Calcul des propositions

Ivan Noyer

Lycée Thiers

Calcul des propositions

Ivan Noyer

Lycée Thiers

Crédits

- Wikipedia ici,
- « Option informatique MPSI, MP/MP* », Roger Mansuy, paru chez Vuibert.

Proposition

- Notion affinée au cours des siècles.

Proposition

- Notion affinée au cours des siècles.
- Une *proposition* est une construction syntaxique censée parler de *vérité*. La définition précise de ce concept est l'un des objectifs du *calcul des propositions*.

Proposition

- Notion affinée au cours des siècles.
- Une *proposition* est une construction syntaxique censée parler de *vérité*. La définition précise de ce concept est l'un des objectifs du *calcul des propositions*.
- ≪ Pourvu que je n'attrape pas le Coronavirus ! ≫ exprime un souhait et pas un fait. Il n'est pas modélisé par une proposition.

Proposition

- Notion affinée au cours des siècles.
- Une *proposition* est une construction syntaxique censée parler de *vérité*. La définition précise de ce concept est l'un des objectifs du *calcul des propositions*.
- « Pourvu que je n'attrape pas le Coronavirus ! » exprime un souhait et pas un fait. Il n'est pas modélisé par une proposition.
- Il n'y a pas de *quantification* en calcul des propositions. Ainsi « M. Noyer marche sur les eaux » est exprimable comme une proposition mais pas « tous les marseillais marchent sur les eaux » ni « il existe un marseillais qui marche sur les eaux » (d'ailleurs c'est M. Noyer !)

Proposition

- Notion affinée au cours des siècles.
- Une *proposition* est une construction syntaxique censée parler de *vérité*. La définition précise de ce concept est l'un des objectifs du *calcul des propositions*.
- ≪ Pourvu que je n'attrape pas le Coronavirus ! ≫ exprime un souhait et pas un fait. Il n'est pas modélisé par une proposition.
- Il n'y a pas de *quantification* en calcul des propositions. Ainsi ≪ M. Noyer marche sur les eaux ≫ est exprimable comme une proposition mais pas ≪ tous les marseillais marchent sur les eaux ≫ ni ≪ il existe un marseillais qui marche sur les eaux ≫ (d'ailleurs c'est M. Noyer !)
- Le *calcul des propositions* est la première étape dans la définition de la logique et du raisonnement. Étape suivante : *calcul des prédicats*.

Contenu des propositions

- Une proposition donne une information *évaluable* sur quelque chose.

Contenu des propositions

- Une proposition donne une information *évaluable* sur quelque chose.
- Exemple : $2+2=4$ ou « la maison de Pierre est rose » ou « si $a \leq b$ et $f(a)f(b) < 0$ alors il existe $c \in]a, b[$ tel que $f(c) = 0$ ».

Contenu des propositions

- Une proposition donne une information *évaluable* sur quelque chose.
- Exemple : $2+2=4$ ou « la maison de Pierre est rose » ou « si $a \leq b$ et $f(a)f(b) < 0$ alors il existe $c \in]a, b[$ tel que $f(c) = 0$ ».
- L'idée est qu'on puisse attribuer une *valeur de vérité* à toute proposition.
  Par exemple $2+2=4$ a pour valeur **Vrai**, la seconde phrase est vérifiable en se rendant chez Pierre, la dernière dépend du type et de la continuité de f.

Contenu des propositions

- Une proposition donne une information *évaluable* sur quelque chose.
- Exemple : $2+2=4$ ou « la maison de Pierre est rose » ou « si $a \leq b$ et $f(a)f(b) < 0$ alors il existe $c \in]a, b[$ tel que $f(c) = 0$ ».
- L'idée est qu'on puisse attribuer une *valeur de vérité* à toute proposition.
  Par exemple $2+2=4$ a pour valeur **Vrai**, la seconde phrase est vérifiable en se rendant chez Pierre, la dernière dépend du type et de la continuité de f.
- Contre-exemple « À quelle heure finit le cours ? » ou « Travaille davantage ! ». Ces phrases n'apportent pas d'information.
  Il est difficile de leur attribuer une valeur.

Contenu des propositions

- Une proposition donne une information *évaluable* sur quelque chose.
- Exemple : $2+2=4$ ou « la maison de Pierre est rose » ou « si $a \leq b$ et $f(a)f(b) < 0$ alors il existe $c \in]a, b[$ tel que $f(c) = 0$ ».
- L'idée est qu'on puisse attribuer une *valeur de vérité* à toute proposition.
  Par exemple $2+2=4$ a pour valeur **Vrai**, la seconde phrase est vérifiable en se rendant chez Pierre, la dernière dépend du type et de la continuité de f.
- Contre-exemple « À quelle heure finit le cours ? » ou « Travaille davantage ! ». Ces phrases n'apportent pas d'information.
  Il est difficile de leur attribuer une valeur.
- Les valeurs attribuées aux propositions sont en général 0 ou 1, **True** ou **False**, **Vrai** ou **Faux**.

Logique classique

- C'est celle du cours de maths en CPGE.

Logique classique

- C'est celle du cours de maths en CPGE.
- Quand le cardinal de l'ensemble des valeurs possibles que peuvent prendre les propositions est 2, on parle de *logique classique* ou *bouléenne*.

Logique classique

- C'est celle du cours de maths en CPGE.
- Quand le cardinal de l'ensemble des valeurs possibles que peuvent prendre les propositions est 2, on parle de *logique classique* ou *bouléenne*.
- Il existe d'autres logiques. Elles sont parfois *tri-valuées*. Par exemple **Vrai, Faux, Je ne sais pas**.

Logique classique

- C'est celle du cours de maths en CPGE.
- Quand le cardinal de l'ensemble des valeurs possibles que peuvent prendre les propositions est 2, on parle de *logique classique* ou *bouléenne*.
- Il existe d'autres logiques. Elles sont parfois *tri-valuées*. Par exemple **Vrai, Faux, Je ne sais pas**.
- Pourquoi faudrait-il d'autres logiques ?
  En logique classique ≪ je suis ici ou ailleurs ≫ (ce qu'on appelle le *tiers exclu*) est toujours vrai. Cependant ≪ si $x \in \mathbb{Q}$ alors machin sinon truc ≫ n'est exploitable que si on possède un test d'appartenance à $\mathbb{Q}$ en temps fini, ce qui n'est pas le cas. Les *logiciens intuitionnistes* refusent le tiers exclu.

Tiers exclu

- Le principe du tiers exclu a été introduit par Aristote comme conséquence du *principe de non-contradiction*. Le principe de non-contradiction stipule que pour toute proposition P on ne peut pas avoir P et $\neg P$ (non P) en même temps.

Tiers exclu

- Le principe du tiers exclu a été introduit par Aristote comme conséquence du *principe de non-contradiction*. Le principe de non-contradiction stipule que pour toute proposition P on ne peut pas avoir P et $\neg P$ (non P) en même temps.
- La proposition $\neg(P \wedge \neg P)$ (toujours vraie dans toute les logiques) équivaut sémantiquement (par utilisation de règle de De Morgan) à $\neg P \vee \neg\neg P$.
  On retrouve le tiers exclu SI ON ACCEPTE que $\neg\neg P$ et P sont sémantiquement équivalents.

D'autres mathématiques

- La *logique intuitionniste* qui engendre les *mathématiques constructives*, admet le principe de non-contradiction mais réfute celui du tiers exclu.

D'autres mathématiques

- La *logique intuitionniste* qui engendre les *mathématiques constructives*, admet le principe de non-contradiction mais réfute celui du tiers exclu.
- Pour un *constructiviste*, le raisonnement $\ll$ si $a \in A$ alors bla sinon bli $\gg$ n'a de sens que s'il existe un test pour déterminer l'appartenance à A.

D'autres mathématiques

- La *logique intuitionniste* qui engendre les *mathématiques constructives*, admet le principe de non-contradiction mais réfute celui du tiers exclu.
- Pour un *constructiviste*, le raisonnement ≪ si $a \in A$ alors bla sinon bli ≫ n'a de sens que s'il existe un test pour déterminer l'appartenance à A.
- Les constructivistes réfutent aussi l'axiome du choix qui dit en substance que ≪ dans un ensemble infini, je peux choisir un élément ≫, au motif que ≪ je ne peux pas choisir si je ne sais pas *comment* choisir ≫.

D'autres mathématiques

- La *logique intuitionniste* qui engendre les *mathématiques constructives*, admet le principe de non-contradiction mais réfute celui du tiers exclu.
- Pour un *constructiviste*, le raisonnement « si $a \in A$ alors bla sinon bli » n'a de sens que s'il existe un test pour déterminer l'appartenance à A.
- Les constructivistes réfutent aussi l'axiome du choix qui dit en substance que « dans un ensemble infini, je peux choisir un élément », au motif que « je ne peux pas choisir si je ne sais pas *comment* choisir ».
- La plupart des théorèmes de CPGE sont adaptables en mathématiques constructives (parfois, les énoncés sont les mêmes, d'autre fois, il faut adapter).
  En revanche l'affirmation « tout $\mathbb{K}$-espace vectoriel admet une base » n'est pas constructive (oui pour la dimension finie mais pas en dimension quelconque).

D'autres mathématiques

- La *logique intuitionniste* qui engendre les *mathématiques constructives*, admet le principe de non-contradiction mais réfute celui du tiers exclu.
- Pour un *constructiviste*, le raisonnement ≪ si $a \in A$ alors bla sinon bli ≫ n'a de sens que s'il existe un test pour déterminer l'appartenance à A.
- Les constructivistes réfutent aussi l'axiome du choix qui dit en substance que ≪ dans un ensemble infini, je peux choisir un élément ≫, au motif que ≪ je ne peux pas choisir si je ne sais pas *comment* choisir ≫.
- La plupart des théorèmes de CPGE sont adaptables en mathématiques constructives (parfois, les énoncés sont les mêmes, d'autre fois, il faut adapter).
- Une preuve de mathématiques constructives est bien adaptée à l'informatique en ce sens qu'elle est pratiquement donnée sous forme d'algorithme.

Structure (d'après Wikipedia)

- Dans les théories de la logique mathématique, on considère deux points de vue dits *syntaxique* et *sémantique*, c'est le cas en calcul des propositions.

Structure (d'après Wikipedia)

- Dans les théories de la logique mathématique, on considère deux points de vue dits *syntaxique* et *sémantique*, c'est le cas en calcul des propositions.
  - La forme : Aspect syntaxique. Il s'agit de définir le langage du calcul des propositions par les règles d'écriture des propositions. On décrit donc ce qu'EST une proposition.

Structure (d'après Wikipedia)

- Dans les théories de la logique mathématique, on considère deux points de vue dits *syntaxique* et *sémantique*, c'est le cas en calcul des propositions.
  - La forme : Aspect syntaxique. Il s'agit de définir le langage du calcul des propositions par les règles d'écriture des propositions. On décrit donc ce qu'EST une proposition.
  - Le fond : Il s'agit de donner un *sens* aux symboles représentant les connecteurs logiques. Il y a deux façons de donner un tel sens :

Structure (d'après Wikipedia)

- Dans les théories de la logique mathématique, on considère deux points de vue dits *syntaxique* et *sémantique*, c'est le cas en calcul des propositions.
  - La forme : Aspect syntaxique. Il s'agit de définir le langage du calcul des propositions par les règles d'écriture des propositions. On décrit donc ce qu'EST une proposition.
  - Le fond : Il s'agit de donner un *sens* aux symboles représentant les connecteurs logiques. Il y a deux façons de donner un tel sens :
    - Soit par la *sémantique* : en munissant d' une *interprétation* (un sens) les connecteurs logique (une fonction à valeur dans un ensemble donné). On peut alors définir inductivement une interprétation pour toute proposition complexe. Ex : table de vérité.

Structure (d'après Wikipedia)

- Dans les théories de la logique mathématique, on considère deux points de vue dits *syntaxique* et *sémantique*, c'est le cas en calcul des propositions.
    - La forme : Aspect syntaxique. Il s'agit de définir le langage du calcul des propositions par les règles d'écriture des propositions. On décrit donc ce qu'EST une proposition.
    - Le fond : Il s'agit de donner un *sens* aux symboles représentant les connecteurs logiques. Il y a deux façons de donner un tel sens :
        - Soit par la *sémantique* : en munissant d' une *interprétation* (un sens) les connecteurs logique (une fonction à valeur dans un ensemble donné). On peut alors définir inductivement une interprétation pour toute proposition complexe. Ex : table de vérité.
        - Soit par *déduction* : On se donne des régles purement syntaxiques qui permettent de déduire une proposition d'un ensemble d'autres : les *règles d'inférence* (Par exemple : *calcul des séquents* ou *déduction naturelle*).

          On ne se préoccupe donc pas du sens, juste de la déductibilité.

Sémantique VS déduction

- La fonction « qui donne du sens » est appelé une *interprétation*.

Sémantique VS déduction

- La fonction ≪ qui donne du sens ≫ est appelé une *interprétation*.
- On construit un *arbre d'inférence* et on dit qu'une formule est *prouvable* si toutes les feuilles de l'arbre sont des *axiomes*.
  On appelle *théorème* une proposition prouvable.

Sémantique VS déduction

- La fonction « qui donne du sens » est appelé une *interprétation*.
- On construit un *arbre d'inférence* et on dit qu'une formule est *prouvable* si toutes les feuilles de l'arbre sont des *axiomes*.
  On appelle *théorème* une proposition prouvable.
- En calcul des propositions, tout ce qui est *vrai* est *prouvable* et réciproquement.

Sémantique VS déduction

- La fonction « qui donne du sens » est appelé une *interprétation*.
- On construit un *arbre d'inférence* et on dit qu'une formule est *prouvable* si toutes les feuilles de l'arbre sont des *axiomes*.
  On appelle *théorème* une proposition prouvable.
- En calcul des propositions, tout ce qui est *vrai* est *prouvable* et réciproquement.
- Si on reste en logique du 1er ordre (en ajoutant quantificateurs, termes et prédicats au cours de 1ere année) alors ce qui est *vrai* est *prouvable* et réciproquement.

Pour avoir mal à la tête

- La logique du 1er ordre (telle qu'enseignée en MP2I/MPI) est cohérente : on ne peut pas démontrer à la fois une formule et son contraire ; ce qui revient à dire qu'on ne peut pas démonrer le faux.

Pour avoir mal à la tête

- La logique du 1er ordre (telle qu'enseignée en MP2I/MPI) est cohérente : on ne peut pas démontrer à la fois une formule et son contraire ; ce qui revient à dire qu'on ne peut pas démonrer le faux.
- Le calcul des prédicats est complet : ce qui est vrai est prouvable et réciproquement.

Pour avoir mal à la tête

- La logique du 1er ordre (telle qu'enseignée en MP2I/MPI) est cohérente : on ne peut pas démontrer à la fois une formule et son contraire ; ce qui revient à dire qu'on ne peut pas démonrer le faux.
- Le calcul des prédicats est complet : ce qui est vrai est prouvable et réciproquement.
- Aucun système logique cohérent (comme la déduction naturelle ou le calcul des séquents) ne peut démontrer (ou infirmer) tous les énoncés de l'arithmétique. Quel que soit le système, s'il est assez riche pour exprimer l'arithmétique, on peut trouver un énoncé valide qui ne sera pas démontrable dans ce système.

Pour avoir mal à la tête

- La logique du 1er ordre (telle qu'enseignée en MP2I/MPI) est cohérente : on ne peut pas démontrer à la fois une formule et son contraire ; ce qui revient à dire qu'on ne peut pas démonrer le faux.
- Le calcul des prédicats est complet : ce qui est vrai est prouvable et réciproquement.
- Aucun système logique cohérent (comme la déduction naturelle ou le calcul des séquents) ne peut démontrer (ou infirmer) tous les énoncés de l'arithmétique. Quel que soit le système, s'il est assez riche pour exprimer l'arithmétique, on peut trouver un énoncé valide qui ne sera pas démontrable dans ce système.
- Aucun système logique cohérent assez riche pour exprimer l'arithmétique ne peut démontrer sa propre cohérence. Par exemple, les mathématiques enséeignées en CPGE ne peuvent pas établir la preuve qu'elles sont sans contradiction.

L'alphabet

Définition

On considère un alphabet Σ constitué :

- de symboles **Vrai** et **Faux** notés **V** et **F**,

Remarque

L'alphabet

Définition

On considère un alphabet Σ constitué :

- de symboles **Vrai** et **Faux** notés **V** et **F**,
- de *variables propositionnelles* en nombre dénombrable notées dans ce cours en lettres romaines $a, b, \dots, a_1, \dots, z_{32}, \dots$

Remarque

L'alphabet

Définition

On considère un alphabet Σ constitué :

- de symboles **Vrai** et **Faux** notés **V** et **F**,
- de *variables propositionnelles* en nombre dénombrable notées dans ce cours en lettres romaines $a, b, \ldots, a_1, \ldots, z_{32}, \ldots$
- de deux symboles de parenthèses « (,) » (ouvrante et fermante).

Remarque

L'alphabet

Définition

On considère un alphabet Σ constitué :

- de symboles **Vrai** et **Faux** notés **V** et **F**,
- de *variables propositionnelles* en nombre dénombrable notées dans ce cours en lettres romaines $a, b, \dots, a_1, \dots, z_{32}, \dots$
- de deux symboles de parenthèses $\ll$ (,) $\gg$ (ouvrante et fermante).
- d'un connecteur unaire $\neg$ dit de *négation*

Remarque

L'alphabet

Définition

On considère un alphabet Σ constitué :

- de symboles **Vrai** et **Faux** notés **V** et **F**,
- de *variables propositionnelles* en nombre dénombrable notées dans ce cours en lettres romaines $a, b, \ldots, a_1, \ldots, z_{32}, \ldots$
- de deux symboles de parenthèses « (,) » (ouvrante et fermante).
- d'un connecteur unaire $\neg$ dit de *négation*
- de trois connecteurs binaires notés $\vee, \wedge, \longrightarrow$ et appelés connecteurs logiques de *disjonction*, *conjonction* et d'*implication*.

Remarque

L'alphabet

Définition

On considère un alphabet Σ constitué :

- de symboles **Vrai** et **Faux** notés **V** et **F**,
- de *variables propositionnelles* en nombre dénombrable notées dans ce cours en lettres romaines $a, b, \ldots, a_1, \ldots, z_{32}, \ldots$
- de deux symboles de parenthèses $\ll$ (,) $\gg$ (ouvrante et fermante).
- d'un connecteur unaire $\neg$ dit de *négation*
- de trois connecteurs binaires notés $\vee, \wedge, \longrightarrow$ et appelés connecteurs logiques de *disjonction*, *conjonction* et d'*implication*.

Remarque

- On dit aussi *opérateur* pour $\ll$ connecteur $\gg$

L'alphabet

Définition

On considère un alphabet Σ constitué :

- de symboles **Vrai** et **Faux** notés **V** et **F**,
- de *variables propositionnelles* en nombre dénombrable notées dans ce cours en lettres romaines $a, b, \dots, a_1, \dots, z_{32}, \dots$
- de deux symboles de parenthèses « (,) » (ouvrante et fermante).
- d'un connecteur unaire $\neg$ dit de *négation*
- de trois connecteurs binaires notés $\vee, \wedge, \longrightarrow$ et appelés connecteurs logiques de *disjonction, conjonction* et d'*implication*.

Remarque

- On dit aussi *opérateur* pour « connecteur »
- Dans certains cours, on ajoute aussi l'opérateur $\iff$ et le XOR. Dans certains autres, seulement $\neg, \vee, \wedge$, voire même $\neg, \wedge$.

L'alphabet

Définition

On considère un alphabet Σ constitué :

- de symboles **Vrai** et **Faux** notés **V** et **F**,
- de *variables propositionnelles* en nombre dénombrable notées dans ce cours en lettres romaines $a, b, \ldots, a_1, \ldots, z_{32}, \ldots$
- de deux symboles de parenthèses « (,) » (ouvrante et fermante).
- d'un connecteur unaire $\neg$ dit de *négation*
- de trois connecteurs binaires notés $\vee, \wedge, \longrightarrow$ et appelés connecteurs logiques de *disjonction*, *conjonction* et d'*implication*.

Remarque

- On dit aussi *opérateur* pour « connecteur »
- Le **NAND** (A NAND B vaut $\neg(A \wedge B)$) est *universel*.

Ensemble des propositions

Définition

On appelle *langage des propositions* le langage défini inductivement par :

- les constantes et les variables propositionnelles sont dans le langage,

Ensemble des propositions

Définition

On appelle *langage des propositions* le langage défini inductivement par :

- les constantes et les variables propositionnelles sont dans le langage,
- si A est dans le langage $(\neg A)$ aussi,

Ensemble des propositions

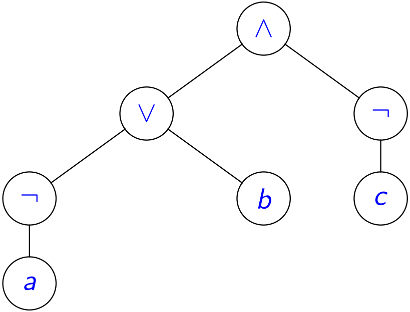
Définition

On appelle *langage des propositions* le langage défini inductivement par :

- les constantes et les variables propositionnelles sont dans le langage,
- si A est dans le langage $(\neg A)$ aussi,
- si A, B sont dans le langage alors $(A \vee B)$, $(A \wedge B)$ et $(A \longrightarrow B)$ aussi.

Représentation arborescente

$(((\neg a) \vee b) \wedge (\neg c))$ se représente sous la forme d'un arbre binaire (dit *arbre syntaxique*) :



Remarque

Puisque les propositions sont construites comme des arbres binaires, il est naturel de parler de leur taille et de leur hauteur.

Simplification de l'écriture

- On ne note pas les parenthèses autour de la racine.

Simplification de l'écriture

- On ne note pas les parenthèses autour de la racine.
- On convient (souvent, mais je ne trouve pas ça si commode) que l'opérateur de négation $\neg$ a priorité sur les autres, que la conjonction et la disjonction ont priorité sur l'implication, et enfin que la conjonction a priorité sur la disjonction.
  C'est très classique !

Simplification de l'écriture

- On ne note pas les parenthèses autour de la racine.
- On convient (souvent, mais je ne trouve pas ça si commode) que l'opérateur de négation $\neg$ a priorité sur les autres, que la conjonction et la disjonction ont priorité sur l'implication, et enfin que la conjonction a priorité sur la disjonction.
  C'est très classique !
- Pour passer des écritures sans parenthèses aux écritures avec parenthèses, une règle simple : plus l'opérateur est prioritaire, plus il a de parenthèses autour de lui.

Priorités en Python

Figure – Règles de priorité de la plus haute (en haut) à la plus basse (en bas). Les opérateurs de même niveau sont évalués de gauche à droite

Operator	Description
`()`	Parentheses (grouping)
`f(args...)`	Function call
`x[index:index]`	Slicing
`x[index]`	Subscription
`x.attribute`	Attribute reference
`**`	Exponentiation
`~x`	Bitwise not
`+x, -x`	Positive, negative
`*, /, %`	Multiplication, division, remainder
`+, -`	Addition, subtraction
`<<, >>`	Bitwise shifts
`&`	Bitwise AND
`^`	Bitwise XOR
`|`	Bitwise OR
`in, not in, is, is not, <, <=, >, >=, <>, !=, ==`	Comparisons, membership, identity
`not x`	Boolean NOT
`and`	Boolean AND
`or`	Boolean OR
`lambda`	Lambda expression

Exemple

Remettre des parenthèses

- $\neg a \vee b \wedge c$

Exemple

Remettre des parenthèses

- $\neg a \vee b \wedge c$
- $((\neg a) \vee (b \wedge c))$

Exemple

Remettre des parenthèses

- $\neg a \vee b \wedge c$
- $((\neg a) \vee (b \wedge c))$
- $\neg a \wedge b \vee c$

Exemple

Remettre des parenthèses

- $\neg a \lor b \land c$
- $((\neg a) \lor (b \land c))$
- $\neg a \land b \lor c$
- $(((\neg a) \land b) \lor c)$

Exemple

Remettre des parenthèses

- $\neg a \vee b \wedge c$
- $((\neg a) \vee (b \wedge c))$
- $\neg a \wedge b \vee c$
- $(((\neg a) \wedge b) \vee c)$
- $a \vee \neg b \wedge c \longrightarrow a \wedge c$

Exemple

Remettre des parenthèses

- $\neg a \vee b \wedge c$
- $((\neg a) \vee (b \wedge c))$
- $\neg a \wedge b \vee c$
- $(((\neg a) \wedge b) \vee c)$
- $a \vee \neg b \wedge c \longrightarrow a \wedge c$
- $[(a \vee ((\neg b) \wedge c)) \longrightarrow (a \wedge c)]$

Contexte

Définition

Un *contexte* (ou une *distribution de vérité*) sur un ensemble de variables propositionnelles $\mathcal{V}$ est une application de $\mathcal{V}$ dans l'ensemble des bouléens $\mathcal{B}$.

Remarque

- Si $|\mathcal{V}| = n$, alors il y a 2^n distributions de vérité.

Contexte

Définition

Un *contexte* (ou une *distribution de vérité*) sur un ensemble de variables propositionnelles $\mathcal{V}$ est une application de $\mathcal{V}$ dans l'ensemble des bouléens $\mathcal{B}$.

Remarque

- Si $|\mathcal{V}| = n$, alors il y a 2^n distributions de vérité.
- L'ensemble des bouléens peut être représenté de différentes façons.

Contexte

Définition

Un *contexte* (ou une *distribution de vérité*) sur un ensemble de variables propositionnelles $\mathcal{V}$ est une application de $\mathcal{V}$ dans l'ensemble des bouléens $\mathcal{B}$.

Remarque

- Si $|\mathcal{V}| = n$, alors il y a 2^n distributions de vérité.
- L'ensemble des bouléens peut être représenté de différentes façons.
- Dans ce cours, on pose $\mathcal{B} = \{0, 1\}$, et on identifie $\mathcal{B}$ avec l'anneau $\mathbb{Z}/2\mathbb{Z}$ (donc $1 + 1 = 0$ en particulier.)

Contexte

Définition

Un *contexte* (ou une *distribution de vérité*) sur un ensemble de variables propositionnelles $\mathcal{V}$ est une application de $\mathcal{V}$ dans l'ensemble des bouléens $\mathcal{B}$.

Remarque

- Si $|\mathcal{V}| = n$, alors il y a 2^n distributions de vérité.
- L'ensemble des bouléens peut être représenté de différentes façons.
- Dans ce cours, on pose $\mathcal{B} = \{0, 1\}$, et on identifie $\mathcal{B}$ avec l'anneau $\mathbb{Z}/2\mathbb{Z}$ (donc $1 + 1 = 0$ en particulier.)
- La multiplication est l'interprétation de la conjonction, 0 celle de **F**, 1 celle de **V**, l'addition celle du XOR.

Interprétation

Définition

Soit μ un contexte sur un ensemble de variables $\mathcal{V}$ à valeur dans $\mathcal{B} = \mathbb{Z}/2\mathbb{Z}$. On appelle *évaluation* (ou encore *interprétation*) associée à μ l'application notée $\mathcal{E}_\mu$ définie sur l'ensemble des propositions par :

- $\mathcal{E}_\mu(V) = 1$, $\mathcal{E}_\mu(F) = 0$

Remarque

Observer l'analogie avec les fonctions caractéristiques.

Interprétation

Définition

Soit μ un contexte sur un ensemble de variables $\mathcal{V}$ à valeur dans $\mathcal{B} = \mathbb{Z}/2\mathbb{Z}$. On appelle *évaluation* (ou encore *interprétation*) associée à μ l'application notée $\mathcal{E}_\mu$ définie sur l'ensemble des propositions par :

- $\mathcal{E}_\mu(V) = 1$, $\mathcal{E}_\mu(F) = 0$
- pour toute variable $v \in \mathcal{V}$, $\mathcal{E}_\mu(v) = \mu(v)$

Remarque

Observer l'analogie avec les fonctions caractéristiques.

Interprétation

Définition

Soit μ un contexte sur un ensemble de variables $\mathcal{V}$ à valeur dans $\mathcal{B} = \mathbb{Z}/2\mathbb{Z}$. On appelle *évaluation* (ou encore *interprétation*) associée à μ l'application notée $\mathcal{E}_\mu$ définie sur l'ensemble des propositions par :

- $\mathcal{E}_\mu(V) = 1$, $\mathcal{E}_\mu(F) = 0$
- pour toute variable $v \in \mathcal{V}$, $\mathcal{E}_\mu(v) = \mu(v)$
- pour toute expression p, $\mathcal{E}_\mu(\neg p) = 1 - \mathcal{E}_\mu(p)$

Remarque

Observer l'analogie avec les fonctions caractéristiques.

Interprétation

Définition

Soit μ un contexte sur un ensemble de variables $\mathcal{V}$ à valeur dans $\mathcal{B} = \mathbb{Z}/2\mathbb{Z}$. On appelle *évaluation* (ou encore *interprétation*) associée à μ l'application notée $\mathcal{E}_\mu$ définie sur l'ensemble des propositions par :

- $\mathcal{E}_\mu(V) = 1$, $\mathcal{E}_\mu(F) = 0$
- pour toute variable $v \in \mathcal{V}$, $\mathcal{E}_\mu(v) = \mu(v)$
- pour toute expression p, $\mathcal{E}_\mu(\neg p) = 1 - \mathcal{E}_\mu(p)$
- pour toutes expressions p_1 et p_2 :

Remarque

Observer l'analogie avec les fonctions caractéristiques.

Interprétation

Définition

Soit μ un contexte sur un ensemble de variables $\mathcal{V}$ à valeur dans $\mathcal{B} = \mathbb{Z}/2\mathbb{Z}$. On appelle *évaluation* (ou encore *interprétation*) associée à μ l'application notée $\mathcal{E}_\mu$ définie sur l'ensemble des propositions par :

- $\mathcal{E}_\mu(V) = 1$, $\mathcal{E}_\mu(F) = 0$
- pour toute variable $v \in \mathcal{V}$, $\mathcal{E}_\mu(v) = \mu(v)$
- pour toute expression p, $\mathcal{E}_\mu(\neg p) = 1 - \mathcal{E}_\mu(p)$
- pour toutes expressions p_1 et p_2 :
    - $\mathcal{E}_\mu(p_1 \wedge p_2) = \mathcal{E}_\mu(p_1)\mathcal{E}_\mu(p_2)$

Remarque

Observer l'analogie avec les fonctions caractéristiques.

Interprétation

Définition

Soit μ un contexte sur un ensemble de variables $\mathcal{V}$ à valeur dans $\mathcal{B} = \mathbb{Z}/2\mathbb{Z}$. On appelle *évaluation* (ou encore *interprétation*) associée à μ l'application notée $\mathcal{E}_\mu$ définie sur l'ensemble des propositions par :

- $\mathcal{E}_\mu(V) = 1$, $\mathcal{E}_\mu(F) = 0$
- pour toute variable $v \in \mathcal{V}$, $\mathcal{E}_\mu(v) = \mu(v)$
- pour toute expression p, $\mathcal{E}_\mu(\neg p) = 1 - \mathcal{E}_\mu(p)$
- pour toutes expressions p_1 et p_2 :
    - $\mathcal{E}_\mu(p_1 \wedge p_2) = \mathcal{E}_\mu(p_1)\mathcal{E}_\mu(p_2)$
    - $\mathcal{E}_\mu(p_1 \vee p_2) = \mathcal{E}_\mu(p_1) + \mathcal{E}_\mu(p_2) - \mathcal{E}_\mu(p_1)\mathcal{E}_\mu(p_2)$

Remarque

Observer l'analogie avec les fonctions caractéristiques.

Interprétation

Définition

Soit μ un contexte sur un ensemble de variables $\mathcal{V}$ à valeur dans $\mathcal{B} = \mathbb{Z}/2\mathbb{Z}$. On appelle *évaluation* (ou encore *interprétation*) associée à μ l'application notée $\mathcal{E}_\mu$ définie sur l'ensemble des propositions par :

- $\mathcal{E}_\mu(V) = 1$, $\mathcal{E}_\mu(F) = 0$
- pour toute variable $v \in \mathcal{V}$, $\mathcal{E}_\mu(v) = \mu(v)$
- pour toute expression p, $\mathcal{E}_\mu(\neg p) = 1 - \mathcal{E}_\mu(p)$
- pour toutes expressions p_1 et p_2 :
  - $\mathcal{E}_\mu(p_1 \wedge p_2) = \mathcal{E}_\mu(p_1)\mathcal{E}_\mu(p_2)$
  - $\mathcal{E}_\mu(p_1 \vee p_2) = \mathcal{E}_\mu(p_1) + \mathcal{E}_\mu(p_2) - \mathcal{E}_\mu(p_1)\mathcal{E}_\mu(p_2)$
  - $\mathcal{E}_\mu(p_1 \longrightarrow p_2) = 1 - \mathcal{E}_\mu(p_1) + \mathcal{E}_\mu(p_2)\mathcal{E}_\mu(p_1)$.

Remarque

Observer l'analogie avec les fonctions caractéristiques.

Implication

- Comme on va le voir $a \longrightarrow b$ est *sémantiquement* équivalent à $\neg a \vee b$, c'est à dire que les deux propositions ont la même valeur de vérité pour toute interprétation.

Implication

- Comme on va le voir $a \longrightarrow b$ est *sémantiquement* équivalent à $\neg a \vee b$, c'est à dire que les deux propositions ont la même valeur de vérité pour toute interprétation.
- L'interprétation de l'implication donnée au transparent précédent s'obtient par calcul :

$$\begin{array}{rcc}
\mathcal{E}_\mu((\neg p_1) \vee p_2) & = & \mathcal{E}_\mu(\neg p_1) + \mathcal{E}_\mu(p_2) - \mathcal{E}_\mu(\neg p_1)\mathcal{E}_\mu(p_2) \\
 & = & (1 - \mathcal{E}_\mu(p_1)) + \mathcal{E}_\mu(p_2) - (1 - \mathcal{E}_\mu(p_1))\mathcal{E}_\mu(p_2) \\
 & = & 1 - \mathcal{E}_\mu(p_1) + \mathcal{E}_\mu(p_2) - \mathcal{E}_\mu(p_2) + \mathcal{E}_\mu(p_1)\mathcal{E}_\mu(p_2) \\
 & = & 1 - \mathcal{E}_\mu(p_1) + \mathcal{E}_\mu(p_2)\mathcal{E}_\mu(p_1) \\
 & = & \mathcal{E}_\mu(p_1 \longrightarrow p_2)
\end{array}$$

Tables de vérité

Les 4 opérateurs de la logique classique

Table du ET		
a	b	$a \wedge b$
1	1	1
1	0	0
0	1	0
0	0	0

Table du OU		
a	b	$a \vee b$
1	1	1
1	0	1
0	1	1
0	0	0

Table du NOT	
a	$\neg a$
1	0
0	1

Table d'implication		
a	b	$a \longrightarrow b$
1	1	1
1	0	0
0	1	1
0	0	1

Equivalence sémantique

Définition

Deux propositions p, q sont dites *sémantiquement équivalentes* si elles ont même table de vérité. Ceci revient à dire que pour tout contexte μ, si ϵ_μ de l'une vaut 1, alors pour l'autre aussi. On note $p \equiv q$.

Définition

On dit qu'une proposition p est une *tautologie* lorsque $p \equiv V$

Remarque

- Par exemple $a \wedge V$ est sémantiquement équivalente à a.

Equivalence sémantique

Définition

Deux propositions p, q sont dites *sémantiquement équivalentes* si elles ont même table de vérité.. On note $p \equiv q$.

Définition

On dit qu'une proposition p est une *tautologie* lorsque $p \equiv V$(c'est à dire que la dernière colonne de sa table de vérité ne contient que des 1).

Remarque

- Par exemple $a \wedge V$ est sémantiquement équivalente à a.
- Montrer que le tiers exclu $a \vee \neg a$ est une tautologie.

Equivalence sémantique

Définition

Deux propositions p, q sont dites *sémantiquement équivalentes* si elles ont même table de vérité.. On note $p \equiv q$.

Définition

On dit qu'une proposition p est une *tautologie* lorsque $p \equiv V$

Remarque

- Par exemple $a \wedge V$ est sémantiquement équivalente à a.
- Montrer que le tiers exclu $a \vee \neg a$ est une tautologie.
- Certains auteurs parlent d'*équivalence logique*.

XOR

Exercice

Définir l'opérateur **XOR** au moyen des opérateurs usuels

Table du XOR OU Exclusif		
a	b	$a \oplus b$
1	1	0
1	0	1
0	1	1
0	0	0

Associativité

- Les opérateurs $\wedge$ et $\vee$ sont associatifs : en conséquence on peut écrire $((a \vee b) \vee (c \vee d))$ comme $a \vee b \vee c \vee d$

Associativité

- Les opérateurs $\wedge$ et $\vee$ sont associatifs : en conséquence on peut écrire $((a \vee b) \vee (c \vee d))$ comme $a \vee b \vee c \vee d$
- $\longrightarrow$ n'est pas associatif : $a \longrightarrow (b \longrightarrow c)$ n'est pas sémantiquement équivalent à $(a \longrightarrow b) \longrightarrow c$

Associativité

- Les opérateurs $\wedge$ et $\vee$ sont associatifs : en conséquence on peut écrire $((a \vee b) \vee (c \vee d))$ comme $a \vee b \vee c \vee d$
- $\longrightarrow$ n'est pas associatif : $a \longrightarrow (b \longrightarrow c)$ n'est pas sémantiquement équivalent à $(a \longrightarrow b) \longrightarrow c$
- Exercice : montrer ces assertions par des tables de vérité.

Associativité

- Les opérateurs $\wedge$ et $\vee$ sont associatifs : en conséquence on peut écrire $((a \vee b) \vee (c \vee d))$ comme $a \vee b \vee c \vee d$
- $\longrightarrow$ n'est pas associatif : $a \longrightarrow (b \longrightarrow c)$ n'est pas sémantiquement équivalent à $(a \longrightarrow b) \longrightarrow c$
- Exercice : montrer ces assertions par des tables de vérité.
- L'implication n'est pas associative mais *par convention* $A \longrightarrow B \longrightarrow C \longrightarrow D$ se lit $A \longrightarrow (B \longrightarrow (C \longrightarrow D))$.
  Observer l'analogie avec une fonction **f** de type **a**$\rightarrow$ **b**$\rightarrow$ **c**$\rightarrow$ **d** en OCAML : **f x** est de type **b**$\rightarrow$ **c**$\rightarrow$ **d**.

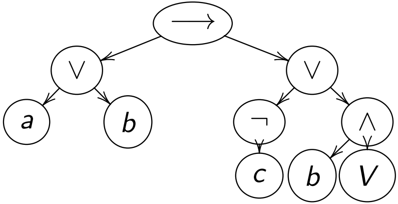
D'autres opérateurs

Exercice

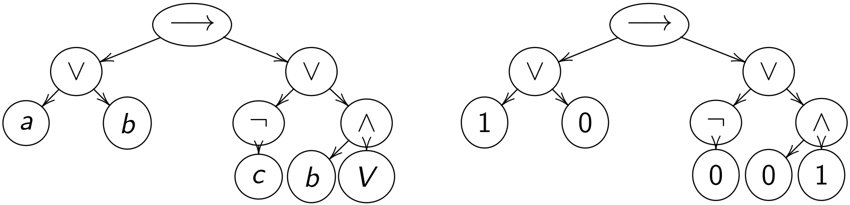
1. Montrer qu'on peut se passer de l'opérateur d'implication.
2. L'opérateur binaire $\iff$ est défini ainsi : $a \iff b$ a la table de $(a \longrightarrow b) \wedge (b \longrightarrow a)$. Montrer que a XOR b est sémantiquement équivalent à $\neg(a \iff b)$.

Calcul à partir de l'arbre syntaxique

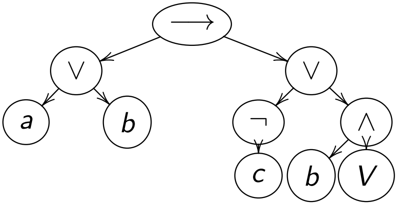
$(a \vee b) \longrightarrow ((\neg c) \vee (b \wedge V))$, $\mu(a) = 1; \mu(b) = \mu(c) = 0$. Commencer par les feuilles et remonter aux pères.

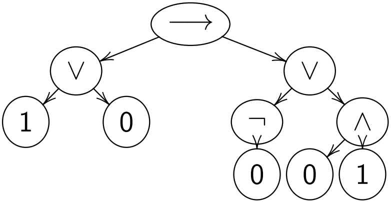


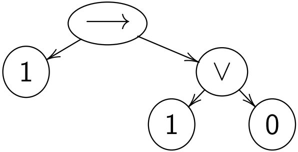
Calcul à partir de l'arbre syntaxique



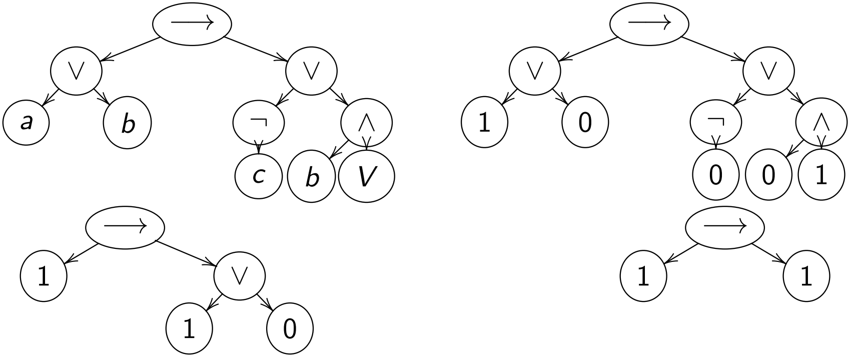
Calcul à partir de l'arbre syntaxique



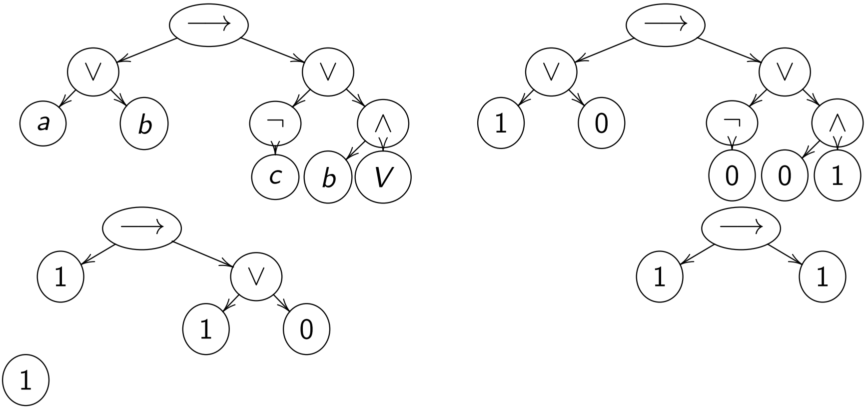




Calcul à partir de l'arbre syntaxique



Calcul à partir de l'arbre syntaxique



Tautologie, proposition satisfiable

Définition

- On appelle *tautologie* toute expression évaluée à 1 dans tout contexte.

Exemple

Tautologie, proposition satisfiable

Définition

- On appelle *tautologie* toute expression évaluée à 1 dans tout contexte.
- Une proposition p est dite *satisfiable* ou *satisfaisable* si il existe un contexte μ tel que $\varepsilon_\mu(p) = 1$. On dit que μ est un *modèle* de p (ou que p est *satisfaite par* μ) et on note $\mu \vDash p$.

Exemple

Tautologie, proposition satisfiable

Définition

- On appelle *tautologie* toute expression évaluée à 1 dans tout contexte.
- Une proposition p est dite *satisfiable* ou *satisfaisable* si il existe un contexte μ tel que $\varepsilon_\mu(p) = 1$. On dit que μ est un *modèle* de p (ou que p est *satisfaite par* μ) et on note $\mu \vDash p$.
- Une proposition qui n'est pas satisfiable est apellée une *antilogie*. On dit aussi que l'expression est *insatisfiable*.

Exemple

Tautologie, proposition satisfiable

Définition

- On appelle *tautologie* toute expression évaluée à 1 dans tout contexte.
- Une proposition p est dite *satisfiable* ou *satisfaisable* si il existe un contexte μ tel que $\varepsilon_\mu(p) = 1$. On dit que μ est un *modèle* de p (ou que p est *satisfaite par* μ) et on note $\mu \models p$.
- Une proposition qui n'est pas satisfiable est apellée une *antilogie*. On dit aussi que l'expression est *insatisfiable*.

Exemple

- Le tiers exclu $a \vee \neg a$ est une tautologie, $(a \wedge b) \longrightarrow b$ aussi.

Tautologie, proposition satisfiable

Définition

- On appelle *tautologie* toute expression évaluée à 1 dans tout contexte.
- Une proposition p est dite *satisfiable* ou *satisfaisable* si il existe un contexte μ tel que $\varepsilon_\mu(p) = 1$. On dit que μ est un *modèle* de p (ou que p est *satisfaite par* μ) et on note $\mu \models p$.
- Une proposition qui n'est pas satisfiable est apellée une *antilogie*. On dit aussi que l'expression est *insatisfiable*.

Exemple

- Le tiers exclu $a \vee \neg a$ est une tautologie, $(a \wedge b) \longrightarrow b$ aussi.
- $a \longrightarrow b$ est satisfiable mais n'est pas une tautologie.

Tautologie, proposition satisfiable

Définition

- On appelle *tautologie* toute expression évaluée à 1 dans tout contexte.
- Une proposition p est dite *satisfiable* ou *satisfaisable* si il existe un contexte μ tel que $\varepsilon_\mu(p) = 1$. On dit que μ est un *modèle* de p (ou que p est *satisfaite par* μ) et on note $\mu \vDash p$.
- Une proposition qui n'est pas satisfiable est apellée une *antilogie*. On dit aussi que l'expression est *insatisfiable*.

Exemple

- Le tiers exclu $a \vee \neg a$ est une tautologie, $(a \wedge b) \longrightarrow b$ aussi.
- $a \longrightarrow b$ est satisfiable mais n'est pas une tautologie.
- $a \wedge \neg a$ est insatisfiable.

Tautologie, proposition satisfiable

Définition

- On appelle *tautologie* toute expression évaluée à 1 dans tout contexte.
- Une proposition p est dite *satisfiable* ou *satisfaisable* si il existe un contexte μ tel que $\varepsilon_\mu(p) = 1$. On dit que μ est un *modèle* de p (ou que p est *satisfaite par* μ) et on note $\mu \models p$.
- Une proposition qui n'est pas satisfiable est apellée une *antilogie*. On dit aussi que l'expression est *insatisfiable*.

Exemple

- Le tiers exclu $a \vee \neg a$ est une tautologie, $(a \wedge b) \longrightarrow b$ aussi.
- $a \longrightarrow b$ est satisfiable mais n'est pas une tautologie.
- $a \wedge \neg a$ est insatisfiable.
- Montrer que $(\neg A \longrightarrow A) \longrightarrow A$ est une tautologie

Notation

- Pour indiquer qu'une proposition a *est sémantiquement équivalente* à une proposition b, il n'est pas suffisant d'écrire $a \iff b$.

Notation

- Pour indiquer qu'une proposition *a est sémantiquement équivalente* à une proposition b, il n'est pas suffisant d'écrire $a \iff b$.
- En effet $a \iff b$ est juste une proposition. Celle-ci peut être satisfiable ou non (peut-être même pas, comme $V \iff F$).

Notation

- Pour indiquer qu'une proposition *a est sémantiquement équivalente* à une proposition b, il n'est pas suffisant d'écrire $a \iff b$.
- En effet $a \iff b$ est juste une proposition. Celle-ci peut être satisfiable ou non (peut-être même pas, comme $V \iff F$).
- La bonne façon d'indiquer cette équivalence est de déclarer « $a \iff b$ est une tautologie ».

Notation

- Pour indiquer qu'une proposition a *est sémantiquement équivalente* à une proposition b, il n'est pas suffisant d'écrire $a \iff b$.
- En effet $a \iff b$ est juste une proposition. Celle-ci peut être satisfiable ou non (peut-être même pas, comme $V \iff F$).
- La bonne façon d'indiquer cette équivalence est de déclarer « $a \iff b$ est une tautologie ».
- Les mathématiciens sont parfois paresseux : il leur arrive d'écrire « $a \iff b$ » au lieu de « $a \iff b$ est *vraie* (ce qui signifie vraie pour tout contexte) ».

Notation

- Pour indiquer qu'une proposition a *est sémantiquement équivalente* à une proposition b, il n'est pas suffisant d'écrire $a \iff b$.
- En effet $a \iff b$ est juste une proposition. Celle-ci peut être satisfiable ou non (peut-être même pas, comme $V \iff F$).
- La bonne façon d'indiquer cette équivalence est de déclarer « $a \iff b$ est une tautologie ».
- Les mathématiciens sont parfois paresseux : il leur arrive d'écrire « $a \iff b$ » au lieu de « $a \iff b$ est *vraie* (ce qui signifie vraie pour tout contexte) ».
- On écrit $a \equiv b$ pour indiquer que $a \iff b$ est une tautologie.

Notation

- Pour indiquer qu'une proposition *a est sémantiquement équivalente* à une proposition b, il n'est pas suffisant d'écrire $a \iff b$.
- En effet $a \iff b$ est juste une proposition. Celle-ci peut être satisfiable ou non (peut-être même pas, comme $V \iff F$).
- La bonne façon d'indiquer cette équivalence est de déclarer « $a \iff b$ est une tautologie ».
- Les mathématiciens sont parfois paresseux : il leur arrive d'écrire « $a \iff b$ » au lieu de « $a \iff b$ est *vraie* (ce qui signifie vraie pour tout contexte) ».
- On écrit $a \equiv b$ pour indiquer que $a \iff b$ est une tautologie.
- Attention, « $\equiv$ » n'est pas un opérateur (il n'appartient pas au langage) mais un *méta-opérateur*.
  A ce propos, un principe empirique est que, pour exprimer des idées sur un langage, il faut souvent « sortir » du langage.

Notion de *conséquence*

Définition

On considère une expression logique p et χ un ensemble d'expressions logiques. On dit que p est *une conséquence de* χ si toute interprétation qui satisfait toutes les formules de l'ensemble χ satisfait p. Si c'est le cas, on note : $\chi \models p$.

Notation

- Si $\chi = \{h_1, \ldots, h_n\}$, on écrit aussi $h_1, \ldots, h_n \models p$.

Notion de *conséquence*

Définition

On considère une expression logique p et χ un ensemble d'expressions logiques. On dit que p est *une conséquence de* χ si toute interprétation qui satisfait toutes les formules de l'ensemble χ satisfait p. Si c'est le cas, on note : $\chi \models p$.

Notation

- Si $\chi = \{h_1, \ldots, h_n\}$, on écrit aussi $h_1, \ldots, h_n \models p$.
- Si $\chi = \{h\}$, on écrit $h \models p$

Notion de *conséquence*

Définition

On considère une expression logique p et χ un ensemble d'expressions logiques. On dit que p est *une conséquence de* χ si toute interprétation qui satisfait toutes les formules de l'ensemble χ satisfait p. Si c'est le cas, on note : $\chi \vDash p$.

Notation

- Si $\chi = \{h_1, \dots, h_n\}$, on écrit aussi $h_1, \dots, h_n \vDash p$.
- Si $\chi = \{h\}$, on écrit $h \vDash p$
- Par convention, $\vDash p$ indique que p est une tautologie.

Propriétés de $\iff$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Réflexivité $p_1 \iff p_1$

Propriétés de $\iff$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Réflexivité $p_1 \iff p_1$

Symétrie $(p_1 \iff p_2) \iff (p_2 \iff p_1)$

Propriétés de $\iff$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Réflexivité $p_1 \iff p_1$

Symétrie $(p_1 \iff p_2) \iff (p_2 \iff p_1)$

Transitivité $((p_1 \iff p_2) \wedge (p_2 \iff p_3)) \iff (p_1 \iff p_3)$

Propriétés de $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Élément neutre $(p_1 \wedge V) \iff p_1$

Propriétés de $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Élément neutre $(p_1 \wedge V) \iff p_1$

Élément absorbant $(p_1 \wedge F) \iff F$

Propriétés de $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Élément neutre $(p_1 \wedge V) \iff p_1$

Élément absorbant $(p_1 \wedge F) \iff F$

Commutativité $(p_1 \wedge p_2) \iff (p_2 \wedge p_1)$

Propriétés de $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Élément neutre $(p_1 \wedge V) \iff p_1$

Élément absorbant $(p_1 \wedge F) \iff F$

Commutativité $(p_1 \wedge p_2) \iff (p_2 \wedge p_1)$

Associativité $(p_1 \wedge (p_2 \wedge p_3)) \iff ((p_1 \wedge p_2) \wedge p_3)$

Propriétés de $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Élément neutre $(p_1 \wedge V) \iff p_1$

Élément absorbant $(p_1 \wedge F) \iff F$

Commutativité $(p_1 \wedge p_2) \iff (p_2 \wedge p_1)$

Associativité $(p_1 \wedge (p_2 \wedge p_3)) \iff ((p_1 \wedge p_2) \wedge p_3)$

Idempotence $(p_1 \wedge p_1) \iff p_1$

Propriétés de $\vee$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Élément neutre $(p_1 \vee F) \iff p_1$

Propriétés de $\vee$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Élément neutre $(p_1 \vee F) \iff p_1$

Élément absorbant $(p_1 \vee V) \iff V$

Propriétés de $\vee$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Élément neutre $(p_1 \vee F) \iff p_1$

Élément absorbant $(p_1 \vee V) \iff V$

Commutativité $(p_1 \vee p_2) \iff (p_2 \vee p_1)$

Propriétés de $\vee$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Élément neutre $(p_1 \vee F) \iff p_1$

Élément absorbant $(p_1 \vee V) \iff V$

Commutativité $(p_1 \vee p_2) \iff (p_2 \vee p_1)$

Associativité $(p_1 \vee (p_2 \vee p_3)) \iff ((p_1 \vee p_2) \vee p_3)$

Propriétés de $\vee$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Élément neutre $(p_1 \vee F) \iff p_1$

Élément absorbant $(p_1 \vee V) \iff V$

Commutativité $(p_1 \vee p_2) \iff (p_2 \vee p_1)$

Associativité $(p_1 \vee (p_2 \vee p_3)) \iff ((p_1 \vee p_2) \vee p_3)$

Idempotence $(p_1 \vee p_1) \iff p_1$

Relations entre $\vee$ et $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Subsomption $(p_1 \vee (p_1 \wedge p_2)) \iff p_1$

Relations entre $\vee$ et $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Subsomption $(p_1 \vee (p_1 \wedge p_2)) \iff p_1$

Subsomption $(p_1 \wedge (p_1 \vee p_2)) \iff p_1$

Relations entre $\vee$ et $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Subsomption $(p_1 \vee (p_1 \wedge p_2)) \iff p_1$

Subsomption $(p_1 \wedge (p_1 \vee p_2)) \iff p_1$

Distributivité $(p_1 \wedge (p_2 \vee p_3)) \iff (p_1 \wedge p_2) \vee (p_1 \wedge p_3)$

Relations entre $\vee$ et $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Subsomption	$(p_1 \vee (p_1 \wedge p_2)) \iff p_1$
Subsomption	$(p_1 \wedge (p_1 \vee p_2)) \iff p_1$
Distributivité	$(p_1 \wedge (p_2 \vee p_3)) \iff (p_1 \wedge p_2) \vee (p_1 \wedge p_3)$
Distributivité	$(p_1 \vee (p_2 \wedge p_3)) \iff (p_1 \vee p_2) \wedge (p_1 \vee p_3)$

Relations entre $\vee$ et $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Subsomption $(p_1 \vee (p_1 \wedge p_2)) \iff p_1$

Subsomption $(p_1 \wedge (p_1 \vee p_2)) \iff p_1$

Distributivité $(p_1 \wedge (p_2 \vee p_3)) \iff (p_1 \wedge p_2) \vee (p_1 \wedge p_3)$

Distributivité $(p_1 \vee (p_2 \wedge p_3)) \iff (p_1 \vee p_2) \wedge (p_1 \vee p_3)$

Première loi de De Morgan $\neg(p_1 \wedge p_2) \iff (\neg p_1) \vee (\neg p_2)$

Relations entre $\vee$ et $\wedge$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Subsomption $(p_1 \vee (p_1 \wedge p_2)) \iff p_1$

Subsomption $(p_1 \wedge (p_1 \vee p_2)) \iff p_1$

Distributivité $(p_1 \wedge (p_2 \vee p_3)) \iff (p_1 \wedge p_2) \vee (p_1 \wedge p_3)$

Distributivité $(p_1 \vee (p_2 \wedge p_3)) \iff (p_1 \vee p_2) \wedge (p_1 \vee p_3)$

Première loi de De Morgan $\neg(p_1 \wedge p_2) \iff (\neg p_1) \vee (\neg p_2)$

Seconde loi de De Morgan $\neg(p_1 \vee p_2) \iff (\neg p_1) \wedge (\neg p_2)$

Propriétés de $\longrightarrow$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Modus ponens $(p_1 \wedge (p_1 \longrightarrow p_2)) \longrightarrow p_2$

Propriétés de $\longrightarrow$

Pour toutes propositions p_1, p_2, p_3, les propositions suivantes sont des tautologies :

Modus ponens $(p_1 \wedge (p_1 \longrightarrow p_2)) \longrightarrow p_2$

Double implication $((p_1 \longrightarrow p_2) \wedge (p_2 \longrightarrow p_1)) \iff (p_1 \iff p_2)$

Raisonnement en mathématiques

Pour toutes propositions p_1, p_2, les propositions suivantes sont des tautologies :

Disjonction de cas $((p_1 \longrightarrow p_2) \wedge (\neg p_1 \longrightarrow p_2)) \iff p_2$

Exercice

Montrer qu'on a bien des tautologies en construisant des tables de vérité.

Raisonnement en mathématiques

Pour toutes propositions p_1, p_2, les propositions suivantes sont des tautologies :

Disjonction de cas $((p_1 \longrightarrow p_2) \wedge (\neg p_1 \longrightarrow p_2)) \iff p_2$

Contraposition $(p_1 \longrightarrow p_2) \iff (\neg p_2 \longrightarrow \neg p_1)$

Exercice

Montrer qu'on a bien des tautologies en construisant des tables de vérité.

Raisonnement en mathématiques

Pour toutes propositions p_1, p_2, les propositions suivantes sont des tautologies :

Disjonction de cas $((p_1 \longrightarrow p_2) \land (\neg p_1 \longrightarrow p_2)) \iff p_2$

Contraposition $(p_1 \longrightarrow p_2) \iff (\neg p_2 \longrightarrow \neg p_1)$

Raisonnement par l'absurde $(\neg p_1 \longrightarrow F) \iff p_1$

Exercice

Montrer qu'on a bien des tautologies en construisant des tables de vérité.

1. Introduction
2. Syntaxe
3. Sémantique
4. Satisfiabilité
   - Tautologies
   - Formes normales
5. Problème SAT
   - Problème SAT et n-SAT
   - Problème 2-SAT

Écriture simplifiée

- Pour travailler avec une proposition p, il est souvent utile de considérer une expression sémantiquement équivalente à p mais plus simple :

Écriture simplifiée

- Pour travailler avec une proposition p, il est souvent utile de considérer une expression sémantiquement équivalente à p mais plus simple :
  - Ce peut être une version standardisée, dite *normale* (Analogie : $\dfrac{6}{4}$ a pour forme normale $\dfrac{3}{2}$)

Écriture simplifiée

- Pour travailler avec une proposition p, il est souvent utile de considérer une expression sémantiquement équivalente à p mais plus simple :
  - Ce peut être une version standardisée, dite *normale* (Analogie : $\frac{6}{4}$ a pour forme normale $\frac{3}{2}$)
  - ou encore une version avec moins de connecteurs : F pour $a \wedge (\neg a)$ (Analogie : 6 pour $1 + 2 + 3$).

Littéraux

Définition

On appelle *littéral* toute proposition de la forme v ou $\neg v$ où v est une variable propositionnelle.

Définition

On appelle *conjonction* (resp. disjonction) des propositions $p_1, \dots, p_n$ ($n \geq 1$), la proposition

$$p_1 \wedge p_2 \cdots \wedge p_n \text{ (resp. } p_1 \vee p_2 \cdots \vee p_n)$$

Remarque

Par convention :

- Une conjonction de 0 proposition est V

Littéraux

Définition

On appelle *littéral* toute proposition de la forme v ou $\neg v$ où v est une variable propositionnelle.

Définition

On appelle *conjonction* (resp. disjonction) des propositions $p_1, \dots, p_n$ ($n \geq 1$), la proposition

$$p_1 \wedge p_2 \cdots \wedge p_n \text{ (resp. } p_1 \vee p_2 \cdots \vee p_n)$$

Remarque

Par convention :

- Une conjonction de 0 proposition est V
- Une disjonction de 0 proposition est F

Littéraux

Définition

On appelle *littéral* toute proposition de la forme v ou $\neg v$ où v est une variable propositionnelle.

Définition

On appelle *conjonction* (resp. disjonction) des propositions $p_1, \ldots, p_n$ ($n \geq 1$), la proposition

$$p_1 \wedge p_2 \cdots \wedge p_n \text{ (resp. } p_1 \vee p_2 \cdots \vee p_n)$$

Remarque

Par convention :

- Une conjonction de 0 proposition est V
- Une disjonction de 0 proposition est F
- Tout littéral est à la fois une conjonction et une disjonction.

Clauses, Monômes

Définition

On appelle *clause* toute disjonction de littéraux. F, disjonction de zéro littéral, est appelé la *clause vide*. (Moyen mnémotechnique : claUse$\leftrightarrow \cup \leftrightarrow \vee$)

Clauses, Monômes

Définition

On appelle *clause* toute disjonction de littéraux. F, disjonction de zéro littéral, est appelé la *clause vide*. (Moyen mnémotechnique : claUse$\leftrightarrow \cup \leftrightarrow \vee$)

Définition

On appelle *monôme* toute conjonction de littéraux. V, conjonction de zéro littéral, est appelé le *monôme vide*. (Moyen mnémotechnique : mon$\widehat{o}$me$\leftrightarrow \wedge$)

Clauses, Monômes

Définition

On appelle *clause* toute disjonction de littéraux. F, disjonction de zéro littéral, est appelé la *clause vide*. (Moyen mnémotechnique : claUse$\leftrightarrow \cup \leftrightarrow \vee$)

Définition

On appelle *monôme* toute conjonction de littéraux. V, conjonction de zéro littéral, est appelé le *monôme vide*. (Moyen mnémotechnique : mon$\widehat{o}$me$\leftrightarrow \wedge$)

Définition

On appelle *forme normale conjonctive* (FNC) toute conjonction de clauses et *forme normale disjonctive* (FND) toute disjonction de monômes.

Forme normale conjonctive/disjonctive

Proposition

Toute proposition est sémantiquement équivalente à une forme normale conjonctive (resp. forme normale disjonctive).

Remarque

- Fait : On n'a besoin que des 3 opérateurs $\neg, \vee, \wedge$ et aucune constante (V, F) pour exprimer toute proposition.

Forme normale conjonctive/disjonctive

Proposition

Toute proposition est sémantiquement équivalente à une forme normale conjonctive (resp. forme normale disjonctive).

Remarque

- Fait : On n'a besoin que des 3 opérateurs $\neg, \vee, \wedge$
- Convertir une proposition en forme normale conjonctive requiert l'utilisation de règles de transformation logiques, comme l'élimination de double négations, les lois de De Morgan, et la loi de distributivité.

Forme normale conjonctive/disjonctive

Proposition

Toute proposition est sémantiquement équivalente à une forme normale conjonctive (resp. forme normale disjonctive).

Remarque

- Fait : On n'a besoin que des 3 opérateurs $\neg, \vee, \wedge$
- Convertir une proposition en forme normale conjonctive requiert l'utilisation de règles de transformation logiques
- Principe : ramener la négation « au contact » des variables (de Morgan), « remonter » les $\wedge$ (distributivité) et supprimer les $\neg\neg$. Par exemple, si v_1, v_2, v_3, v_4 sont des variables :

$$\begin{aligned}
\underbrace{\neg(v_1 \wedge (v_2 \vee \neg v_3)) \vee v_4}_{\text{F. ni conjonctive ni disjonctive}} &\equiv \neg v_1 \vee \neg(v_2 \vee \neg v_3) \vee v_4 \equiv \\
\neg v_1 \vee (\neg v_2 \wedge \neg\neg v_3) \vee v_4 &\equiv (\neg v_1 \vee \neg v_2 \vee v_4) \wedge (\neg v_1 \vee v_3 \vee v_4)
\end{aligned}$$

Forme normale conjonctive/disjonctive

Proposition

Toute proposition est sémantiquement équivalente à une forme normale conjonctive (resp. forme normale disjonctive).

Remarque

- Fait : On n'a besoin que des 3 opérateurs $\neg, \vee, \wedge$
- Convertir une proposition en forme normale conjonctive requiert l'utilisation de règles de transformation logiques
- Principe : ramener la négation « au contact » des variables (de Morgan), « remonter » les $\wedge$ (distributivité) et supprimer les $\neg\neg$.
- Les formes normales conjonctives ou disjonctives de littéraux ne sont en général pas uniques !

Exemples

Les propositions suivantes, dans lesquelles a, b, c, d, e, f sont des variables, sont des formes conjonctives :

- $a \wedge b$

Exemples

Les propositions suivantes, dans lesquelles a, b, c, d, e, f sont des variables, sont des formes conjonctives :

- $a \wedge b$
- $a \vee b$ (oui car c'est une clause - il y a zéro conjonction-)

Exemples

Les propositions suivantes, dans lesquelles a, b, c, d, e, f sont des variables, sont des formes conjonctives :

- $a \wedge b$
- $a \vee b$ (oui car c'est une clause - il y a zéro conjonction-)
- a

Exemples

Les propositions suivantes, dans lesquelles a, b, c, d, e, f sont des variables, sont des formes conjonctives :

- $a \wedge b$
- $a \vee b$ (oui car c'est une clause - il y a zéro conjonction-)
- a
- $(a \vee b) \wedge c$

Exemples

Les propositions suivantes, dans lesquelles a, b, c, d, e, f sont des variables, sont des formes conjonctives :

- $a \wedge b$
- $a \vee b$ (oui car c'est une clause - il y a zéro conjonction-)
- a
- $(a \vee b) \wedge c$
- $(a \vee \neg b \vee \neg c) \wedge (\neg d \vee e \vee f)$

Contre exemples

Les propositions suivantes, dans lesquelles a, b, c, d, e, f sont des variables, ne sont pas des formes conjonctives :

- $\neg(a \wedge b)$: la négation $\neg$ est devant la proposition $(a \wedge b)$ qui n'est pas atomique (ce n'est pas un littéral)

Contre exemples

Les propositions suivantes, dans lesquelles a, b, c, d, e, f sont des variables, ne sont pas des formes conjonctives :

- $\neg(a \wedge b)$: la négation $\neg$ est devant la proposition $(a \wedge b)$ qui n'est pas atomique (ce n'est pas un littéral)
- $a \wedge (b \vee (c \wedge d))$: un $\wedge$ est imbriqué dans un $\vee$.

Réécriture

Pour obtenir une FNC à partir d'une proposition p :

- Supprimer les constantes : V est remplacé par $v \vee \neg v$; F par $v \wedge \neg v$ où v est une variable.

Réécriture

Pour obtenir une FNC à partir d'une proposition p :

- Supprimer les constantes : V est remplacé par $v \vee \neg v$; F par $v \wedge \neg v$ où v est une variable.
- Supprimer les implications : on exprime la proposition uniquement avec des $\neg, \vee, \wedge$

Réécriture

Pour obtenir une FNC à partir d'une proposition p :

- Supprimer les constantes : V est remplacé par $v \vee \neg v$; F par $v \wedge \neg v$ où v est une variable.
- Supprimer les implications : on exprime la proposition uniquement avec des $\neg, \vee, \wedge$
- On descend $\neg$ par les lois de De Morgan au contact des variables.

Réécriture

Pour obtenir une FNC à partir d'une proposition p :

- Supprimer les constantes : V est remplacé par $v \vee \neg v$; F par $v \wedge \neg v$ où v est une variable.
- Supprimer les implications : on exprime la proposition uniquement avec des $\neg, \vee, \wedge$
- On descend $\neg$ par les lois de De Morgan au contact des variables.
- On applique la distributivité pour faire descendre les $\vee$ et remonter les $\wedge$.

Preuve d'existence d'une forme normale conjonctive

Récurrence sur la taille de la proposition p. HR : p est équivalente à FNC.

- Si p est un littéral : ok.

1.

Preuve d'existence d'une forme normale conjonctive

Récurrence sur la taille de la proposition p. HR : p est équivalente à FNC.

- Si p est un littéral : ok.
- Si $p = \neg q$, on applique l'hypothèse de récurrence à q qui s'écrit donc $c_1 \wedge \cdots \wedge c_n$ où les c_i sont des clauses.

1.

Preuve d'existence d'une forme normale conjonctive

Récurrence sur la taille de la proposition p. HR : p est équivalente à FNC.

- Si p est un littéral : ok.
- Si $p = \neg q$, on applique l'hypothèse de récurrence à q qui s'écrit donc $c_1 \wedge \cdots \wedge c_n$ où les c_i sont des clauses.
  - Par les lois de De Morgan, chaque clause c_i se réécrit [1] comme un $\neg m_i$ où m_i est un monôme $m_i = \bigwedge_{k_i=1}^{n_i} \ell_{i,k_i}$ (les ℓ_{i,k_i} sont des littéraux).

1. En exercice (il faut tout de même faire attention aux doubles négations).

Preuve d'existence d'une forme normale conjonctive

Récurrence sur la taille de la proposition p. HR : p est équivalente à FNC.

- Si p est un littéral : ok.
- Si $p = \neg q$, on applique l'hypothèse de récurrence à q qui s'écrit donc $c_1 \wedge \cdots \wedge c_n$ où les c_i sont des clauses.
  - Par les lois de De Morgan, chaque clause c_i se réécrit [1] comme un $\neg m_i$ où m_i est un monôme $m_i = \bigwedge_{k_i=1}^{n_i} \ell_{i,k_i}$ (les ℓ_{i,k_i} sont des littéraux).
  - Alors par De Morgan (encore) :
    $p \equiv \neg(\neg m_1 \wedge \cdots \wedge \neg m_n) \equiv (\neg\neg m_1) \vee \cdots \vee (\neg\neg m_n) \equiv m_1 \vee \cdots \vee m_n$.

1. En exercice (il faut tout de même faire attention aux doubles négations).

Preuve d'existence d'une forme normale conjonctive

Récurrence sur la taille de la proposition p. HR : p est équivalente à FNC.

- Si p est un littéral : ok.
- Si $p = \neg q$, on applique l'hypothèse de récurrence à q qui s'écrit donc $c_1 \wedge \cdots \wedge c_n$ où les c_i sont des clauses.
  - Par les lois de De Morgan, chaque clause c_i se réécrit [1] comme un $\neg m_i$ où m_i est un monôme $m_i = \bigwedge_{k_i=1}^{n_i} \ell_{i,k_i}$ (les ℓ_{i,k_i} sont des littéraux).
  - Alors par De Morgan (encore) : $p \equiv \neg(\neg m_1 \wedge \cdots \wedge \neg m_n) \equiv (\neg\neg m_1) \vee \cdots \vee (\neg\neg m_n) \equiv m_1 \vee \cdots \vee m_n$.
  - On a bien ce qu'on veut :

$$p \equiv \bigvee_{i=1}^{n} \left(\bigwedge_{k_i=1}^{n_i} \ell_{i,k_i} \right) \underbrace{\equiv}_{\text{distributivité}} \overbrace{\bigwedge_{\substack{k_1 \in [\![1,n_1]\!] \\ \vdots \\ k_n \in [\![1,n_n]\!]}} \underbrace{\ell_{1,k_1} \vee \cdots \vee \ell_{n,k_n}}_{\text{clause}}}^{\text{conjonction de clauses}}$$

1. En exercice (il faut tout de même faire attention aux doubles négations).

Preuve d'existence d'une forme normale conjonctive

Récurrence sur la taille de la proposition p. HR : p est équivalente à une conjonction de clause.

- Si p est un littéral : OK.

Preuve d'existence d'une forme normale conjonctive

Récurrence sur la taille de la proposition p. HR : p est équivalente à une conjonction de clause.

- Si p est un littéral : OK.
- Si $p = \neg q$: OK

Preuve d'existence d'une forme normale conjonctive

Récurrence sur la taille de la proposition p. HR : p est équivalente à une conjonction de clause.

- Si p est un littéral : OK.
- Si $p = \neg q$: OK
- Si $p = f_1 \wedge f_2$, par HR $f_1 \equiv p_1 \wedge \cdots \wedge p_{n_1}$ et $f_2 \equiv q_1 \wedge \cdots \wedge q_{n_2}$ où les p_k, q_r sont des clauses.

$$p \underbrace{\equiv}_{\text{assoc. de } \wedge} \underbrace{p_1 \wedge \cdots \wedge p_{n_1} \wedge q_1 \wedge \cdots \wedge q_{n_2}}_{\text{conjonction de clauses}}$$

Ce qu'on veut.

Preuve d'existence d'une forme normale conjonctive

Récurrence sur la taille de la proposition p. HR : p est équivalente à une conjonction de clause.

- Si p est un littéral : OK.
- Si $p = \neg q$: OK
- Si $p = f_1 \wedge f_2$, par HR $f_1 \equiv p_1 \wedge \cdots \wedge p_{n_1}$ et $f_2 \equiv q_1 \wedge \cdots \wedge q_{n_2}$ où les p_k, q_r sont des clauses.

$$p \underbrace{\equiv}_{\text{assoc. de } \wedge} \underbrace{p_1 \wedge \cdots \wedge p_{n_1} \wedge q_1 \wedge \cdots \wedge q_{n_2}}_{\text{conjonction de clauses}}$$

  Ce qu'on veut.
- Si $p = f_1 \vee f_2$ alors, par HR, p est équivalent à une formule de la forme $(p_1 \wedge \cdots \wedge p_{n_1}) \vee (q_1 \wedge \cdots \wedge q_{n_2})$ où les p_i, q_j sont des clauses. Et par distributivité :

$$p \equiv \bigwedge_{(k_1,k_2)\in[\![1,n_1]\!]\times[\![1,n_2]\!]} (p_{k_1} \vee q_{k_2}) \text{ : OK}$$

Forme normale disjonctive

- On sait que toute proposition peut s'exprimer comme conjonction de clause.

Forme normale disjonctive

- On sait que toute proposition peut s'exprimer comme conjonction de clause.
- Soit p une proposition, alors $\neg p$ s'écrit sous la forme $\neg p \equiv c_1 \wedge \cdots \wedge c_n$ avec $c_i = \ell_{i,1} \vee \cdots \vee \ell_{i,n_i}$.

Forme normale disjonctive

- On sait que toute proposition peut s'exprimer comme conjonction de clause.
- Soit p une proposition, alors $\neg p$ s'écrit sous la forme $\neg p \equiv c_1 \wedge \cdots \wedge c_n$ avec $c_i = \ell_{i,1} \vee \cdots \vee \ell_{i,n_i}$.
- Donc $p \equiv \neg\neg p \equiv \neg c_1 \vee \cdots \vee \neg c_n$ par De Morgan.

Forme normale disjonctive

- On sait que toute proposition peut s'exprimer comme conjonction de clause.
- Soit p une proposition, alors $\neg p$ s'écrit sous la forme $\neg p \equiv c_1 \wedge \cdots \wedge c_n$ avec $c_i = \ell_{i,1} \vee \cdots \vee \ell_{i,n_i}$.
- Donc $p \equiv \neg\neg p \equiv \neg c_1 \vee \cdots \vee \neg c_n$ par De Morgan.
- Or chaque $\neg c_i$ est équivalent sémantiquement à une conjonction de littéraux (par De Morgan) : $\neg c_i \equiv \neg\ell_{i,1} \wedge \cdots \wedge \neg\ell_{i,n_i}$. (attention toutefois à simplifier les doubles négations).

Forme normale disjonctive

- On sait que toute proposition peut s'exprimer comme conjonction de clause.
- Soit p une proposition, alors $\neg p$ s'écrit sous la forme $\neg p \equiv c_1 \wedge \cdots \wedge c_n$ avec $c_i = \ell_{i,1} \vee \cdots \vee \ell_{i,n_i}$.
- Donc $p \equiv \neg\neg p \equiv \neg c_1 \vee \cdots \vee \neg c_n$ par De Morgan.
- Or chaque $\neg c_i$ est équivalent sémantiquement à une conjonction de littéraux (par De Morgan) : $\neg c_i \equiv \neg \ell_{i,1} \wedge \cdots \wedge \neg \ell_{i,n_i}$. (attention toutefois à simplifier les doubles négations).
- Donc toute proposition peut s'écrire comme disjonction de monômes.

Forme normale disjonctive

- On sait que toute proposition peut s'exprimer comme conjonction de clause.
- Soit p une proposition, alors $\neg p$ s'écrit sous la forme $\neg p \equiv c_1 \wedge \cdots \wedge c_n$ avec $c_i = \ell_{i,1} \vee \cdots \vee \ell_{i,n_i}$.
- Donc $p \equiv \neg\neg p \equiv \neg c_1 \vee \cdots \vee \neg c_n$ par De Morgan.
- Or chaque $\neg c_i$ est équivalent sémantiquement à une conjonction de littéraux (par De Morgan) : $\neg c_i \equiv \neg \ell_{i,1} \wedge \cdots \wedge \neg \ell_{i,n_i}$. (attention toutefois à simplifier les doubles négations).
- Donc toute proposition peut s'écrire comme disjonction de monômes.
- PB : ces réécritures ne sont pas uniques.

Explosion combinatoire

- Considérons la proposition en forme disjonctive : $(x_1 \wedge y_1) \vee (x_2 \wedge y_2) \vee \cdots \vee (x_n \wedge y_n)$ de taille linéaire

Explosion combinatoire

- Considérons la proposition en forme disjonctive : $(x_1 \wedge y_1) \vee (x_2 \wedge y_2) \vee \cdots \vee (x_n \wedge y_n)$ de taille linéaire
- Par distributivité, sa FNC, de taille 2^n, est de la forme :

$$
\begin{gathered}
(x_1 \vee x_2 \vee \cdots \vee x_{n-1} \vee x_n) \wedge (x_1 \vee x_2 \vee \cdots \vee x_{n-1} \vee y_n) \wedge \cdots \\
\cdots \wedge (x_1 \vee x_2 \vee \cdots \vee x_k \vee y_{k+1} \vee \cdots \vee y_{n-1} \vee y_n) \wedge \cdots \\
\wedge (x_1 \vee y_2 \vee \cdots \vee y_{n-1} \vee y_n) \wedge (y_1 \vee y_2 \vee \cdots \vee y_{n-1} \vee y_n)
\end{gathered}
$$

Mintermes, maxtermes

Définition

Soient $v_1, \ldots, v_n$ des variables distinctes.

- On appelle *minterme* de $v_1, \ldots, v_n$, tout monôme où chaque variable v_i apparaît exactement une fois (et il n'y a pas d'autre variable).

Remarque

Les 4 (à l'ordre près) maxtermes de v_1 et v_2 sont

$$v_1 \vee v_2,\ v_1 \vee \neg v_2,\ \neg v_1 \vee v_2,\ \neg v_1 \vee \neg v_2.$$

On confond les maxtermes $v_1 \vee v_2$ et $v_2 \vee v_1$.

Mintermes, maxtermes

Définition

Soient $v_1, \ldots, v_n$ des variables distinctes.

- On appelle *minterme* de $v_1, \ldots, v_n$, tout monôme où chaque variable v_i apparaît exactement une fois (et il n'y a pas d'autre variable).
- On appelle *mAxterme* de $v_1, \ldots, v_n$, une clAuse où chaque variable v_i apparaît exactement une fois (et il n'y a pas d'autre variable).

Remarque

Les 4 (à l'ordre près) maxtermes de v_1 et v_2 sont

$$v_1 \vee v_2,\ v_1 \vee \neg v_2,\ \neg v_1 \vee v_2,\ \neg v_1 \vee \neg v_2.$$

On confond les maxtermes $v_1 \vee v_2$ et $v_2 \vee v_1$.

Forme normale conjonctive

Définition

- On dit qu'une proposition est en *forme normale conjonctive complète* (*i.e.* avec maxtermes) (FNCC), si elle s'écrit comme une conjonction de maxtermes tous distincts.
- On dit qu'une proposition est en *forme normale disjonctive complète* (*i.e.* avec mintremes) (FNDC), si elle s'écrit comme une disjonction de mintermes tous distincts.

Remarque

- Mettre une proposition sous FNCC, c'est donner une expression en FNCC sémantiquement équivalente à cette proposition.

Forme normale conjonctive

Définition

- On dit qu'une proposition est en *forme normale conjonctive complète* (*i.e.* avec maxtermes) (FNCC), si elle s'écrit comme une conjonction de maxtermes tous distincts.
- On dit qu'une proposition est en *forme normale disjonctive complète* (*i.e.* avec mintremes) (FNDC), si elle s'écrit comme une disjonction de mintermes tous distincts.

Remarque

- Mettre une proposition sous FNCC, c'est donner une expression en FNCC sémantiquement équivalente à cette proposition.
- $(v_1 \vee \neg v_2 \vee v_3) \wedge (v_1 \vee v_2 \vee \neg v_3)$ est une FNCC sur les variables v_1, v_2, v_3 mais pas $(v_1 \vee \neg v_2 \vee v_3) \wedge \neg(v_1 \vee v_2 \vee \neg v_3)$ car la négation n'est pas au contact des variables. $(v_1 \vee v_2 \vee \neg v_3) \wedge (v_1 \vee v_3)$ non plus car il manque une variable.

Forme normale conjonctive

Définition

- On dit qu'une proposition est en *forme normale conjonctive complète* (*i.e.* avec maxtermes) (FNCC), si elle s'écrit comme une conjonction de maxtermes tous distincts.
- On dit qu'une proposition est en *forme normale disjonctive complète* (*i.e.* avec mintremes) (FNDC), si elle s'écrit comme une disjonction de mintermes tous distincts.

Remarque

- Mettre une proposition sous FNCC, c'est donner une expression en FNCC sémantiquement équivalente à cette proposition.
- $(v_1 \vee \neg v_2 \vee v_3) \wedge (v_1 \vee v_2 \vee \neg v_3)$ est une FNCC sur les variables v_1, v_2, v_3 mais pas $(v_1 \vee \neg v_2 \vee v_3) \wedge \neg(v_1 \vee v_2 \vee \neg v_3)$ car la négation n'est pas au contact des variables.
- Utilisation : démonstration automatique de théorèmes ou PB SAT.

Exemple

Soit p une proposition à 3 variables dont la table est :

Table de p			
a	b	c	p
1	1	1	0
1	1	0	1
1	0	1	1
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	0
0	0	0	1

p est sémantiquement équivalente à la disjonction de mintermes q suivante : $(a \wedge b \wedge \neg c) \vee (a \wedge \neg b \wedge c) \vee (\neg a \wedge b \wedge c) \vee (\neg a \wedge \neg b \wedge \neg c)$
En effet, p prend la valeur 1 pour les mêmes contextes exactement que la FNDC q (une disjonction vaut 1 si et seulement si un terme vaut 1).

Exemple

Soit p une proposition à 3 variables dont la table est :

Table de p			
a	b	c	p
1	1	1	0
1	1	0	1
1	0	1	1
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	0
0	0	0	1

$\neg p$ est sémantiquement équivalente à la disjonction de mintermes q' suivante : $(a \wedge b \wedge c) \vee (a \wedge \neg b \wedge \neg c) \vee (\neg a \wedge b \wedge \neg c) \vee (\neg a \wedge \neg b \wedge c)$

Donc $\neg\neg p$ est sémantiquement équivalente à

$(\neg a \vee \neg b \vee \neg c) \wedge (\neg a \vee b \vee c) \wedge (a \vee \neg b \vee c) \wedge (a \vee b \vee \neg c)$

Exemple

Soit p une proposition à 3 variables dont la table est :

Table de p			
a	b	c	p
1	1	1	0
1	1	0	1
1	0	1	1
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	0
0	0	0	1

Lire positivement les variables des lignes à 1 pour la FND

Lire négativement les variables des lignes à 0 pour la FNC

Forme normale conjonctive

Proposition

Toute proposition est sémantiquement équivalente à une FNCC (resp. FNDC) unique à l'ordre des maxtermes (resp. mintermes) près (et à ordre des littéraux près dans chaque maxterme -resp. minterme-).

Remarque

- Soit P une antilogie : la disjonction à zéro minterme est F par convention. Or $F \equiv P$, donc P est bien une disjonction de minterme.
- Soit P une tautologie : la conjonction à zéro maxterme est V par convention. Or $V \equiv P$, donc P est bien une conjonction de maxterme.
- Un minterme est satisfait par un seul contexte ; un maxterme par tous les contextes sauf 1.
- Dans ce qui suit on dit que deux FNDC sont égales si elles ont les mêmes mintermes (les mintermes sont donnés à l'ordre des littéraux près).

Existence de la FNDC

Démonstration.

- On construit la table de vérité de la proposition : il y a 2^n lignes correspondants aux valeurs de vérité des n variables de p.

Existence de la FNDC

Démonstration.

- On construit la table de vérité de la proposition : il y a 2^n lignes correspondants aux valeurs de vérité des n variables de p.
- Chaque ligne satisfaisant l'expression (1 dans la dernière colonne) donne un minterme. Toutes les variables sont présentes et une variable v y apparaît positivement (v) si sa valeur de vérité dans la ligne est 1, négativement ($\neg v$) sinon.

Existence de la FNDC

Démonstration.

- On construit la table de vérité de la proposition : il y a 2^n lignes correspondants aux valeurs de vérité des n variables de p.
- Chaque ligne satisfaisant l'expression (1 dans la dernière colonne) donne un minterme. Toutes les variables sont présentes et une variable v y apparaît positivement (v) si sa valeur de vérité dans la ligne est 1, négativement ($\neg v$) sinon.
- On considère la disjonction de tous les mintermes ainsi obtenus. C'est une expression q en FNDC. Possiblement q est vide.

Existence de la FNDC

Démonstration.

- On construit la table de vérité de la proposition : il y a 2^n lignes correspondants aux valeurs de vérité des n variables de p.
- Chaque ligne satisfaisant l'expression (1 dans la dernière colonne) donne un minterme. Toutes les variables sont présentes et une variable v y apparaît positivement (v) si sa valeur de vérité dans la ligne est 1, négativement ($\neg v$) sinon.
- On considère la disjonction de tous les mintermes ainsi obtenus. C'est une expression q en FNDC. Possiblement q est vide.
- Soit μ un contexte. Si $\varepsilon_\mu(p) = 1$, alors le minterme associé à μ est vrai pour μ donc $\varepsilon_\mu(q) = 1$.

Existence de la FNDC

Démonstration.

- On construit la table de vérité de la proposition : il y a 2^n lignes correspondants aux valeurs de vérité des n variables de p.
- Chaque ligne satisfaisant l'expression (1 dans la dernière colonne) donne un minterme. Toutes les variables sont présentes et une variable v y apparaît positivement (v) si sa valeur de vérité dans la ligne est 1, négativement ($\neg v$) sinon.
- On considère la disjonction de tous les mintermes ainsi obtenus. C'est une expression q en FNDC. Possiblement q est vide.
- Soit μ un contexte. Si $\varepsilon_\mu(p) = 1$, alors le minterme associé à μ est vrai pour μ donc $\varepsilon_\mu(q) = 1$.
- Si $\varepsilon_\mu(p) = 0$, alors q ne contient pas le minterme associé à μ. Ainsi, pour tout minterme m de q, on a $\varepsilon_\mu(m) = 0$ et donc $\varepsilon_\mu(q) = 0$.

Existence de la FNDC

Démonstration.

- On construit la table de vérité de la proposition : il y a 2^n lignes correspondants aux valeurs de vérité des n variables de p.
- Chaque ligne satisfaisant l'expression (1 dans la dernière colonne) donne un minterme. Toutes les variables sont présentes et une variable v y apparaît positivement (v) si sa valeur de vérité dans la ligne est 1, négativement ($\neg v$) sinon.
- On considère la disjonction de tous les mintermes ainsi obtenus. C'est une expression q en FNDC. Possiblement q est vide.
- Soit μ un contexte. Si $\varepsilon_\mu(p) = 1$, alors le minterme associé à μ est vrai pour μ donc $\varepsilon_\mu(q) = 1$.
- Si $\varepsilon_\mu(p) = 0$, alors q ne contient pas le minterme associé à μ. Ainsi, pour tout minterme m de q, on a $\varepsilon_\mu(m) = 0$ et donc $\varepsilon_\mu(q) = 0$.
- Donc p et q prennent bien la même valeur pour tout contexte.

Unicité de la FNDC à l'ordre près

Soient $p \equiv q$ deux FNDC. Elles ont donc même table.

- Si $p \neq q$ et p, q ne possèdent qu'un minterme :

Unicité de la FNDC à l'ordre près

Soient $p \equiv q$ deux FNDC. Elles ont donc même table.

- Si $p \neq q$ et p, q ne possèdent qu'un minterme :
  - Si p, q ont deux littéraux ℓ, ℓ' de la même variable v distincts, l'un des littéraux vaut $\neg v$ l'autre v. Autrement dit $\ell \equiv \neg \ell'$

Unicité de la FNDC à l'ordre près

Soient $p \equiv q$ deux FNDC. Elles ont donc même table.

- Si $p \neq q$ et p, q ne possèdent qu'un minterme :
  - Si p, q ont deux littéraux ℓ, ℓ' de la même variable v distincts, l'un des littéraux vaut $\neg v$ l'autre v. Autrement dit $\ell \equiv \neg \ell'$
  - Comme p est une conjonction, il n'y a qu'un seul contexte c (*i.e.* une seule ligne) dans lequel p vaut 1.

Unicité de la FNDC à l'ordre près

Soient $p \equiv q$ deux FNDC. Elles ont donc même table.

- Si $p \neq q$ et p, q ne possèdent qu'un minterme :
  - Si p, q ont deux littéraux ℓ, ℓ' de la même variable v distincts, l'un des littéraux vaut $\neg v$ l'autre v. Autrement dit $\ell \equiv \neg \ell'$
  - Comme p est une conjonction, il n'y a qu'un seul contexte c (*i.e.* une seule ligne) dans lequel p vaut 1.
  - Alors ℓ vaut aussi 1 dans ce contexte (p étant une conjonction).

Unicité de la FNDC à l'ordre près

Soient $p \equiv q$ deux FNDC. Elles ont donc même table.

- Si $p \neq q$ et p, q ne possèdent qu'un minterme :
  - Si p, q ont deux littéraux ℓ, ℓ' de la même variable v distincts, l'un des littéraux vaut $\neg v$ l'autre v. Autrement dit $\ell \equiv \neg \ell'$
  - Comme p est une conjonction, il n'y a qu'un seul contexte c (*i.e.* une seule ligne) dans lequel p vaut 1.
  - Alors ℓ vaut aussi 1 dans ce contexte (p étant une conjonction).
  - Donc ℓ' vaut 0 dans c. Et donc q vaut 0 dans c. p et q ne peuvent avoir même table.

Unicité de la FNDC à l'ordre près

Soient $p \equiv q$ deux FNDC. Elles ont donc même table.

- Si $p \neq q$ et p, q ne possèdent qu'un minterme :
  - Si p, q ont deux littéraux ℓ, ℓ' de la même variable v distincts, l'un des littéraux vaut $\neg v$ l'autre v. Autrement dit $\ell \equiv \neg \ell'$
  - Comme p est une conjonction, il n'y a qu'un seul contexte c (*i.e.* une seule ligne) dans lequel p vaut 1.
  - Alors ℓ vaut aussi 1 dans ce contexte (p étant une conjonction).
  - Donc ℓ' vaut 0 dans c. Et donc q vaut 0 dans c. p et q ne peuvent avoir même table.
  - De ce qui précède, on déduit aussi que des mintermes (ayant les mêmes variables) distincts ne valent jamais 1 en même temps.

Unicité de la FNDC(suite)

Soient p, q deux FNDC de même table.

- On a vu que si $p \neq q$ et p, q ne possèdent qu'un minterme, leurs tables sont différentes. On en déduit que des mintermes distincts ne valent jamais 1 en même temps.

Unicité de la FNDC(suite)

Soient p, q deux FNDC de même table.

- On a vu que si $p \neq q$ et p, q ne possèdent qu'un minterme, leurs tables sont différentes. On en déduit que des mintermes distincts ne valent jamais 1 en même temps.
- Si p et q sont des disjonctions de mintermes, et si un minterme m de p (par exemple) ne se retrouve pas dans q :

Unicité de la FNDC(suite)

Soient p, q deux FNDC de même table.

- On a vu que si $p \neq q$ et p, q ne possèdent qu'un minterme, leurs tables sont différentes. On en déduit que des mintermes distincts ne valent jamais 1 en même temps.
- Si p et q sont des disjonctions de mintermes, et si un minterme m de p (par exemple) ne se retrouve pas dans q :
  - Lorsque m vaut 1, p aussi (comme disjonction). Tous les autres mintermes valent 0 (puisque un seul minterme vaut 1 pour un contexte donné), ceux de p comme ceux de q.

Unicité de la FNDC(suite)

Soient p, q deux FNDC de même table.

- On a vu que si $p \neq q$ et p, q ne possèdent qu'un minterme, leurs tables sont différentes. On en déduit que des mintermes distincts ne valent jamais 1 en même temps.
- Si p et q sont des disjonctions de mintermes, et si un minterme m de p (par exemple) ne se retrouve pas dans q :
  - Lorsque m vaut 1, p aussi (comme disjonction). Tous les autres mintermes valent 0 (puisque un seul minterme vaut 1 pour un contexte donné), ceux de p comme ceux de q.
  - Alors dans ce contexte, comme disjonction de 0, q vaut 0.

Unicité de la FNDC(suite)

Soient p, q deux FNDC de même table.

- On a vu que si $p \neq q$ et p, q ne possèdent qu'un minterme, leurs tables sont différentes. On en déduit que des mintermes distincts ne valent jamais 1 en même temps.
- Si p et q sont des disjonctions de mintermes, et si un minterme m de p (par exemple) ne se retrouve pas dans q :
  - Lorsque m vaut 1, p aussi (comme disjonction). Tous les autres mintermes valent 0 (puisque un seul minterme vaut 1 pour un contexte donné), ceux de p comme ceux de q.
  - Alors dans ce contexte, comme disjonction de 0, q vaut 0.
  - Et donc p et q ne peuvent avoir même table.

FNCC

Soit une expression p à n variables qui n'est pas une tautologie.

FNCC

Soit une expression p à n variables qui n'est pas une tautologie.

Existence

- $\neg p$ a une FNDC $c_1 \vee \cdots \vee c_n$ où les c_i sont des mintermes.

FNCC

Soit une expression p à n variables qui n'est pas une tautologie.

Existence

- $\neg p$ a une FNDC $c_1 \vee \cdots \vee c_n$ où les c_i sont des mintermes.
- Donc p se réécrit comme $(\neg c_1) \wedge \cdots \wedge (\neg c_n)$. Or les négations de mintermes se réécrivent en maxtermes (par De Morgan et suppression des doubles négations).

FNCC

Soit une expression p à n variables qui n'est pas une tautologie.

Existence

- $\neg p$ a une FNDC $c_1 \vee \cdots \vee c_n$ où les c_i sont des mintermes.
- Donc p se réécrit comme $(\neg c_1) \wedge \cdots \wedge (\neg c_n)$. Or les négations de mintermes se réécrivent en maxtermes (par De Morgan et suppression des doubles négations).
- Et donc p se réécrit en une FNCC. Voilà pour l'existence.

FNCC

Soit une expression p à n variables qui n'est pas une tautologie.

Existence
- $\neg p$ a une FNDC $c_1 \vee \cdots \vee c_n$ où les c_i sont des mintermes.
- Donc p se réécrit comme $(\neg c_1) \wedge \cdots \wedge (\neg c_n)$. Or les négations de mintermes se réécrivent en maxtermes (par De Morgan et suppression des doubles négations).
- Et donc p se réécrit en une FNCC. Voilà pour l'existence.

Unicité Si p avait deux FNCC distinctes, par De Morgan $\neg p$ aurait deux FNDC distinctes. Or il y a une seule FNDC possible pour une formule.

Présentation du problème 2-SAT

- Le problème SAT qui consiste à déterminer si une proposition est satisfiable ou non est en général de complexité exponentielle en le nombre de variables.

Présentation du problème 2-SAT

- Le problème SAT qui consiste à déterminer si une proposition est satisfiable ou non est en général de complexité exponentielle en le nombre de variables.
- Le problème CNF-SAT est la restriction du problème SAT aux formes normales conjonctives.

Présentation du problème 2-SAT

- Le problème SAT qui consiste à déterminer si une proposition est satisfiable ou non est en général de complexité exponentielle en le nombre de variables.
- Le problème CNF-SAT est la restriction du problème SAT aux formes normales conjonctives.
- Le problème n-SAT est la restriction du problème SAT aux formes normales conjonctives avec au plus n littéraux par clause. Nous verrons *l'algorithme de Quine* qui résoud le problème CNF-STAT : il est souvent meilleur que la recherche exhaustive en force brute.

Présentation du problème 2-SAT

- Le problème SAT qui consiste à déterminer si une proposition est satisfiable ou non est en général de complexité exponentielle en le nombre de variables.
- Le problème CNF-SAT est la restriction du problème SAT aux formes normales conjonctives.
- Le problème n-SAT est la restriction du problème SAT aux formes normales conjonctives avec au plus n littéraux par clause. Nous verrons *l'algorithme de Quine* qui résoud le problème CNF-STAT : il est souvent meilleur que la recherche exhaustive en force brute.
- Le problème 2-SAT consiste à déterminer si une forme conjonctive dont les clauses ont seulement deux littéraux est satisfiable.
  Il admet des solutions polynômiales.

Problème SAT

- Comment tester qu'une proposition p est une tautologie ? Si on sait résoudre le problème SAT, il suffit de montrer que $\neg p$ est insatisfiable.

Problème SAT

- Comment tester qu'une proposition p est une tautologie ? Si on sait résoudre le problème SAT, il suffit de montrer que $\neg p$ est insatisfiable.
- Pour le problème SAT, on peut simplement calculer sa table. Mais on a alors une complexité en $O(2^n)$, si n est le nombre de variables.

Problème SAT

- Comment tester qu'une proposition p est une tautologie ? Si on sait résoudre le problème SAT, il suffit de montrer que $\neg p$ est insatisfiable.
- Pour le problème SAT, on peut simplement calculer sa table. Mais on a alors une complexité en $O(2^n)$, si n est le nombre de variables.
- Des algorithmes plus efficaces (en pratique, c'est à dire sauf cas pathologiques) existent (DPLL) et passent par une mise sous FNC. Mais dans le pire des cas ce passage à la FNC est lui-même de complexité exponentielle.

2-SAT

Cette section est laissée pour info mais sera traitée en seconde année.

Problème 2-SAT (Aspvall-Plass-Tarjan)

Soit p une proposition mise sous forme conjonctive avec des 2-clauses

- On construit un graphe orienté :

Problème 2-SAT (Aspvall-Plass-Tarjan)

Soit p une proposition mise sous forme conjonctive avec des 2-clauses

- On construit un graphe orienté :
  - Ses sommets sont tous les littéraux formés avec les variables apparaissant dans la proposition.

Problème 2-SAT (Aspvall-Plass-Tarjan)

Soit p une proposition mise sous forme conjonctive avec des 2-clauses

- On construit un graphe orienté :
  - Ses sommets sont tous les littéraux formés avec les variables apparaissant dans la proposition.
  - Pour deux littéraux a, b, l'arc $(\neg a, b)$ est présent si et seulement si la disjonction $a \vee b$ est dans p. Et dans ce cas $(\neg b, a)$ est présent aussi.

Problème 2-SAT (Aspvall-Plass-Tarjan)

Soit p une proposition mise sous forme conjonctive avec des 2-clauses

- On construit un graphe orienté :
  - Ses sommets sont tous les littéraux formés avec les variables apparaissant dans la proposition.
  - Pour deux littéraux a, b, l'arc $(\neg a, b)$ est présent si et seulement si la disjonction $a \vee b$ est dans p. Et dans ce cas $(\neg b, a)$ est présent aussi.
  - Ceci correspond à $a \vee b \equiv (\neg a \longrightarrow b) \wedge (\neg b \longrightarrow a)$

Problème 2-SAT (Aspvall-Plass-Tarjan)

Soit p une proposition mise sous forme conjonctive avec des 2-clauses

- On construit un graphe orienté :
  - Ses sommets sont tous les littéraux formés avec les variables apparaissant dans la proposition.
  - Pour deux littéraux a, b, l'arc $(\neg a, b)$ est présent si et seulement si la disjonction $a \vee b$ est dans p. Et dans ce cas $(\neg b, a)$ est présent aussi.
  - Ceci correspond à $a \vee b \equiv (\neg a \longrightarrow b) \wedge (\neg b \longrightarrow a)$
- Une fois ce graphe construit, on examine la composante fortement connexe de toute variable v de p. Si $\neg v$ est dedans, il y a un chemin d'implications $v \longrightarrow \ldots \longrightarrow \neg v$ et un autre $\neg v \longrightarrow \ldots \longrightarrow v$.

Problème 2-SAT (Aspvall-Plass-Tarjan)

Soit p une proposition mise sous forme conjonctive avec des 2-clauses

- On construit un graphe orienté :
  - Ses sommets sont tous les littéraux formés avec les variables apparaissant dans la proposition.
  - Pour deux littéraux a, b, l'arc $(\neg a, b)$ est présent si et seulement si la disjonction $a \vee b$ est dans p. Et dans ce cas $(\neg b, a)$ est présent aussi.
  - Ceci correspond à $a \vee b \equiv (\neg a \longrightarrow b) \wedge (\neg b \longrightarrow a)$
- Une fois ce graphe construit, on examine la composante fortement connexe de toute variable v de p. Si $\neg v$ est dedans, il y a un chemin d'implications $v \longrightarrow \ldots \longrightarrow \neg v$ et un autre $\neg v \longrightarrow \ldots \longrightarrow v$.
- Alors $v \iff \neg v$ est une conséquence de p (Il faudrait le prouver !) : $p \models v \iff \neg v$ (ce qui signifie que tout modèle satisfaisant p, satisfait $v \iff \neg v$). Donc si p est satisfiable, $v \iff \neg v$ aussi : ABSURDE.

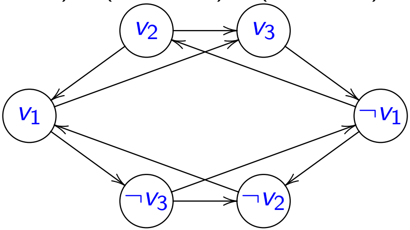
Problème 2-SAT (Aspvall-Plass-Tarjan)

Soit p une proposition mise sous forme conjonctive avec des 2-clauses

- On construit un graphe orienté :
  - Ses sommets sont tous les littéraux formés avec les variables apparaissant dans la proposition.
  - Pour deux littéraux a, b, l'arc $(\neg a, b)$ est présent si et seulement si la disjonction $a \vee b$ est dans p. Et dans ce cas $(\neg b, a)$ est présent aussi.
  - Ceci correspond à $a \vee b \equiv (\neg a \longrightarrow b) \wedge (\neg b \longrightarrow a)$
- Une fois ce graphe construit, on examine la composante fortement connexe de toute variable v de p. Si $\neg v$ est dedans, il y a un chemin d'implications $v \longrightarrow \ldots \longrightarrow \neg v$ et un autre $\neg v \longrightarrow \ldots \longrightarrow v$.
- Alors $v \iff \neg v$ est une conséquence de p (Il faudrait le prouver !) : $p \vDash v \iff \neg v$ (ce qui signifie que tout modèle satisfaisant p, satisfait $v \iff \neg v$). Donc si p est satisfiable, $v \iff \neg v$ aussi : ABSURDE.
- Donc si v et $\neg v$ sont dans la même composante connexe, p n'est pas satisfiable (Aspvall-Plass-Tarjan).

Exemple (Mansuy)

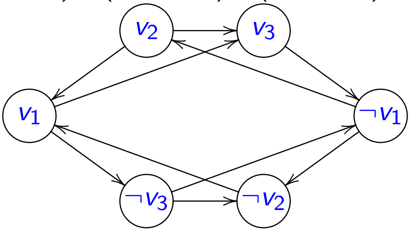
$p = (v_1 \vee v_2) \wedge (\neg v_1 \vee v_3) \wedge (v_1 \vee \neg v_2) \wedge (\neg v_2 \vee v_3) \wedge (\neg v_1 \vee \neg v_3)$ donne le graphe



- $\neg v_1$ accède à v_1 via v_2 et

Exemple (Mansuy)

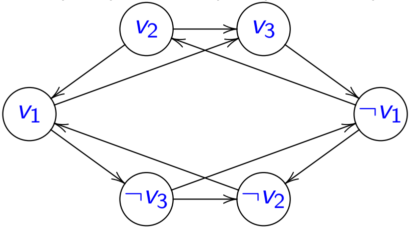
$p = (v_1 \vee v_2) \wedge (\neg v_1 \vee v_3) \wedge (v_1 \vee \neg v_2) \wedge (\neg v_2 \vee v_3) \wedge (\neg v_1 \vee \neg v_3)$ donne le graphe



- $\neg v_1$ accède à v_1 via v_2 et
- v_1 accède à $\neg v_1$ via $\neg v_3$

Exemple (Mansuy)

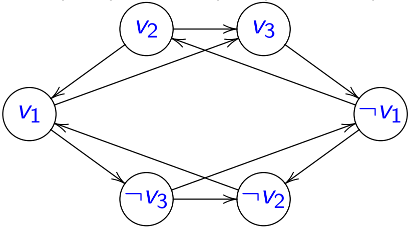
$p = (v_1 \vee v_2) \wedge (\neg v_1 \vee v_3) \wedge (v_1 \vee \neg v_2) \wedge (\neg v_2 \vee v_3) \wedge (\neg v_1 \vee \neg v_3)$ donne le graphe



- $\neg v_1$ accède à v_1 via v_2 et
- v_1 accède à $\neg v_1$ via $\neg v_3$
- Donc v_1 et $\neg v_1$ sont dans la même composante connexe.

Exemple (Mansuy)

$p = (v_1 \vee v_2) \wedge (\neg v_1 \vee v_3) \wedge (v_1 \vee \neg v_2) \wedge (\neg v_2 \vee v_3) \wedge (\neg v_1 \vee \neg v_3)$ donne le graphe



- $\neg v_1$ accède à v_1 via v_2 et
- v_1 accède à $\neg v_1$ via $\neg v_3$
- Donc v_1 et $\neg v_1$ sont dans la même composante connexe.
- Par suite p n'est pas satisfiable.

Problème SAT

- Le problème SAT est le problème algorithmique qui consiste, étant donnée une formule à décider en FNC si elle valide ou non.

Problème SAT

- Le problème SAT est le problème algorithmique qui consiste, étant donnée une formule à décider en FNC si elle valide ou non.
- Le problème est *NP-complet*, même pour le cas particulier 3-SAT où on n'autorise que les clauses d'au plus trois littéraux.

Problème SAT

- Le problème SAT est le problème algorithmique qui consiste, étant donnée une formule à décider en FNC si elle valide ou non.
- Le problème est *NP-complet*, même pour le cas particulier 3-SAT où on n'autorise que les clauses d'au plus trois littéraux.
- Un problème $\mathcal{P}$ NP-complet vérifie :

Problème SAT

- Le problème SAT est le problème algorithmique qui consiste, étant donnée une formule à décider en FNC si elle valide ou non.
- Le problème est *NP-complet*, même pour le cas particulier 3-SAT où on n'autorise que les clauses d'au plus trois littéraux.
- Un problème $\mathcal{P}$ NP-complet vérifie :
  - Il est possible de vérifier une solution de $\mathcal{P}$ efficacement (en temps polynomial) : on me donne un candidat solution, et je peux vérifier en temps polynomial ce qu'il en est. La classe des problèmes vérifiant cette propriété est notée *NP*.

Problème SAT

- Le problème SAT est le problème algorithmique qui consiste, étant donnée une formule à décider en FNC si elle valide ou non.
- Le problème est *NP-complet*, même pour le cas particulier 3-SAT où on n'autorise que les clauses d'au plus trois littéraux.
- Un problème $\mathcal{P}$ NP-complet vérifie :
  - Il est possible de vérifier une solution de $\mathcal{P}$ efficacement (en temps polynomial) : on me donne un candidat solution, et je peux vérifier en temps polynomial ce qu'il en est. La classe des problèmes vérifiant cette propriété est notée NP.
  - tous les problèmes de la classe NP se ramènent à $\mathcal{P}$ via une réduction polynomiale. Cela signifie que le problème est au moins aussi difficile que tous les autres problèmes de la classe NP (aspect « complet »). Appliqué à la satisfiabilité, cela signifie que si on me donne un problème de la classe NP, je peux le transformer en temps polynomial en un problème de satisfiabilité.

Problème SAT

- Le problème SAT est le problème algorithmique qui consiste, étant donnée une formule à décider en FNC si elle valide ou non.
- Le problème est *NP-complet*, même pour le cas particulier 3-SAT où on n'autorise que les clauses d'au plus trois littéraux.
- Un problème $\mathcal{P}$ NP-complet vérifie :
  - Il est possible de vérifier une solution de $\mathcal{P}$ efficacement (en temps polynomial) : on me donne un candidat solution, et je peux vérifier en temps polynomial ce qu'il en est. La classe des problèmes vérifiant cette propriété est notée NP.
  - tous les problèmes de la classe NP se ramènent à $\mathcal{P}$ via une réduction polynomiale. Cela signifie que le problème est au moins aussi difficile que tous les autres problèmes de la classe NP (aspect « complet »). Appliqué à la satisfiabilité, cela signifie que si on me donne un problème de la classe NP, je peux le transformer en temps polynomial en un problème de satisfiabilité.
- Bien sûr, un problème de la classe P (polynomial) vérifie la première condition. On a donc $P \subset NP$, mais a-t-on $NP \subset P$?