.4 Adrénaline	43
11.4.1 Utilisation d'un cycle	44
11.4.2 Utilisation de 2 cycles	44
11.5 Guanine	45
12 Liste des commandes	46

Première PARTIE

Introduction

1 Nouveautés de la version 0.3

Nouveau Suite à des demandes d'utilisateurs et des lacunes qui devaient de toutes façons être comblées, voici les nouveautés de la version 0.3 :

1. Amélioration de la commande `\definesubmol` qui accepte les séquences de contrôle comme alias. On peut aussi choisir un alias dont la substitution est différente selon l'orientation de la liaison qui lui arrive dessus. Voir page 33 ;
2. Le caractère “|” force la fin d'un atome. Si on écrit “D|ef” alors, `ChemFig` verra deux atomes “D” et “ef”. Lire page 28 ;
3. Le caractère “#” est reconnu lorsqu'il suit un caractère de liaison. Il doit être suivi d'un argument entre parenthèses qui contient le retrait de début et de fin qui s'appliqueront à cette liaison. Voir page 11.
4. La macro `\chemfig` admet des arguments optionnels qui seront passés à l'environnement “tikzpicture” dans lequel elle est dessinée. Lire page 29 ;
5. Mise en place de la représentation des mouvement d'électrons et des mécanismes réactionnels avec la syntaxe “@{<nom>}” devant un atome ou “@{<nom>,<coeff>}” au tout début de l'argument d'une liaison.
Cette syntaxe permet de placer un nœud (au sens de `tikz`) qui deviendra l'extrémité des flèches dessinées. Le tracé des flèches est fait par la macro `\chemmove` dont l'argument optionnel devient celui de l'environnement “tikzpicture” dans lequel sont faites les flèches. Lire page 23 ;
6. Pour le mécanisme d'alignement vertical des atomes sur une même horizontale, la commande `\chemskipalign` permet d'ignorer ce mécanisme pour le groupe d'atomes dans lequel elle est écrite. Voir page 30 ;
7. La commande `\chemname` permet d'afficher un nom sous une molécule. La commande `\chemnameinit` initialise l'alignement des noms. Lire page 26 ;
8. les dessins des décorations qu'effectue la commande `\lewis` ont été améliorés ;
9. Corrections de bugs et optimisation du code.

2 Avant propos

Cette extension a pu voir le jour grâce à l'aide de Christophe CASSEAU qui en a eu l'idée après s'être confronté à la complexité de la syntaxe de l'extension `ppchtex`.

Il m'a aidé tout au long de l'écriture du code à trouver des fonctionnalités intéressantes. Il m'a souvent encouragé à coder des fonctionnalités plus avancées alors que j'étais parfois (presque toujours ?) réticent et donc, si **ChemFig** a les fonctionnalités qu'il a, c'est en grande partie grâce à lui. Je lui adresse également mes remerciements pour les tests qu'il a effectués lors des versions beta de cette extension et pour sa contribution à l'écriture de ce manuel.

*L'expérience montre qu'il est toujours difficile d'associer le dessin des molécules à la qualité typographique d'un logiciel comme L^AT_EX, de ce fait l'utilisateur exigeant et désireux d'avoir un format vectoriel pour ses molécules n'a que peu de choix. Après avoir abandonné en raison de sa complexité le package ppctex développé pour conT_EXt et disponible sous L^AT_EX, je me suis tourné vers le monde des logiciels extérieurs à L^AT_EX. La difficulté est dans ce cas de trouver un compromis entre qualité et gratuité. Après de nombreux essais infructueux la seule alternative était qu'un nouveau package pour les molécules voit le jour et je remercie vivement Christian TELLECHEA pour la réalisation de **ChemFig**. Il a su être attentif à toutes mes exigences, **ChemFig** devait être simple d'utilisation mais posséder des fonctionnalités étendues autant dire que ce n'était pas gagné. Il a pourtant su composer un code T_EX d'une grande souplesse avec lequel je prends beaucoup de plaisir à écrire mes molécules. J'espère qu'il en sera de même pour vous lecteur à la recherche d'un package efficace dans ce domaine de la chimie.*

Christophe CASSEAU

Enfin, je tiens à chaleureusement remercier Theo HOPMAN qui m'a spontanément proposé d'effectuer la traduction de ce manuel en anglais.

3 Présentation

Pour utiliser ce package, il faut commencer par placer dans le préambule le code suivant :

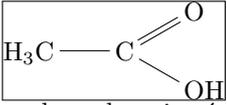
```
\usepackage{chemfig}
```

La commande principale permettant de dessiner les molécules est `\chemfig{<code>}`. L'argument `<code>` est la suite de caractères décrivant le dessin de la molécule selon les règles qui seront exposées dans ce manuel.

Tout a été fait pour qu'il soit possible de dessiner le plus grand nombre de configurations de molécules chimiques, tout en privilégiant une syntaxe simple, souple et intuitive. Malgré tout, le `<code>` qui décrit le dessin en 2D de la molécule voit sa complexité augmenter proportionnellement à celle de la molécule à dessiner.

La commande `\chemfig` dessine une molécule en se servant de commandes de l'extension `tikz`, placées à l'intérieur de l'environnement `tikzpicture`. Le choix de `tikz` implique que :

- l'utilisateur a le choix pour le moteur de compilation : pdfL^AT_EX peut indifféremment être utilisé en mode dvi (`tex` → `dvi` → `ps` → `pdf`) ou en mode pdf (`tex` → `pdf`). En effet, `tikz`, via la sous-couche `pgf`, donne des résultats graphiques identiques dans les deux modes ;
- la boîte englobante est automatiquement calculée par `tikz` et l'utilisateur n'a pas à se préoccuper d'éventuels chevauchements avec le texte. En revanche il faut faire attention à la régularité des interlignes lorsque la molécule est dessinée dans un paragraphe. À titre d'exemple, on a tracé la

boîte englobante pour cette molécule : . **ChemFig** placera toujours le premier atome de la molécule sur la ligne de base du code qui précède.

Deuxième PARTIE

ChemFig pour l' impatient

Cette partie est un condensé non exhaustif des fonctionnalités de **ChemFig**. L'objectif est d'introduire les notions de base permettant au lecteur de dessiner ses premières molécules le plus vite possible. Cette partie ne donne pas de connaissance approfondie. Une utilisation plus avancée et une approche nettement plus formelle des commandes de **ChemFig** sera abordée dans les parties suivantes.

1 Syntaxe

La commande `\chemfig` s'utilise de la manière suivante :

```
\chemfig{<atome1><type de liaison>[<angle>,<coeff>,<n1>,<n2>,<code tikz>]<atome2>}
```

- `<angle>` est l'angle de la liaison entre deux atomes ;
- `<coeff>` est le coefficient multiplicateur de la longueur de liaison par défaut ;
- `<n1>` et `<n2>` sont les numéros de l'atome de départ et de l'atome d'arrivée de la liaison ;
- `<code tikz>` est l'option supplémentaire de couleurs ou de formes pour une liaison.

Chaque liaison admet des arguments optionnels qui se mettent entre crochets. Ces arguments permettent de régler tout ce dont on a besoin pour la liaison. Chaque argument possède une valeur par défaut. On peut écrire très simplement :



Pour cet exemple comme pour tous les autres, la ligne grise représente la ligne de base.

2 Les différents types de liaisons

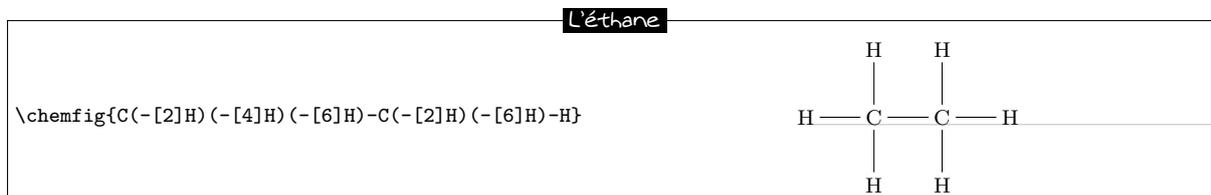
Pour **ChemFig**, les liaisons entre 2 atomes sont de 9 types, correspondant aux caractères `—`, `=`, `\equiv`, `\rhd`, `\lhd`, `\hdashline`, `\dashrightarrow`, `\dashleftarrow` et `\triangleleft` :

N° liaison	Code	Résultat	Type de liaison
1	<code>\chemfig{A-B}</code>	A — B	Simple
2	<code>\chemfig{A=B}</code>	A = B	Double
3	<code>\chemfig{A\equiv B}</code>	A ≡ B	Triple
4	<code>\chemfig{A\rhd B}</code>	A ► B	Cram pleine droite
5	<code>\chemfig{A\lhd B}</code>	A ◄ B	Cram pleine gauche
6	<code>\chemfig{A\hdashline B}</code>	A - - - B	Cram pointillée droite
7	<code>\chemfig{A\dashrightarrow B}</code>	A - - -> B	Cram pointillée gauche
8	<code>\chemfig{A\triangleleft B}</code>	A ▷ B	Cram évidée droite
9	<code>\chemfig{A\triangleleft B}</code>	A ◁ B	Cram évidée gauche

La commande `\setdoublesep{<dim>}` permet de régler l'espacement entre les traits des liaisons doubles ou triples. Cet espacement vaut 2pt par défaut.

3 Les différents types de formules

3.1 Formule développée

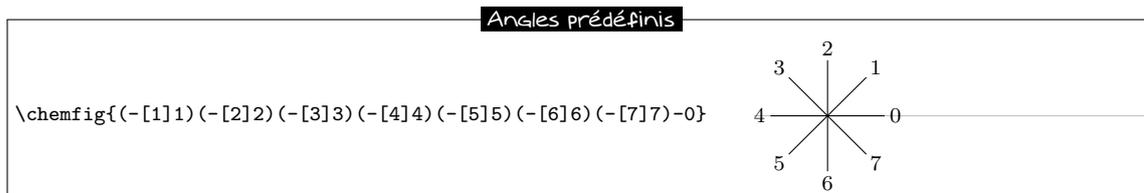


Chaque liaison admet des arguments optionnels. Le premier des arguments optionnels définit l'<angle> de la liaison. Les angles vont croissant dans le sens trigonométrique. Si l'angle n'est pas indiqué, alors sa valeur par défaut vaut 0°.

Remarque : les parenthèses permettent d'autoriser plusieurs liaisons à partir d'un même atome, voir "Molécules ramifiées", page 7.

Il y a plusieurs façons de spécifier l'angle d'une liaison.

Angles prédéfinis Lorsque l'argument optionnel contient un entier celui-ci représente l'angle que fait la liaison avec l'horizontale, en multiples de 45°.



Angle absolu Si on veut spécifier un angle en degrés avec l'horizontale, alors l'argument optionnel doit prendre cette forme : [:<angle absolu>]. L'<angle absolu> peut être positif, négatif, décimal. Il est ramené dans l'intervalle [0 ; 360].

Angle relatif Il est souvent intéressant de spécifier pour une liaison l'angle qu'elle fera relativement à la précédente. On doit alors employer cette syntaxe pour l'argument optionnel : [::<angle relatif>]. L'<angle relatif> peut être positif, négatif, décimal.

3.2 Formule semi-développée



Il peut-être utile dans certains cas de pouvoir modifier la longueur d'une liaison. Dans l'exemple précédent, il est nécessaire d'allonger les deux premières liaisons, beaucoup trop courtes par rapport à la double liaison. Pour cela il faut utiliser l'argument <coeff> des arguments optionnels :

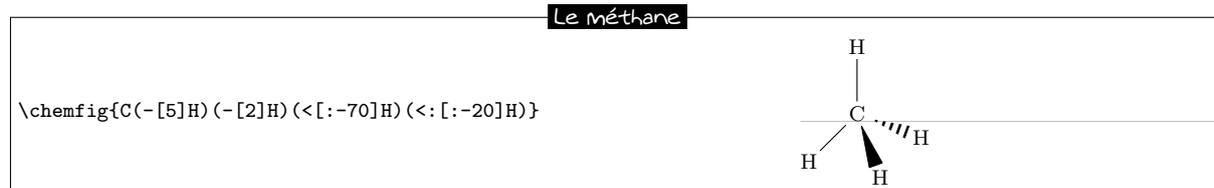


Remarques :

- dans la notation [, 1.5], la virgule sert à indiquer que la valeur placée entre crochet correspond au deuxième argument (<coeff>). Pour donner une valeur au quatrième argument on écrira [, , , 2] ;
- les caractères entre accolades ne sont pas interprétés par ChemFig, ce qui permet par exemple d'écrire des groupements d'atomes entre parenthèses sans que celles-ci ne soient considérées comme une molécule ramifiée (voir page 7).

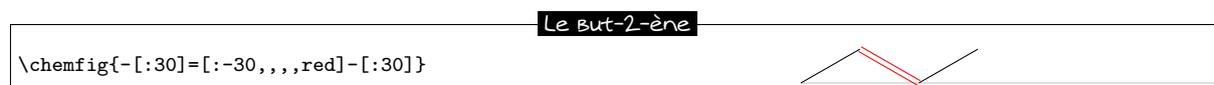
3.3 Représentation de CRAM

Une autre façon d'indiquer l'angle de la liaison entre deux atomes.

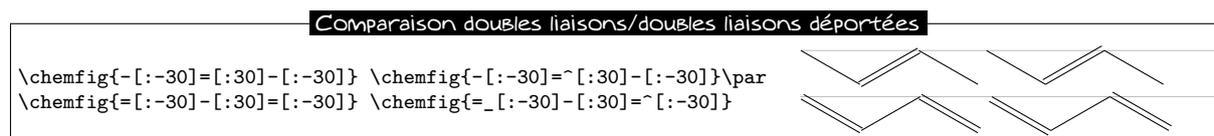


3.4 Formule topologique

Une saisie simplifiée à son maximum, seule les liaisons covalentes sont indiquées ainsi que leurs éventuels paramètres.



La liaison double est tracée de part et d'autre de la ligne de liaison simple. Pour garder une cohérence graphique des formules topologiques de certains composés, il est intéressant de déporter cette liaison double soit au dessus de cette ligne, soit au dessous. Il suffit alors de faire suivre le signe "=" de "~" ou "~_" :



3.5 Formule de Lewis

La syntaxe est la suivante :

```
\lewis{<indice de position><état électronique>,<atome>}
```

Les états électroniques pouvant être un doublet non liant, une lacune électronique ou un électron célibataire. Par défaut l'état électronique est un doublet non liant. L'électron est représenté par le caractère \cdot et la lacune par \square .



Remarque : les positions prennent des valeurs entières allant de 0 à 7.



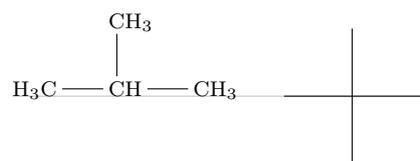
4 Molécules ramifiées

Pour indiquer une ramification, il suffit de faire suivre l'atome portant la ramification d'un <code> entre parenthèses. Ce <code> est le code de la sous molécule qui viendra s'attacher sur l'atome. Il peut bien sûr

y avoir plusieurs ramifications sur un même atome. Celles-ci peuvent aussi être imbriquées.

Alcane

```
\chemfig{H_3C-CH(-[2]CH_3)-CH_3}
\hspace{.5cm}\chemfig{-(-[2])(-[6])-}
```



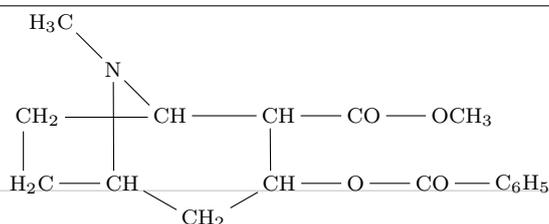
Dans ce type de représentation il est parfois nécessaire de relier deux atomes distants. Pour cela on dispose du caractère ? qui va servir de crochet entre deux atomes. La fonction

?[<nom>, <liaison>, <tikz>]

accepte trois arguments, le nom du crochet, le type de liaison et du code tikz. L'exemple suivant indique comment poser deux crochets différents.

Cocaïne

```
\chemfig{H_2C(-C?[a]H-[: -30]CH_2-[:30]C?[b]H-O-CO-C_6H_5)
-[2]CH_2-[,1.7]CH(-[3]N?[a]-[3]H_3C)(-[,1.35]C?[b]H-CO-OCH_3)}
```



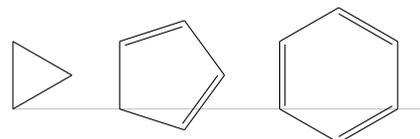
5 Cycles

ChemFig peut tracer des polygones réguliers facilement. La syntaxe est la suivante :

```
\chemfig{*n(<code de la molécule>)}
```

Quelques cycles

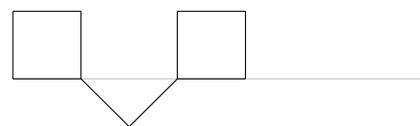
```
\chemfig{*3(---)}\hspace{.5cm}
\chemfig{*5(-----)}\hspace{.5cm}
\chemfig{*6(-----=)}
```



Les ramifications s'utilisent de la même façon

Cycles et ramifications

```
\chemfig{*4(-[--[1]*4(-----)]---)}
```



Remarque : un cycle ne pas commencer ou finir par l'atome ou le groupe d'atomes sur lequel on veut refermer le cycle. L'entité chimique sur laquelle on veut boucler doit impérativement se trouver en dehors du cycle.

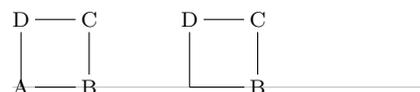
Le bon codage

```
\chemfig{A*4(-B-C-D-)}
```



Les mauvais codages

```
\chemfig{*4(A-B-C-D-)}\hspace{1cm}
\chemfig{*4(-B-C-D-A)}
```



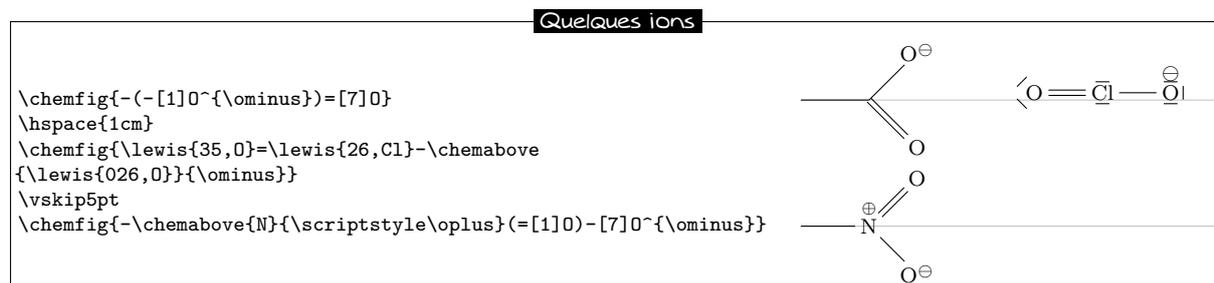
6 Ions

La commande `\chemfig` vous place directement dans l'environnement mathématique¹ de \TeX , il est donc très simple d'écrire un ion. Il faut toutefois placer le signe $-$ entre accolades pour éviter à `ChemFig` qu'il le confonde avec le symbole d'une liaison covalente.

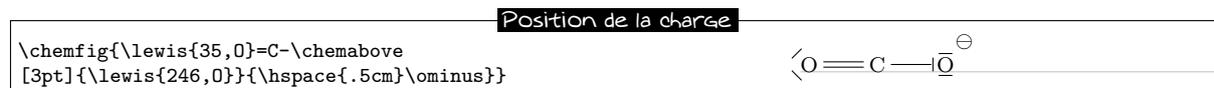


Pour les puristes il est possible d'entourer la charge de l'ion en utilisant les commandes `\ominus` et `\oplus`.

Pour répondre à toutes les exigences, il existe deux autres commandes `\chemabove` et `\chembelow` qui permettent de placer les charges au dessus et en dessous de l'atome considéré.

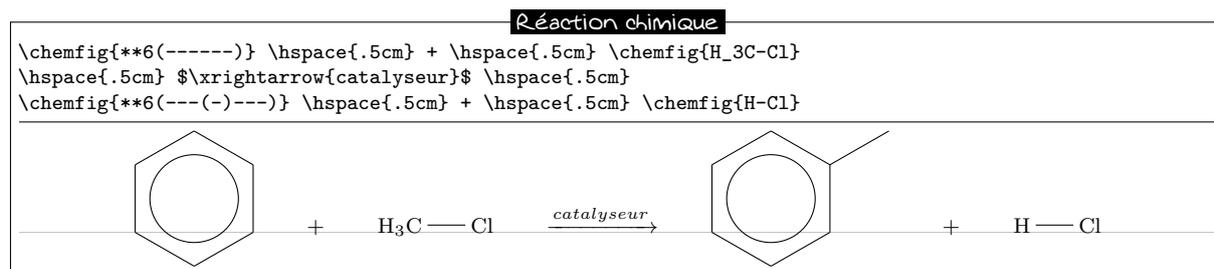


Pour ceux qui font du zèle, les commandes `\chemabove` et `\chembelow` acceptent un argument optionnel qui règle la distance entre la charge et l'atome.

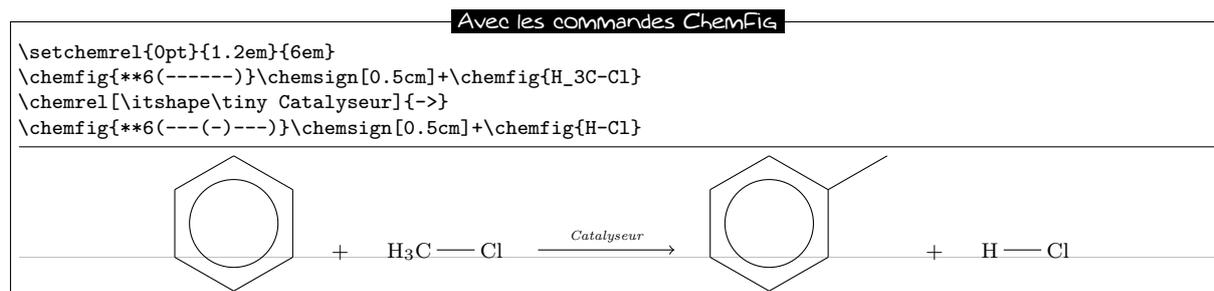


7 Équations chimiques

Voici un exemple de réaction chimique :



`ChemFig` introduit également deux nouvelles commandes `\chemsign` et `\chemrel` qui permettent de simplifier légèrement la syntaxe précédente.



1. Il existe un problème de placement des groupes d'atomes contenant des exposants ou des indices. Voir page 30.

Troisième PARTIE

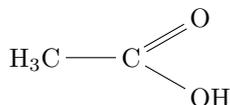
Fonctionnement de ChemFig

Cette partie est consacrée à la description des fonctionnalités les plus courantes de **ChemFig**. Le cadre de cette description dépasse largement celui de « **ChemFig** pour l' impatient » mais l'utilisateur trouvera ici les explications suffisantes pour dessiner la plupart des molécules. La présentation des fonctionnalités est faite sous un angle théorique, et le but de cette partie n'est pas de dessiner de vraies molécules chimiques mais de donner à l'utilisateur une description formelle des fonctionnalités de **ChemFig**. La partie "Utilisation avancée", page 28, sera plus pratique et dévoilera des fonctionnalités plus avancées pour les utilisations les plus pointues. On y mettra aussi en avant des méthodes pour construire de vraies molécules chimiques, page 38. Enfin, la dernière partie présentera des molécules chimiques et le code utilisé pour les dessiner.

1 Groupes d'atomes

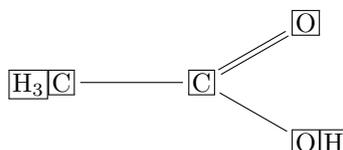
Intrinsèquement, le dessin d'une molécule chimique consiste à relier par des traits de différents types des groupes d'atomes. Ainsi, dans la molécule $O=O$, il y a 2 groupes d'atomes, chacun constitué d'un seul atome "O".

Mais dans cette molécule



on compte 4 groupes d'atomes : "H₃C", "C", "O" et "OH". Pour des raisons que nous verrons plus tard, **ChemFig** examine chaque groupe d'atomes et le découpe en atomes. Chaque atome s'étend jusqu'à rencontrer une lettre majuscule ou un de ces caractères spéciaux : \square \square \sim \square \square \square \square \square \square . Pour **ChemFig**, tous les caractères entre accolades sont ignorés pour le découpage en atomes.

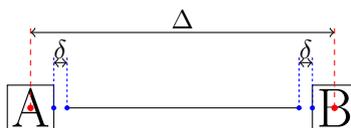
Par conséquent, le premier groupe d'atomes "H₃C" est découpé en 2 atomes : \square et \square . Chimiquement, il arrive que ce ne soient pas de vrais atomes puisque H₃ par exemple, est constitué de 3 atomes d'hydrogène. Par abus de langage, on parlera d'atomes par la suite. Par conséquent, **ChemFig** voit la molécule précédente ainsi :



2 Différents types de liaisons

Comme on l'a vu (voir page 5), les liaisons peuvent être de 9 types, chacune correspondant aux caractères \square , \square , \sim , \square , \square , \square , \square , \square et \square .

Lorsqu'une liaison est faite entre 2 atomes, il faut comprendre que ces atomes sont contenus dans des boîtes invisibles rectangulaires. Les centres des deux rectangles sont séparés par une distance réglable Δ , appelée "distance interatome". De plus, les liaisons ne relient pas exactement les frontières des rectangles : une distance δ , réglable elle aussi, sépare le bord des rectangles du début et de la fin de la liaison. Pour aider à la compréhension, les boîtes rectangulaires sont rendues visibles dans ce schéma :



La macro `\setatomsep{<dimension>}` règle cette distance interatome Δ . Si la `<dimension>` est vide, elle prend sa valeur par défaut : 3em. Cette commande, comme toutes les commandes de réglage, agit sur toutes les molécules qui vont suivre.

Distance interatome

<code>\setatomsep{2em}\chemfig{A-B}\par</code>	A — B
<code>\setatomsep{50pt}\chemfig{A-B}</code>	A ——— B

La commande `\setbondoffset{<dimension>}` permet de régler l'espacement δ entre le trait représentant la liaison et l'atome. Si la `<dimension>` est vide, alors $\delta = 2$ pt.

Retrait des liaisons

<code>\setbondoffset{0pt}\chemfig{A-B}\par</code>	A ——— B
<code>\setbondoffset{5pt}\chemfig{A-B}</code>	A — B

Si deux liaisons se suivent, alors `ChemFig` insère un groupe vide {}, et autour de ce groupe vide, l'espacement δ est nul :

Groupes vides

<code>\chemfig{A-B==C}</code>	A — B = = = C
-------------------------------	---------------

Nouveau

On peut spécifier l'espacement δ pour une seule liaison avec le caractère #. Ce caractère doit se situer *immédiatement* après le signe de liaison et dispose d'un argument obligatoire entre parenthèses de la forme "#(<dim1>,<dim2>)", où `<dim1>` représente l'espacement δ au début de la liaison et `<dim2>` celui de la fin. Si `<dim2>` est omis, l'espacement en fin de liaison prend la valeur de δ en vigueur à ce moment. On peut observer sur cet exemple comment le retrait, réglé à 4pt pour être plus visible, est rendu nul pour la liaison arrivant sur "B", puis partant de "B" et enfin pour les deux à la fois :

Réglage fin du retrait des liaisons

<code>\setbondoffset{4pt}</code>	A — B — C
<code>\chemfig{A-B-C}\par</code>	A — B — C
<code>\chemfig{A-#(,0pt)B-C}\par</code>	A — B — C
<code>\chemfig{A-B-#(0pt)C}\par</code>	A — B — C
<code>\chemfig{A-#(,0pt)B-#(0pt)C}</code>	A — B — C

Par défaut, tous les atomes se trouvant dans les groupes d'atomes sont composés en mode mathématique (les espaces sont ignorés). Ils peuvent donc contenir des caractères propres à ce mode comme la mise en indice ou en exposant ² :

Mode mathématique

<code>\chemfig{A_1B^2-C _ 3 ^ 4}</code>	A ₁ B ² — C ₃ ⁴
---	---

Il existe des réglages spécifiques aux liaisons de Cram. On utilise cette syntaxe :

$$\setcrambond{<dim1>}{<dim2>}{<dim3>}$$

Si l'un des argument est vide, il prend sa valeur par défaut. Les 3 arguments sont :

- `<dim1>` est la largeur de la base des triangles et vaut 1.5ex par défaut ;
- `<dim2>` est l'épaisseur des pointillés et vaut 1pt par défaut ;
- `<dim3>` est l'espacement entre les pointillés et vaut 2pt par défaut.

Voici un exemple où les 3 dimensions sont modifiées :

Liaison de Cram modifiée

<code>\setcrambond{10pt}{0.4pt}{1pt}</code>	A B C D
<code>\chemfig{A>B>:C> D}</code>	

2. Il existe un problème de placement des groupes d'atomes contenant des exposants ou des indices. Voir page 30.

3 Angle d'une liaison

Chaque liaison admet un argument optionnel qui se met entre crochet. Le contenu de cet argument optionnel permet de régler tout ce dont on a besoin pour la liaison. Cet argument optionnel est constitué de 5 champs séparés par des virgules qui sont autant d'arguments optionnels pour la liaison. Le premier de ces arguments définit l'angle optionnel de la liaison. Les angles vont croissant dans le sens trigonométrique et sont définis par rapport à l'horizontale. Si l'argument optionnel est vide, alors l'angle par défaut vaut 0°. Nous verrons plus loin comment modifier cet angle par défaut.

Il y a plusieurs façons de spécifier l'angle d'une liaison.

3.1 Angle prédéfini

Lorsque l'argument optionnel contient un entier celui-ci représente l'angle que fait la liaison avec l'horizontale, en multiples de 45°. Ainsi, [0] spécifie un angle de 0°, [1] de 45°, ainsi de suite jusqu'à [7] qui spécifie un angle de 315°. L'entier peut trouver en dehors de l'intervalle [0 ; 7] auquel cas l'angle est ramené dans l'intervalle [0 ; 360].

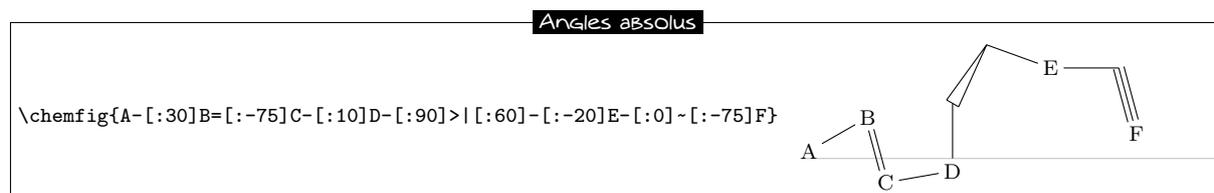


Ces angles restent valable si les atomes sont vides et il en sera de même par la suite pour toutes les fonctionnalités que nous verrons :



3.2 Angle absolu

Si on veut spécifier un angle en degrés avec l'horizontale, alors l'argument optionnel doit prendre cette forme : [: <angle absolu>]. S'il le faut, l'<angle absolu> est ramené dans l'intervalle [0 ; 360] :



3.3 Angle relatif

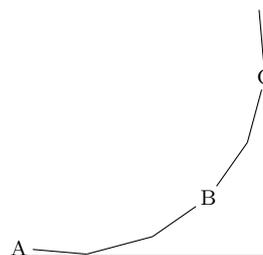
Il est souvent intéressant de spécifier pour une liaison l'angle qu'elle fera relativement à la précédente. On doit alors employer cette syntaxe pour l'argument optionnel : [: : <angle relatif>]. Le signe de l'<angle relatif> peut être omis s'il s'agit d'un +.

Voici une molécule où les angles des liaisons augmentent de 20° à chacune après la première dont l'angle

est absolu de -5° :

Suite d'angles relatifs

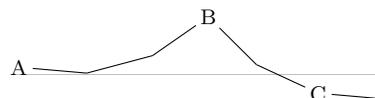
```
\chemfig{A-[: -5]-[::+20]-[::20]B-[::+20]-[::20]C-[::20]}
```



On peut “casser” un enchaînement d’angles relatifs en mettant un angle absolu ou prédéfini lorsqu’on le souhaite. Ici, c’est à l’atome “B” dont la liaison suivante un angle absolu de 315° .

Suite d'angles relatif puis angle absolu

```
\chemfig{A-[: -5]-[::20]-[::20]B-[7]-[::20]C-[::20]}
```

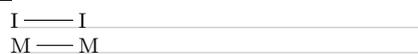


4 Longueur d’une liaison

En fait, il faudrait plutôt parler d’espace interatome que de longueur d’une liaison. En effet, seul l’espace interatome est réglable avec `\setatomsep` comme on l’a vu page 10. Une fois ce paramètre fixé, la longueur d’une liaison dépend du contenu des atomes et, dans une moindre mesure, de l’angle que fait la liaison avec l’horizontale. Il est bien évident que deux atomes peu “encombrants” auront leurs bords plus lointains que s’il avaient été encombrants. On s’en rend très bien compte sur cet exemple où les atomes “I” sont plus étroits que les atomes “M” ce qui entraîne que la liaison entre les “I” est plus longue que celle entre les “M” :

Influence de la largeur de l'atome

```
\chemfig{I-I}\par
\chemfig{M-M}
```



Cet aspect de l’encombrement des atomes se fait particulièrement sentir lorsqu’ils comportent les exposants ou des indices. Dans cet exemple, la liaison est bien plus courte, au point de se confondre avec un signe mathématique $-$:

Liaison trop courte

```
\chemfig{A^{++}_2-B^{-}_3}
```

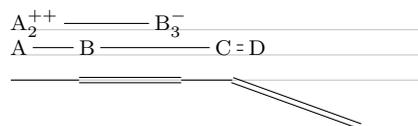


Il est important de remarquer que l’exposant $-$ est *mis entre accolades*. En effet si ce n’était pas le cas, **ChemFig** stopperait l’atome sur ce caractère qui est un caractère de liaison. L’atome serait donc “ B^- ”, ce qui conduirait à des résultats inattendus.

On voit sur l’exemple ci-dessus qu’il est donc nécessaire de pouvoir augmenter (ou parfois réduire) la distance interatome par l’intermédiaire de leur liaison. Pour cela, l’argument optionnel des liaisons est en réalité constitué de plusieurs champs séparés par des virgules. Comme on l’a vu, le premier champ spécifie l’angle. Le deuxième champ, s’il est non vide, est un coefficient qui multipliera la distance interatome Δ par défaut. Ainsi, écrire $-[,2]$ demandera à ce que cette liaison ait l’angle par défaut (1^{er} champ vide) et que les atomes qu’elle relie soit espacés d’une distance double de celle par défaut.

Longueur de liaison modifiée

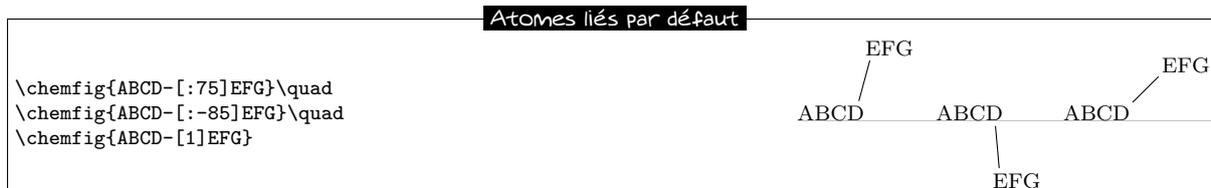
```
\chemfig{A^{++}_2-[ ,2]B^{-}_3}\par
\chemfig{A-B-[ ,2]C=[ ,0.5]D}\par
\chemfig{=-[,1.5]-[,0.75]=[:-20,2]}
```



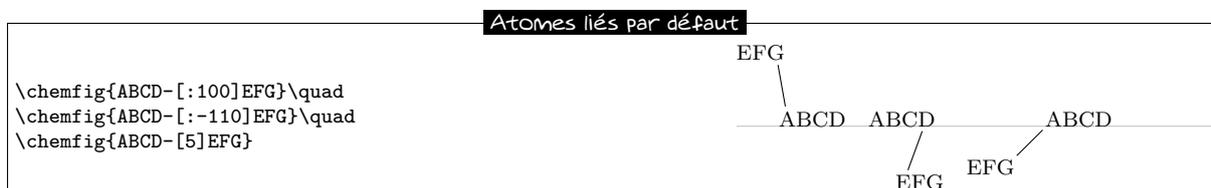
5 Atome de départ et d'arrivée

Un groupe d'atomes peut contenir plusieurs atomes. Mettons que l'on veuille relier le groupe "ABCD" au groupe "EFG" avec une liaison. **ChemFig** calcule quel atome du premier groupe et quel atome de second groupe il faut relier en fonction de l'angle que fait la liaison avec l'horizontale. Si l'angle est compris entre -90° et 90° (modulo 360°) ces valeurs étant *non comprises*, alors, la liaison se fait entre le dernier atome du premier groupe et de premier atome du second groupe. Dans tous les autres cas, la liaison se fait par défaut entre le premier atome du premier groupe et le dernier atome du second groupe.

Voici des exemples où les angles sont dans l'intervalle $] -90 ; 90[$, et où la liaison se fait donc entre D et E :



Dans les exemples suivants, les angles appartiennent à $[90 ; 270]$ et donc la liaison se fait entre A et G :



Dans certains cas, on peut désirer qu'une liaison parte d'autres atomes que ceux calculés par **ChemFig**. On peut forcer un atome de départ ou un atome d'arrivée avec l'argument optionnel de la liaison. Il faut écrire :

[,,<entier 1>,<entier 2>]

où <entier 1> et <entier 2> sont les numéros des atomes de départ et d'arrivée souhaités. Il faut que ces atomes existent, sinon, un message d'erreur sera émis.



6 Personnalisation des liaisons

Il existe un 5^e et dernier argument optionnel pour les liaisons qui se trouve à droite de la 4^e virgule :

[,,,,<code tikz>]

Ce <code tikz> est passé directement à *tikz* lorsque la liaison est tracée. On peut y mettre des attributs de couleur comme "red", de pointillés comme "dash pattern=on 2pt off 2pt", d'épaisseur comme "line width=2pt" ou même de décoration si on a chargé une librairie de décoration de *tikz*. On peut aussi rendre une liaison invisible en écrivant "draw=none". Si on veut spécifier plusieurs attributs, on utilise la syntaxe de *tikz* en les séparant par une virgule :



On peut ainsi avoir accès aux nombreuses librairies de décoration de *tikz*. On peut par exemple utiliser la librairie "pathmorphing" en écrivant `\usetikzlibrary{decorations.pathmorphing}` dans le préambule

pour tracer des liaisons ondulées :



Les liaisons de Cram sont insensibles aux attributs d'épaisseur et de pointillés.

7 Valeurs par défaut

Au début de chaque molécule, les valeurs par défaut des arguments optionnels des liaisons sont initialisées. Elles valent :

- 0° pour l'angle des liaisons ;
- 1 pour le coefficient multiplication des longueurs ;
- `<vide>` pour les numéros de départ et d'arrivée des atomes, ce qui laisse à **ChemFig** le soin de calculer ceux-ci en fonction de l'angle de la liaison ;
- `<vide>` pour le paramètre passé à *tikz*.

On peut changer ces valeurs par défaut pour toute la molécule en débutant le code de la molécule par

`[<angle>,<coeff>,<n1>,<n2>,<code tikz>]`

Ainsi, si le code de la molécule commence par `[:20,1.5]`, alors toutes les liaisons auront un angle de 20° par défaut et les distances interatomes une longueur qui vaut 1,5 fois la distance interatomes par défaut. On peut à tout moment ignorer ces valeurs par défaut en spécifiant un argument optionnel, comme pour la liaison qui suit l'atome "C" de cet exemple :



Si on écrit un improbable `[1,1.5,2,2,red,thick]`, sauf indication contraire, toutes les liaisons auront un angle de 45° avec l'horizontale, les distances interatomes vaudront 1,5 fois la distance par défaut, les liaisons partiront et arriveront sur le deuxième atome de chaque groupe et seront rouges et épaisses :



8 Ramifications

8.1 Principe

Jusqu'à présent, toutes les molécules étaient linéaires, ce qui est rare. On peut attacher une sous molécule à un atome faisant suivre cet atome d'un `<code>` entre parenthèses. Ce `<code>` est le code d'une sous molécule qui viendra d'attacher sur l'atome.

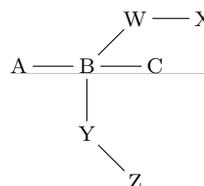
Dans cet exemple, la sous molécule "`-[1]W-X`" vient s'attacher sur l'atome "B" :



On peut avoir plusieurs sous molécules qui viennent s'attacher sur un même atome. Il suffit de le faire suivre de plusieurs parenthèses contenant le code de chaque sous molécule :

Ramifications multiples

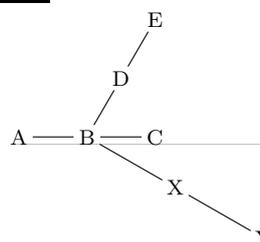
```
\chemfig{A-B(-[1]W-X)(-[6]Y-[7]Z)-C}
```



Le code de chaque sous molécule peut définir ses propres valeurs par défaut qui seront valables dans toute l'étendue de la sous molécule. Ici, on attache sur "B" une sous molécule "[:60]-D-E" dont l'angle par défaut vaut 60° absolu. On attache également sur "B" une sous molécule "[::-30,1.5]-X-Y" dont l'angle par défaut vaut 30° de moins que celui de la liaison précédente (qui était celle qui allait de "A" à "B") et dont la distance interatome est 1,5 fois celle par défaut :

Valeur par défaut dans les ramifications

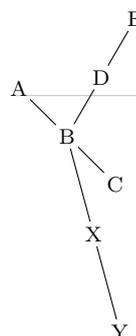
```
\chemfig{A-B([:60]-D-E)([::-30,1.5]-X-Y)-C}
```



Maintenant, que se passe t-il si au début de la molécule principale, on écrit "[:-45]" :

Influence de l'angle par défaut

```
\chemfig{[: -45]A-B([:60]-D-E)([::-30,1.5]-X-Y)-C}
```



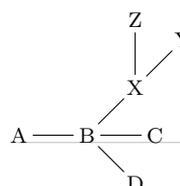
On constate que l'angle entre la liaison B-C et la liaison B-X reste de 30° puisqu'il s'agissait d'un angle relatif pour la sous molécule "-X-Y". Par contre, la branche "-D-E" reste inclinée à 60° avec l'horizontale et n'a pas suivi le mouvement de rotation imprimé par l'angle de -45° du début, ce qui est normal puisque "-D-E" a un angle absolu. Pour faire pivoter la totalité d'une molécule il est donc important que tous les angles soient relatifs.

8.2 Imbrication

Les sous molécules peuvent être imbriquées, et les règles vues au paragraphe précédent restent valables :

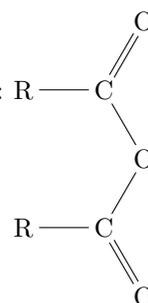
Ramifications imbriquées

```
\chemfig{A-B([1]-X([2]-Z)-Y)(-[7]D)-C}
```



8.3 Méthode

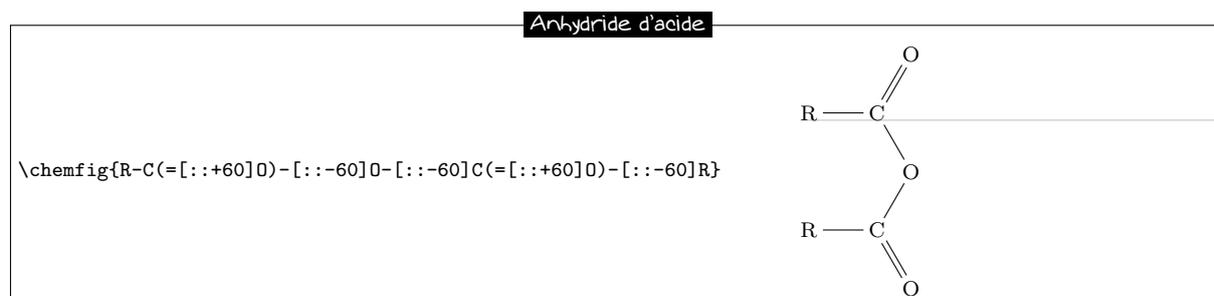
Mettons maintenant que nous voulions dessiner la molécule d'anhydride d'acide :



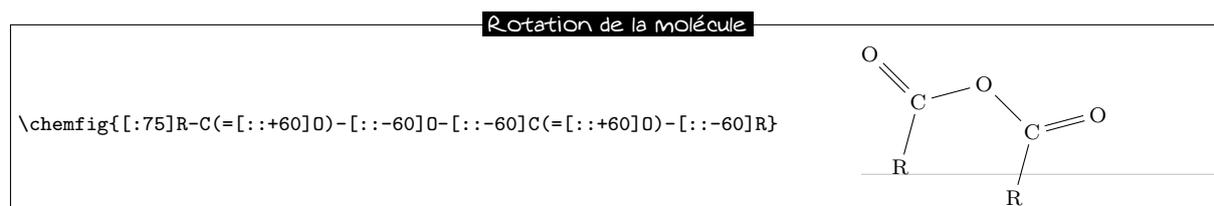
La meilleure méthode pour y arriver est de choisir le plus long chemin. Ici nous pouvons par exemple dessiner le chemin R-C-O-C-R en tenant compte des angles et en n'utilisant que les angles relatifs :



Partant de cette structure, il suffit de rajouter deux sous molécules “=O” sur les atomes de carbone :



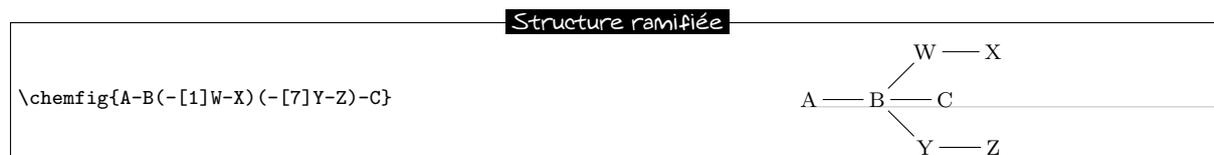
N'ayant utilisé que des angles relatifs, on peut faire pivoter cette molécule, en spécifiant un angle par défaut de 75° par exemple :



9 Lier des atomes éloignés

Nous avons vu comment relier des atomes *qui se suivent dans le code*. Il est parfois indispensable de relier entre-eux des atomes ne se suivant pas dans le code. Appelons “liaisons distantes” ces liaisons particulières.

Prenons cette molécule :

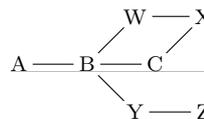


Et mettons que l'on veuille relier les atomes X et C. Dans ce cas, **ChemFig** permet de poser un “crochet” *immédiatement* après l'atome qui nous intéresse. Le caractère utilisé pour poser ce crochet est “?”, pour

sa ressemblance avec un crochet. Ainsi, si l'on écrit $X?$, alors, l'atome X portera ce crochet. Par la suite dans le code, tous les atomes suivis d'un $?$ seront reliés à X :

Liaison éloignée

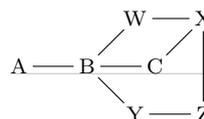
```
\chemfig{A-B(-[1]W-X?)-([7]Y-Z)-C?}
```



On aurait pu relier d'autres atomes à X en les faisant suivre de $?$. Ici, ce sont les atomes C et Z :

Plusieurs liaisons éloignées

```
\chemfig{A-B(-[1]W-X?)-([7]Y-Z?) -C?}
```



Maintenant, imaginons que nous devons laisser ces liaisons distantes $X-C$ et $X-Z$, tout en ajoutant une autre : $A-W$. Il faut donc poser deux crochets *différents*, l'un sur A et l'autre sur X . En fait, le caractère $?$ a un argument optionnel :

`?[<nom>,<liaison>,<tikz>]`

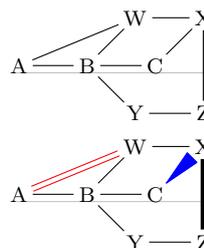
où chaque champ prend sa valeur par défaut s'il est vide :

- le `<nom>` est le nom du crochet : tous les caractères alphanumériques (a..z, A..Z, 0..9) sont admis³. Ce nom vaut "a" par défaut. Seul ce champ est pris en compte lors de la première occurrence de ce crochet portant ce nom.
- `<liaison>` spécifie avec quelle liaison l'atome portant cette occurrence du crochet doit être reliée à l'atome portant la première occurrence. Deux cas de figure sont possibles ; soit ce champ est un entier représentant le type de liaison voulu : 1=liaison simple, 2=liaison double, etc. Voir le tableau page 5 pour les codes des liaisons. Soit la `<liaison>` est constituée des caractères codant la liaison, à condition que ces caractères soient *entre accolades* ;
- `<tikz>` sera passé directement à `tikz` comme on l'a vu avec les liaisons classiques.

Voici notre molécule avec les liaisons distantes requises, puis avec les liaisons $A-W$ et $X-C$ personnalisées :

Plusieurs liaisons éloignées

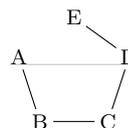
```
\chemfig{A?[a]-B(-[1]W?[a]-X?[b])(-[7]Y-Z?[b])-C?[b]}\par\medskip
\chemfig{A?[a]-B(-[1]W?[a,2,red]-X?[b])(-[7]Y-Z?[b,1,{line width=2pt}])-C?[b,{>},blue]}
```



On peut écrire plusieurs crochets différents après un atome. Mettons que dans ce pentagone incomplet, on veuille relier $A-E$, $A-C$ et $E-C$:

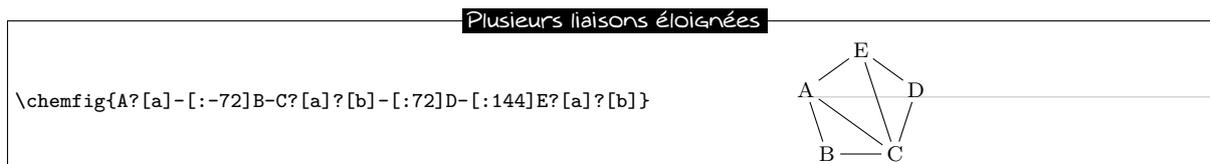
Un cycle incomplet

```
\chemfig{A-[:72]B-C-[:72]D-[:144]E}
```



3. Ce n'est pas totalement exact. En réalité, tous les caractères pouvant être mis entre `\csname... \endcsname` sont autorisés.

Alors, il faut procéder de cette façon :



10 Cycles

L'exemple précédent montre comment tracer un polygone régulier, mais la méthode est fastidieuse puisque les angles dépendent du nombre de côté du polygone.

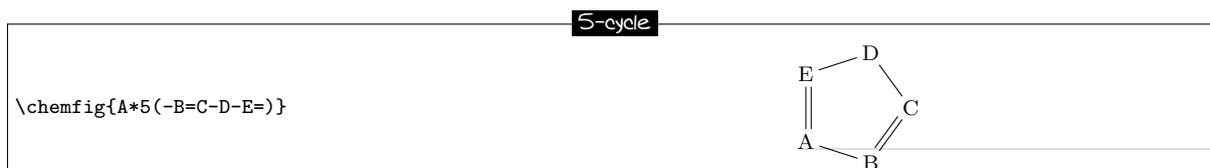
10.1 Syntaxe

ChemFig peut tracer des polygones réguliers facilement. L'idée est d'attacher un cycle à un `<atome>` extérieur à ce cycle avec cette syntaxe :

`<atome>*<n>(<code>)`

`<n>` est le nombre de côtés du polygone et le `<code>` représente les liaisons et les groupes d'atomes qui constituent ses sommets et arêtes. Ce code *doit* commencer par une liaison puisque l'atome se trouve à l'extérieur du cycle.

Voici un 5-cycle, attaché à l'atome "A" :



On peut également dessiner un cycle avec un, plusieurs, ou tous les groupes d'atomes vides, comme c'est le cas pour les dessins hors des cycles :



Un cycle peut être incomplet :



Si un cycle a un code qui contient trop de liaisons et de groupes d'atomes pour le nombre de sommets spécifiés, toutes les liaisons et les groupes au delà du maximum admissible sont ignorés :



Il est possible de dessiner un cercle ou un arc de cercle à l'intérieur d'un cycle. Pour cela, on utilise cette syntaxe :

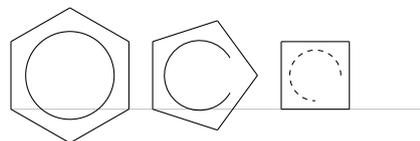
`<atome>**[<angle 1>,<angle 2>,<tikz>]<n>(<code>)`

où chaque champ de l'argument optionnel prend sa valeur par défaut s'il est vide :

- `<angle 1>` et `<angle 2>` sont les angles absolus de départ et de fin de l'arc de cercle. Ceux-ci valent 0° et 360° de telle sorte qu'un cercle complet est tracé par défaut ;
- `<tikz>` est le code qui sera passé à `tikz` pour le dessin de l'arc de cercle.

Cycles et arcs de cercle

```
\chemfig{**6(-----)}\quad
\chemfig{**[30,330]5(-----)}\quad
\chemfig{**[0,270,dash pattern=on 2pt off 2pt]4(-----)}
```



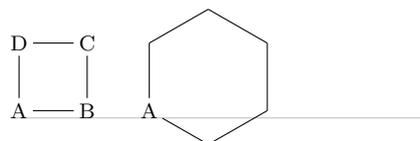
10.2 Position angulaire

10.2.1 Au départ

Comme on le voit dans les exemples ci-dessous, la règle est que l'atome d'attache "A" se trouve toujours au sud ouest du cycle. De plus, le cycle est toujours construit dans le sens trigonométrique et la dernière liaison tombe verticalement sur l'atome de rattachement :

Position angulaire des cycles

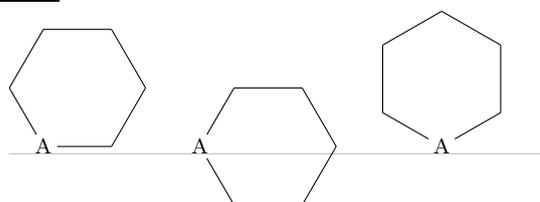
```
\chemfig{A*4(-B-C-D-)}\quad\chemfig{A*6(-----)}
```



Si cette position angulaire ne convient pas, il est possible de spécifier via l'argument optionnel du début de la molécule un autre angle par défaut. Voici un 6-cycle qui a été pivoté de $+30^\circ$, de -30° puis de $+60^\circ$:

Rotation des cycles

```
\chemfig{[:30]A*6(-----)}\quad
\chemfig{[:-30]A*6(-----)}\quad
\chemfig{[:60]A*6(-----)}
```



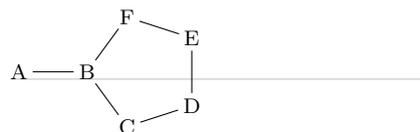
10.2.2 Après une liaison

Lorsqu'un cycle ne commence pas une molécule et qu'une (ou plusieurs) liaisons ont été auparavant tracées, la position angulaire par défaut change : le cycle est tracé de telle sorte que la liaison qui arrive sur l'atome de rattachement soit la bissectrice du premier et du dernier côté du cycle.

Voici un cas simple :

Liaison arrivant sur un cycle

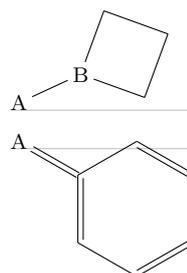
```
\chemfig{A-B*5(-C-D-E-F-)}
```



La règle reste valable, quelque soit l'angle de la liaison précédente :

Liaisons arrivant sur un cycle

```
\chemfig{A-[:25]B*4(----)}\vskip5pt
\chemfig{A=[:30]*6(=----)}
```



10.3 Ramifications partant d'un cycle

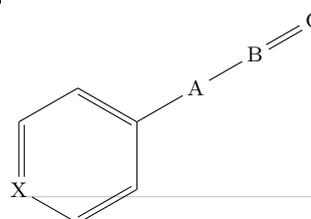
Pour faire partir une ramification d'un des sommets du cycle, on emploie la syntaxe déjà vue :

`<atome>(<code>)`

où le `<code>` est celui de la sous molécule et l'`<atome>` occupe le sommet. Une chose particulière aux cycles est que l'angle par défaut de la sous molécule n'est pas 0° mais est calculé de telle sorte qu'il soit la bissectrice des côtés partant du sommet :

Ramification partant d'un cycle

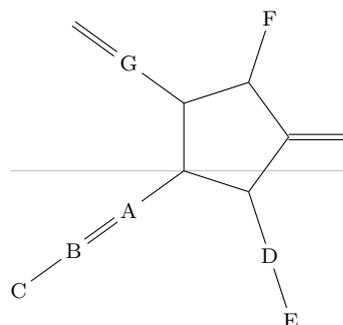
```
\chemfig{X*6(---(-A-B=C)---)}
```



On peut faire partir une sous molécule du premier sommet du cycle, ainsi que de tous les autres sommets :

Cycle et ramifications

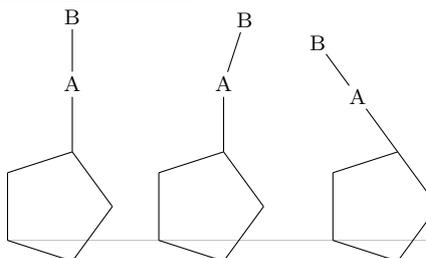
```
\chemfig{*5((-A=B-C)-(-D-E)-(=)-(-F)-(-G=)-)}
```



Si on souhaite que la liaison partant d'un sommet ne soit pas la bissectrice de ses côtés, on peut jouer sur le paramètre optionnel global ou le paramètre optionnel de la liaison :

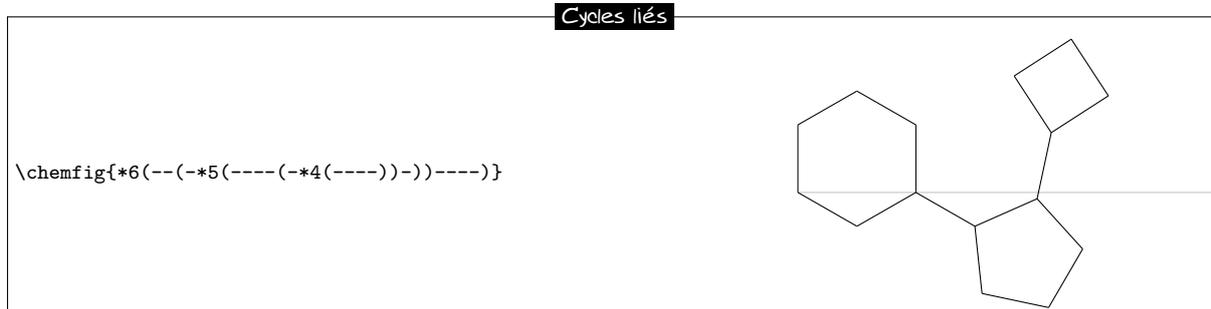
Ramifications et angles spécifiés

```
\chemfig{*5(---[:90]-A-B---)}\quad
\chemfig{*5(---(-[:90]A-B---)}\quad
\chemfig{*5(---[:+0]-A-B---)}
```



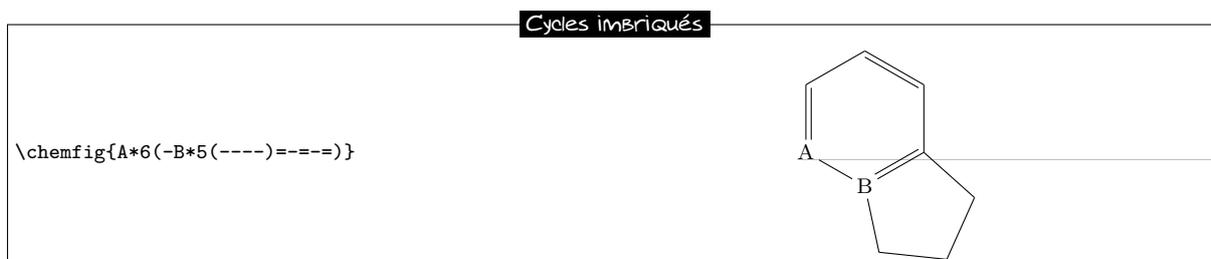
Il est intéressant de constater au troisième exemple que si on spécifie un angle relatif de 0° , la liaison se fait dans le prolongement de la liaison qui précédait dans le cycle. C'est la règle de la page 12 qui spécifiait que l'angle de référence était celui de la dernière liaison tracée.

On peut désormais lier des cycles entre eux par des liaisons :



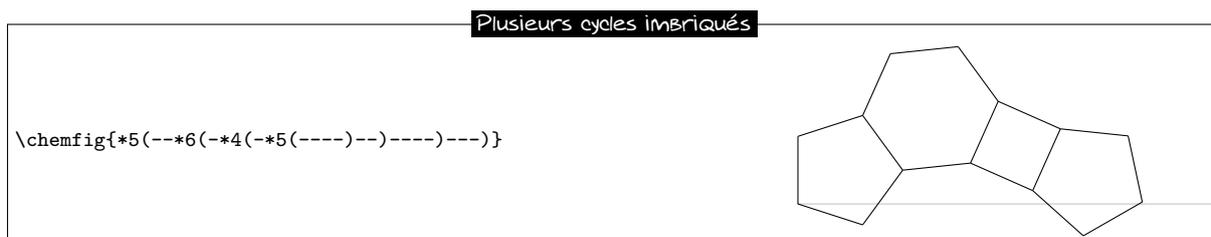
10.4 Cycles emboîtés

Pour “coller” 2 cycles entre eux, la syntaxe est légèrement différente : on identifie le sommet d'où va commencer l'autre cycle. Il suffit de faire suivre ce sommet par la syntaxe habituelle d'un cycle. Voici par exemple un 5-cycle qui part du deuxième sommet d'un 6-cycle :



On remarque que le cycle qui vient se greffer sur le cycle principal a une position angulaire telle que deux de leurs côtés coïncident. De plus, le 5-cycle n'a que 4 liaisons “----”. En effet, la 5^e serait inutile puisqu'il s'agit du deuxième côté du 6-cycle qui est déjà tracé.

Il est bien entendu possible de coller plusieurs cycles entre eux :



Il y a un cas où on doit employer une ruse. On voit sur cet exemple que le quatrième côté du deuxième 5-cycle vient boucler sur le centre de l'atome “E”.



Ceci est normal puisque le deuxième 5-cycle (qui part de l'atome “D”) est dessiné *avant* que ChemFig n'ait pris connaissance de l'atome “E”. Dans ce cas, il faut se servir de deux crochets pour tracer la liaison

Z-E :

On aurait aussi pu utiliser un `` au dernier sommet du 5-cycle :

10.5 Cycles et groupes d'atomes

Il faut prendre des précautions avec les cycles lorsque un ou plusieurs sommets sont constitués de plusieurs atomes :



Pour que le cycle ait une forme régulière, il faut passer outre le mécanisme de **ChemFig** qui calcule automatiquement les atomes de départ et d'arrivée des liaisons. Ici, il faut relier C-F et F-G en le spécifiant avec l'argument optionnel de ces liaisons :



11 Représentation des déplacements d'électrons

Nouveau

Depuis la version 0.3 de **ChemFig**, on peut représenter les déplacements d'électrons des effets mésomères ou des mécanismes réactionnels. Cela est rendu possible en marquant le point de départ et le point d'arrivée de la flèche que l'on utilise pour indiquer la migration des électrons à l'aide de la syntaxe “`@{<argument>}`”. Cette syntaxe permet de poser un nœud (au sens de *tikz*) en rendant ce nœud accessible en dehors de l'argument de la commande `\chemfig` grâce à l'option “**remember picture**” qui est passée à tous les environnements “*tikzpicture*”. Cela suppose que le visualiseur prenne en charge le “*picture remembering*” et que la compilation soit lancée deux fois.

Deux cas de figure peuvent se présenter, on peut poser :

- un nœud de dimension nulle sur une liaison en utilisant la syntaxe “`@{<nom>,<coeff>}`” placée immédiatement au début de l'argument optionnel de la liaison concernée sans être suivi d'une virgule s'il y a un premier argument optionnel. Dans ce cas, le nœud portera le nom “`<nom>`” et le `<coeff>`, compris entre 0 et 1, déterminera où le nœud se situera sur la liaison. Si on utilise “`@{<nom>}`”, le `<coeff>` prend la valeur de 0.5 par défaut ce qui signifie que le nœud est posé au milieu de la liaison ;
- un nœud sur un atome en utilisant la syntaxe “`@{<nom>}`” immédiatement avant l'atome concerné. Dans ce cas, le nœud a exactement le même encombrement que l'atome, il peut notamment être vide et donc avoir des dimensions nulles.

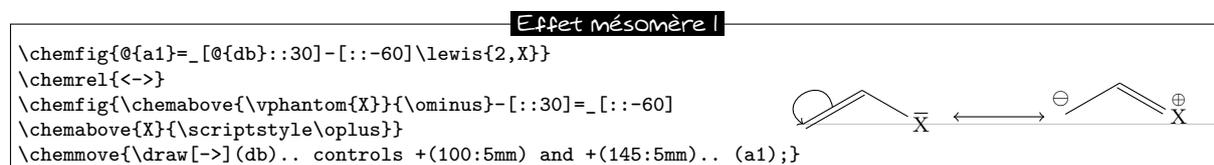
Une fois que la commande `\chemfig` a dessiné la (ou les) molécule(s) et a posé les nœuds avec la syntaxe décrite ci dessus, on peut relier ces nœuds entre eux avec les instructions de `tikz`. Ces instructions seront placées dans l'argument de la commande `\chemmove`⁴ et auront la syntaxe suivante si par exemple, on doit relier un nœud portant le nom “<nom1>” au nœud portant le nom “<nom2>” :

```
\chemmove[<opt>]{\draw[<opt tikz>](<nom1><link tikz>(<nom2>);}
```

L'argument optionnel `<opt>` de la commande `\chemmove` sera ajouté à l'argument de l'environnement `tikzpicture` dans lequel seront tracées les liaisons entre les nœuds. Les instructions `<opt tikz>` et `<link tikz>` sont exhaustivement décrites dans la documentation de l'extension `tikz`.

11.1 Effets mésomères

Pour fixer les idées prenons l'exemple d'un effet mésomère impliquant une double liaison et un doublet non liant conjugués. Commençons par la possible délocalisation des électrons de la double liaison. On va poser un nœud nommé “db” (double bond) au milieu de la double liaison et un nœud nommé “a1” sur l'extrémité de la double liaison.

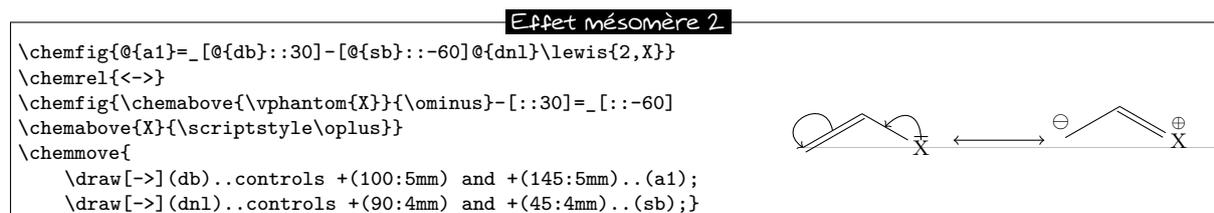


Comme on l'a dit, on remarque qu'il n'y pas de virgule derrière le nœud posé dans les arguments optionnels d'une liaison. On écrit “=_@{db}:::30]” et non pas “=_@{db},:::30]” comme on serait tenté de le faire.

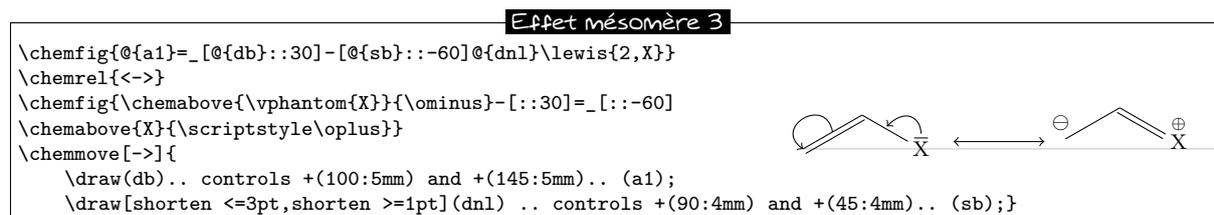
Pour relier les nœuds “db” et “a1”, nous avons utilisé la syntaxe suivante :

```
\chemmove{\draw[->](db)..controls +(100:5mm) and +(145:5mm)..(a1);}
```

Dans cet exemple nous demandons une flèche (`[->]`) et nous utilisons 2 points de contrôle⁵. Ceux-ci seront situés en coordonnées polaires à 100° et 5 mm de “db” pour le premier et à 145° et 5 mm de “a1” pour le second. Il ne faut pas être effrayé par cette syntaxe qui peut paraître compliquée en première lecture car son utilisation se réduit dans la plupart des cas à un simple copier-coller dans lequel on modifie juste le nom des nœuds et les coordonnées polaires des points de contrôle. Ce que nous allons vérifier tout de suite avec l'ajout d'une flèche partant du doublet non liant (nœud “dnl”) vers la liaison simple (nœud “sb”).



Pour notre nouvelle flèche nous avons fixé les points de contrôle comme suit : un angle de 90° à 4 mm de “dnl” et un angle de 45° à 4 mm de “sb”. Mais nous ne sommes pas entièrement satisfaits car nous aimerions que la flèche ne touche pas le trait représentant le doublet non liant. Pour cela nous allons ajouter quelques options à notre flèche.

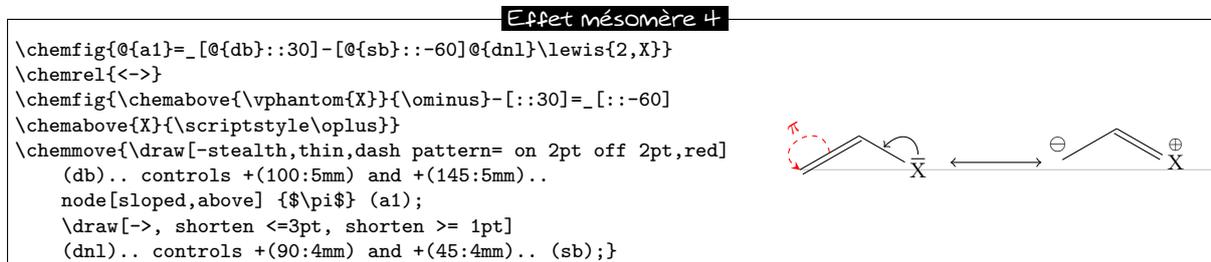


4. En réalité, la commande `\chemmove` met son argument dans un environnement “tikzpicture” avec les options “remember picture, overlay”.

5. Pour connaître toutes les façons de relier deux nœuds avec `tikz`, lire la documentation de ce package.

L'option “shorten <=3pt” indique que le point de départ de la flèche doit être raccourci de 3 pt de même “shorten >=2pt” indique que le point d'arrivée doit être raccourci de 2 pt.

On peut utiliser toute la puissance des instructions de *tikz* pour modifier le dessin de la flèche. Ici, nous changeons l'extrémité de la flèche partant de la double liaison, non plus écrite “->” mais “-stealth”, nous la dessinons en pointillés d'épaisseur fine et rouge. Nous rajoutons également la lettre π au dessus de la flèche en son milieu :



Dans l'exemple suivant nous allons voir comment indiquer la position de l'ancre de départ ou d'arrivée. Si nous écrivons



Nous constatons que l'ancre de départ de notre flèche ne pointe pas correctement sur nos électrons. La flèche part du milieu du bord supérieur du nœud. En effet, nous avons choisi un angle de départ de 90° et *tikz* fait donc partir la flèche de l'ancre “x1.90” qui correspond à l'intersection de la demi droite partant du centre du nœud “x1” et faisant un angle avec l'horizontale de 90° avec le bord du nœud qui est un rectangle. Pour obtenir le départ de la flèche d'où nous voulons, nous devons spécifier sa position. Après quelques tâtonnements, c'est “x1.57” :



Dans certains cas il sera plus facile d'utiliser les coordonnées cartésiennes pour les points de contrôle. Ici, nous n'utilisons qu'un seul point de contrôle situé à 1 cm horizontalement de “x1” et 1,5 cm verticalement :



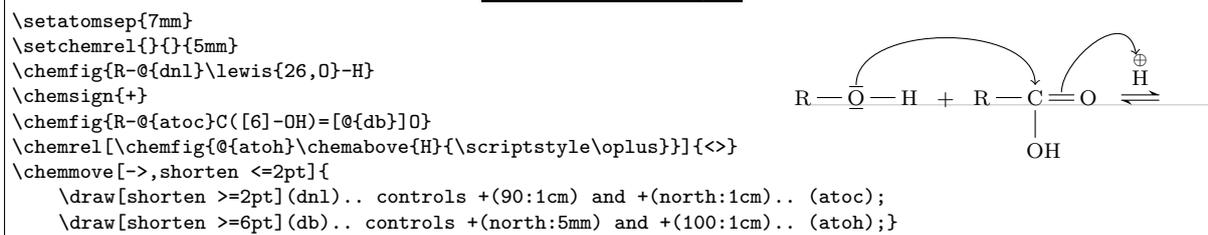
Dans ce cas nous plaçons un point de contrôle à 1 cm horizontalement et 2 cm verticalement de “x1”. Tous les objets graphiques dessinés par l'intermédiaire de la commande `\chemmove` sont faits en surimpression et ne seront pas comptés dans les boîtes englobantes. On peut le constater sur l'exemple précédent.

11.2 Mécanismes réactionnels

Grâce à l'option `remember picture` qui est passée à tous les environnements “tikzpicture” nous pouvons facilement dessiner les flèches indiquant les mécanismes réactionnels. Prenons comme exemple la première

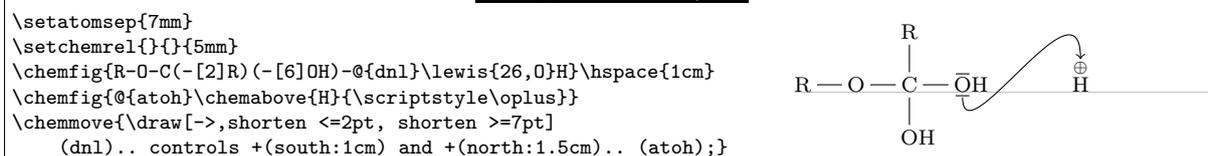
étape de la réaction d'estérification.

Estérification : étape 1



L'utilisation de la commande `\chemabove{<code>}{<materiel>}` ne change pas les dimensions de la boîte englobante du `<code>`. Pour cette raison on peut rencontrer certaines difficultés pour pointer sur le matériel indiquant la charge portée (\oplus ou \ominus). Dans l'exemple ci-dessus la solution est de créer un point de contrôle avec un angle de 110° à 1 cm de "atoh" et de raccourcir la flèche de 6pt. Dans l'exemple suivant, seconde étape de la réaction d'estérification, on peut voir que la flèche peut prendre des formes plus compliquées sans forcément surcharger le code.

Estérification : étape 2



Nous laissons le soin au lecteur d'écrire la suite...

12 Écrire un nom sous une molécule

Nouveau

Pour plus de commodité, **ChemFig** permet d'écrire le nom d'une molécule sous celle-ci avec la commande

```
\chemname[<dim>]{\chemfig{<code de la molécule>}}{<nom>}
```

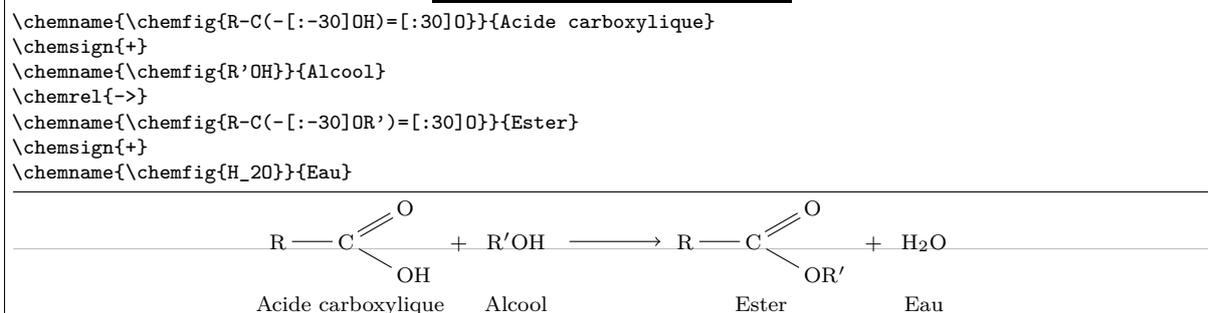
La `<dimension>`, qui vaut 1.5ex par défaut, sera insérée verticalement entre la ligne de base de la molécule et le haut des lettres du `<nom>`. Le `<nom>` sera centré par rapport à la molécule mais ce `<nom>` ne peut pas contenir plusieurs paragraphes. Comme on le voit sur cet exemple : H — O — H, le `<nom>` qui est

La molécule d'eau : H₂O

affiché sous la molécule est pris en compte dans la boîte englobante mais uniquement en ce qui concerne sa dimension *verticale*. L'encombrement horizontal du `<nom>` est toujours nul.

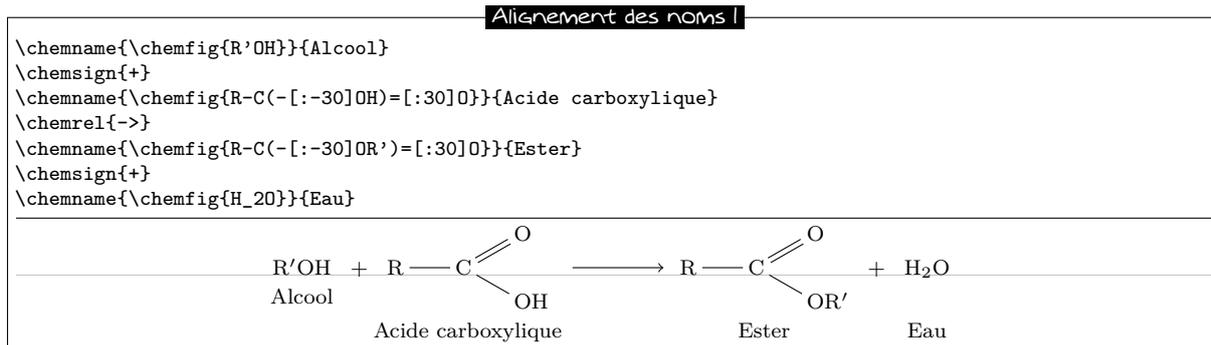
Voici une réaction avec les noms sous les molécules :

Affichage du nom des molécules



Il y a quelques contraintes avec cette commande. Pour en prendre conscience, mettons que l'on ait inversé

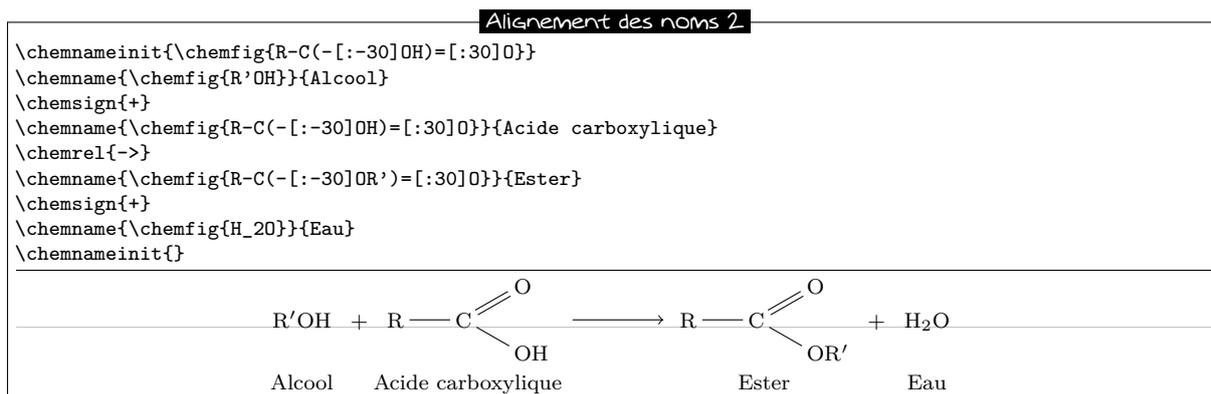
l'acide et l'alcool dans le membre de gauche :



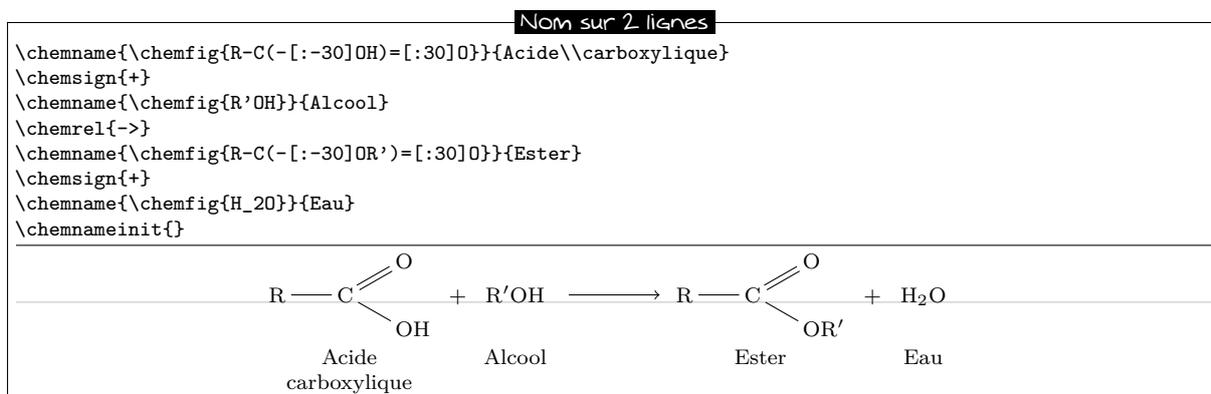
En fait, au dessous de la ligne de base de chaque molécule (en gris clair sur les exemples de ce manuel), pour écrire le <nom>, la commande `\chemname` insère 1.5ex + la plus grande des profondeurs⁶ des molécules rencontrées. La macro `\chemnameinit{<argument>}` initialise cette plus grande profondeur avec le contenu de l'<argument>. Il convient donc :

- d'écrire `\chemnameinit{<plus profonde molécule>}` avant d'impliquer cette commande dans une réaction, sauf si cette réaction commence par la plus profonde molécule ;
- d'écrire `\chemnameinit{}` après avoir écrit tous les noms d'une réaction chimique afin d'éviter que la plus grande profondeur trouvée dans cette réaction n'interfère dans une réaction future.

Le code correct est donc d'utiliser `\chemnameinit` avant et après la réaction :



Enfin, pour écrire un nom sur plusieurs lignes, la commande `\` rencontrée dans un <nom> effectue un retour à la ligne⁷ :



6. En langage TeX, la profondeur est la dimension qui s'étend verticalement sous la ligne de base.

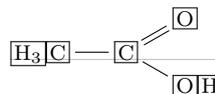
7. Par contre, la commande `\par` est interdite et provoquera une erreur à la compilation.


```
\newcommand*\printatom[1]{\ensuremath{\mathrm{#1}}}
```

On peut modifier le code de cette macro pour personnaliser l’affichage de atomes. Dans l’exemple ci-dessous, on redéfinit `\printatom` de façon à ce que chaque atome soit contenu dans une boîte rectangulaire :

Redéfinition de `\printatom`

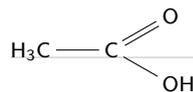
```
\fboxsep=1pt
\renewcommand*\printatom[1]{\fbox{\ensuremath{\mathrm{#1}}}}
\chemfig{H_3C-C(=[:30]O)(-[:-30]OH)}
```



Voici comment la redéfinir pour utiliser de la police “sf” du mode math :

Atomes affichés en police “sf”

```
\renewcommand*\printatom[1]{\ensuremath{\mathsf{#1}}}}
\chemfig{H_3C-C(=[:30]O)(-[:-30]OH)}
```



3 Arguments optionnels

Nouveau La commande `\chemfig` admet deux arguments optionnels et sa syntaxe complète est la suivante :

```
\chemfig[<opt1>][<opt2>]{<code molécule>}
```

Le premier argument optionnel `<opt1>` contient des instructions `tikz` qui seront passées à l’environnement `tikzpicture` dans lequel est dessinée la molécule. Le deuxième argument optionnel `<opt2>` contient des instructions `tikz` qui seront exécutées lors du dessin de chaque nœud⁸.

Par l’intermédiaire du premier argument optionnel, on peut choisir par exemple la couleur générale ou l’épaisseur des lignes :

Choix du style

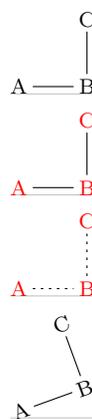
```
\chemfig{A-B-[2]C}\par\medskip
\chemfig[line width=1.5pt]{A-B-[2]C}\par\medskip
\chemfig[red]{A-B-[2]C}
```



Avec le second argument optionnel, on peut choisir la couleur des nœud dessinés par `tikz`, et même modifier l’inclinaison du dessin :

Choix des styles

```
\chemfig{A-B-[2]C}\par\medskip
\chemfig[red]{A-B-[2]C}\par\medskip
\chemfig[dash pattern=on 1pt off 2pt][red]{A-B-[2]C}\par\medskip
\chemfig[rotate=20]{A-B-[2]C}
```

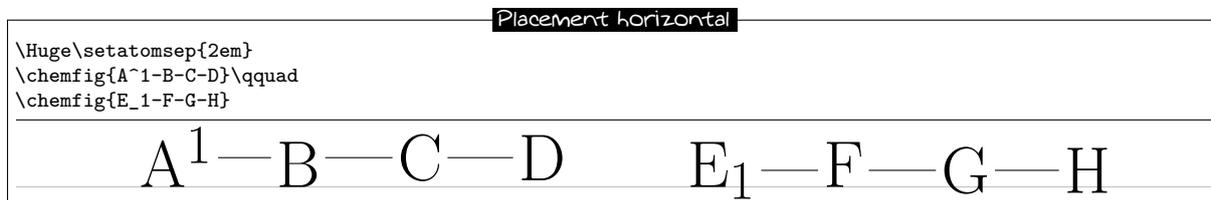


8. Ces instructions sont ajoutées à la fin de l’argument de `every node/.style{<argument>}`. Cet argument contient par défaut les instructions suivantes : “`anchor=base,inner sep=0pt,outer sep=0pt,minimum size=0pt`”.

4 Alignement vertical

Dans certains cas de molécules semi développées dont les liaisons sont horizontales, le placement des groupes d'atomes est incorrect.

Un examen attentif de l'exemple suivant révèle que les groupes d'atomes ne sont pas correctement alignés sur la ligne de base :

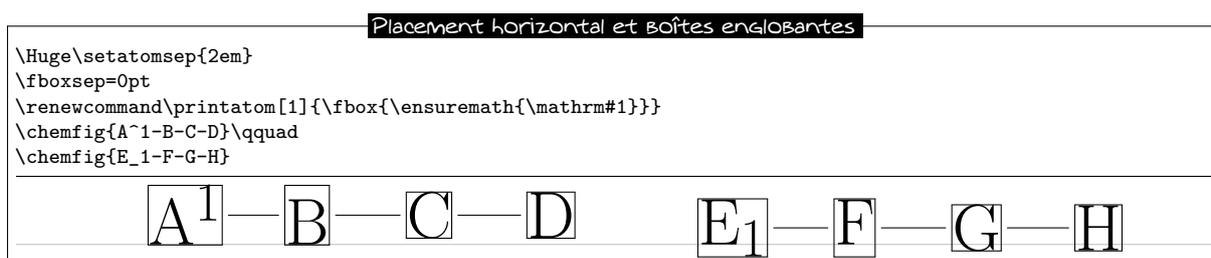


Il est étonnant que le deuxième atome soit correctement aligné alors que les deux derniers subissent un décalage vertical qui semblent être la conséquence de la hauteur différente de la boîte englobant le premier atome “A¹” et “E₁”.

Pour comprendre ce phénomène, il faut examiner comment **ChemFig** place les groupes d'atomes les uns par rapport aux autres. Pour cela, limitons nous aux liaisons horizontales ce qui permet d'employer un vocabulaire particulier, mais l'algorithme est le même pour les autres inclinaisons. En fait, une liaison horizontale part du milieu du côté droit de la boîte englobant l'atome d'où part cette liaison. L'atome d'arrivée est positionné de telle sorte que le milieu du côté gauche de sa boîte englobante soit à l'extrémité de la liaison. Il en découle que le placement vertical de l'atome d'arrivée dépend de l'encombrement vertical de l'atome de départ. Pour limiter ce phénomène, **ChemFig** ajoute devant chaque atome d'arrivée le `\vphantom` de l'atome de départ, mais sans l'inclure dans le contenu de cet atome d'arrivée : ce `\vphantom` n'est donc pas destiné à se répercuter sur les atomes suivants. Les atomes qui restent dans chaque groupe sont alignés de telle sorte que leur ligne de base coïncide avec la ligne de base de l'atome précédent.

On peut donc mieux expliquer l'alignement défectueux constaté. Les atomes “B” et “F” sont correctement alignés car ils tiennent compte de la hauteur des atomes qui les précèdent grâce à leur `\vphantom`. Pour les atomes “C” et “G”, la hauteur des atomes précédents est prise en compte, mais celle des atomes “A¹” et “E₁” est oubliée ! Il en résulte que, dépendant de l'altitude de ces liaisons, ces atomes sont un peu trop haut ou trop bas.

On peut mettre en évidence ce phénomène en rendant visible les boîtes englobantes des atomes où l'on voit clairement que les atomes “B” et “F” ont des boîtes englobantes dont la hauteur qui tient compte des hauteurs des atomes précédents :



Aucune solution automatique n'étant satisfaisante, on peut contourner manuellement ce problème en plaçant à l'intérieur du troisième atome un `\vphantom` ayant la même hauteur que le premier pour que cette hauteur se répercuter sur les suivants :



Nouveau

Il est possible pour n'importe quel groupe d'atomes de désactiver momentanément le mécanisme d'ajustement d'alignement et neutraliser le `\vphantom`. Il suffit pour cela de placer dans le groupe d'atomes

la commande `\chemskipalign` : l’alignement reprendra au groupe d’atomes suivant comme si le groupe d’atome contenant `\chemskipalign` n’avait pas existé. On peut se rendre compte sur l’exemple suivant de l’effet de cette instruction qui a pour effet de placer le point de référence de la boîte contenant le premier atome au niveau de la liaison qui arrive de gauche. Les boîtes englobant les atomes ont été dessinées à la seconde ligne :

Désactivation du mécanisme d'alignement	
<pre>\large \chemfig{A-.B}\quad \chemfig{A-\chemskipalign.-B}\par\bigskip \fbboxsep=0pt \renewcommand\printatom[1]{\fbox{\ensuremath{\mathrm{#1}}}} \chemfig{A-.B}\quad \chemfig{A-\chemskipalign.-B}</pre>	

Cette commande est à utiliser avec précaution car l’alignement des atomes dans le groupe d’atomes à venir peut être perturbé. En règle générale, tout se passera bien si le groupe d’atomes dans lequel figure `\chemskipalign` contient *un seul atome* dont la hauteur et la profondeur sont *inférieures* à celles de l’atome qui précède et qui suit, et si les atomes qui précèdent et suivent ont leur profondeur et hauteur égales. Voici par exemple la mésaventure qui arrive lorsque le groupe d’atomes contient 2 atomes, ici “`\chemskipalign.`” et “`B`” :

Conséquence de la commande \chemskipalign	
<pre>\large \fbboxsep=0pt \renewcommand\printatom[1]{\fbox{\ensuremath{\mathrm{#1}}}} \chemfig{A-\chemskipalign.B-C}</pre>	

Cette fonctionnalité peut parfois s’avérer utile. Mettons que l’on veuille dessiner la molécule



On peut définir les commandes qui vont dessiner les disques vides et pleins avec `tikz`. Afin que ces disques soit à la bonne hauteur, c’est-à-dire à la hauteur de la liaison qui leur arrive dessus, on se servira de la commande `\chemskipalign`. Pour que, à la deuxième ligne de l’exemple ci dessous, les liaisons “collent” aux disques, nous utiliserons la possibilité de modifier le retrait d’une liaison avec le caractère “`#`”, fonctionnalité qui a été vue à la page 11. Pour les utilisateurs ayant chargé le package `babel` avec l’option “`frenchb`”, il est nécessaire de désactiver le caractère actif “`;`” qui marque la fin d’une instruction `tikz` et qui sinon ne serait pas reconnu en tant que tel. C’est pourquoi on se place dans un groupe où l’on écrit l’instruction `\shorthandoff{;}` :

Utilisation de \chemskipalign et #	
<pre>\begingroup \shorthandoff{;} \def\emptydisk{\chemskipalign\tikz\draw(0,0)circle(2pt);} \def\fulldisk{\chemskipalign\tikz\fill(0,0)circle(2pt);} \chemfig{A-\emptydisk-\fulldisk-B}\par \chemfig{A-#(,Opt)\emptydisk-#(Opt,Opt)\fulldisk-#(Opt)B} \endgroup</pre>	

5 Liaisons doubles déportées

Toutes les liaisons doubles sont composées de 2 traits et ces traits sont tracés de part et d’autre de la ligne théorique que prendrait la liaison simple. Il est possible de déporter cette liaison double de telle sorte qu’un des deux traits soit dans le prolongement de cette ligne théorique. L’autre trait étant alors au dessus ou au dessous de la liaison. En fait, il est plus rigoureux de dire “à gauche” ou “à droite” de la ligne théorique lorsqu’on parcourt la liaison dans le sens du tracé.

Pour déporter la liaison vers la gauche, il suffit d’écrire “`=^`” et pour la déporter vers la droite “`=_`” :

Liaisons doubles déportées	
<pre>\chemfig{A==B}\par \chemfig{A=^B}\par \chemfig{A=_B}</pre>	

Dans les cycles, les liaisons doubles sont automatiquement déportées vers la gauche. On peut cependant les déporter vers la droite en le spécifiant avec “=” :



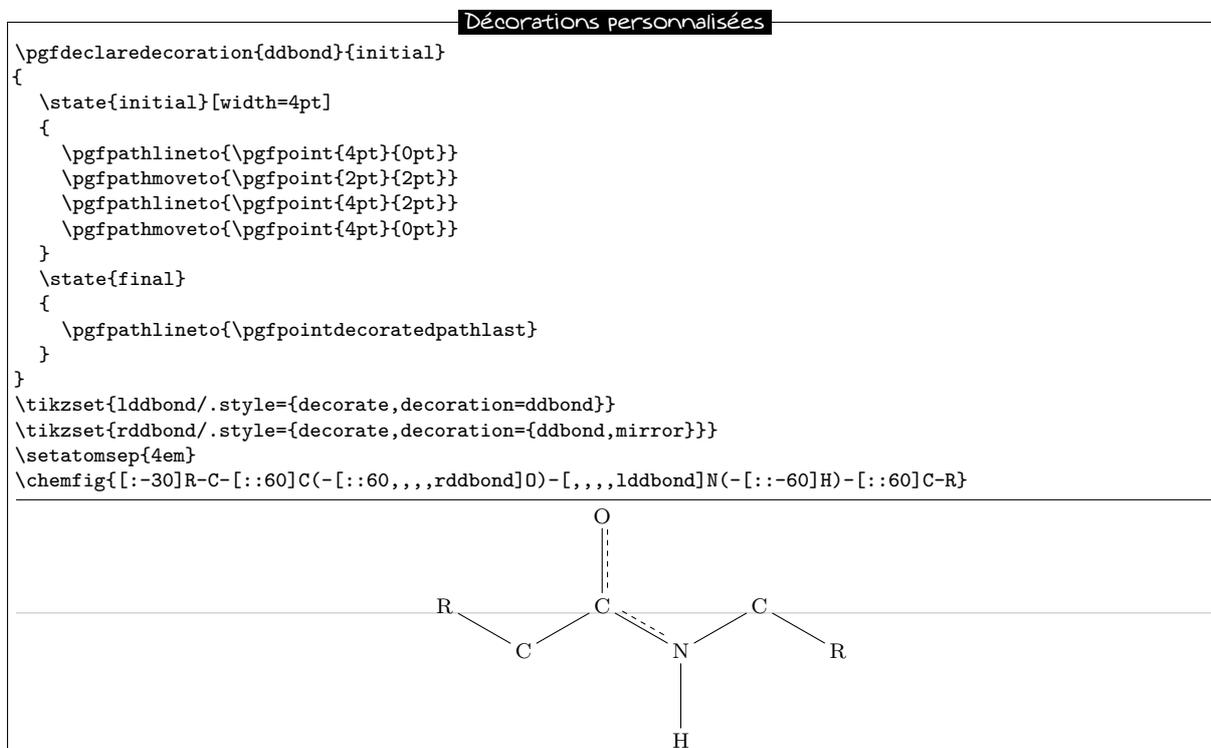
Les liaisons déportées sont particulièrement utiles dans le tracé de formules topologiques de molécules comprenant chaînes carbonées avec des liaisons doubles. Elles permettent d’avoir une ligne brisée continue, alors que cette ligne brisée serait discontinue avec les liaisons doubles normales :



6 Liaisons doubles délocalisées

Il est parfois nécessaire de tracer une liaison double dont un trait serait plein et l’autre en pointillé⁹. Cette fonctionnalité n’est pas codée en dur dans **ChemFig** puisque *tikz*, sa librairie “decorations” et ses styles programmables ont tout ce qu’il faut pour le rendre possible.

Tout d’abord, après avoir chargé la librairie “decorations” en mettant `\usetikzlibrary{decorations}` dans le préambule, il faut définir (lignes 1 à 14) une décoration baptisée “ddbond” pour « Dashed Double Bond » puis définir (lignes 15 et 16) deux styles de *tikz* que l’on appelle “lddbond” et “rddbond”. Pour le premier, le trait pointillé se fera à gauche du trait médian et pour le second à droite. Ces styles peuvent désormais être appelés avec le 5^e argument optionnel des liaisons :



9. Merci à Andreas BRÖERMANN qui m’a suggéré cette fonctionnalité et m’a donné la solution à ce problème.

7 Sauvegarde d'une sous molécule

ChemFig est capable de sauvegarder un `<code>` sous forme d'un alias pour le réutiliser sous forme compacte dans le code d'une molécule. Ceci est particulièrement utile lorsque le `<code>` apparaît plusieurs fois.

Pour cela, on dispose de la commande

```
\definesubmol{<nom>}{<code>}
```

qui sauvegarde le `<code>` de façon à l'appeler dans le code de la molécule par le raccourci “!`<nom>`”. Ce `<nom>` peut être :

- une suite de caractères : tous les caractères alphanumériques pouvant se trouver entre `\csname` et `\endcsname` sont acceptés ;
- une séquence de contrôle.

Nouveau

Dans tous les cas, si l'alias est déjà défini il est déconseillé de l'écraser avec une nouvelle définition faite avec `\definesubmol`, d'ailleurs, un “warning” sera émis avertissant l'utilisateur que cet alias sera écrasé par le nouveau. Pour écraser la définition d'un alias faite au préalable, il faut utiliser :

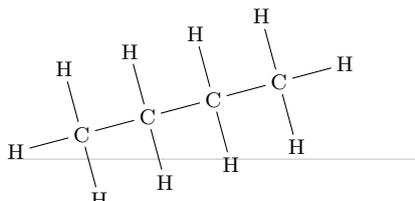
```
\redefinesubmol{<nom>}{<code>}
```

Voici un code qui dessine la molécule de pentane. On a pris soin auparavant de définir un alias “xy” pour le code `CH_2` :

Pentane	
<pre>\definesubmol{xy}{CH_2} \chemfig{H_3C-!{xy}-!{xy}-!{xy}-CH_3}</pre>	$\text{H}_3\text{C} \text{---} \text{CH}_2 \text{---} \text{CH}_2 \text{---} \text{CH}_2 \text{---} \text{CH}_3$

Ici, la manœuvre n'est pas très intéressante puisque “!`{xy}`” est aussi long à taper que le code qu'il remplace.

Mais dans certains cas, cette fonctionnalité fait gagner beaucoup de place dans le code de la molécule et en améliore la lisibilité. Dans l'exemple suivant, on dessine la molécule développée de butane. Pour cela, on va définir un alias avec la séquence de contrôle “`\xx`” pour la sous molécule `CH2`. Comme on n'emploie que des angles relatifs, il est possible de faire pivoter la molécule entière de l'angle que l'on veut via l'angle du paramètre optionnel global qui spécifie l'angle par défaut des liaisons de la molécule principale. On met ici 15° :

Butane	
<pre>\definesubmol\xx{C(-[:+90]H)(-[:-90]H)} \chemfig{[:15]H-!\xx-!\xx-!\xx-!\xx-H}</pre>	

Nouveau

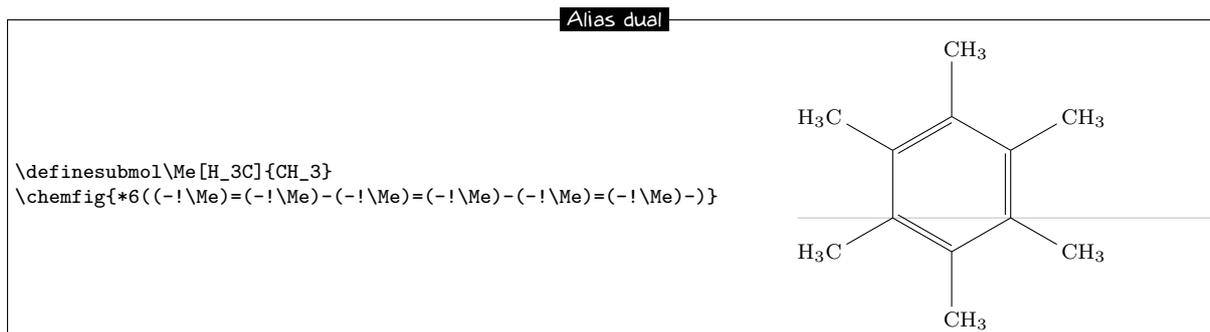
La commande `\definesubmol` admet un argument optionnel et sa syntaxe complète est la suivante :

```
\definesubmol{<nom>}[<code1>]{<code2>}
```

Dans la cas où l'argument optionnel est présent, l'alias “!`<nom>`” sera remplacé par `<code1>` si la liaison qui arrive sur l'alias vient de droite, c'est à dire si l'angle que fait la liaison entrante est compris entre -90° et 90° , ces valeurs étant non comprises. Pour tous les autres cas où la liaison arrive de gauche ou verticalement, l'alias sera remplacé par `<code2>`.

On va définir une séquence de contrôle `\Me` pour “méthyl” de telle sorte que l'alias “!`\Me`” soit remplacé par “`H_3C`” lorsque la liaison arrive de droite et par “`CH_3`” lorsqu'elle arrive de gauche. Avec cet alias, on peut observer sur l'exemple que l'on n'a plus à s'occuper de l'angle (que dans certaines molécules plus

complexes on ne connaît même pas) :



8 Décorations

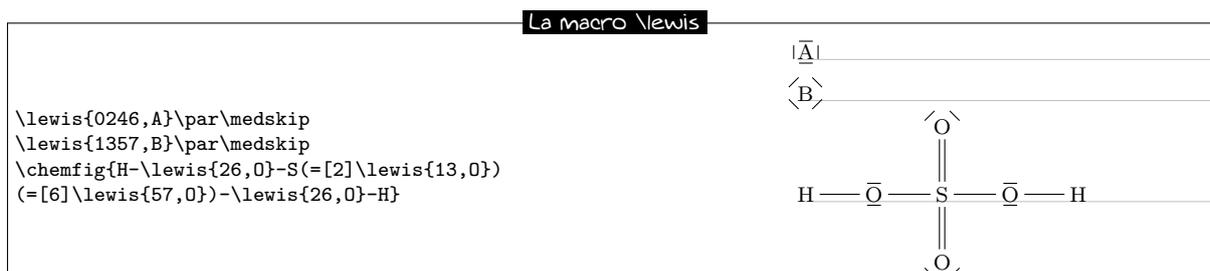
8.1 Formule de Lewis

La macro `\lewis` permet de placer des paires d'électrons, des électrons ou une lacune. On utilise cette syntaxe :

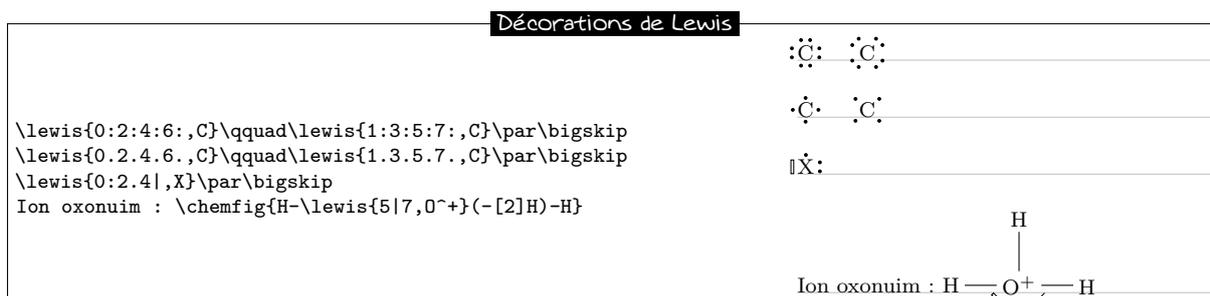
$$\backslash\text{lewis}\{<n1><n2>\dots<ni>,<atome>\}$$

où les `<n1>...<ni>` représentent les positions (en multiples de 45°) désirées autour de l'`<atome>`. Ces entiers doivent être compris entre 0 et 7.

Cette commande peut également être utilisée en dehors de l'argument de `\chemfig` :



Si au lieu d'une paire représentée par une ligne, on souhaite deux points, on fait suivre l'entier par “:”. Si on veut dessiner un électron, il suffit de le faire suivre par un “.”. Pour dessiner une lacune, on le fait suivre d'un “|” :



Toutes les décorations dessinées par l'intermédiaire de `\lewis` ne sont pas comptabilisées dans la boîte englobante de l'atome. Les décorations sont faites en surimpression. On en voit une conséquence dans les deux exemples ci-dessus où les cadre ne semblent ne plus être correctement ajustés au dessin de la molécule qui dépasse légèrement vers le bas. Ceci sera parfois observable dans ce chapitre “Décorations” où l'on présente des commandes qui ne modifient pas la boîte englobante.

On peut le constater de façon plus évidente en traçant une `\fbox` autour des atomes décorés :



On peut régler plusieurs paramètres à l'aide la macro

$$\backslash\text{setlewis}\{\langle\text{dim1}\rangle\}\{\langle\text{dim2}\rangle\}\{\langle\text{code tikz}\rangle\}$$

Si un argument est vide, il prend sa valeur par défaut.

- $\langle\text{dim1}\rangle$ est la distance entre la boîte englobant l'atome et la décoration. Elle vaut 0.4ex par défaut ;
- $\langle\text{dim2}\rangle$ est la longueur du trait représentant la paire d'électrons. Elle vaut 1.5ex par défaut ;
- $\langle\text{code tikz}\rangle$ est le code qui sera passé directement à *tikz*. Ce code est vide par défaut.

Les paramètres de la macro $\backslash\text{lewis}$

$\backslash\text{setlewis}\{4\text{pt}\}\{1.5\text{em}\}\{\text{red}\}$	$A \text{ --- } \overline{B} \text{ --- } C$
$\backslash\text{chemfig}\{A-\backslash\text{lewis}\{26,B\}-C\}\backslash\text{par}\backslash\text{bigskip}$	$A \text{ --- } \overline{B} \text{ --- } C$
$\backslash\text{setlewis}\{\}\{\}\{line\ width=0.4\text{pt}\}$	$A \text{ --- } \overline{B} \text{ --- } C$
$\backslash\text{chemfig}\{A-\backslash\text{lewis}\{2 ,B\}-C\}$	$A \text{ --- } \overline{B} \text{ --- } C$

8.2 Empilement de caractères

Les macros

$$\backslash\text{chemabove}[\langle\text{dim}\rangle]\{\langle\text{code}\rangle\}\{\langle\text{matériel}\rangle\}$$

et

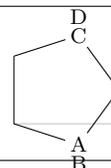
$$\backslash\text{chembelow}[\langle\text{dim}\rangle]\{\langle\text{code}\rangle\}\{\langle\text{matériel}\rangle\}$$

placent le $\langle\text{matériel}\rangle$ respectivement au dessus et au dessous du $\langle\text{code}\rangle$ à une distance verticale $\langle\text{dim}\rangle$, et cela sans changer la boîte englobante du $\langle\text{code}\rangle$. La dimension $\langle\text{dim}\rangle$ vaut 1.5pt par défaut.

Ces commandes sont indépendantes de la macro $\backslash\text{chemfig}$ et peuvent aussi bien être appelées à l'intérieur ou à l'extérieur de son argument.

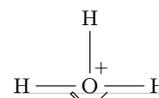
On peut les utiliser notamment dans les cycles en prenant soin de mettre des accolades autour des lettres A, B, C et D pour éviter que **ChemFig** ne stoppe la lecture de l'atome sur ces lettres :

Superposition dans les cycles

$$\backslash\text{chemfig}\{*5(-\backslash\text{chembelow}\{A\}\{B\}--\backslash\text{chemabove}\{C\}\{D\}--)\}$$


Elles sont parfois utiles pour placer des pseudo exposants qui ne changent pas la boîte englobante de l'atome, de façon à ce que les liaisons n'en soient pas trop éloignées :

ion oxonium

$$\backslash\text{chemfig}\{H-\backslash\text{chemabove}\{\backslash\text{lewis}\{5|7,0\}\}\{\quad\scriptstyle+\}\{-[2]H\}-H\}$$


8.3 Réactions chimiques

Pour écrire des réactions chimiques, **ChemFig** fournit une commande pour les signes et une commande pour les flèches.

La commande $\backslash\text{chemsign}[\langle\text{dim}\rangle]\langle\text{signe}\rangle$ écrit le $\langle\text{signe}\rangle$, celui-ci étant entouré de part et d'autre d'un espacement horizontal insécable $\langle\text{dim}\rangle$ qui vaut 0.5em par défaut.

La commande $\backslash\text{chemrel}[\langle\text{arg1}\rangle][\langle\text{arg2}\rangle]\{\langle\text{code flèche}\rangle\}$ trace une flèche où les argument optionnels $\langle\text{arg1}\rangle$ et $\langle\text{arg2}\rangle$ seront placés respectivement au dessus et au dessous de la flèche, sans que sa boîte englobante soit modifiée. Le $\langle\text{code flèche}\rangle$ est directement passé à *tikz* sauf dans le cas où il comporte " $\langle\rangle$ " qui permet de tracer deux flèches l'une au dessus de l'autre.

Types de flèches

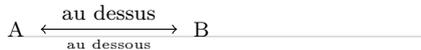
$$\begin{aligned} &A\backslash\text{chemrel}\{->\}B\backslash\text{par} \\ &A\backslash\text{chemrel}\{<->\}B\backslash\text{par} \\ &A\backslash\text{chemrel}\{<->\}B\backslash\text{par} \\ &A\backslash\text{chemrel}\{<>\}B\backslash\text{par} \\ &A\backslash\text{chemrel}\{->,red,thick\}B \end{aligned}$$


La commande `\setchemrel{<dim1>}{<dim2>}{<dim3>}` permet de régler les dimensions dans le tracé de la flèche :

- `<dim1>` est l'espace vertical entre la flèche et les textes optionnels écrits au dessus et/ou au dessous. Si cet argument est vide, il prend la valeur 2pt par défaut ;
- `<dim2>` est l'espace horizontal insécable inséré avant et après la flèche. Si cet argument est vide, il prend sa valeur par défaut de 0,7em ;
- `<dim3>` est la longueur de la flèche. Si l'argument est vide, il prend sa valeur par défaut de 4em.

Textes au dessus des flèches

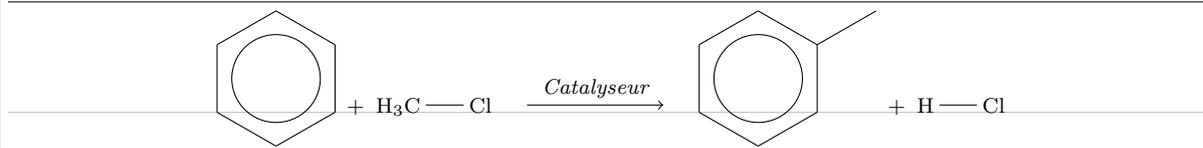
```
\setchemrel{1pt}{6em}
A\chemrel[\footnotesize au dessus][\tiny au dessous]{<->}B
```



Voici un exemple de réaction chimique :

Réaction chimique

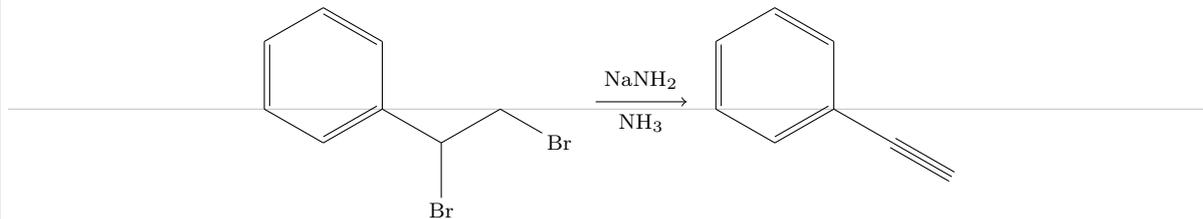
```
\setchemrel{0pt}{1.2em}{6em}
\chemfig{*6(-----)}\chemsign+\chemfig{H_3C-Cl}
\chemrel[\itshape\footnotesize Catalyseur]{->}
\chemfig{*6(---(-)---)}\chemsign+\chemfig{H-Cl}
```



Et en voici une autre :

Synthèse du phényléthyne

```
\chemfig{*6(==*6(-(-Br)-(-Br))===)}
\chemrel[\chemfig{NaNH_2}][\chemfig{NH_3}]{->}
\chemfig{*6(==(-)===)}
```

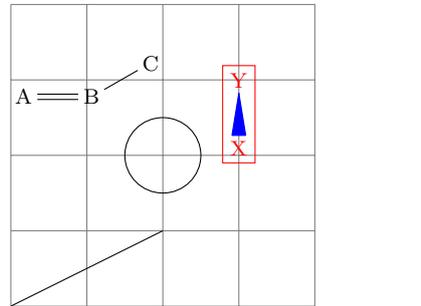


9 Utilisation de `\chemfig` dans l'environnement `tikzpicture`

Il est possible d'appeler la commande `\chemfig` à l'intérieur d'un environnement `tikzpicture` :

`\chemfig` dans `tikzpicture`

```
\begin{tikzpicture}[help lines/.style={thin,draw=black!50}]
\draw[help lines] (0,0) grid (4,4);
\draw(0,0) -- (2,1);
\draw(2,2) circle (0.5);
\node at (1,3) {\chemfig{A=B-[:30]C}};
\node[draw,red,anchor=base] at(3,2){\chemfig{X}>[2,,,blue]Y}};
\end{tikzpicture}
```



10 Au delà de la chimie

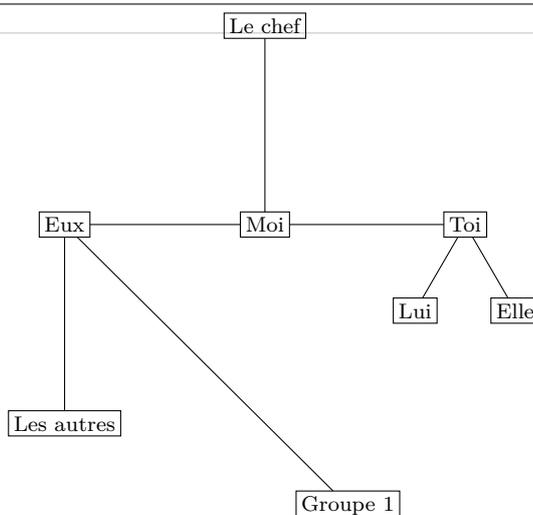
Intrinsèquement, **ChemFig** est un outil pour tracer des graphes, et cet outil a été programmé pour qu'il soit adapté à la chimie. Dans une certaine mesure, il est possible de détourner **ChemFig** de son usage premier pour tracer des organigrammes ou autres schémas se ramenant à des graphes.

Chaque atome est contenu dans un nœud de *tikz*. Par défaut, ces nœuds ont un “inner sep” et un “outer sep” égal à 0pt. Ils sont rectangulaires comme on l’a vu page 10. Ces valeurs par défaut peuvent être écrasées avec la macro `\setnodestyle` dont l’argument est passé à *tikz* et spécifie le style des nœuds contenant les atomes.

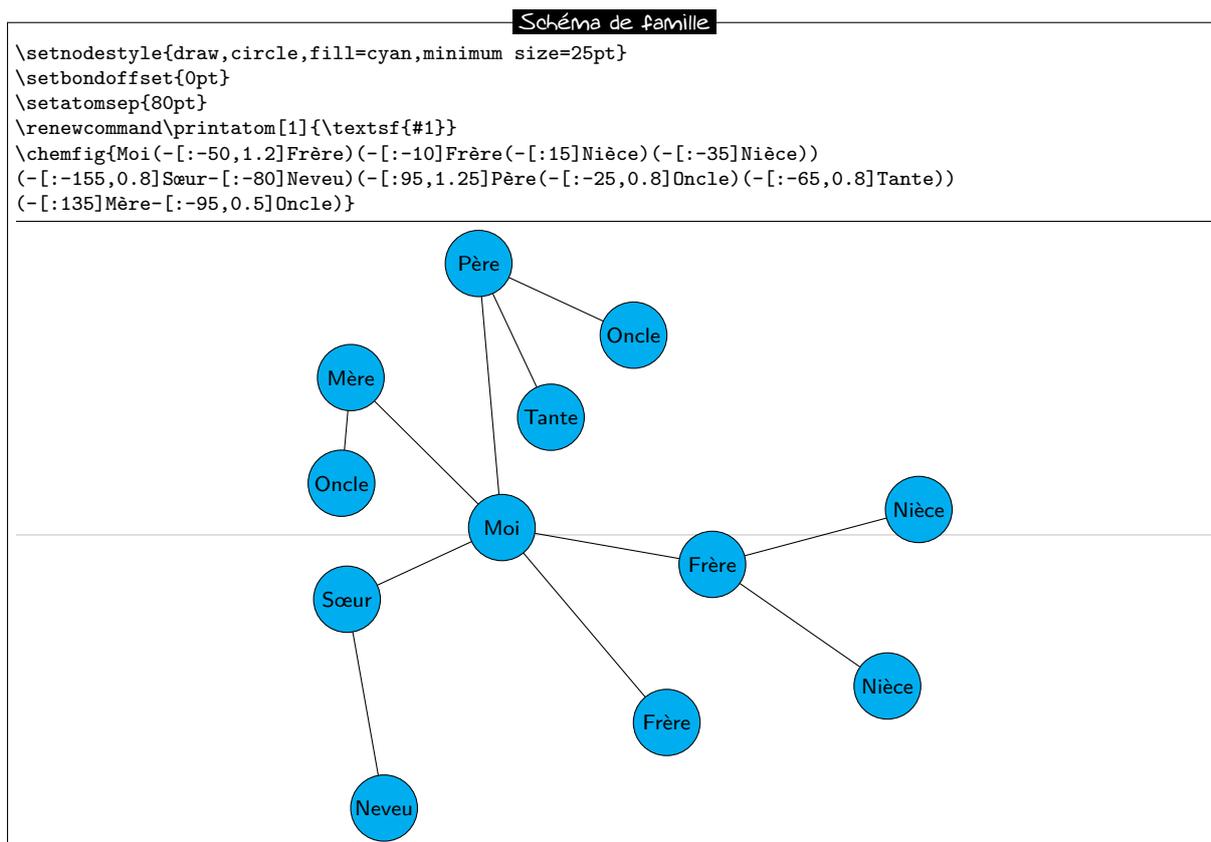
Dans cet exemple, on spécifie simplement “draw,inner sep=2pt” ce qui a pour effet de tracer le contour des nœuds et espacer de 2pt leur contour de leur contenu. On spécifie également `\setbondoffset{0pt}` pour que les liaisons touchent les frontières des nœuds. L’espace interatome est allongé à 75pt. Enfin, la commande `\printatom` est réduite à sa plus simple expression de façon à ne plus se placer en mode mathématique pour afficher les atomes et donc tenir compte des espaces :

Un organigramme

```
\setnodestyle{draw,inner sep=2pt}
\setbondoffset{0pt}
\setatomsep{75pt}
\renewcommand\printatom[1]{#1}
\chemfig{Le chef-[6]Moi(-[4]Eux(-[6]Les autres)-[7,2]Groupe 1))-Toi(-[: -120,0.5]Lui)(-[: -60,0.5]Elle)}
```



Voici un autre organigramme où les nœuds sont circulaires et colorés en bleu cyan :

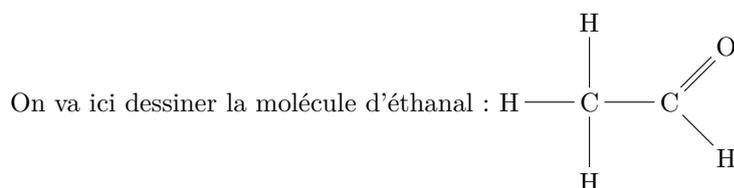


11 Exemples commentés

Dans ce chapitre, plusieurs molécules seront dessinées en mettant en œuvre les méthodes précédemment exposées. Le but recherché ici est de montrer dans quel ordre logique peut se construire une molécule de façon à ce que l'utilisateur peu familier avec **ChemFig** acquière une méthode pour construire des molécules complexes. Pour l'y aider, les étapes de la construction seront montrées.

De plus, on mettra en évidence que plusieurs possibilités s'offrent à l'utilisateur pour un même résultat graphique, certaines intuitives et d'autres beaucoup moins, l'essentiel étant de montrer que **ChemFig** permet une certaine souplesse dans le codage des molécules. À chacun ensuite de se construire et s'appropriier les méthodes avec lesquelles il est le plus à l'aise.

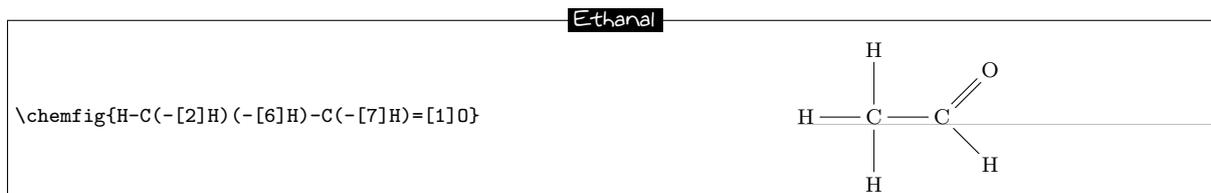
11.1 L'éthanal



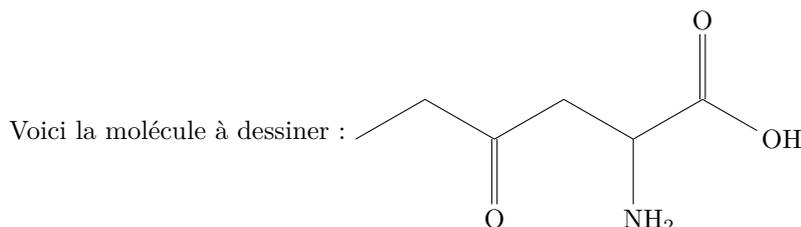
La meilleure méthode pour les molécules non cycliques est de choisir la plus longue chaîne. Ici on peut prendre "H-C-C=O" par exemple. Il faut incliner la liaison C=O de 45° en utilisant l'angle prédéfini "[1]". On obtient la "structure" de la molécule sur laquelle il suffira de rajouter les ramifications :



Il reste à placer 3 atomes d'hydrogène avec les bonnes inclinaisons à l'aide des angles prédéfinis. Le premier à 90° avec la ramification “(-[1]H)”, le second à 270° avec “(-[6]H)” et celui de droite à 315° avec “(-[7]H)” :



11.2 L'acide 2-amino-4-oxohexanoïque



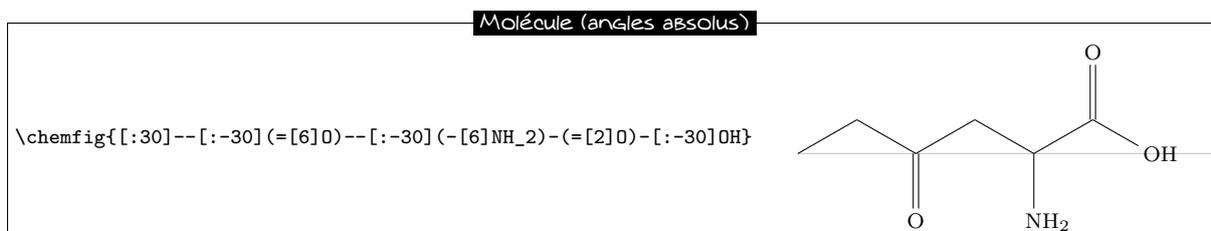
Il existe — comme c'est le souvent cas pour la plupart des molécules — plusieurs méthodes, et pour chacune, plusieurs façons différentes d'arriver au résultat. Ici, nous allons examiner 4 méthodes différentes.

11.2.1 Angles absolus

On va tout d'abord tracer la chaîne médiane avec des angles absolus. On règle l'angle par défaut à $+30^\circ$ avec l'argument optionnel, ainsi seules les liaisons “descendantes” auront besoin que l'on spécifie leur angle absolu de -30° :



Il reste à ajouter les ramifications “(=[6]O)”, “(-[6]NH₂)” et “(=[2]O)” sur les bons sommets :

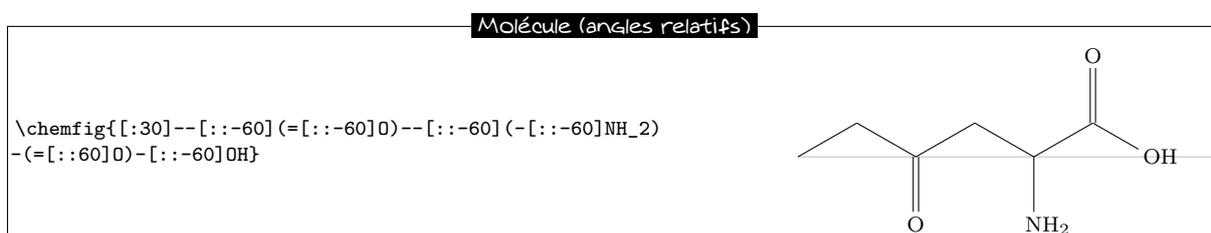


11.2.2 Angles relatifs

Une approche plus générale aurait été de n'utiliser que des angles relatifs. Il aurait fallu procéder de cette façon :

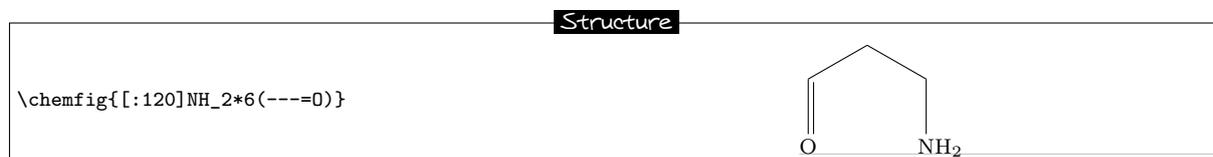


puis

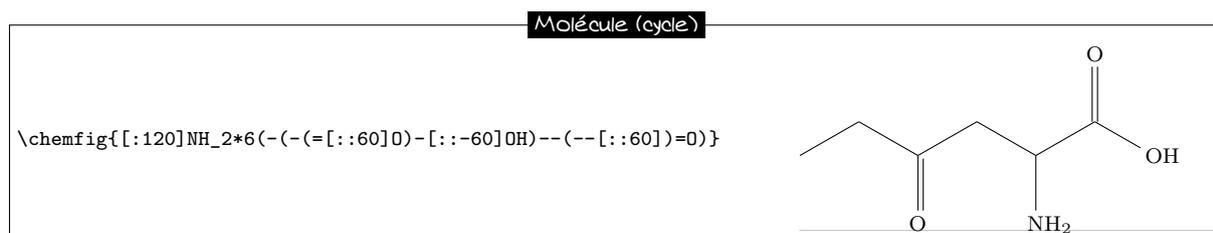


11.2.3 Cycle

Les angles entre les liaisons étant de 120° , on peut penser à utiliser un 6-cycle, quoique cette méthode soit moins naturelle. Il faut ici profiter du fait qu'un cycle peut être incomplet. Il est également nécessaire de faire pivoter le cycle de 120° pour que le premier sommet soit au sud-est du cycle :

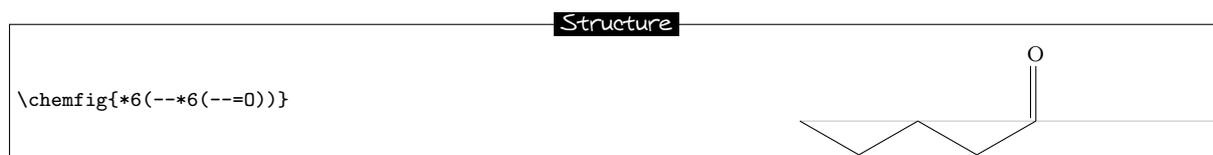


Il faut maintenant faire partir les ramifications des sommets adéquats :

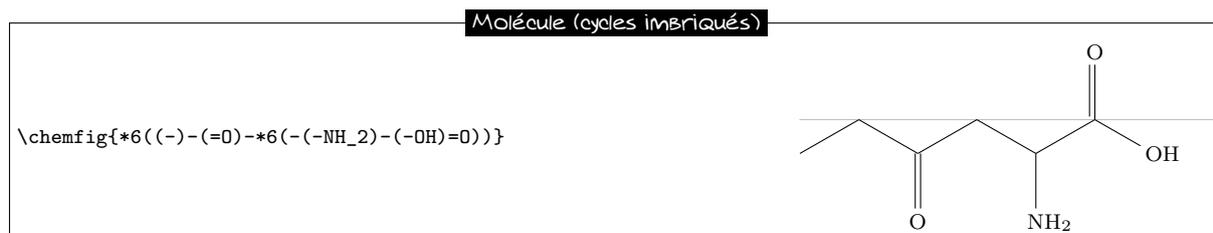


11.2.4 Cycles imbriqués

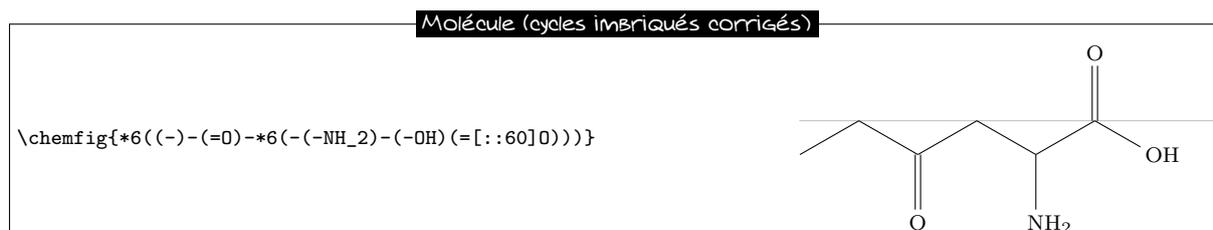
En approfondissant la méthode avec les cycles, on peut aussi penser à imbriquer des 6-cycles incomplets. On pourrait partir de cette structure :



Et ensuite ajouter les liaisons qui partent des sommets des ces cycles. Il n'y a pas à se préoccuper des angles puisque les liaisons qui partent des cycles sont les bissectrices des côtés du cycle, ce que justement, on cherche ici :



Un examen attentif révèle cependant que la seconde double liaison vers l'atome d'oxygène se trouve à l'intérieur du 6-cycle incomplet¹⁰. Malgré sa concision, ce code ne conduit donc pas à un dessin parfait. On peut bien sûr corriger ce défaut en allongeant un peu le code :



10. C'était aussi le cas pour la méthode précédente avec un seul cycle.

11.3 Glucose

Le but ici est de représenter selon différentes conventions la molécule de glucose.

11.3.1 Formule topologique

Le code ressemble ici à celui de l'acide 2-amino-4-oxohexanoïque. On obtient quasiment la même structure avec des angles absolus sauf qu'ici, l'angle par défaut est de -30° :

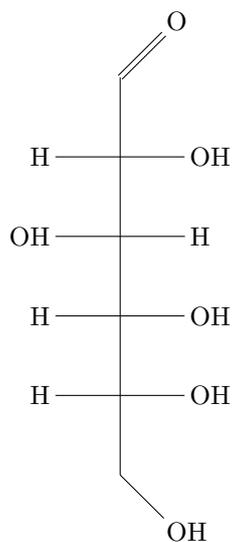


Et l'ajout des ramifications ne pose pas de problème particulier. On utilise des angles absolus :

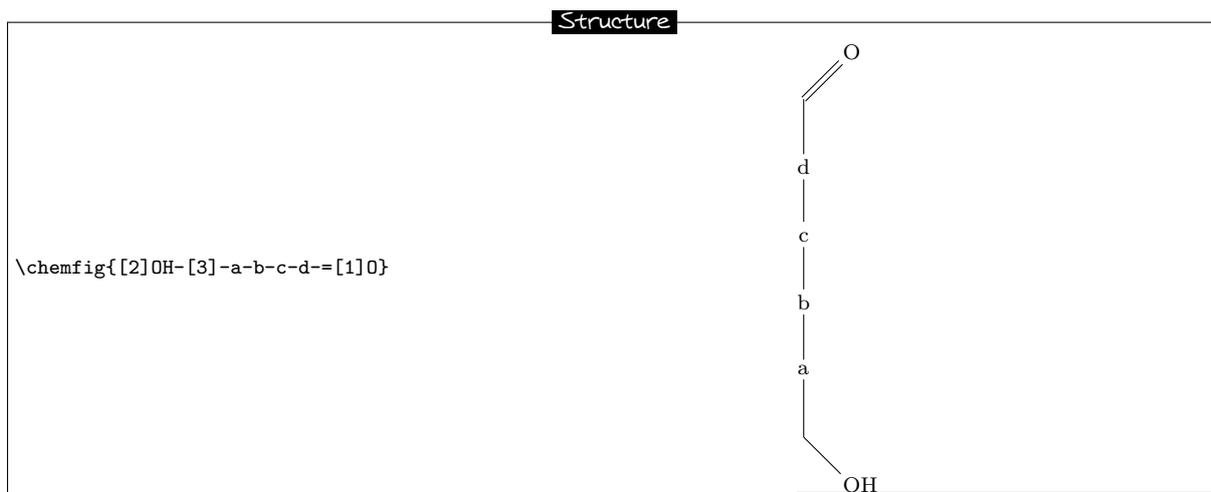


11.3.2 Projection de Fisher

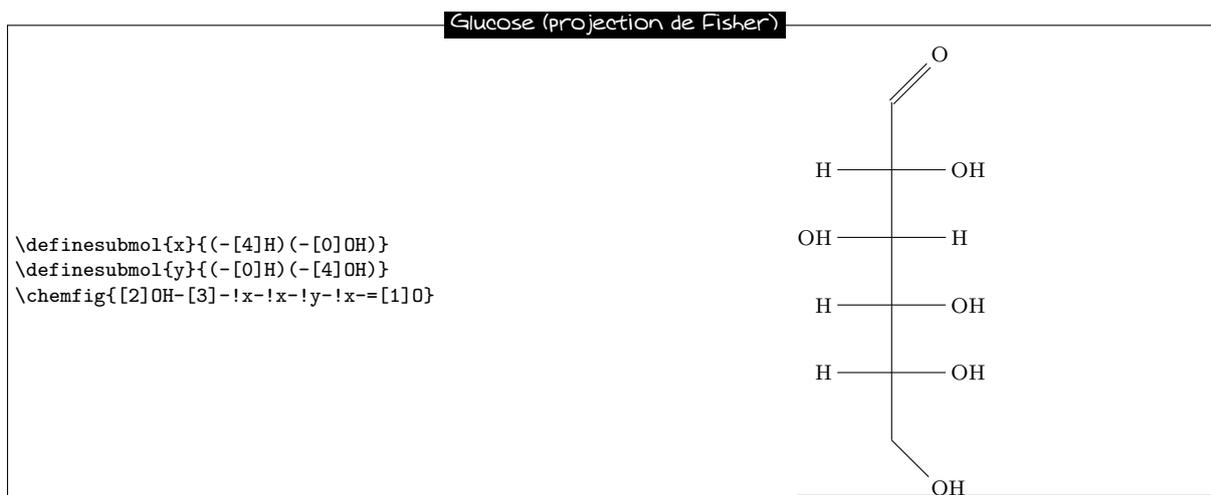
Le but est d'obtenir la molécule ci dessous :



L'idée est de commencer à dessiner la plus longue chaîne verticale en indiquant un angle "[2]" par défaut. Voici la structure où l'on met volontairement des lettres minuscules au bout de chaque liaison verticale :

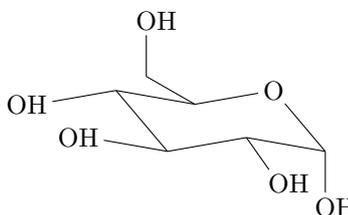


Ensuite, on définit deux alias pour les liaisons horizontales et les atomes qui les termineront. Prenons "x" que nous mettrons à la place des minuscules a, c et d et "y" qui prendra la place de la lettre c. Comme les alias ne comportent qu'un seul caractère, on peut se passer d'accolade et écrire "!x" au lieu de "!{x}" :



11.3.3 Représentation « chaise »

On va représenter cette molécule d' α -D-glucose :



Pour cela, on va tout d'abord tracer 5 côtés de la chaise et relier le premier sommet au dernier avec un crochet "?". On adopte les angles absolus suivants, donnés dans l'ordre de parcourt trigonométrique : -50° , 10° , -10° , 130° , 190° .

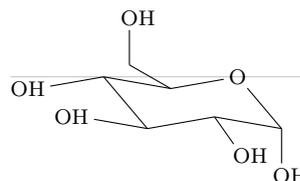


Maintenant, il suffit de rajouter les ramifications entre parenthèses. Les angles sont choisis au mieux pour

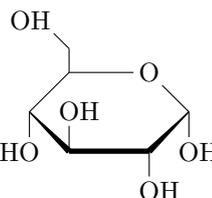
rendre une impression de perspective, et certaines liaisons sont raccourcis d'un coefficient de 0,7 :

Représentation chaise

```
\chemfig{?(-[:190]OH)-[: -50](-[:170]OH)-[:10](-[: -55,0.7]OH)
-[: -10](-[6,0.7]OH)-[:130]O-[:190]?(-[:150,0.7]-[2,0.7]OH)}
```



11.3.4 Projection de Haworth



Le but est de représenter cette molécule de D-glucopyranose :

Tout d'abord, on va choisir la plus longue chaîne qui part du groupe "HO" de gauche et continue sur 5 côtés du cycle. Le cycle sera fermé avec un crochet. Pour la liaison verticale qui part du premier groupe "HO", il faut spécifier qu'elle doit partir du second atome avec l'argument optionnel. De plus, elle sera raccourcie avec un coefficient de 0,5. Son argument optionnel sera donc "[2,0.5,2]".

Ensuite, pour donner une impression de perspective au cycle, les liaisons inclinées seront réduites d'un coefficient de 0,7. Pour les traits gras inclinés, on va se servir des liaisons de Cram en ayant redéfini la largeur de la base des triangles à 2pt. Pour la liaison horizontale en trait gras, il faudra la tracer avec une épaisseur de 2pt et son argument optionnel sera donc "[0,,,line width=2pt]". Voici la structure de la molécule :

Structure

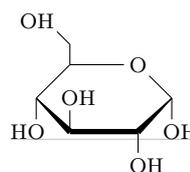
```
\setcrambond{2pt}{-}{-}
\chemfig{HO-[2,0.5,2]?<[7,0.7]-[,,,
line width=2pt]>[1,0.7]-[3,0.7]O-[4]?}
```



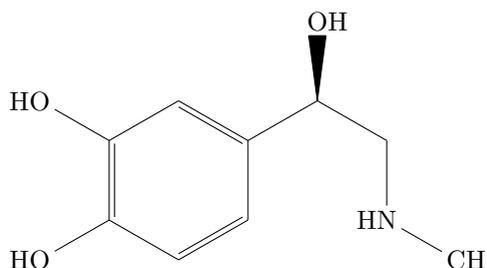
Il ne reste plus qu'à rajouter les ramifications, mises au bon endroit, en spécifiant des angles absolus correctement choisis et des longueurs parfois réduites pour mieux donner l'illusion de la perspective :

Projection de Haworth

```
\setcrambond{2pt}{-}{-}
\chemfig{HO-[2,0.5,2]?<[7,0.7](-[2,0.5]OH)-[,,,
line width=2pt](-[6,0.5]OH)>[1,0.7](-[6,0.5]OH)-[3,0.7]
O-[4]?(-[2,0.3]-[3,0.5]OH)}
```



11.4 Adrénaline



On cherche à dessiner la molécule d'adrénaline :

Nous allons utiliser deux méthodes différentes.

11.4.1 Utilisation d'un cycle

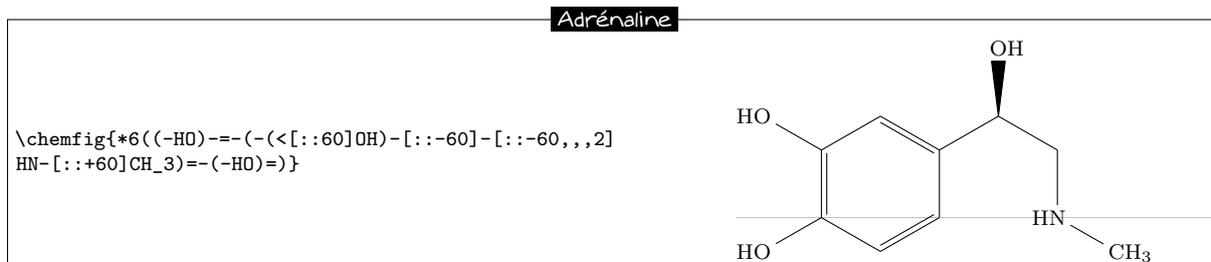
Tout d'abord, on part du 6-cycle et nous dessinons le début des ramifications qui en partent :



Il faut maintenant compléter la ramification de droite en utilisant, par exemple, des angles relatifs :



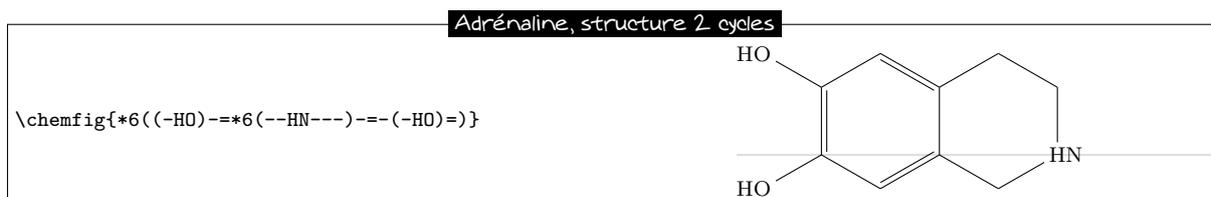
Puis, il faut rajouter la liaison de Cram vers OH et spécifier que la liaison qui arrive sur "NH" le fait sur le second atome "N". Nous utilisons le 4^e argument optionnel de la liaison :



11.4.2 Utilisation de 2 cycles

Cette méthode est moins naturelle, mais le but est ici d'expliquer comment rendre une liaison invisible.

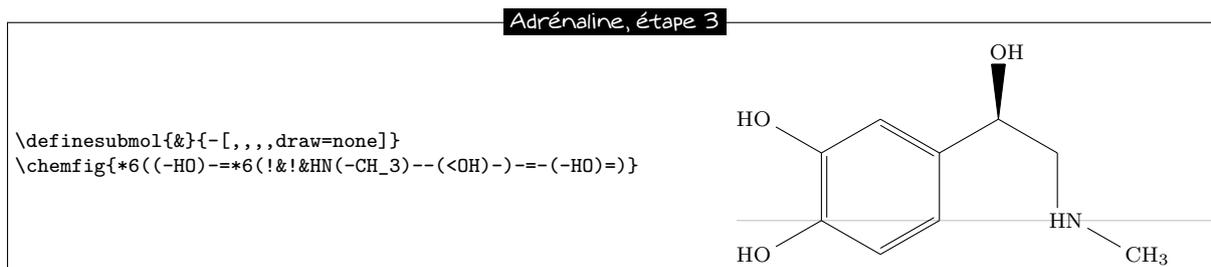
On pourrait améliorer ce code en considérant que le dessin de la molécule d'adrénaline est constitué de deux 6-cycles accolés l'un à l'autre :



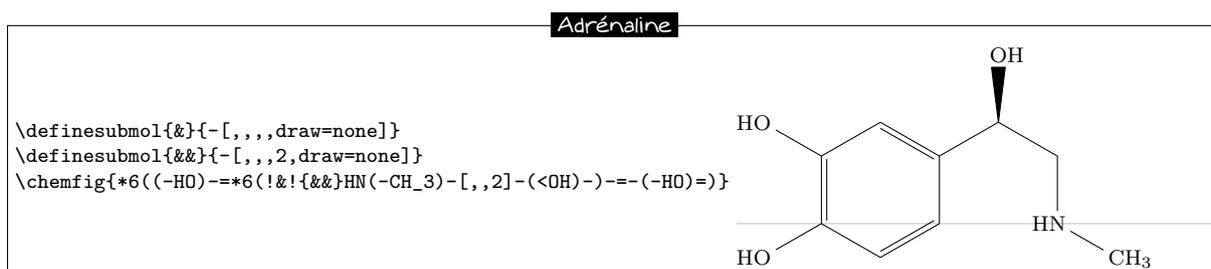
Il faut donc rendre invisible les deux premières liaisons du cycle de droite. Pour cela, on se sert de l'argument qui est passé à `tikz` en spécifiant "`draw=none`". Ces liaisons ont donc ce code : "`-[,,,,draw=none]`". Pour que le code reste lisible, on définit un alias nommé "&" pour ces liaisons :



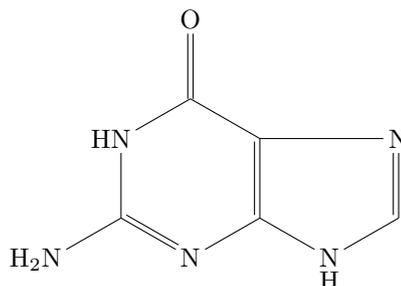
Le reste devient facile, il suffit d'ajouter les ramifications aux bons sommets :



Pour finir, on spécifie que les liaisons qui *arrivent et partent* de “HN” doivent le faire sur deuxième atome. On définit donc un autre alias pour la liaison invisible qui arrive sur “HN” :

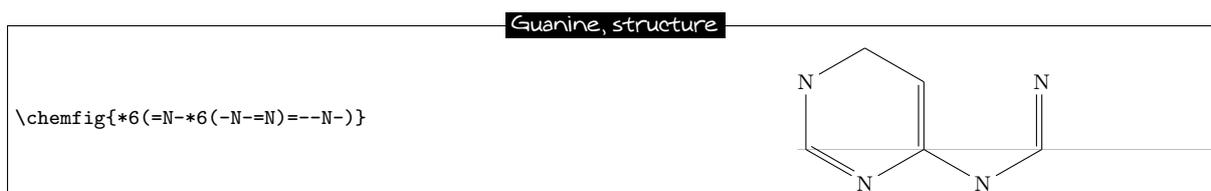


11.5 Guanine



Nous allons dessiner la molécule de guanine :

Tout d'abord, commençons par dessiner les cycles imbriqués en mettant seulement les atomes d'azote aux sommets :



Puis nous allons tracer la liaison horizontale dans le cycle de droite avec un crochet. Nous allons aussi positionner un atome d'hydrogène sous l'atome d'azote du 5-cycle avec la commande `\chembelow{N}{H}`. Il faut aussi écrire “HN” au lieu de “N” au sommet en haut à gauche de la molécule :



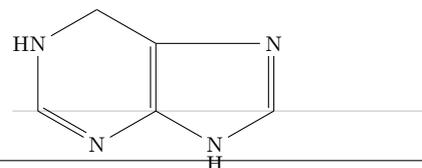
On constate qu'une liaison part du mauvais atome¹¹ ! Il faut corriger le mécanisme de calcul automatique

11. Ceci semble illogique puisque l'angle de la liaison du groupe HN vers le premier sommet du 6-cycle est comprise entre -90° et 90° ; ChemFig devrait donc partir du 2^e atome. Pour expliquer cette contradiction, il faut savoir que dans les cycles, la dernière liaison relie toujours le dernier sommet au premier sommet en ignorant l'angle *calculé théorique* de cette liaison

pour que la liaison parte du 2^e atome “N” au lieu du premier. Pour cela, on spécifie un argument optionnel pour la dernière liaison du premier 6-cycle “[, , 2]” :

Guanine, étape 3

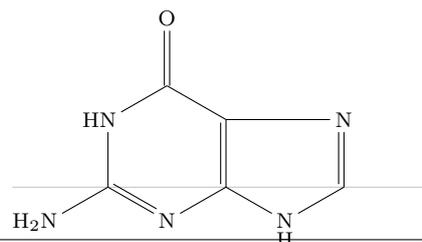
```
\chemfig{*6(=N-*6(-\chembelow{N}{H}-=N?)=?--HN-[ , , 2])}
```



Il suffit de rajouter les ramifications aux bons sommets. On peut notamment remarquer la ramification qui part du premier sommet du premier 6-cycle “(-H_2N)” :

Guanine

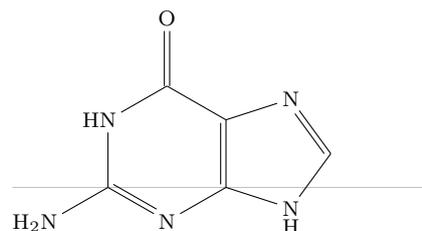
```
\chemfig{*6((-H_2N)=N-*6(-\chembelow{N}{H}-=N?)=?-(=O)-HN-[ , , 2])}
```



On aurait aussi pu dessiner cette même molécule avec un 5-cycle *régulier*, comme cela se fait parfois :

Guanine avec 5-cycle

```
\chemfig{*6((-H_2N)=N-*5(-\chembelow{N}{H}-=N)-=(=O)-HN-[ , , 2])}
```



12 Liste des commandes

Les commandes créées par **ChemFig** sont :

(qui est ici de -90°). **ChemFig** utilise cet angle théorique pour déterminer les atomes de départ et d'arrivée, mais ne s'en sert pas pour tracer la liaison puisque les deux extrémités sont déjà définies. L'atome de départ de la dernière liaison est donc le n° 1.

Commandes	Description
<code>\chemfig<code></code>	dessine la molécule dont le dessin est décrit par le <code><code></code>
<code>\printatom</code>	cette macro affiche les atomes dans les molécules. Elle peut être redéfinie pour personnaliser l’affichage, voir 28
<code>\setnodestyle{<style tikz>}</code>	à l’aide de la syntaxe de tikz, cette macro définit le style de nœuds contenant les atomes, voir 36
<code>\definesubmol{<nom>}[<code1>]{<code2>}</code>	créé un alias <code>!<nom></code> que l’on peut placer dans le code des molécules à dessiner qui remplace le <code><code1></code> ou le <code><code2></code> selon l’inclinaison de la dernière liaison. Voir page 33
<code>\chemskipalign</code>	Ignore le groupe d’atomes en cours pour le mécanisme d’alignement vertical. Voir page 30.
<code>\redefinesubmol{<nom>}{<code>}</code>	remplace l’alias déjà existant <code>!<nom></code> par le nouveau <code><code></code> . Voir page 33
<code>\setcrambond{<dim1>}{<dim2>}{<dim3>}</code>	règle les dimensions des triangles représentant les liaisons de Cram : <code><dim1></code> est la largeur de la base, <code><dim2></code> est l’espace entre les pointillés et <code><dim3></code> est la largeur des pointillés. Voir page 11
<code>\setatomsep{<dim>}</code>	règle la distance interatome. Voir page 11
<code>\setbondoffset{<dim>}</code>	règle l’espace entre les atomes liés et la liaison. Voir page 11
<code>\setdoublesep{<dim>}</code>	règle l’espace entre les deux traits d’une double liaison. Voir page 5
<code>\lewis{<codes>,<atome>}</code>	affiche l’ <code><atome></code> et positionne les décorations de Lewis selon ce qui est spécifié dans le <code><code></code> . Les décorations dessinées ne modifient pas la boîte englobante de l’ <code><atome></code> . Voir page 34
<code>\setlewis{<dim1>}{<dim2>}{<code tikz>}</code>	règle les décoration de Lewis ; <code><dim1></code> est la distance entre l’atome et la décoration, <code><dim2></code> est la longueur du trait représentant la paire d’électrons et <code><code tikz></code> est le code qui sera passé directement à tikz. Voir page 35
<code>\chemmove[<options tikz>]<code tikz></code>	Ouvre un environnement <code>tikzpicture</code> en y ajoutant à celles qui existent déjà les <code><options tikz></code> , et relie avec le <code><code tikz></code> les nœuds posés dans les molécules à l’aide du caractère “@”. Voir page 23.
<code>\chemsign[<dim>]<signe></code>	affiche le <code><signe></code> en plaçant avant et après un espace insécable de longueur <code><dim></code> . Voir page 35
<code>\chemrel[<txt1>][<txt2>]{<code fleche>}</code>	affiche une flèche décrite par son <code><code fleche></code> en positionnant les textes optionnels <code><txt1></code> et <code><txt2></code> respectivement au dessus et au dessous de la flèche. Voir page 35
<code>\setchemrel{<dim1>}{<dim2>}{<dim3>}</code>	règle les dimensions des flèches dessinées avec la commande <code>\chemrel</code> : <code><dim1></code> est l’espace vertical entre la flèche et les textes optionnels, <code><dim2></code> est l’espace horizontal insécable inséré avant et après la flèche et <code><dim3></code> est la longueur de la flèche. Voir page 36
<code>\chemabove[<dim>]{<txt1>}{<txt2>}</code>	écrit le <code><txt1></code> et positionne le <code><txt2></code> au dessus en laissant <code><dim></code> d’espace vertical. Cette commande ne change pas la boîte englobante de <code><txt1></code> . Voir page 35
<code>\chembelow[<dim>]{<txt1>}{<txt2>}</code>	écrit le <code><txt1></code> et positionne le <code><txt2></code> au dessous en laissant <code><dim></code> d’espace vertical. Cette commande ne change pas la boîte englobante de <code><txt1></code> . Voir page 35
<code>\chemname[<dim>]{<molécule>}{<nom>}</code>	Affiche le <code><nom></code> sous la <code><molécule></code>
<code>\chemnameinit</code>	Initialise la plus grande profondeur des molécules rencontrées pour avoir un alignement correct de leurs noms

Cinquième PARTIE

Galerie

Ce manuel s'achève avec des dessins de molécules plus ou moins complexes.

L'utilisateur curieux pourra s'intéresser au `<code>` de chaque molécule, bien que celui-ci devienne parfois rebutant lorsqu'elles deviennent complexes. En effet, au delà d'un certain niveau de complexité, bien qu'il soit assez facile d'écrire un `<code>` pour dessiner une molécule, il est assez ardu de relire ce `<code>` pour l'analyser a posteriori. On atteint rapidement les limites de la lisibilité immédiate du code d'un dessin complexe.

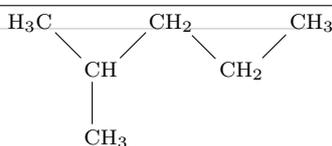
Quoi qu'il en soit, j'espère que cette extension aidera tous les utilisateurs de \LaTeX qui souhaitent dessiner des molécules chimiques. Compte tenu de la jeunesse de cette extension, j'espère que vous serez indulgent quant aux bugs rencontrés. Un **email** pour me signaler tout dysfonctionnement ou toute proposition d'amélioration sera bienvenu.

Christian TELLECHEA

*
* *

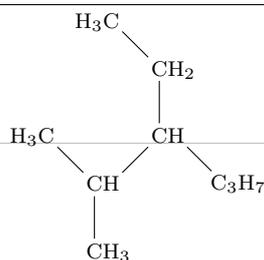
2-méthylpentane

```
\chemfig{[7]H_3C-CH(-[6]CH_3)-[1]CH_2-CH_2-[1]CH_3}
```



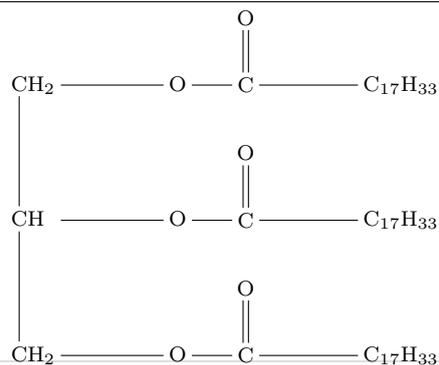
3-éthyl-2-méthylhexane

```
\chemfig{H_3C-[7]CH(-[6]CH_3)-[1]CH(-[7]C_3H_7)-[2]CH_2-[3]H_3C}
```

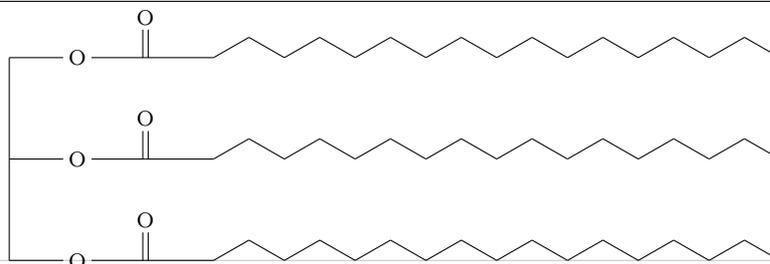


Stéarine, formule semi développée

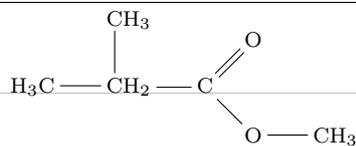
```
\definesubmol{@}{([0,2]-O-[0,1]C(=[2,1]O)-C_{17}H_{33}))}
\chemfig{[2,2]CH_2!@-CH_{\phantom 2}!@-CH_2!@}
```

**Stéarine, formule topologique**

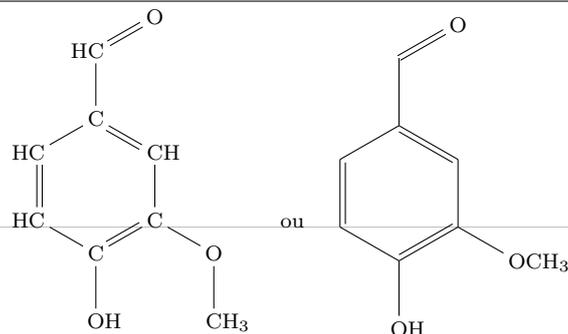
```
\definesubmol{x}{-[:+30,.6]-[: -30,.6]}
\definesubmol{y}{-O-(=[2,.6]O)-!x!x!x!x!x!x!x}
\chemfig{[2](!y)-[,1.5](!y)-[,1.5](!y)}
```

**2-Méthylpropanoate de méthyle**

```
\chemfig{H_3C-CH_2(-[2]CH_3)-C(=[1]O)-[7]O-CH_3}
```

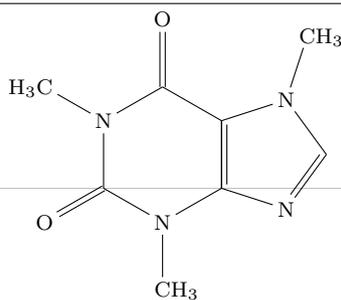
**La vanilline**

```
\chemfig{HC*6(-C(-OH)=C(-O-[: -60]CH_3)-CH=C(-[, , 2]HC=[: -60]O)-HC=[, , 2])} \quad ou \quad
\chemfig{*6(-(-OH)=(-OCH_3)-(-[: -60]O)-=)}
```



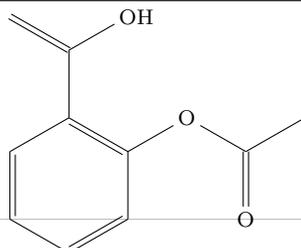
La caféine

```
\chemfig{*6((=O)-N(-CH_3)-*5(-N=N(-CH_3)-=)--(=O)-N(-H_3C)-)}
```



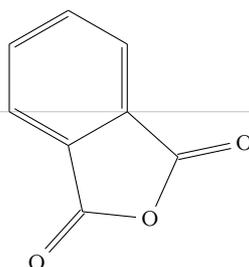
L'aspirine

```
\chemfig{*6(=-(-O-[::-60](=[::-60]O)-[::+60])=-(=[::+60])-[::-60]OH)-=)}
```



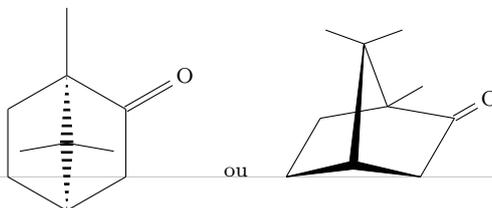
Anhydride phtalique

```
\chemfig{*6(=*5(-=O)-O-(=O)-=-=-)}
```



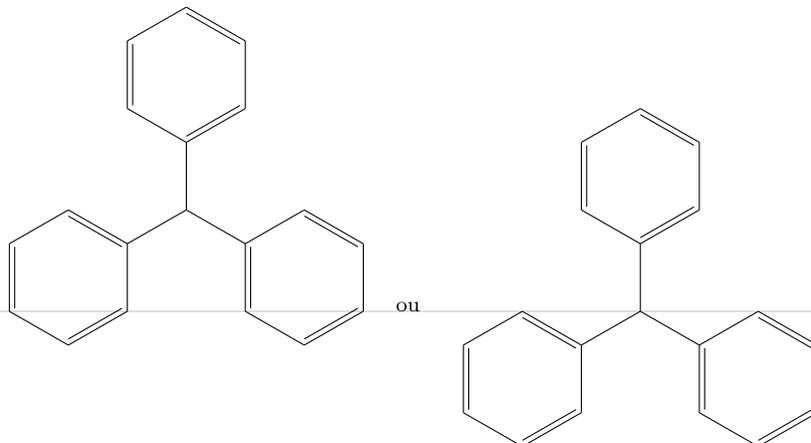
Camphre

```
\chemfig{*6(-(<[::120](-[::-100,0.7])(-[::100,0.7]))--(=O)-(-<[::120])--)}
\quad ou \quad
\setcrambond{3pt}{-}{-}
\chemfig{<[:10](>[:85,1.8]?(-[:160,0.6])-[:20,0.6])
>[:-10]-[:60](=[:30,0.6]O)-[:170]?(-[:30,0.6])-[:190]-[:240]}
```



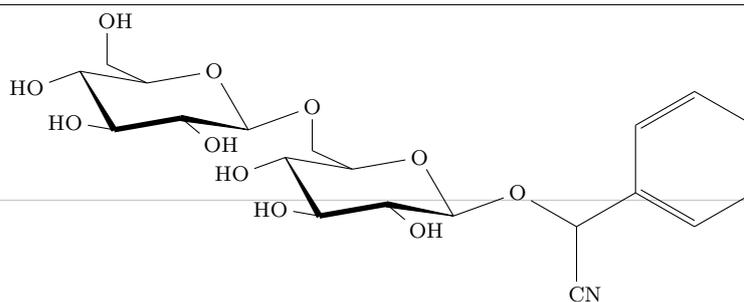
Triphenylméthane

```
\chemfig{*6(==*6(-=*6(==--=))-*6(==--=))=--=}
\quad ou \quad
\definesubmol{@}{*6(==--=)}
\chemfig{(-[:30]!@)(-[:90]!@)(-[:210]!@)}
```



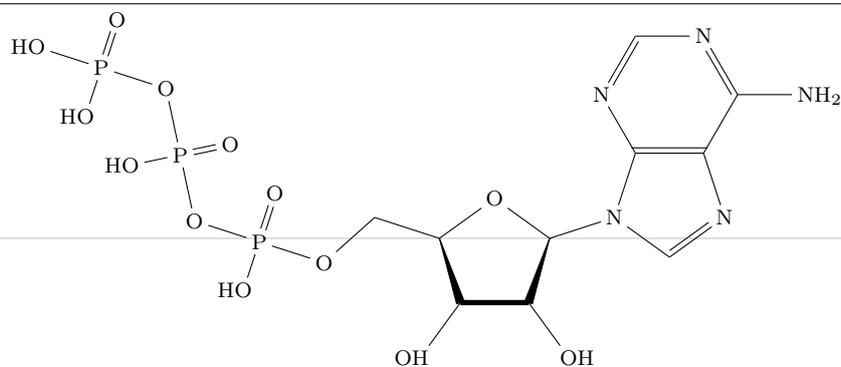
Amygdaline

```
\setcrambond{2pt}{-}{-}
\definesubmol{c1}{-[:200]-[:120]O-[:190]}
\definesubmol{c2}{-[:170](-[:200,0.7]HO)<[:300](-[:170,0.6]HO)
-[:10,,,line width=2pt](-[:40,0.6]OH)>[:10]}
\definesubmol{csub}{-[:155,0.65]-[:90,0.65]}
\chemfig{O(!{c1}(!{csub}O(!{c1}(!{csub}OH)!{c2}))!{c2})-[:30](-[:90]CN)-[:30]*6(==--=)}
```



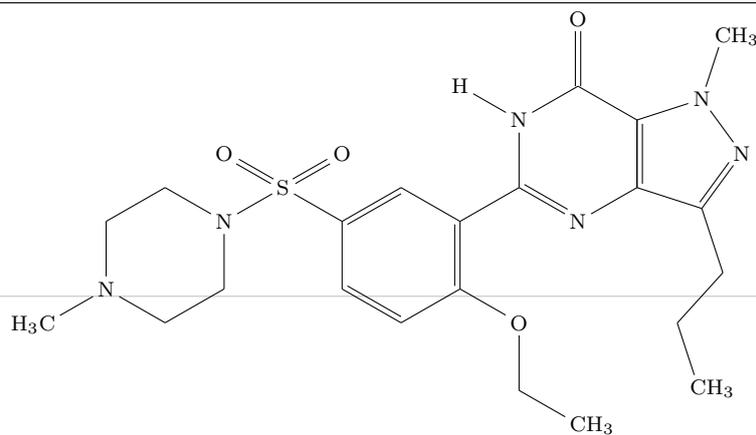
Adénosine triphosphate

```
\setcrambond{3pt}{-}{-}
\definesubmol{a}{-P(=[:90,0.75]O)(-[:90,0.75]HO)-}
\chemfig{[:54]*5((-[:60]O([::-60]!aO([::-60]!aO([::-60]!aHO))))<(-OH)
-[,,,,line width=2pt](-OH)>(-N*5(-N-*6(-(-NH_2)=N=-N-)-)-)-0-)}
```



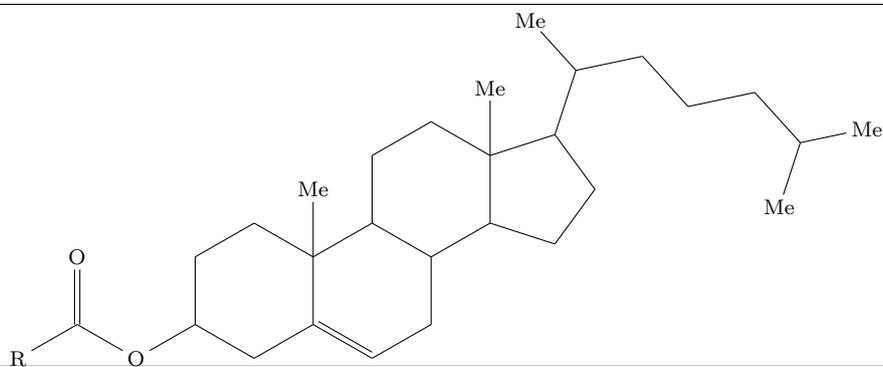
Viagra

```
\chemfig{N*6((-H_3C)---N(-S(=[::+120]O)(=[::+0]O)-[::-60]*6(==(-O-[::-60]-[::+60]CH_3)
=(-*6(=N-*5(-(--[::-60]-[::+60]CH_3)=N-N(-CH_3)=)--(=O)-N(-H)-))---))}
```



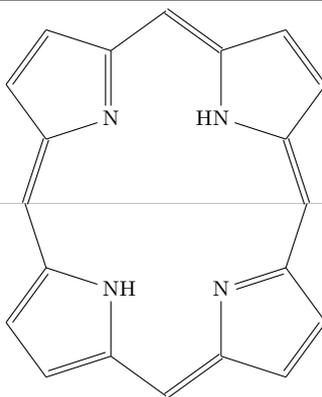
Ester de cholestérol

```
\chemfig{[:30]R-(=[::+60]O)-[::-60]O-*6(--*6(==*6(-*5(---(-[::+60]Me)
-[::-60]-[::-60]-[::+60]-[::-60](-[::-60]Me)-[::+60]Me)-(-[::+0]Me)---)))-(-[::+0]Me)---)}
```



Porphyrine

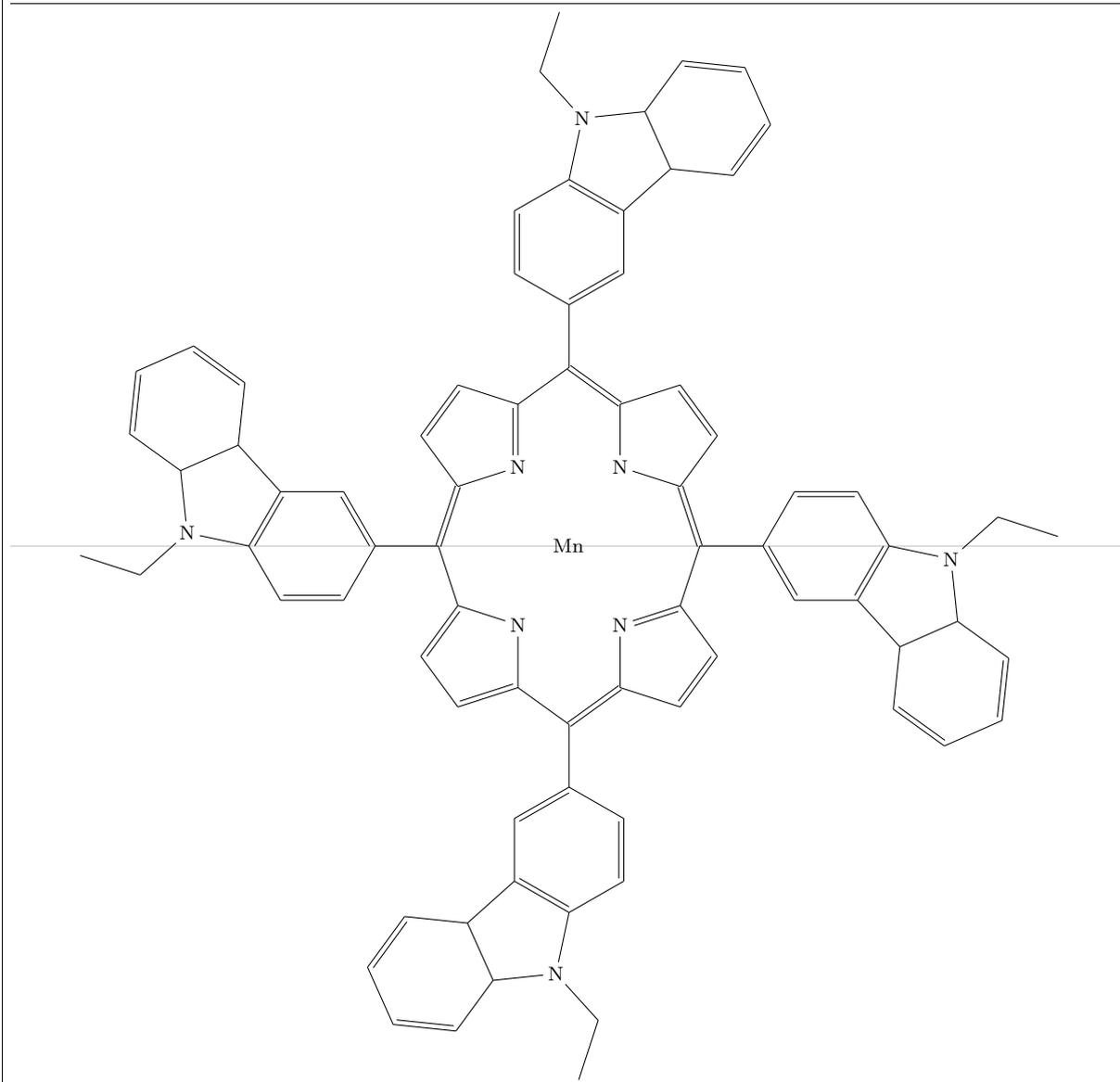
```
\chemfig{?=[::+72]*5(-N(=-[::-72]*5(-[, , , 2]HN-[ , , 2](-=[::-36]*5(=N(-=[::-72]*5(-NH-[ , , 1]?=-))
=-)))-=-))=-)}
```



Manganese 5,10,15,20-tetra(N-ethyl-3-carbazolyl) porphyrine

```
\definesubmol{A}{*6(=*5(*6(==*)--N(--[: -60])--*)--*)}
\chemfig{([: +180]!A)=[: +72]*5(-N(-([: +54]!A)=[: -72]*5(-N(-[: -33,1.5,,draw=none]Mn)
-(=([: +72]!A)-[: -36]*5(=N(=([: +54]!A)-[: -72]*5(-N(-)=))--))--))--)}

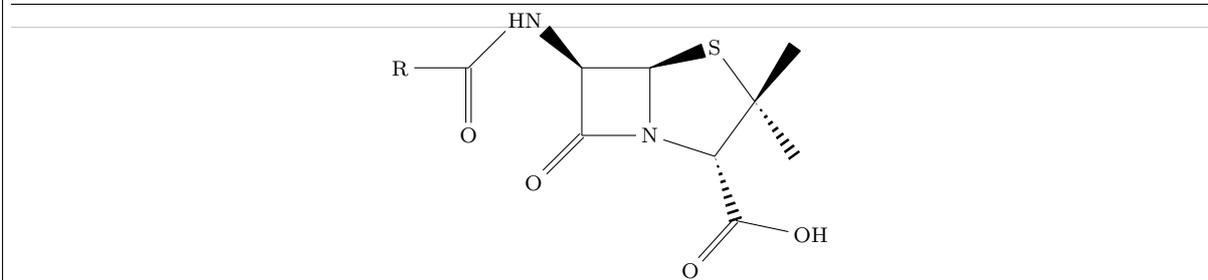
```



Pénicilline

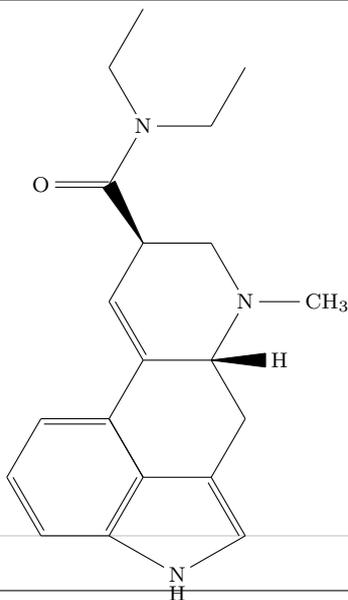
```
\chemfig{[: -90]HN(-[: -45](-[: -45]R)=[: +45]O)>[: +45]*4(-=O)-N*5(-(<[: -60]O)
-[: +60]OH)-(<[: +0])(<[: -108]-S>--)}

```

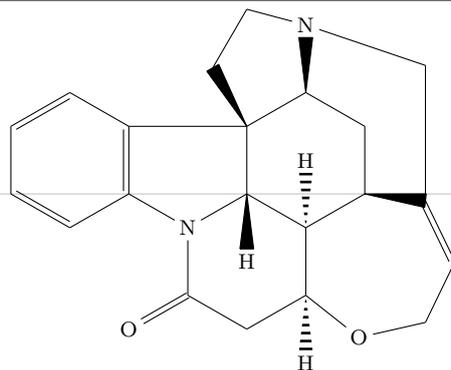


LSD

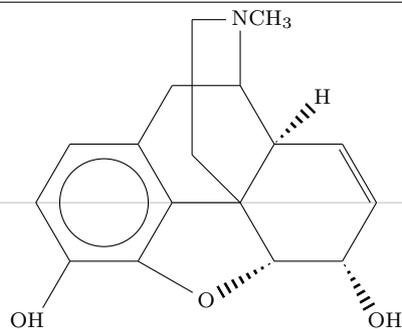
```
\chemfig{[:150]?*6(=*6(--*6(-N(-CH_3)--(<[::+60]O)-[::-60]N(-[:+60]-[::-60])
-[::-60]-[:+60])=>([::-120]<H)---)*6(---(-[::-30,1.155]\chembelow{N}{H}?=))}
```

**Strychnine**

```
\chemfig{*6(=*6(-N*6(-=O)--([::-120]<:H)*7(-0---?[0]([::-25.714]-[,2]?[1]))
-*6(-?[0,{>}--(<N?[1]?[2])-([::-90]-[::-60]?[2]))(<[:+0]H)-([::+120]<H)--?)=?---)}
```

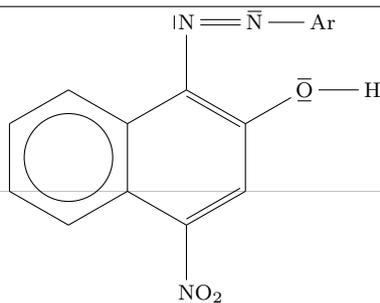
**Codéine**

```
\chemfig{[: -30]**6(-(-OH)-?*6(-[-3]-[2,2]-[0,.5])*6(-(<[: -150,1.155]O?)
-(<:OH)=)-(<:[1]H)-(-[2]NCH_3)---)}
```



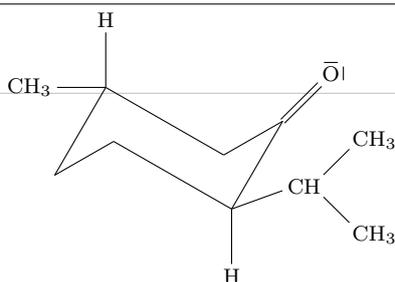
Colorant(rouge)

```
\chemfig{*6(--*6(-(-NO_2)=(-\lewis{26,0}-[0]H)=(-\lewis{4,N}=[0]\lewis{2,N}-[0]Ar)-)----)}
```



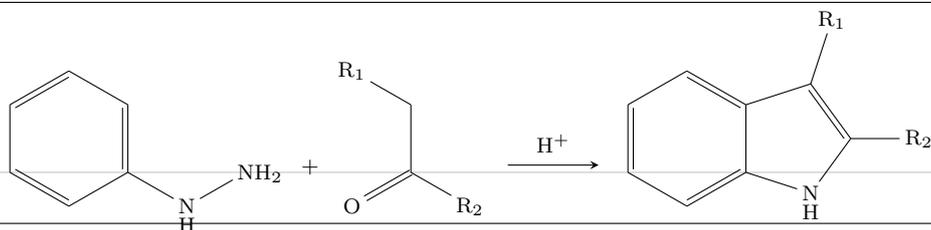
Menthone

```
\chemfig{CH_3-?(-[2]H)(-[::-30,2]-[::+60](=[1]\lewis{20,0})-[::-150,1.5](-[:20]CH(-[1]CH_3)(-[7]CH_3))(-[6]H)-[::-90,2]-[::+60]?)}
```



Synthèse de Fischer de l'indole

```
\chemfig{*6(=*6(-\chembelow{N}{H}-NH_2)=---)}
\chemsign+
\chemfig{([:-150]O)(-[::-30]R_2)-[2]-[:150]R_1}
\chemrel{[\mathrm{H^+}]}{\stealth}
\chemfig{*6(=*5(-\chembelow{N}{H})-(R_2)=(R_1)-)=-)}
```



Mécanisme réactionnel : groupement carbonyle

```
\chemfig{C([3-])([5-])=[@{db,.5}]@{atoo}\lewis{06,0}}
\chemrel{<>}
\chemfig{\chemabove{C}{\scriptstyle\oplus}([3-])([5-])-\chemabove{\lewis{026,0}}{\hspace{5mm}\scriptstyle\ominus}}
\chemmove{\draw[->,shorten <=2pt,shorten >=2pt](db)..controls +(up:5mm) and +(up:5mm)..(atoo);}
```

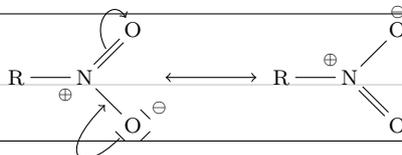


Mécanismes réactionnels : dérivés nitrés

```

\chemfig{R-\chembelow{N}{\hspace{-5mm}\scriptstyle\oplus}([\@{db}]@{atoo1}O)([7]-[\@{sb}]@{atoo2}
\chemabove{\lewis{157,0}}{\hspace{7mm}\scriptstyle\ominus})}
\chemrel{<->}
\chemfig{R-\chemabove{N}{\hspace{-5mm}\scriptstyle\oplus}([\@{db}]@{atoo1}O)([7]=O)}
\chemmove{
\draw[->,shorten <=2pt, shorten >=2pt](db) ..controls +(120:5mm) and +(120:5mm)..(atoo1);
\draw[->,shorten <=3pt, shorten >=2pt](atoo2) ..controls +(225:10mm) and +(225:10mm)..(sb);
}

```

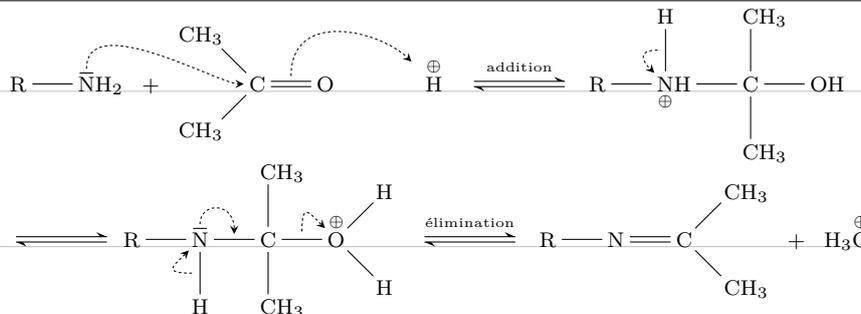


Addition nucléophile. Amines primaires

```

\chemfig{R-\@{aton}\lewis{2,N}H_2}
\chemsign{+}
\chemfig{@{atoc}C([3]-CH_3)([5]-CH_3)=[\@{atoo1}]O}
\hspace{1cm}
\chemfig{@{atoo2}\chemabove{H}{\scriptstyle\oplus}}
\chemmove[-stealth,shorten <=3pt,dash pattern= on 1pt off 1pt,thin]{
\draw[shorten >=2pt](aton) ..controls +(up:10mm) and +(left:5mm)..(atoc);
\draw[shorten >=8pt](atoo1) ..controls +(up:10mm) and +(north west:10mm)..(atoo2);}
\chemrel[\tiny addition]{<>}
\chemfig{R-\@{aton}\chembelow{N}{\scriptstyle\oplus}H([2]-[\@{sb}]H)-C(-[2]CH_3)(-[6]CH_3)-OH}
\par
\chemmove{
\draw[-stealth,dash pattern= on 1pt off 1pt,shorten <=3pt, shorten >=2pt]
(sb)..controls +(left:5mm) and +(135:2mm)..(aton);}
\chemrel{<>}
\chemfig{R-\@{aton}\lewis{2,N}([\@{sbh}]H)-[\@{sb}]C(-[2]CH_3)(-[6]CH_3)-[\@{sbo}]@{atoo}}
\chemabove{O}{\scriptstyle\oplus}(-[1]H)(-[7]H)}
\chemmove[-stealth,shorten <=3pt,shorten >=2pt,dash pattern= on 1pt off 1pt,thin]{
\draw(aton) ..controls +(up:5mm) and +(up:5mm)..(sb);
\draw(sbh) ..controls +(left:5mm) and +(south west:5mm)..(aton);
\draw(sbo) ..controls +(up:5mm) and +(north west:5mm)..(atoo);}
\chemrel[\tiny élimination]{<>}
\chemfig{R-N=C(-[1]CH_3)(-[7]CH_3)}
\chemsign{+}
\chemfig{H_3\chemabove{O}{\scriptstyle\oplus}}

```

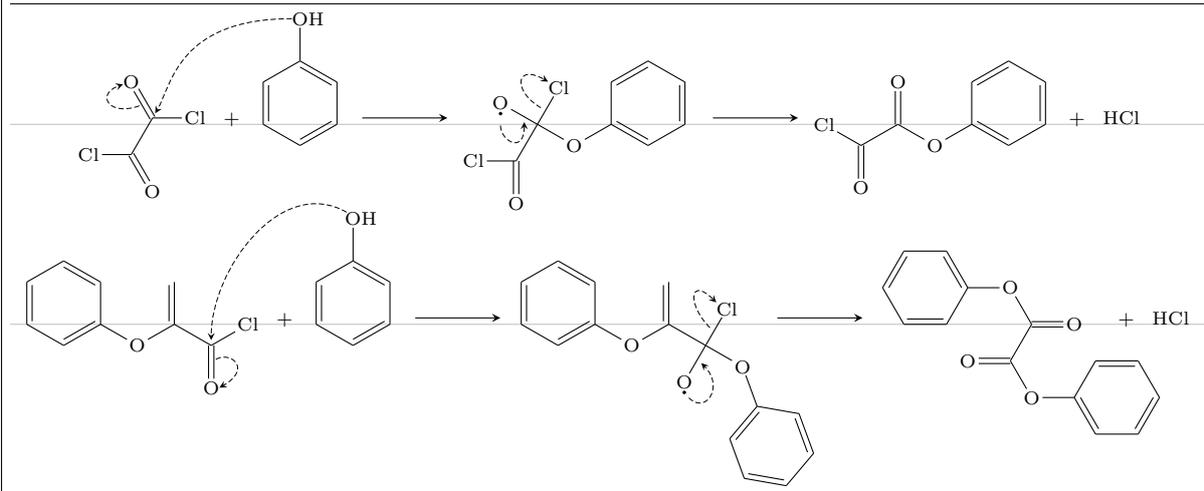


Mécanismes réactionnels de la chloration

```

\scriptsize
\setbondoffset{1pt}\setatomsep{2em}
\chemfig{Cl-[4]@{a0}{=[@{a1}:120]@{a2}O)-[: -120] (=[: -60]O)-[4]Cl}
\chemsign{+}
\chemfig{*6(==(-@{oh1}OH)-=)}
\chemrel{-stealth}
\chemfig{*6((-O-[:150](-@{o0}:150)@{o1}\lewis{6.,0})(-@{c10}:60)@{c11}Cl)-[:240](-[4]Cl)=[6]O)====)}
\chemrel{-stealth}
\chemfig{*6((-O-[:150](=[2]O)-[: -150]([6]O)-[:150]Cl)====)}
\chemsign{+}
\chemfig{HCl}
\par
\chemfig{*6(==*6(-O-*6(-@{o2}(=[@{o3}]@{o4}O)-Cl)=)====)}
\chemsign{+}
\chemfig{*6(==(-@{oh2}OH)-=)}
\chemrel{-stealth}
\chemfig{*6(==*6(-O-(-@{c12}:60)@{c13}Cl)-(@{o5}:-120)@{o6}\lewis{6.,0})-O-[: -40]*6(====))====)}
\kern-3em \chemrel{-stealth}
\chemfig{[:30]*6(=-(-O-[: -60](=O)-[: -120]([4]O)-[: -60]O-*6(====))====)}
\kern-3em \chemsign{+}
\chemfig{HCl}
\chemmove[line width=0.2pt,-stealth,dash pattern = on 2pt off 1pt]{
  \draw[shorten <=2pt](a1)..controls+(200:5mm)and+(200:5mm)..(a2);
  \draw[shorten >=2pt](oh1.west)..controls+(180:15mm)and+(60:5mm)..(a0);
  \draw[shorten <=6pt,shorten >=2pt](o1)..controls+(270:5mm)and+(270:5mm)..(o0);
  \draw[shorten <=2pt](c10)..controls+(150:5mm)and+(150:5mm)..(c11.150);
  \draw[shorten <=2pt](o3)..controls+(30:3mm)and+(30:5mm)..(o4.east);
  \draw[shorten >=2pt](oh2.135)..controls+(150:10mm)and+(90:10mm)..(o2);
  \draw[shorten >=2pt,shorten <=5pt]([xshift=-1.5mm]o6.315)..controls+(315:5mm)and+(315:5mm)..(o5);
  \draw[shorten <=2pt](c12)..controls+(135:5mm)and+(135:5mm)..(c13.north west);}

```

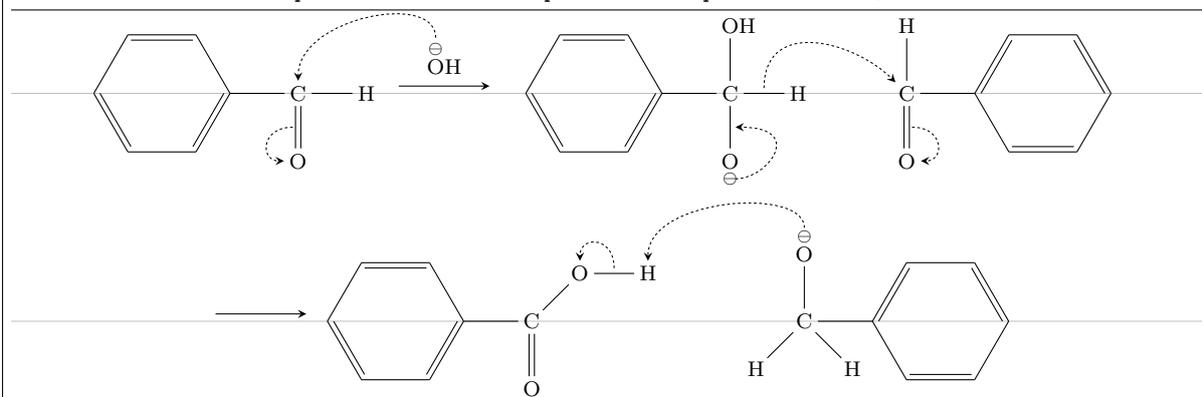


Cannizzaro

```

\chemfig{[: -30]*6(==(-@{atoc}C([6]=[@{db}]@{atoo1}0)-H)-==)}
\chemrel[\chemfig{@{atoo2}\chemabove{0}{\scriptstyle\ominus}}H]{-stealth}
\chemmove[-stealth,shorten >=2pt,dash pattern=on 1pt off 1pt,thin]{
  \draw[shorten <=8pt](atoo2) ..controls +(up:10mm) and +(up:10mm)..(atoc);
  \draw[shorten <=2pt](db) ..controls +(left:5mm) and +(west:5mm)..(atoo1);}
\chemfig{[: -30]*6(==(-C([6]-[@{sb1}]@{atoo1}\chembelow{0}{\scriptstyle\ominus})([2]-OH)-[@{sb2}]H)-==)}
\hspace{1cm}
\chemfig{[: -30]*6((-@{atoc}C([6]=[@{db}]@{atoo2}0)-[2]H)-==)}
\chemmove[-stealth,shorten <=2pt,shorten >=2pt,dash pattern=on 1pt off 1pt,thin]{
  \draw([yshift=-4pt]atoo1.270) ..controls +(0:5mm) and +(right:10mm)..(sb1);
  \draw(sb2) ..controls +(up:10mm) and +(north west:10mm)..(atoc);
  \draw(db) ..controls +(right:5mm) and +(east:5mm)..(atoo2);}
\vspace{1cm}
\par
\chemrel{-stealth}
\chemfig{[: -30]*6(==(-C([1]-@{atoo2}0-[@{sb}0]@{atoh}H)([6]=0)-==)} \hspace{1cm}
\chemfig{[: -30]*6((-C(-[5]H)(-[7]H)-[2]@{atoo1}\chemabove{0}{\scriptstyle\ominus})-==)}
\chemmove[-stealth,shorten >=2pt,dash pattern=on 1pt off 1pt,thin]{
  \draw[shorten <=7pt](atoo1.90) ..controls +(90:8mm) and +(up:10mm)..(atoh);
  \draw[shorten <=2pt](sb) ..controls +(up:5mm) and +(up:5mm)..(atoo2);}

```

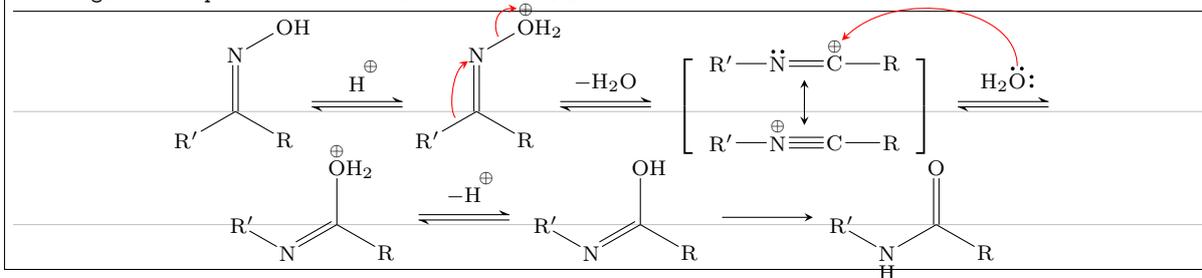


Réarrangement de Beckmann

```

\setbondoffset{1pt}
\setatomsep{2.5em}
\chemfig{(-[: -150]R')(-[: -30]R)=[2]N-[:30]OH}
\kern-1em
\chemrel[\chemfig{\chemabove{H}{\kern10pt\scriptstyle\oplus}}]{<>}
\kern-0.5em
\chemfig{(-[@{a0}:-150]R')(-[: -30]R)=[2]@{a1}N-[@{b0}:30]@{b1}\chemabove{O}{\scriptstyle\oplus}H_2}
\chemmove[red,-stealth,red,shorten <=2pt]{
  \draw(a0)..controls +(135:2mm) and +(215:4mm).. (a1);
  \draw(b0)..controls +(120:2mm) and +(180:3mm).. ([yshift=7pt]b1.180);
}
\kern-1.5em
\chemrel[\chemfig{-H_2O}]{<>}
\left[\begin{tabular}{c}
  \chemfig{R'-\lewis{2:,N}=@{a1}\chemabove{C}{\scriptstyle\oplus}-R}\backslash
  \tikz\draw[stealth-stealth](0,3mm)--(0,-3mm);\backslash
  \chemfig{R'-\chemabove{N}{\scriptstyle\oplus}-C-R}
\end{tabular}\right]
\chemrel[\chemfig{H_2O@{a0}\lewis{0:2:,O}}]{<>}\par
\chemmove[red,-stealth,red,shorten <=3pt]{
  \draw(a0)..controls+(90:10mm)and+(45:10mm)..([yshift=6pt]a1.45);
}
\chemfig{*6(R\rlap{\$'\$}-N=(-R)-\chemabove{O}{\scriptstyle\oplus} H_2)}
\chemrel[\chemfig{-}\chemabove{H}{\kern10pt\scriptstyle\oplus}]{<>}
\chemfig{*6(R\rlap{\$'\$}-N=(-R)-OH)}
\chemrel{-stealth}
\chemfig{*6(R\rlap{\$'\$}-\chembelow{N}{H}-(-R)(=[2]O))}

```



Sulfonation de la naphthalène

```

\newcommand\putchemnode[1]{\begin{tikzpicture}[remember picture]\node(#1){};\end{tikzpicture}}
\definesubmol\cycleoplus{-[,0.25,,draw=none]\oplus}
\definesubmol{so2oh}{S(=[:90]O)(=[::-90]O)-OH}
\footnotesize
\setatomsep{2.5em}
\begin{tabular}{l}
&\&\putchemnode{11}\chemname{\chemfig{*6(=*6(--(!\cycleoplus)-(-SO_3H)-=--)}}}{Ion 1-arenium}
&\chemrel{-stealth}
&\chemname{\chemfig{*6(=*6(---(-!\so2oh)-)=--)}}{Acide 1-naphthalenesulfonique}\\
&\chemname{\chemfig{*6(=*6(---)=--)}}{Naphtalène}
&\chemsign{+}
&\chemfig{H_2SO_4}\putchemnode{a}\kern1cm&\[-5ex]
&\&\putchemnode{12}\chemname{\chemfig{*6(=*6(--(-SO_3H)-(!\cycleoplus)-)=--)}}{Ion 2-arenium}\kern-3em
&\chemrel{-stealth}
&\chemname{\chemfig{*6(=*6(--(-!\so2oh)-)=--)}}{Acide 2-naphthalenesulfonique}
\end{tabular}
\chemmove[-stealth,yshift=1cm]{
\draw(a)--node[sloped,above]{80\degre C}(11);\draw(a)--node[sloped,above]{160\degre C}(12);}

```

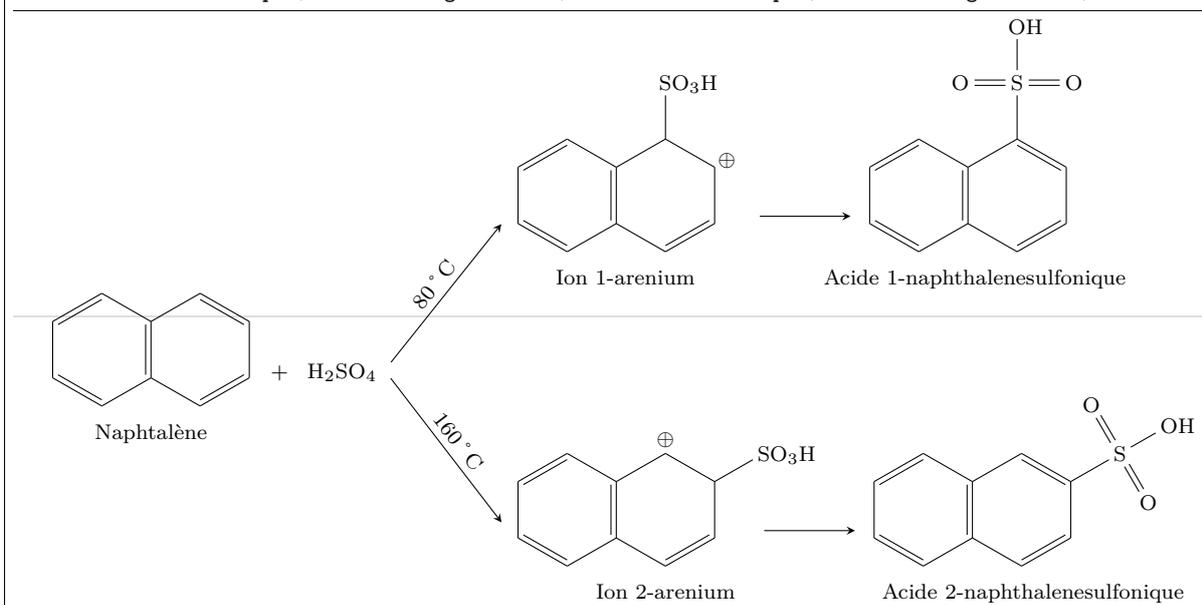
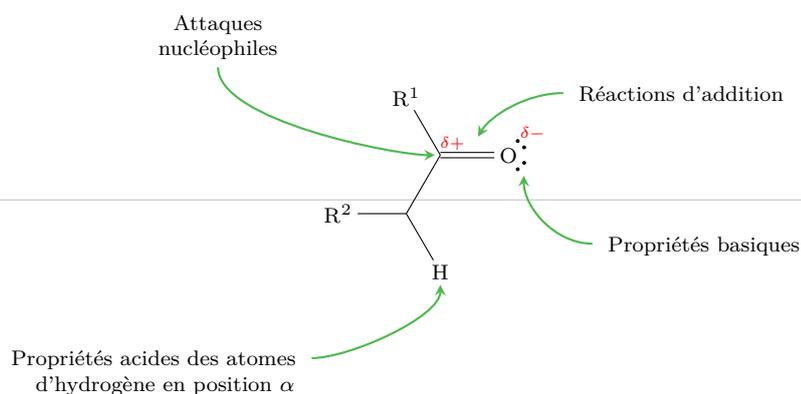


Schéma explicatif

```

\parbox{0.8\linewidth}{
\hspace{10em}
\tikz[remember picture]\node(n0){\chemname{}}{Attaques\nucléophiles}};\par
\vspace{2ex}\hspace{15em}
\chemfig{R^2-(-[:60]@{m3}H)-[:60]@{m0}(-[:120]R^1)(-[:1,0.25,,,draw=none]\scriptstyle\color{red}\delta+)=@{m1}@{m2}\lewis{1:7:,0}-[:1,0.5,,,draw=none]\scriptstyle\color{red}\delta-}}
\hspace{5em}
\chemname[-15ex]{}\tikz[remember picture]\node(n1){};Réactions d'addition}\kern1em
\chemname{}\tikz[remember picture]\node(n2){};Propriétés basiques}\par
\vspace{6ex}\hspace{8em}
\chemname{}\tikz[remember picture]\node(n3){};
  \d'hydrogène en position $\alpha$
\chemmove[-stealth,line width=0.8pt,green!60!black!70]{
  \draw[shorten >=2pt](n0)..controls+(270:4em)and+(180:2em)..(m0);
  \draw[shorten >=8pt](n1)..controls+(180:2em)and+(60:2em)..(m1);
  \draw[shorten >=5pt](n2)..controls+(180:2em)and+(270:2em)..([xshift=3pt]m2.315);
  \draw[shorten >=2pt](n3)..controls+(0:2em)and+(270:2em)..(m3);
}}

```

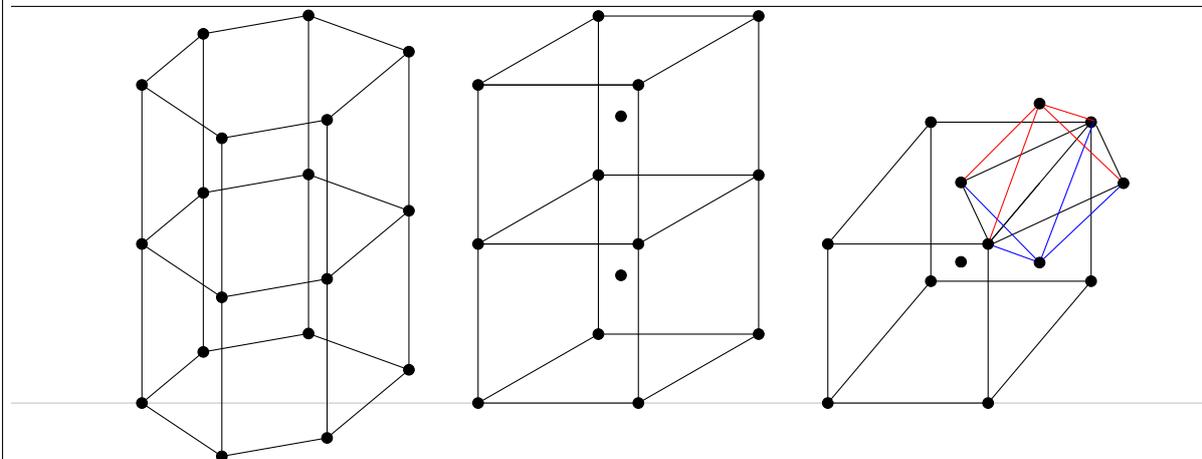


Cristallographie

```

\newcommand\disk{\tikz\draw[fill=black,overlay](0,0)circle(2pt);}
\setatomsep{20pt}
\renewcommand\printatom[1]{#1}
\setbondoffset{2pt}
\definesubmol\hat{-[:40,1.5]\disk-[:-30,2]\disk-[:-30,2]\disk-[:-120,2]\disk-[:-30,2]\disk}
\definesubmol\motif{-[:40,1.5]\disk(-[2,3])-[:-30,2]\disk
(-[2,3])-[:-30,2]\disk(-[2,3])-[:-120,2]\disk(-[2,3])-[:-30,2]\disk}
\chemfig{\disk?{a}(-[2,3]\disk?{b}(-[2,3]\disk?{c}!\hat}?{c})!\motif}?{b}(-[2,3])!\motif}?{a}(-[2,3])}
\qqquad
\redefinesubmol\motif{\disk(-[2,3])(-[:42,3.6,,draw=none]\disk)-[:30,2.6]\disk
(-[2,3])-[0,3]\disk(-[2,3])-[:-150,2.6]\disk(-[4,3])-[2,3]-[4,3]}
\redefinesubmol\hat{\disk-[:30,2.6]\disk-[0,3]\disk-[:-150,2.6]\disk-[4,3]}
\chemfig{!\motif}!\motif}!\hat}}
\qqquad
\redefinesubmol\motif{(-[2,3])(-[:25,2.75,,white]-[2,1.5,,white]\disk)-[:50,3]\disk
(-[2,3])-[:-50,3]\disk(-[2,3])-[:-130,3]\disk(-[2,3])-[4,3]\disk}
\redefinesubmol\hat{-[:50,3]\disk-[:-50,3]\disk-[:-130,3]\disk}
\chemfig{-[4,3]\disk!\motif}(-[:25,2.75,,draw=none]\disk?[uat]?[dat](-[:0,2.75]?[uat1]?[dat1]
-[:-90,1.3]\disk?[uat2]?[dat2])-[:-90,1.3]-[:25,2.75]!\hat}?[uat3]?[dat3]-[:50,1.5]
(-[6,1.5,,draw=none]\disk?[dat,,blue]?[dat1,,blue]?[dat2,,blue]?[dat3,,blue])
-[:2,1.5,,draw=none]\disk?[uat,,red]?[uat1,,red]?[uat2,,red]?[uat3,,red]}

```



Taxotère

```

\chemfig{-[:-30](-[5])(-[7])-[:+60]-[:-60]0-[:+60](=[:-45]0)-[:+90]HN>[:-60](-[:+60]**6(-----))
-[:-30](<[:2]OH)-[:-60](=[6]0)-[:+60]0>[:-60]*7(---?(<[:-120]OH)-(<[1]CH_3)(<[:-90]CH_3)
-([1](<[:+80]HO)-[0](=[:+60]0)-[7](<[[:+130]CH_3)-[:+75](<[2]OH)-[:-60]-[:-60](<[:+30]0-[:-90])
-[:-60](<[:+90])(<[:+30]0-[7](-[6]CH_3)=[0]0)-[:-60])-[6]-[5,1.3]?(<[7]0-[5](=[:-60]0)
-[6]**6(-----))=(-[2]CH_3-)}

```

