1 Notions

1. On dit qu'une branche de notre arbre *ferme* quand on trouve le long de cette branche une formule ϕ et sa négation $\neg\phi$. Nous représentons le fait qu'une branche est fermée par l'ajout d'une croix sous la branche ($\otimes$), et ce, dès lors que nous avons cette contradiction. Quand une branche est fermée, il ne faut plus rien écrire sous cette branche. Une branche commence *tout en haut de l'arbre*.
2. On dira qu'une branche de notre arbre est *ouverte* si elle n'est pas fermée.
3. Quand toutes les formules ont été traitées, si une branche est ouverte, c'est que nous avons trouvé un moyen de rendre vraie la formule qui se trouve en haut de l'arbre (et donc de rendre fausse la formule de départ).
4. Quand toutes les branches ferment, c'est que nous n'avons pas trouvé un moyen de rendre vraie la formule qui se trouve en haut de l'arbre (donc qu'elle est nécessairement fausse) ; c'est donc que la formule de départ est tautologique.
5. Il y a deux façons de construire un arbre :
   (a) En faisant une *pile*. On fait une pile quand la condition de vérité de la formule que l'on traite correspond à une conjonction.
   (b) En faisant une *branche*. On fait une branche quand la condition de vérité de la formule que l'on traite correspond à une disjonction.
6. Dresser l'arbre implique d'identifier, ici encore, le *connecteur principal* des formules que l'on traite.

2 Règles de construction pour les arbres

1. Règle $\vee$ (*branche*) : Quand une formule dont le connecteur principal est une disjonction apparaît, i.e. quand on a affaire à une formule de la forme $(\phi \vee \psi)$ au cours de notre arbre, on écrira en bas de chaque branche où cette formule apparaît :

$$\begin{array}{cc} \diagup & \diagdown \\ \phi & \psi \end{array}$$

2. Règle $\neg\vee$ (*pile*) : Quand une formule dont le connecteur principal est une négation de disjonction apparaît, i.e. quand on a affaire à une formule de la forme $\neg(\phi \vee \psi)$ au cours de notre arbre, on écrira en bas de chaque branche où cette formule apparaît :

$$\begin{array}{c} | \\ \neg\phi \\ | \\ \neg\psi \end{array}$$

3. Règle $\neg\wedge$ (*branche*) : Quand une formule dont le connecteur principal est une négation de conjonction apparaît, i.e. quand on a affaire à une formule de la forme $\neg(\phi \wedge \psi)$ au cours de notre arbre, on écrira en bas de chaque branche où cette formule apparaît :

$$\begin{array}{cc} \diagup & \diagdown \\ \neg\phi & \neg\psi \end{array}$$

4. Règle $\wedge$ (*pile*) : Quand une formule dont le connecteur principal est une conjonction apparaît, i.e. quand on a affaire à une formule de la forme $(\phi \wedge \psi)$ au cours de notre arbre, on écrira en bas de chaque branche où cette formule apparaît :

$$\begin{array}{c} | \\ \phi \\ | \\ \psi \end{array}$$

5. Règle $\rightarrow$ (*branche*) : Quand une formule dont le connecteur principal est une implication apparaît, i.e. quand on a affaire à une formule de la forme $(\phi \rightarrow \psi)$ au cours de notre arbre, on écrira en bas de chaque branche où cette formule apparaît :

$$\begin{array}{cc} \diagup & \diagdown \\ \neg\phi & \psi \end{array}$$

6. Règle $\neg \rightarrow$ (*pile*) : Quand une formule dont le connecteur principal est une négation d'implication apparaît, i.e. quand on a affaire à une formule de la forme $\neg(\phi \rightarrow \psi)$ au cours de notre arbre, on écrira en bas de chaque branche où cette formule apparaît :

$$\begin{array}{c} | \\ \phi \\ | \\ \neg\psi \end{array}$$

7. Règle $\leftrightarrow$ (*branche*) : Quand une formule dont le connecteur principal est une bi-implication apparaît, i.e. quand on a affaire à une formule de la forme $(\phi \leftrightarrow \psi)$ au cours de notre arbre, on écrira en bas de chaque branche où cette formule apparaît :

$$\begin{array}{cc} \multicolumn{2}{c}{\frown} \\ \phi & \neg\phi \\ | & | \\ \psi & \neg\psi \end{array}$$

8. Règle $\neg \leftrightarrow$ (*branche*) : Quand une formule dont le connecteur principal est une bi-implication apparaît, i.e. quand on a affaire à une formule de la forme $(\phi \leftrightarrow \psi)$ au cours de notre arbre, on écrira en bas de chaque branche où cette formule apparaît :

$$\begin{array}{cc} \multicolumn{2}{c}{\frown} \\ \phi & \neg\phi \\ | & | \\ \neg\psi & \psi \end{array}$$

9. Règle $\neg\neg$ (*pile*) : Quand une formule dont le connecteur principal est une négation de négation apparaît, i.e. quand on a affaire à une formule de la forme $\neg\neg\phi$ au cour de notre arbre, on écrira en bas de chaque branche où cette formule apparaît :

$$\begin{array}{c} | \\ \phi \end{array}$$

3 Méthode

Pour tester la validité d'un argument, on peut donc (au choix, puisque cela revient au même – la première vous épargne une étape) :

1. Empiler les prémisses de l'argument et la négation de la conclusion de l'argument. OU :
2. Prendre la négation de l'implication dont l'antécédent est la conjonction des prémisses et le conséquent la conclusion de l'argument. C'est ce que nous avons appelé la « formule en haut de l'arbre ».

   Vous pouvez bien entendu dresser un arbre pour n'importe quelle formule bien formée.
3. Vous pouvez appliquer les règles dans l'ordre que vous voulez (mais on va voir qu'il y a des stratégies pour faciliter la construction de l'arbre).
4. Une fois qu'une règle a été appliquée, vous cochez la formule sur laquelle vous avez appliqué la règle d'un petit $\checkmark$. Cela veut dire que cette formule a été traitée et qu'il ne faudra plus s'en occuper.
5. À chaque application d'une règle, vérifiez s'il y a ou non des branches qui ferment. S'il y a des branches qui ferment, notez le petit $\otimes$ sous la branche. Cela signifie qu'il ne faudra plus rien écrire en dessous.
6. Continuez à traiter les phrases non traitées jusqu'à ce que l'arbre soit complété, ce qui peut arriver dans deux cas :

   (a) Toutes les branches ferment.

   (b) Il ne reste plus de formules à traiter.

   On constate alors qu'il peut se faire que – i.e. qu'il est logiquement possible que – toutes les branches ferment *avant* que toutes les formules aient été traitées.
7. Une branche ouverte fournit une interprétation des lettres de proposition qui rend vraie la formule du haut de l'arbre. Pour cela, il faut attribuer la valeur « V » aux lettres de propositions élémentaires qui apparaissent sur cette branche en tant que formule complète. Il faut attribuer la valeur « F » aux lettres de propositions élémentaires qui apparaissent sur cette branche précédée d'un signe de négation (quand cette négation est la formule complète).
8. Si toutes les branches ferment, cela veut dire qu'il n'y a pas d'interprétation qui rende vraie la formule en haut de l'arbre. Puisqu'il n'y a pas d'interprétation qui rende vraie cette formule, cela veut dire que cette formule est une contradiction. Et puisque la formule en haut de l'arbre est la négation de notre formule de départ, cela veut dire que notre formule de départ est tautologique.
9. Si l'on cherche à évaluer un argument et que l'on a trouvé une branche ouverte en dressant l'arbre pour cet argument, c'est que nous avons trouvé une interprétation qui rende vraie(s) la ou les prémisses et fausse la conclusion. Nous avons donc trouvé un *contre-exemple* à la validité supposée de l'argument. Cela nous prouve que l'argument n'est pas valide (puisqu'un argument est valide quand il est impossible que les prémisses soient vraies et la conclusion fausse).
10. On écrira enfin, à côté de l'arbre : « Valide » ou « Invalide » en guise de conclusion de l'évaluation. Si l'argument est invalide, on pourra alors fournir un contre-exemple en donnant l'interprétation fournie par la ou une des branches ouvertes.

4 Conseils

Il est bon de suivre les conseils suivants :

1. Il est préférable de toujours commencer, si c'est possible, par les règles qui donnent lieu à des piles plutôt qu'à des branches.
2. Si les formules à traiter sont toutes des formules qui donnent lieu à des branches, il est préférable de choisir une formule qui va donner une ou des branches qui vont fermer (puisque cela évitera de devoir réécrire *sous* ces branches).

5 Exemples

Prenons un exemple. Soit la formule suivante : $(A \wedge B) \rightarrow (B \wedge A)$; nous avons plusieurs méthodes pour évaluer cette formule et décider en dernier lieu s'il s'agit d'une formule tautologique ou non. Utilisons la méthode des arbres :

$\checkmark \neg((A \wedge B) \rightarrow (B \wedge A))$

Nous nions la formule de départ

|

$\checkmark (A \wedge B)$

|

$\checkmark \neg(B \wedge A)$

Nous mettons ces deux formules en dessous car
la négation d'une implication est équivalente à la conjonction
de l'antécédent et la négation du conséquent

|

A

|

B

Ici nous avons traité la première conjonction
Reste la négation de conjonction

$\neg B$	$\neg A$
$\otimes$	$\otimes$

Toutes les branches de l'arbre sont fermées, la formule en haut de l'arbre est donc contradictoire, ce qui signifie que la formule de départ (la formule à évaluer) est tautologique.

Cherchons à démontrer si la formule suivante est tautologique en utilisant la méthode des arbres :

$$[(B \wedge C) \vee (\neg B \wedge A)] \rightarrow (C \vee A)$$

Nous partirons donc de la négation de cette formule et nous chercherons à la rendre vraie.

$\checkmark \neg\{[(B \wedge C) \vee (\neg B \wedge A)] \rightarrow (C \vee A)\}$

|

$\checkmark [(B \wedge C) \vee (\neg B \wedge A)]$

|

$\checkmark \neg(C \vee A)$

|

$\neg C$

|

$\neg A$

$\checkmark (B \wedge C)$	$\checkmark (\neg B \wedge A)$
|	|
B	$\neg B$
|	|
C	A
$\otimes$	$\otimes$

Nouvelle formule :

$$[(A \rightarrow B) \wedge (\neg B \rightarrow \neg A)] \leftrightarrow (\neg A \vee B)$$

Comme toujours, nous partons donc de sa négation :

$$\checkmark\neg\{[(A \to B) \wedge (\neg B \to \neg A)] \leftrightarrow (\neg A \vee B)\}$$

$$\begin{array}{c|c}
\checkmark[(A \to B) \wedge (\neg B \to \neg A)] & \checkmark\neg[(A \to B) \wedge (\neg B \to \neg A)] \\
\checkmark\neg(\neg A \vee B) & \checkmark(\neg A \vee B) \\
\checkmark(A \to B) & \begin{array}{c|c} \neg A & B \end{array} \\
(\neg B \to \neg A) & \begin{array}{c|c|c|c} \checkmark\neg(A \to B) & \checkmark\neg(\neg B \to \neg A) & \checkmark\neg(A \to B) & \checkmark\neg(\neg B \to \neg A) \\ A & \neg B & A & \neg B \\ \neg B & \neg\neg A & \neg B & \neg\neg A \\ \otimes & A & \otimes & A \\ & \otimes & & \otimes \end{array} \\
\checkmark\neg\neg A & \\
A & \\
\neg B & \\
\begin{array}{c|c} \neg A & B \\ \otimes & \otimes \end{array} &
\end{array}$$

Soit la formule suivante ; évaluons-la avec la méthode des arbres de vérité.

$$[(A \wedge B) \to (C \vee F)] \vee \neg[(A \to B) \wedge \neg(A \to B)]$$

Nous partons donc de sa négation :

$$\begin{array}{c}
\checkmark\neg\{[(A \wedge B) \to (C \vee F)] \vee \neg[(A \to B) \wedge \neg(A \to B)]\} \\
| \\
\neg[(A \wedge B) \to (C \vee F)] \\
| \\
\checkmark\neg\neg[(A \to B) \wedge \neg(A \to B)] \\
| \\
\checkmark[(A \to B) \wedge \neg(A \to B)] \\
| \\
(A \to B) \\
| \\
\neg(A \to B) \\
\otimes
\end{array}$$

Ici on voit qu'il n'est pas nécessaire d'aller plus loin dans l'arbre : toutes les branches ferment (il n'y en a qu'une). Il n'est pas nécessaire de traiter les formules non traitées. Nous avons ainsi prouvé que la formule de départ était tautologique (l'ordre choisi nous épargne ici 8 lignes superflues).

Autre exemple similaire :

$$[(A \wedge B) \vee (D \wedge E)] \to \{[(A \to B) \wedge C] \to C\}$$

Nous partons donc de la négation de notre formule :

$$\begin{array}{c}
\checkmark\neg\{[(A \wedge B) \vee (D \wedge E)] \to \{[(A \to B) \wedge C] \to C\}\} \\
| \\
[(A \wedge B) \vee (D \wedge E)] \\
| \\
\checkmark\neg\{[(A \to B) \wedge C] \to C\} \\
| \\
\checkmark[(A \to B) \wedge C] \\
| \\
\neg C \\
| \\
(A \to B) \\
| \\
C \\
\otimes
\end{array}$$