Logique des propositions partie 3

S. Mazouz & K. Akli
USTHB
2015-2016

Sémantique
Formes normales

Définition (Littéral)

On appelle littéral associé à une variable propositionnelle A chacune des deux expressions A et $\neg A$ (i.e la variable elle-même et sa négation).

Etant données deux variables propositionnelles A et B, on peut construire les conjonctions et les disjonctions de littéraux suivants :

A	B	Conjonctions	disjonctions
V	V	$\varphi 1 = A \wedge B$	$\psi 1 = \neg A \vee \neg B$
V	F	$\varphi 2 = A \wedge \neg B$	$\psi 2 = \neg A \vee B$
F	V	$\varphi 3 = \neg A \wedge B$	$\psi 3 = A \vee \neg B$
F	F	$\varphi 4 = \neg A \wedge \neg B$	$\psi 4 = A \vee B$

Noter que φi=V et ψi=F pour une unique et même instanciation (i=1, 4)

Sémantique

Formes normales

Définition (Forme normale disjonctive et forme normale conjonctive)

- On appelle forme normale disjonctive (FND) toute disjonction de conjonctions de littéraux.
- On appelle forme normale conjonctive (FNC) toute conjonction de disjonctions de littéraux.

Exemples

- $(A \wedge \neg B) \vee (\neg A \wedge B)$ est une forme normale disjonctive
- $(\neg A \vee B) \wedge (A \vee B)$ est une forme normale conjonctive

Théorème

Toute formule α est logiquement équivalente à une forme normale disjonctive (et à une forme normale conjonctive).

Sémantique

Formes normales

Exemple Mise en forme normale disjonctive pour la formule $\alpha = (P \vee Q \rightarrow R) \wedge (P \leftrightarrow R)$

1. Dresser la table de vérité de la formule
2. Construire une disjonction des conjonctions de littéraux associées aux instanciations rendant Vraie α

P	Q	R	$P \vee Q$	$P \vee Q \rightarrow R$	$P \leftrightarrow R$	α	Conjonctions
V	**V**	**V**	V	V	V	V	
V	**V**	**F**	V	F	F	F	
V	**F**	**V**	V	V	V	V	
V	**F**	**F**	V	F	F	F	
F	**V**	**V**	V	V	F	F	
F	**V**	**F**	V	F	V	F	
F	**F**	**V**	F	V	F	F	
F	**F**	**F**	F	V	V	V	

$FND(\alpha) = (P \wedge Q \wedge R) \vee (P \wedge \neg Q \wedge R) \vee (\neg P \wedge \neg Q \wedge \neg R)$

Relation entre syntaxe et Sémantique

Théorème de consistance

Soient $\alpha_1, \ldots, \alpha_n, \alpha$ des formules du langage $L_p(\neg, \wedge)$.

1. $\alpha_1, \ldots, \alpha_n \vdash \alpha \;\; \Rightarrow \;\; \models (\alpha_1 \wedge \ldots \wedge \alpha_n) \rightarrow \alpha$
2. $\vdash \;\; \Rightarrow$ ssi $\models \alpha$

Relation entre syntaxe et Sémantique

Preuve du théorème de consistance

On appelle longueur de la déduction le nombre d'applications de règles de déductions.

Une longueur de déduction peut prendre la valeur minimale zéro pour une déduction du type

$$\alpha_1, \ldots, \alpha_n \vdash \alpha_i \ (i=1, n).$$

En d'autres termes, la conclusion est l'une des prémisses de la déduction.

Le théorème sera démontré par récurrence sur la longueur m de la déduction D de α à partir de $\{\alpha_1, \ldots, \alpha_n\}$.

Relation entre syntaxe et Sémantique

Cas de base (m = 0)

Comme la déduction est de longueur zéro , α est l'un des α_i càd il existe i tel que $\alpha_i = \alpha$

Montrons que $\models (\alpha_1 \wedge \ldots \wedge \alpha_n) \rightarrow \alpha_i$ (i=1,n).

Deux cas se présentent :

- $\alpha_i = F \Rightarrow \alpha_1 \wedge \ldots \wedge \alpha_n = F \Rightarrow (\alpha_1 \wedge \ldots \wedge \alpha_n) \rightarrow \alpha_i = V$
- $\alpha_i = V \Rightarrow$ quelque soit la valeur de vérité de $\alpha_1 \wedge \ldots \wedge \alpha_n$, on a $(\alpha_1 \wedge \ldots \wedge \alpha_n) \rightarrow \alpha_i = V$

Donc $\models (\alpha_1 \wedge \ldots \wedge \alpha_n) \rightarrow \alpha$

Relation entre syntaxe et Sémantique

Cas général

Supposons le résultat est vrai jusqu'à l'ordre m et montrons qu'il reste vrai à l'ordre m+1.

Soit D une déduction de longueur m+1. Il s'agira de décomposer la déduction D en une déduction D' de longueur m et la dernière règle utilisée. La règle de déduction peut être l'une des 4 règles : E$\wedge$, I $\wedge$, $\neg$E, $\neg$ I.

Relation entre syntaxe et Sémantique

<u>supposons que la règle est $E\wedge$</u>

La déduction D peut être décomposée en une déduction D' de longueur m suivi d''une règle d''élimination comme illustrée ci-dessous :

$$\left.\begin{array}{c} \dfrac{\left.\begin{array}{c}\alpha_1 \ldots\ldots\ldots\ldots \alpha_n \\ \\ \\ \alpha\wedge\beta\end{array}\right\}\text{Déduction D'}}{\alpha}\; E\wedge \end{array}\right\}\text{Déduction D}$$

On a.. $\alpha_1, \ldots, \alpha_n \vdash \alpha\wedge\beta$

Donc par hypothèse de récurrence sur la déduction D' qui est de longueur m, on a alors

$$\models (\alpha_1\wedge\ldots \wedge \alpha_n)\rightarrow \alpha \wedge \beta$$

$\Rightarrow$ Pour chaque instanciation, $(\alpha_1\wedge\ldots \wedge \alpha_n) = F$ ou $\alpha \wedge \beta = V$

$\Rightarrow$ Pour chaque instanciation, $(\alpha_1 \wedge\ldots \wedge \alpha_n) = F$ ou $\alpha = V$

$\Rightarrow \models (\alpha_1\wedge\ldots \wedge \alpha_n)\rightarrow \alpha$

Les 3 autres cas sont similaires

Relation entre syntaxe et Sémantique

Théorème de complétude

Soient $\alpha_1, \ldots, \alpha_n, \alpha$ des formules du langage $L_p(\neg, \wedge)$.

1. $\models (\alpha_1 \wedge \ldots \wedge \alpha_n) \rightarrow \alpha \quad \Rightarrow \quad \alpha_1, \ldots, \alpha_n \vdash \alpha$
2. $\models \alpha \quad \Rightarrow \quad \vdash \alpha$

Relation entre syntaxe et Sémantique

Lemme 1

$\alpha_1, \ldots, \alpha_n \vdash \alpha \iff \{\alpha_1, \ldots, \alpha_n, \neg\alpha\}$ est inconsistant

c-à-d $\alpha_1, \ldots, \alpha_n, \neg\alpha \vdash \perp$

Relation entre syntaxe et Sémantique

Lemme 2

Tout ensemble de formules consistant est satisfiable

Remarque

Consistant $\Rightarrow$ Satisfiable

donc Non satisfiable $\Rightarrow$ Inconsistant (Contraposée)

Relation entre syntaxe et Sémantique

Preuve du théorème de complétude

Hypothèse $\models (\alpha_1 \wedge \ldots \wedge \alpha_n) \rightarrow \alpha$

Pour montrer que $\alpha_1, \ldots, \alpha_n \vdash \alpha$, on tentera d'établir d'abord que $\{\alpha_1, \ldots, \alpha_n, \neg\alpha\}$ est non satisfiable.

Nous avons deux cas :

1er cas : $\alpha_1, \ldots, \alpha_n$ sont vraies en même temps

Donc $\alpha_1 \wedge \ldots \wedge \alpha_n$ est vraie

Or $(\alpha_1 \wedge \ldots \wedge \alpha_n) \rightarrow \alpha$ est vraie par hypothèse

Donc la seule valeur de vérité possible que peut prendre α est vraie

Donc $\neg\alpha$ est à faux

Par conséquent $\{\alpha_1, \ldots, \alpha_n, \neg \alpha\}$ est non satisfiable.

Relation entre syntaxe et Sémantique

(suite preuve théorème de complétude)

2ème cas : l'un des α_i (i=1,n) est faux

Donc $\{\alpha_1, \ldots, \alpha_n, \neg\alpha\}$ est non satisfiable

D'après la contraposée du lemme 2, on déduit que:

$\{\alpha_1, \ldots, \alpha_n, \neg\alpha\}$ est inconsistant.

Rappel Contraposée du lemme 2

Non satisfiable $\Rightarrow$ Inconsistant

D'après le lemme 1, on obtient :

$\alpha_1, \ldots, \alpha_n \vdash \alpha$

Rappel du Lemme 1

$\alpha_1, \ldots, \alpha_n \vdash \alpha \Leftrightarrow \{\alpha_1, \ldots, \alpha_n, \neg\alpha\}$ est inconsistant

Relation entre syntaxe et Sémantique

Preuve du lemme 1

(⇒) Hypothèse $\alpha_1, \ldots, \alpha_n \vdash \alpha$

Montrons que $\{\alpha_1, \ldots, \alpha_n, \neg\alpha\}$ est inconsistant

$$\dfrac{\text{(Hyp)}\ \dfrac{\alpha_1 \ \ldots\ldots\ \alpha_n}{\alpha} \qquad \neg\alpha}{\bot}$$

Rappel du Lemme 1

$\alpha_1, \ldots, \alpha_n \vdash \alpha \iff \{\alpha_1, \ldots, \alpha_n, \neg\alpha\}$ est inconsistant

Relation entre syntaxe et Sémantique

(suite preuve lemme 1)

($\Leftarrow$) Hypothèse $\{\alpha_1, \ldots, \alpha_n, \neg\alpha\}$ est inconsistant

Montrer que $\alpha_1, \ldots, \alpha_n \vdash \alpha$

Relation entre syntaxe et Sémantique

Algorithme de réfutation

Preuve du lemme 2

La preuve est donnée par l'algorithme de réfutation suivant :

On considère un ensemble Γ de formules ayant une propriété donnée (satisfiable ou inconsistant). Transformer Γ en plusieurs ensembles de littéraux en préservant la satisfiabilité et l'inconsistance.

Soit $\Gamma = \{\alpha_1, \ldots, \alpha_n\}$

- **1er cas** Γ est composé de littéraux uniquement, donc $\Gamma = \{A_1, \ldots, A_p, \neg B_1, \ldots, \neg B_q\}$

Il existe deux possibilités qui s'excluent :

1. **$\exists i, \exists j, \ A_i = B_j$** ie que Γ contient une variable propositionnelle et sa négation.

   Donc Γ est non satisfiable $\Rightarrow \Gamma$ est inconsistant

2. **$\forall i, \forall j, A_i \neq B_j$** ie Γ est consistant

$\Rightarrow \Gamma$ est satisfiable, il suffit de prendre A_i=V (i=1..p) et B_j=F (j=1…q)

Conclusion : un ensemble de littéraux Γ est inconsistant s'il comporte un couple de littéraux de la forme $(A, \neg A)$ sinon Γ est satisfiable.

Relation entre syntaxe et Sémantique
Algorithme de réfutation

2ème cas Γ comporte au moins une formule autre qu'un littéral.

On a trois cas possibles : cette formule est

- soit de la forme $\boldsymbol{\alpha \wedge \beta}$,
- soit de la forme $\neg\neg\alpha$,
- **et** soit de la forme $\neg(\alpha \wedge \beta)$

1. Règle R1 :

$\Gamma = \Sigma \cup \{\alpha \wedge \beta\}$ **sera remplacé par** $\Gamma' = \Sigma \cup \{\alpha, \beta\}$

On peut montrer que

- Γ' est satisfiable $\Rightarrow \Gamma$ est satisfiable
- Γ' est inconsistant $\Rightarrow \Gamma$ est inconsistant

Relation entre syntaxe et Sémantique
Algorithme de réfutation

2. Règle R2 :

$\Gamma = \Sigma \cup \{\neg\neg\alpha\}$ sera remplacé par $\Gamma' = \Sigma \cup \{\alpha\}$

On peut montrer aussi que

- Γ' est satisfiable $\Rightarrow$ Γ est satisfiable
- Γ' est inconsistant $\Rightarrow$ Γ est inconsistant

Relation entre syntaxe et Sémantique
Algorithme de réfutation

3. **Règle R3 :**

$\Gamma = \Sigma \cup \{\neg(\alpha \wedge \beta)\}$ **sera remplacé par**

$\Gamma_1 = \Sigma \cup \{\neg \alpha\}$ **et** $\Gamma_2 = \Sigma \cup \{\neg \beta\}$

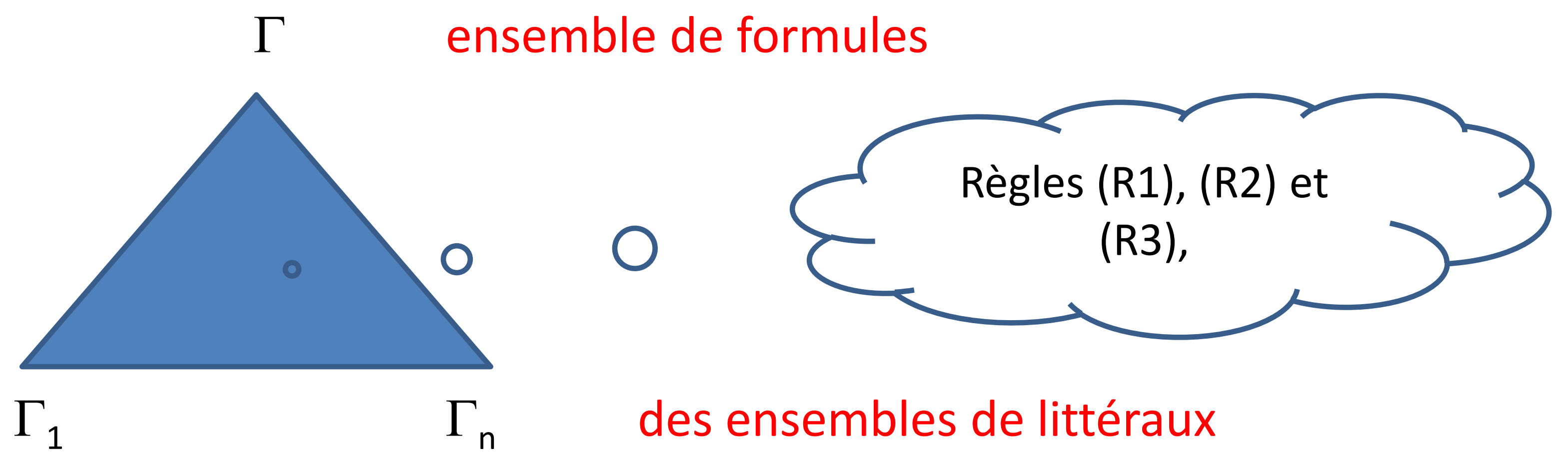
On peut montrer que

- Γ_1 est satisfiable ou Γ_2 est satisfiable
  $\Rightarrow \Gamma$ est satisfiable
- Γ_1 est inconsistant et Γ_2 est inconsistant
  $\Rightarrow \Gamma$ est inconsistant

Relation entre syntaxe et Sémantique

Algorithme de réfutation

En appliquant à Γ les règles (R1), (R2) et (R3), tant que cela possible, on obtient un arbre dont les feuilles sont des ensembles Γ_i de littéraux. On revient alors au 1^{er} cas. Ainsi on peut décider pour chaque Γ_i s'il est satisfiable ou inconsistant.



Pour Γ, on a deux possibilités :

- Il existe au moins un Γ_i satisfiable alors Γ est satisfiable
- Tous les Γ_i sont inconsistants alors Γ est inconsistant

Relation entre syntaxe et Sémantique
Algorithme de réfutation

Exemple

Appliquer l'algorithme de réfutation sur

$\Gamma = \{\neg C \rightarrow A \wedge B,\ B \vee \neg C,\ \neg A \wedge B\}$

1. Remplacer chaque connecteur d'abréviation par sa définition dans le langage Lp($\neg$, $\wedge$).

On obtient $\Gamma = \{\neg (\neg C \wedge \neg(A \wedge B)),\ \neg(\neg B \wedge \neg\neg C),\ \neg A \wedge B\}$

2. A chaque étape, on doit choisir dans Γ une formule qui ne soit pas un littéral puis appliquer la règle appropriée.

Relation entre syntaxe et Sémantique

Algorithme de réfutation

$$\Gamma = \{\neg (\neg C \wedge \neg(A\wedge B)), \neg(\neg B \wedge \neg\neg\neg C), \underline{\neg A \wedge B}\}$$

R1

$$\{\neg (\neg C \wedge \neg(A\wedge B)), \underline{\neg(\neg B \wedge \neg\neg\neg C)}, \neg A, B\}$$

R3

$$\{\underline{\neg (\neg C \wedge \neg(A\wedge B)), \neg\neg B}, \neg A, B\} \quad \{\neg (\neg C \wedge \neg(A\wedge B)), \neg\neg\neg C, \neg A, B\}$$

R3

$$\{\underline{\neg\neg C}, \neg\neg B, \neg A, B\} \quad \{\neg\neg(A\wedge B), \neg\neg B, \neg A, B\}$$

R2

$$\{C, \underline{\neg\neg B}, \neg A, B\}$$

R2

$$\Gamma' = \{C, B, \neg A\}$$

Γ' est un ensemble de littéraux qui est satisfaisable (il suffit de prendre C=V, B=V et A=F). Donc Γ est satisfaisable.

Relation entre syntaxe et Sémantique

Algorithme de réfutation

Exemple

Montrer la déduction suivante en utilisant l'algorithme de réfutation :

$(P \vee \neg R) \rightarrow \neg Q \wedge R, \; P \; |\text{--} \; \neg Q \wedge R$

D'après le lemme 1, la déduction est équivalente à montrer que l'ensemble $\Gamma = \; (P \vee \neg R) \rightarrow \neg Q \wedge R, \; P, \neg(\neg Q \wedge R)\}$

est inconsistant.

On remplace les connecteurs d'abréviations par leur définition

$\Gamma = \; \{\neg(\neg(\neg P \wedge \neg \neg R) \wedge \neg(\neg Q \wedge R)), \; P, \; \neg(\neg Q \wedge R)\}$

Relation entre syntaxe et Sémantique

Algorithme de réfutation

$\Gamma = \{\neg(\neg(\neg P \wedge \neg\neg R) \wedge \neg(\neg Q \wedge R)),\ P,\ \neg(\neg Q \wedge R)\}$

$\Gamma 1 = \{\neg\neg(\neg P \wedge \neg\neg R),\ P,\ \neg(\neg Q \wedge R)\}$

$\Gamma 2 = \{\neg\neg(\neg Q \wedge R),\ P,\ \neg(\neg Q \wedge R)\}$

$\Gamma 3 = \{(\neg P \wedge \neg\neg R),\ P,\ \neg(\neg Q \wedge R)\}$

$\Gamma 4 = \{\neg P,\ \neg\neg R,\ P,\ \neg(\neg Q \wedge R)\}$

$\Gamma 5 = \{\neg P,\ R,\ P,\ \neg(\neg Q \wedge R)\}$

$\Gamma 6 = \{\neg P,\ R,\ P,\ \neg\neg Q\}$

$\Gamma 7 = \{\neg P,\ R,\ P,\ \neg R)\}$

$\Gamma 8 = \{\neg P,\ R,\ P,\ Q\}$

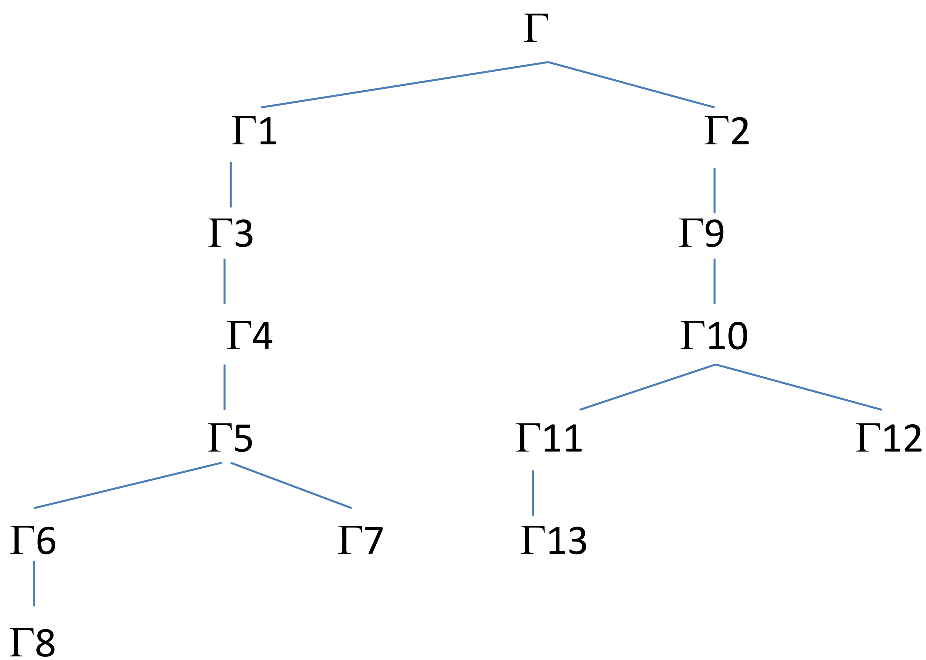
$\Gamma 9 = \{(\neg Q \wedge R),\ P,\ \neg(\neg Q \wedge R)\}$

$\Gamma 10 = \{\neg Q,\ R,\ P,\ \neg(\neg Q \wedge R)\}$

$\Gamma 11 = \{\neg Q,\ R,\ P,\ \neg\neg Q\}$

$\Gamma 12 = \{\neg Q,\ R,\ P,\ \neg R\}$

$\Gamma 13 = \{\neg Q,\ R,\ P,\ Q\}$



Tous les ensembles de littéraux obtenus Γ7, Γ8, Γ12 et Γ13 sont inconsistants donc Γ est inconsistant.

Relation entre syntaxe et Sémantique

Exercice

Dans un rapport d'enquête, on trouve les affirmations suivantes :

α_1 : « Si Ali dit la vérité alors Ali n'est pas l'assassin »

α_2 : « Si Ali ne dit pas la vérité alors le crime a eu lieu après minuit et Ali est l'assassin »

α_3 : « Ali n'est pas l'assassin ou le crime n'a pas eu lieu après minuit »

Soient les variables propositionnelles suivantes :

A : Ali est l'assassin D : Ali dit la vérité C : le crime a eu lieu après minuit

Les affirmations sont représentées par les formules suivantes :

$\alpha_1 = D \rightarrow \neg A$ $\alpha_2 = \neg D \rightarrow C \wedge A$ $\alpha_3 = \neg A \vee \neg C$

On veut répondre à la question : « Ali est-il l'assassin ? »

1. Tableau de vérité : Vérifier si $\alpha_1, \alpha_2, \alpha_3 \models A$

Vérifier, pour toute instanciation où les formules α_1, α_2 et α_3 sont vraies, que la variable propositionnelle A est aussi vraie.

Relation entre syntaxe et Sémantique

A	D	C	¬A	D→ ¬A	¬D	C∧A	¬D→C∧A	¬C	¬A∨¬C
V	V	V							
V	V	F							
V	F	V							
V	F	F							
F	V	V							
F	V	F							
F	F	V							
F	F	F							

Relation entre syntaxe et Sémantique

Déduction : Démontrer que $\alpha_1, \alpha_2, \alpha_3 \vdash \neg A$

On a :

$\alpha_1 = D \rightarrow \neg A = \neg(D \wedge \neg\neg A)$

$\alpha_2 = \neg D \rightarrow C \wedge A = \neg(\neg D \wedge \neg (C \wedge A))$

$\alpha_3 = \neg A \vee \neg C = \neg(\neg \neg A \wedge \neg\neg C)$

Montrons que

$\neg(D \wedge \neg\neg A), \neg(\neg D \wedge \neg (C \wedge A)), \neg(\neg \neg A \wedge \neg\neg C) \vdash \neg A$

(Utiliser la déduction $\alpha \vdash \neg\neg \alpha$)

Relation entre syntaxe et Sémantique

3. Utiliser l'algorithme de réfutation pour savoir si Ali est l'assassin ou non, s'il dit la vérité ou non et si le crime a eu lieu ou non après minuit.

Soit L un littéral tel que $L \in \{A, \neg A, D, \neg D, C, \neg C\}$ (ces littéraux correspondent aux différentes situations possibles)

Appliquer l'algorithme de réfutation sur l'ensemble $\Gamma = \{\alpha_1, \alpha_2, \alpha_3, \neg L\}$