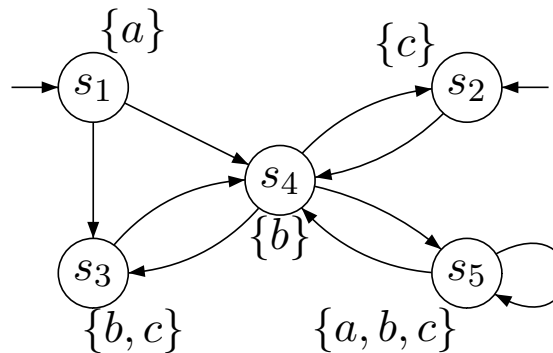


3.Algorithmes de Model- Checking

Model-Checking de LTL

- **Données** : Une structure de Kripke $M=(Q,T,A, q_0,AP, I)$ et une formule LTL φ .
- **Question** : Est-ce que $M \models \varphi$?
 - $M \models \varphi$ ssi $t,0 \models \varphi$ pour toute trace initiale t de M .

Exercise



$M \models \varphi$?

- $\varphi = FGc$
- $\varphi = GFc$
- $\varphi = Ga$
- $\varphi = aU(G(b \vee c))$
- $X\neg c \rightarrow XXc$

Model-Checking de LTL: principe

- Soit Σ un alphabet. On note Σ^* l'ensemble des mots finis et Σ^ω les mots infinis.
- Modèles de φ = mots infinis. Soit $\llbracket \varphi \rrbracket$ le langage des modèles de la formule : $\llbracket \varphi \rrbracket = \{t \in (2^{AP})^\omega \mid t, 0 \models \varphi\}$
- Soit $\llbracket M \rrbracket$ le langage des traces initiales de M : $\llbracket M \rrbracket = \{t \in (2^{AP})^\omega \mid t \text{ est une trace initiale de } M\}$
- Le problème du model-checking revient donc à vérifier si : $\llbracket M \rrbracket \subseteq \llbracket \varphi \rrbracket$

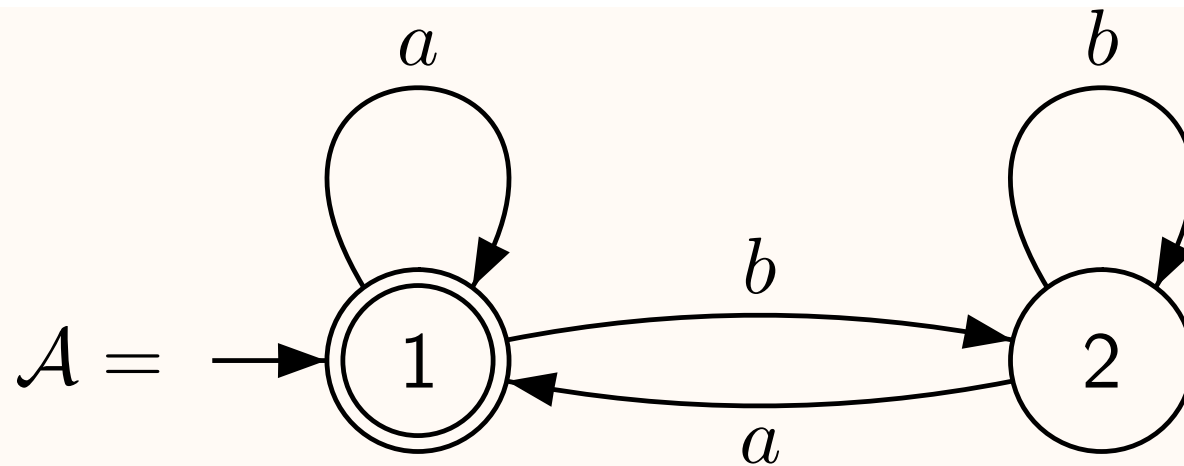
Outil : les automates de Büchi

- **Définition** : Un automate de Büchi est un n-uplet $A=(Q, \Sigma, I, T, F)$ avec
 - Q un ensemble fini d'états
 - Σ un alphabet fini
 - $I \subseteq Q$ les états initiaux
 - $T \subseteq Q \times \Sigma \times Q$ la relation de transition
 - $F \subseteq Q$ un ensemble d'états acceptants (ou répétés)

Outil : les automates de Büchi

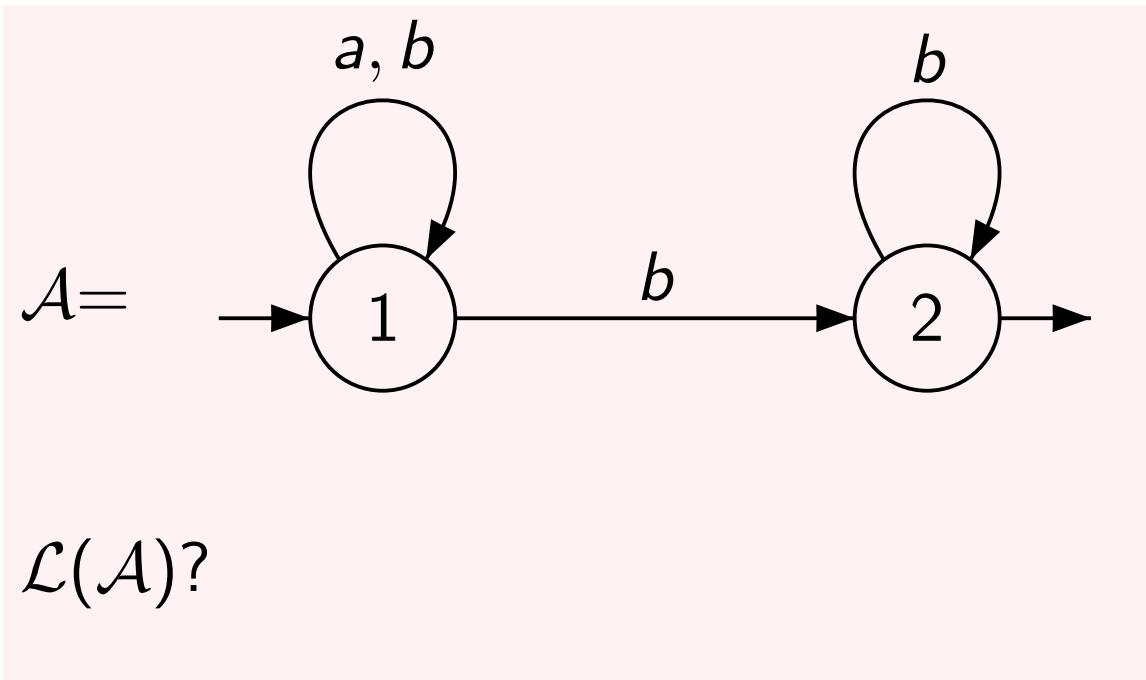
- Une **exécution** de A sur un mot infini $w=w_0w_1w_2\dots$ de Σ^ω est une séquence $r=q_0q_1q_2q_3\dots$ telle que $q_0 \in I$ et $(q_i, w_i, q_{i+1}) \in T$, pour tout $i \geq 0$.
- r est **acceptante** si $q_i \in F$ pour un nombre infini de i .
- w est **accepté par A** s'il existe une exécution acceptante de A sur w .
- $L(A) = \{w \in \Sigma^\omega \mid w \text{ accepté par } A\}$.

Automate de Büchi: exemple



$$\mathcal{L}(\mathcal{A}) = \{w \in \{a, b\}^\omega \mid |w|_a = \omega\}$$

Automate de Büchi: exemple



Automates de Büchi non-déterministes

- Les automates de Büchi non déterministes sont plus expressifs que les automates de Büchi déterministes
- Les langages reconnus par un NBA forment les ω -réguliers
- Toute formule de LTL peut être reconnue par un NBA

Les automates de Büchi pour LTL

- Définition : Un automate de Büchi est un n-uplet $A=(Q, \Sigma, I, T, F)$ avec
 - Q un ensemble fini d'états
 - Σ un alphabet fini $\Sigma = 2^{AP}$
 - $I \subseteq Q$ les états initiaux
 - $T \subseteq Q \times \Sigma \times Q$ la relation de transition
 - $F \subseteq Q$ un ensemble d'états acceptants (ou répétés)

Exercice

- Exemple : automate de Büchi reconnaissant p, Xp .
- Construire des automates de Büchi reconnaissant $Fp, XXp, Gp, FGp, GFp, pUq, pRq$.

Automates de Büchi et LTL

- Les formules de LTL sont moins expressives que les automates de Büchi
- Exemple : «Un instant sur deux, l'événement a arrive.» est une propriété ω régulière non exprimable en LTL

Automates de Büchi

Théorème : Les automates de Büchi sont clos par union, intersection, et complément.

Théorème : on peut tester le vide d'un automate de Büchi.

Automates de Büchi - Test du vide

- Chercher si un état acceptant est accessible depuis l'état initial
- Chercher si cet état appartient à un cycle

Model-Checking LTL : approche par automates

- Donnée: Structure de Kripke M , formule LTL φ .
- Etapes de l'algorithme :
 - Transformer M en un automate A_M tel que $L(A_M) = \llbracket M \rrbracket$ (assez facile)
 - Transformer φ en un automate A_φ tel que $L(A_\varphi) = \llbracket \varphi \rrbracket$ (difficile)
 - Tester si $L(A_M) \subseteq L(A_\varphi)$, i.e., si $L(A_M) \cap L(A_\varphi)^c = \emptyset$.

Difficile de compléter un automate de Büchi!!

Model-Checking LTL : approche par automates

- Donnée: Structure de Kripke M , formule LTL φ .
- Etapes de l'algorithme :
 - Transformer M en un automate A_M tel que $L(A_M) = \llbracket M \rrbracket$
 - Transformer φ en un automate $A_{\neg\varphi}$ tel que $L(A_{\neg\varphi}) = \llbracket \neg\varphi \rrbracket$
 - Tester si $L(A_M) \cap L(A_{\neg\varphi}) = \emptyset$.

Transformer φ en un automate de Büchi

- I. Automates de Büchi généralisés
- II. Réduire la formule
 - I. Forme normale négative
 - II. Réduire les connecteurs temporels
- III. Construire un graphe
- IV. Transformation en automate de Büchi

Transformer φ en un automate de Büchi

- I. Automates de Büchi généralisés
- II. Réduire la formule
 - I. Forme normale négative
 - II. Réduire les connecteurs temporels
- III. Transformation en automate de Büchi généralisé

Automates de Büchi généralisés

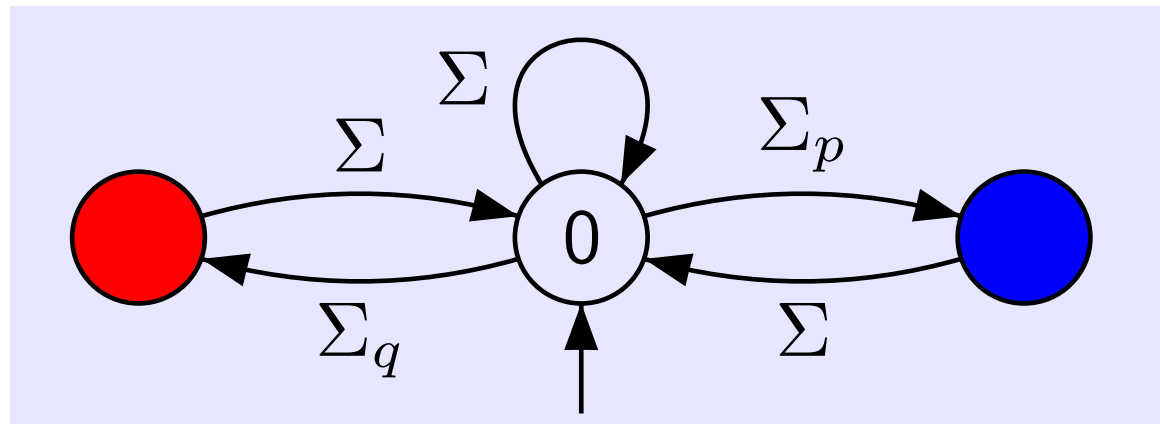
- Définition : Un automate de Büchi généralisé est un n-uplet $A=(Q, \Sigma, I, T, F)$ avec
 - Q un ensemble fini d'états
 - Σ un alphabet fini
 - $I \subseteq Q$ les états initiaux
 - $T \subseteq Q \times \Sigma \times Q$ la relation de transition
 - $F=\{F_1, F_2, \dots, F_k\} \subseteq 2^Q$ un ensemble d'ensemble d'états acceptants (ou répétés)

Automates de Büchi généralisés

- Une **exécution** de A sur un mot infini $w = w_0w_1w_2\dots$ de Σ^ω est une séquence $r = q_0q_1q_2q_3\dots$ telle que $q_0 \in I$ et $(q_i, w_i, q_{i+1}) \in T$, pour tout $i \geq 0$.
- r est **acceptante** si **pour tout $\mathcal{F} \in \mathcal{F}$, $q_i \in \mathcal{F}$ pour un nombre infini de i .**
- w est **accepté par A** s'il existe une exécution acceptante de A sur w .
- $L(A) = \{w \in \Sigma^\omega \mid w \text{ accepté par } A\}$.

Automates de Büchi généralisés : exemple

$GFp \wedge GFq$:

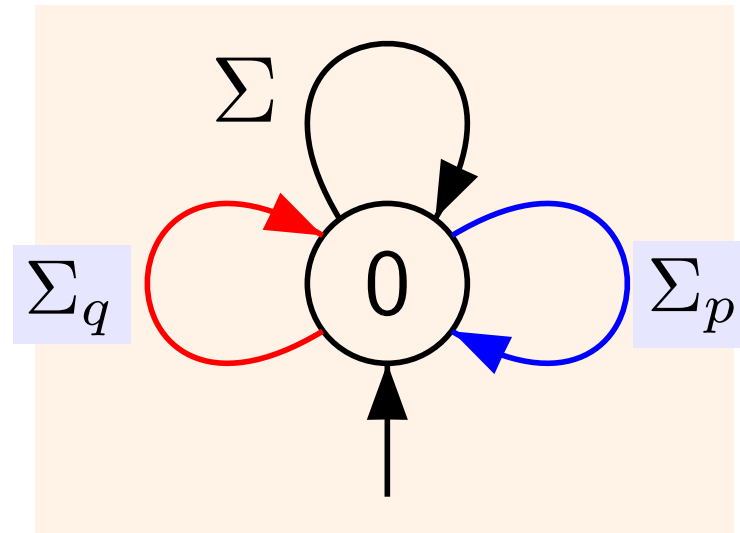


Automates de Büchi généralisés avec condition sur les transitions

- Définition : Un automate de Büchi généralisé avec condition sur les transitions est un n-uplet $A=(Q, \Sigma, I, T, F)$ avec
 - Q un ensemble fini d'états
 - Σ un alphabet fini
 - $I \subseteq Q$ les états initiaux
 - $T \subseteq Q \times \Sigma \times Q$ la relation de transition
 - $T=\{T_1, T_2, \dots, T_k\} \subseteq 2^T$ un ensemble d'ensemble de transitions acceptantes (ou répétées)

Automates de Büchi généralisés avec condition sur les transitions - exemple

$GFp \wedge GFq$:



Des ABG aux AB

Théorème : Tout automate de Büchi généralisé A peut être transformé en un automate de Büchi A' tel que $L(A) = L(A')$

Transformer φ en un automate de Büchi

- I. Automates de Büchi généralisés
- II. Réduire la formule
 - I. Forme normale négative
 - II. Réduire les connecteurs temporels
- III. Transformation en automate de Büchi généralisé

Forme normale négative

$\varphi ::= \perp \mid \top \mid p \mid \neg p \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid$
 $X\varphi \mid \varphi U \varphi \mid \varphi R \varphi$

- $\neg \neg p = p$
- $\neg(\varphi_1 \vee \varphi_2) = \neg \varphi_1 \wedge \neg \varphi_2$
- $\neg(\varphi_1 \wedge \varphi_2) = \neg \varphi_1 \vee \neg \varphi_2$
- $\neg(X\varphi) = X(\neg \varphi)$
- $\neg(\varphi_1 U \varphi_2) = \neg \varphi_1 R \neg \varphi_2$
- $\neg(\varphi_1 R \varphi_2) = \neg \varphi_1 U \neg \varphi_2$

Exercice

- Transformer $G(p \rightarrow Fq)$ en forme normale négative

Réduire les connecteurs temporels

- Idée : Un état de notre graphe va représenter l'ensemble des propositions atomiques vérifiées au prochain instant de la séquence, et l'ensemble des sous-formules qu'il «promet» de vérifier à l'état suivant.
- Pour cela, on ne veut que des propositions atomiques (ou négations), et des sous-formules commençant par X (next).

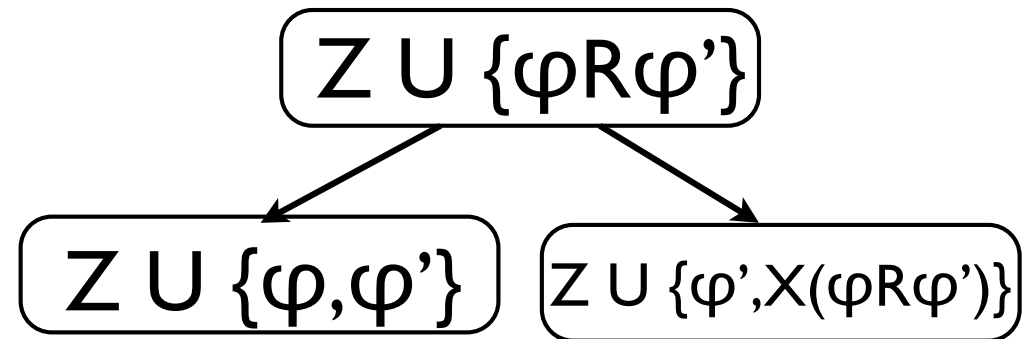
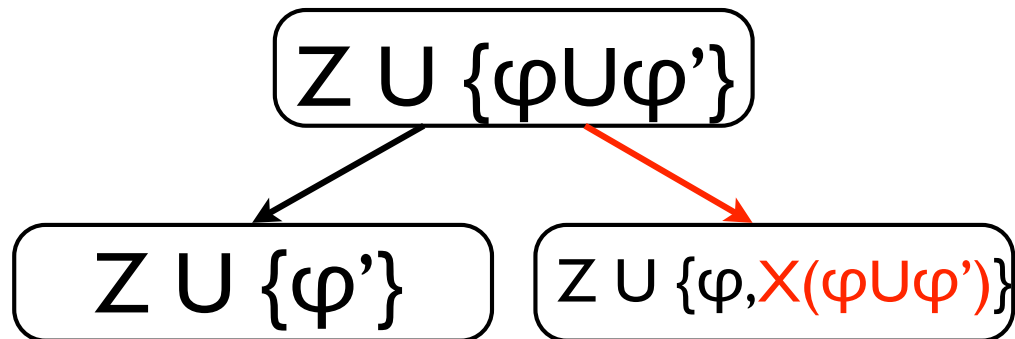
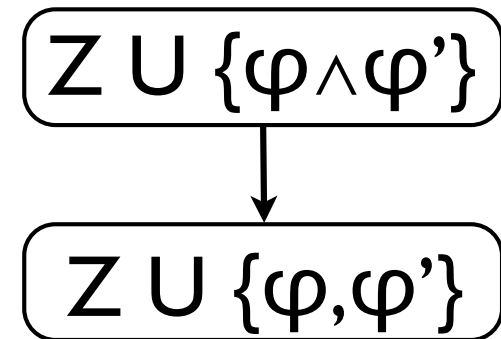
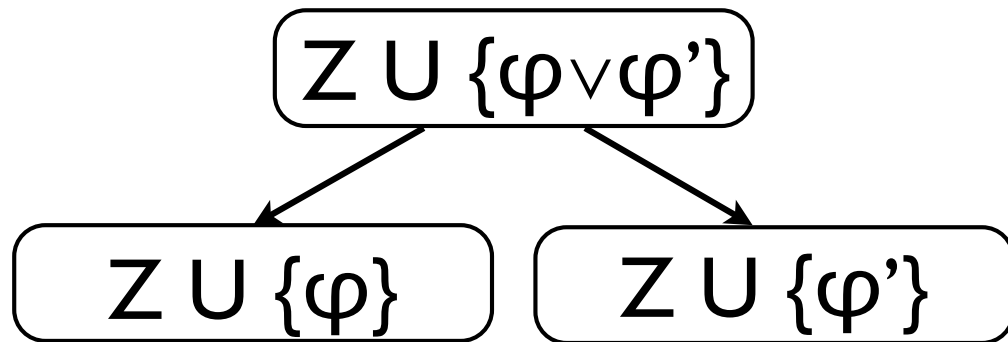
Réduire les connecteurs temporels

- Un ensemble Z de formules en forme normale négative est **réduit** si
 1. pour tout $z \in Z$, z est de la forme p , $\neg p$ ou $X(z')$
 2. il est **cohérent** : $\perp \notin Z$, $\{p, \neg p\} \not\subseteq Z$, pour tout $p \in AP$.

Réduire les connecteurs temporels

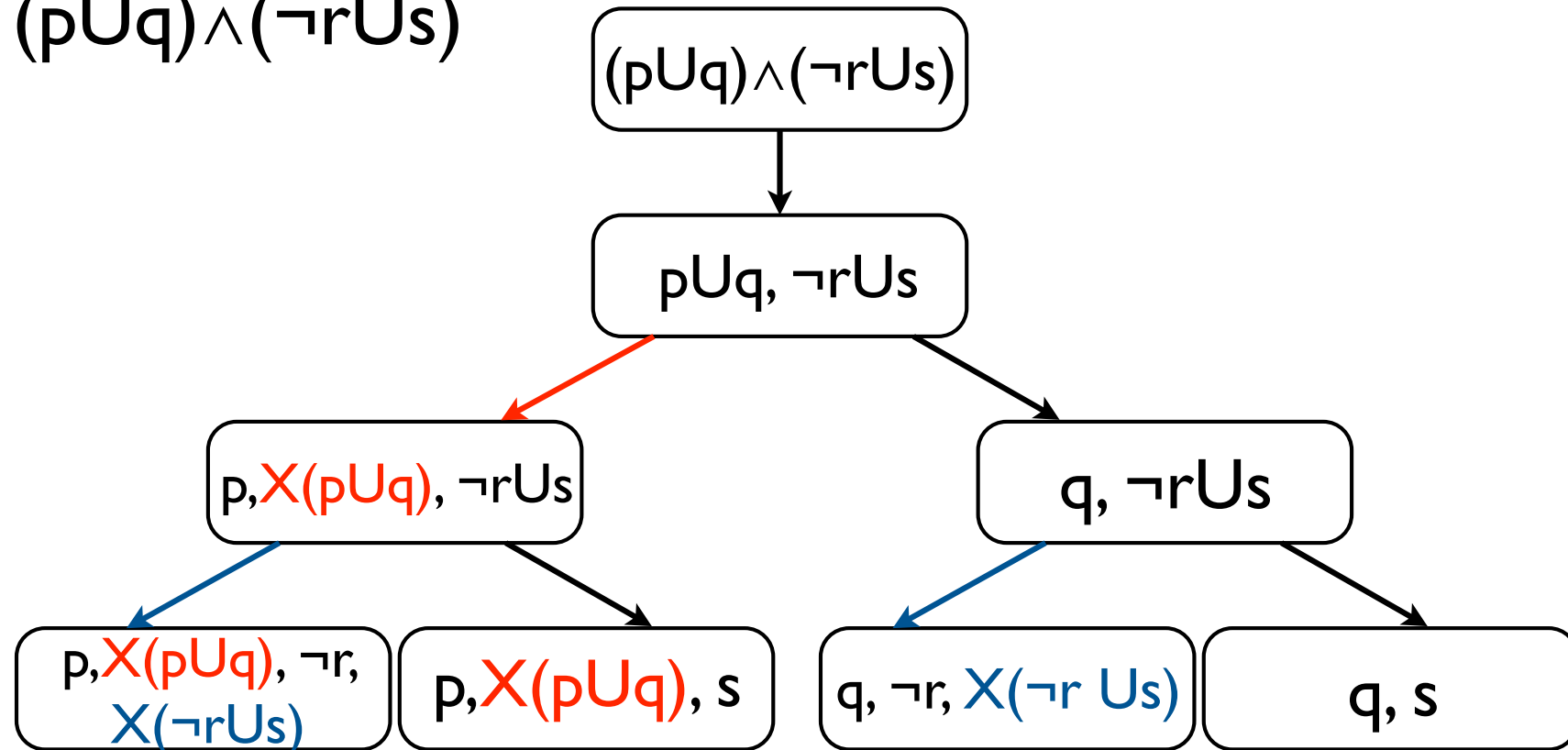
- On utilise les équivalences suivantes :
 - $\varphi \cup \varphi' \equiv \varphi' \vee (\varphi \wedge X(\varphi \cup \varphi'))$
 - $\varphi R \varphi' \equiv (\varphi \wedge \varphi') \vee (\varphi' \wedge X(\varphi R \varphi'))$

Construction du graphe de réduction



Exemple de réduction d'une formule

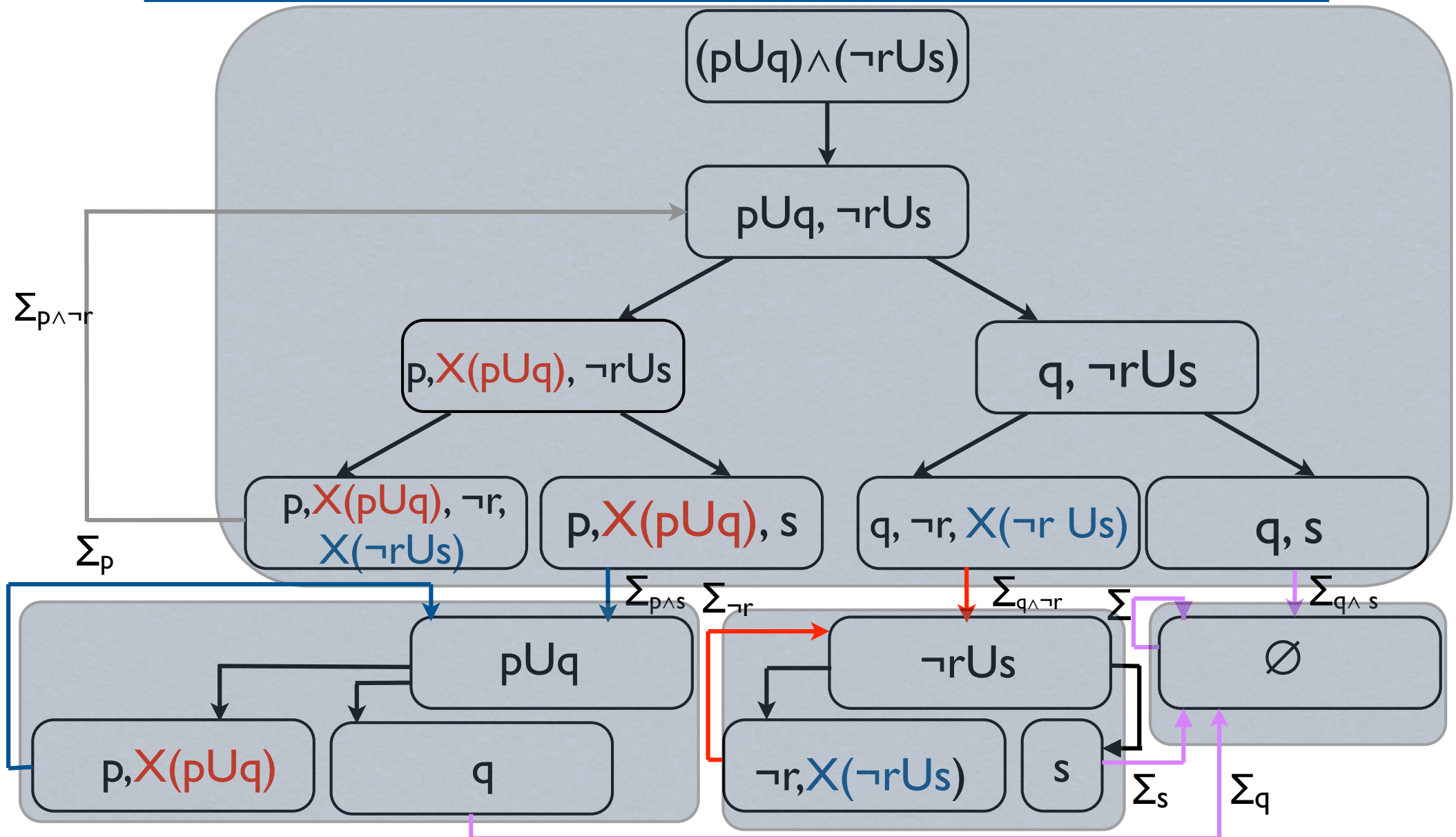
$(p \cup q) \wedge (\neg r \cup s)$



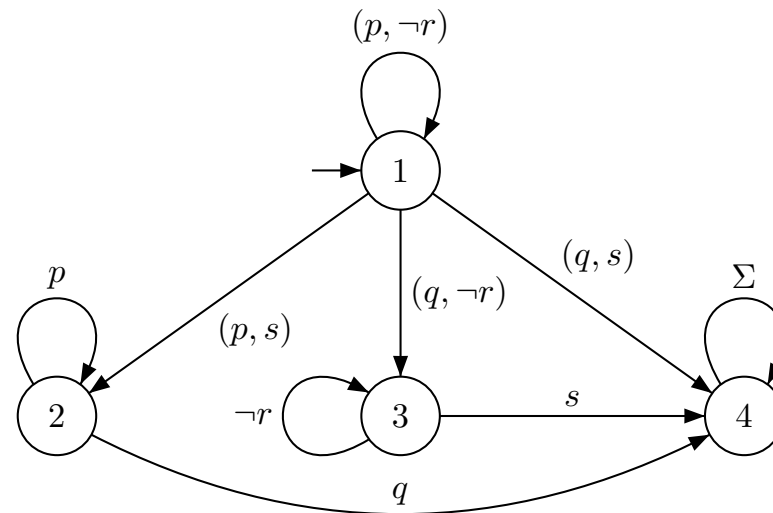
Transformer φ en un automate de Büchi

- I. Automates de Büchi généralisés
- II. Réduire la formule
 - I. Forme normale négative
 - II. Réduire les connecteurs temporels
- III. Transformation en automate de Büchi généralisé

Exemple (suite)



Example (fin)



$$T_{p \cup q} = \{(1,3)(3,3), (3,4), (4,4)\}$$

$$T_{\neg r \cup s} = \{(1,2), (2,2), (2,4), (1,4), (4,4)\}$$