



# FO sur structures relationnelles : graphes

Aujourd'hui :

1 évaluation de formules FO  
(model-checking :  $G \models \varphi$ )

2 pb. de satisfaisabilité de FO  
( $\exists G$  tq.  $G \models \varphi$  ?)

1. Model-checking

(1)  $\leq$  (2)  
 $G \mapsto \varphi_G$   
 $\text{Sat}(\varphi_G \wedge \varphi)$ ?

— jeu à 2 joueurs ( $\exists/\forall$ ) :

— configurations / positions du jeu  
appartiennent soit à  $\exists$ , ou à  $\forall$

— chaque joueur joue dans les  
configurations lui appartenant, et  
choisit la confg. suivante

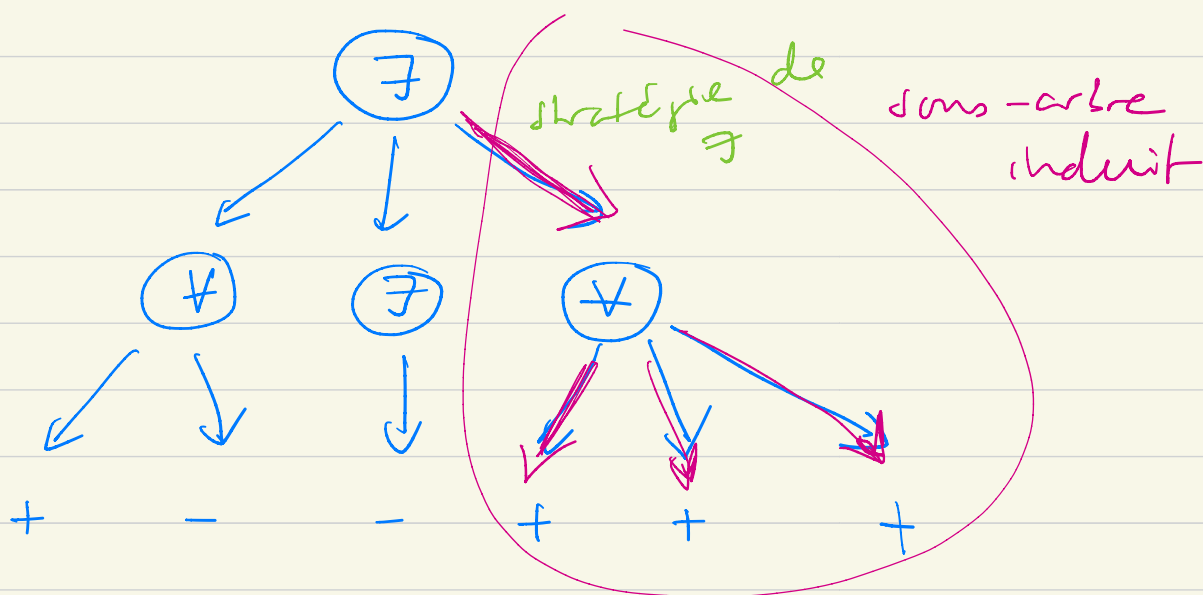
- ici : jeu fini  $\leadsto$

arène de jeu  $\hat{=}$  arbre avec

noeuds étiquetés  $\exists/\forall$ , et

feuilles étiquetées  $+/-$

( $+$  : gagnant pour joueur  $\exists$   
- perdant  $\leadsto$  -  $\exists$ )



Q: est-ce que  $\exists$  peut choisir ses  
coup tq. pour tous les choix de  $\forall$ ,  
le partie termine dans une feuille  $+$ ?

Ex : MC (FO)

$$\varphi = \exists x, y, z (x \neq y \wedge y \neq z \wedge z \neq x \\ \wedge E(x, y) \wedge E(y, z) \wedge E(z, x))$$

$\varphi:$

$\exists x$

$\varphi_1$

$\exists y$

$\varphi_2$

$\exists z$

$\varphi_3$

$\wedge$

$x \neq y$

$\varphi_4$

$\wedge$

$y \neq z$

$\varphi_5$

$\wedge$

$z \neq x$

$\varphi_6$

$\wedge$

$E(x, y)$

$\varphi_7$

$\wedge$

$E(y, z)$

$E(z, x)$

$$\varphi = \exists x \varphi_1$$

$$\varphi_1 = \exists y \varphi_2$$

$\vdots$

$$\varphi_3 = x \neq y \wedge \varphi_4$$

$G \models \varphi$

$G:$

$a - b - c$

$\neg \varphi$

→ forme normale  
en poussant les  $\neg$

feuilles : formules atomiques

Negation-free formula :

$\neg$  pushed to the leaves using  
De Morgan & co.

$$\neg (\exists x. \varphi) \equiv \forall x. (\neg \varphi)$$

$$\neg (\varphi_1 \wedge \varphi_2) \equiv \neg \varphi_1 \vee \neg \varphi_2$$

wlog, formula  $\varphi$  has  $\neg$  only on  
leaves :  $x \neq y$ ,  $\neg E(x, y)$

Rem size of the formula does not  
change, modulo constant factor.

Game graph : tree with

- inner nodes labelled  $\exists, \forall, \wedge, \vee$

- leaves labelled by

$$x = y, x \neq y, E(x, y), \neg E(x, y)$$

→ syntactic tree of the formula



$$\# \{ \vec{a} : \vec{a} \in V^{\leq |\varphi|} \}$$

Arbre de jeu est

- poly dans  $|G|$

- exponentiel dans  $|\varphi|$

$(\Psi, \vec{a})$

$\Psi$  sous-formule  
de  $\varphi$

$n^{|\varphi|}$

On veut savoir si joueur  $\exists$  a une  
stratégie gagnante : elle doit choisir  
un successeur pour chaque nœud  
 $\exists$  ou  $\forall$  tq. le sous-arbre  
induit a toutes les feuilles +

(linéaire dans taille arbre)

Algo exponentiel : bottom-up,

étiqueter tous les nœuds par +/-

- les feuilles gardent leur étiquette

- un nœud de  $\exists$  est étiqueté +

si il a au moins un enfant +

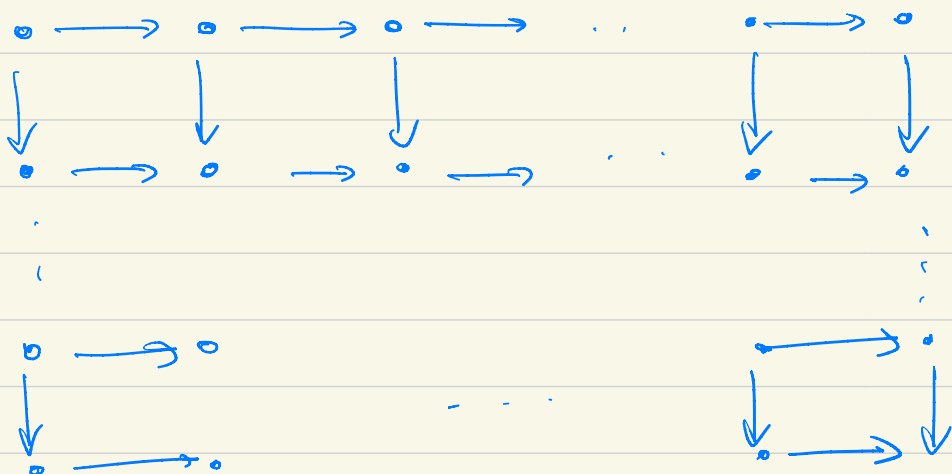
- un nœud de  $T$  est étiqueté  $+$  si tous ses enfants sont  $+$

Joueur 7 a une stratégie gagnante  
ssi la racine de l'arbre a l'étiquette  $+$ .

2. Satisfaisabilité  $\rightarrow$  indécidable

Etant donnée formule FO  $\varphi$ , est-ce  
qu'il ex. graphe  $G$  tq.  $G \models \varphi$ ?

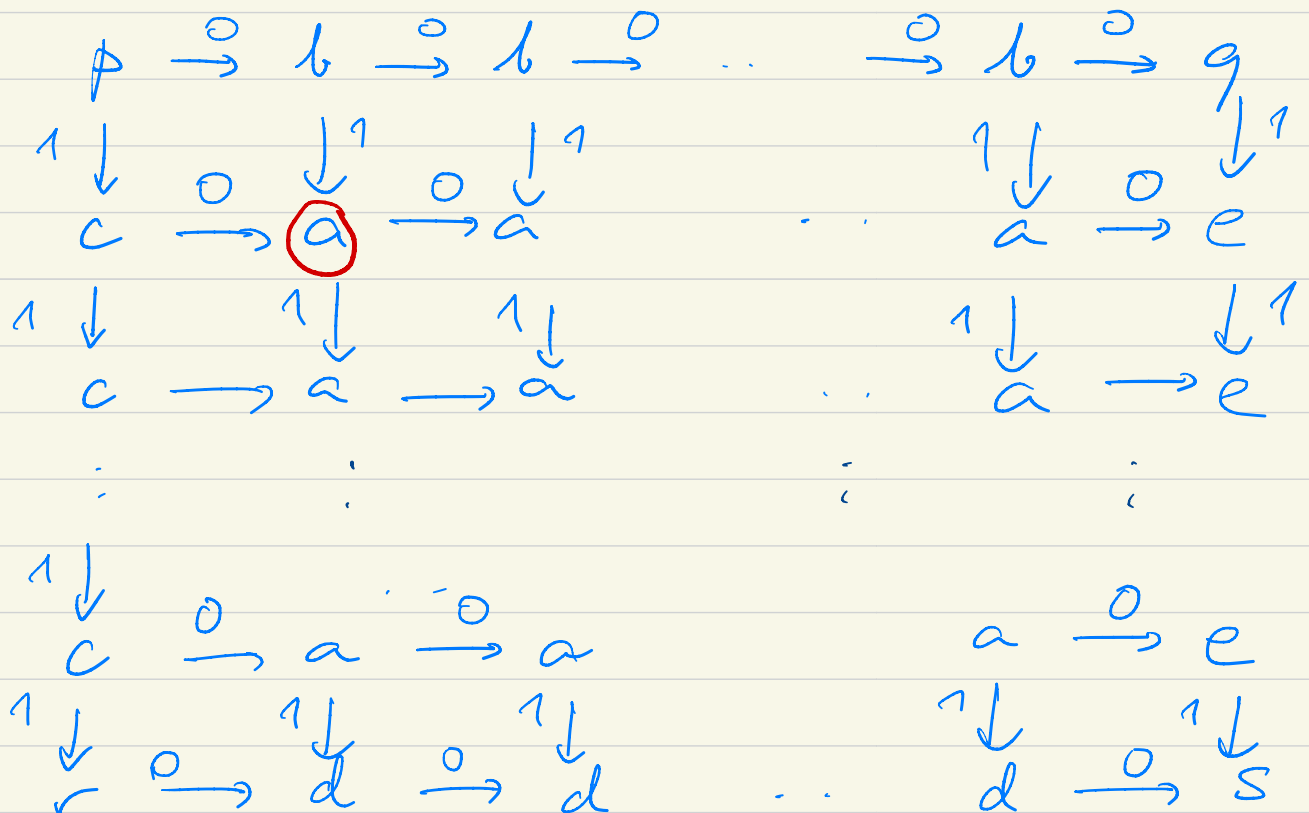
Idee : en FO on peut coder des  
graphes - grilles



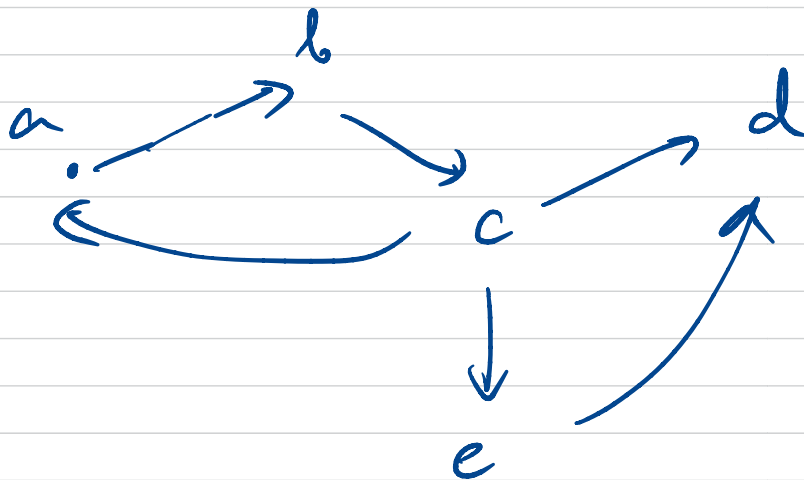
Pour simplifier, on va considérer des graphes dont les noeuds sont étiquetés (ou arêtes)

Ceci n'est pas une "vraie" extension, on peut coder des étiquettes de noeuds / arêtes par des "gadgets".

Graphes qu'on veut décrire en FO :



For a fixed graph we can easily write a FO formula that describes the graph:



$$\exists x_1, \dots, x_5 \left( \bigwedge_{i \neq j} x_i \neq x_j \wedge \right.$$

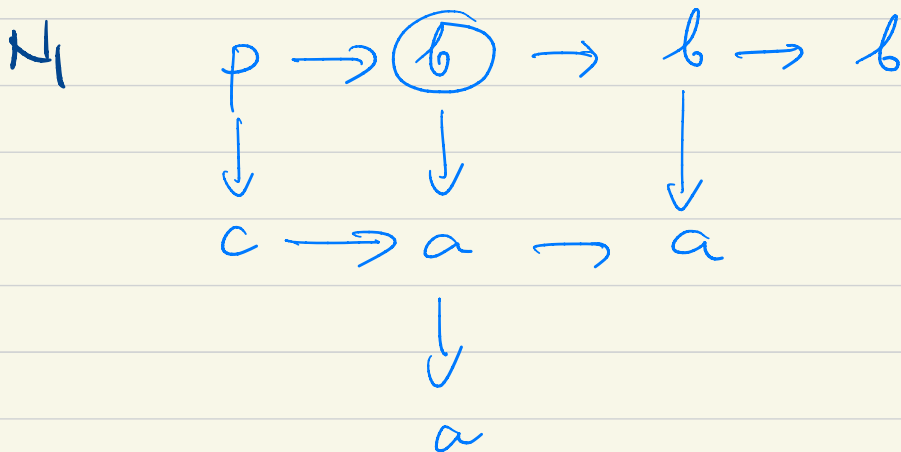
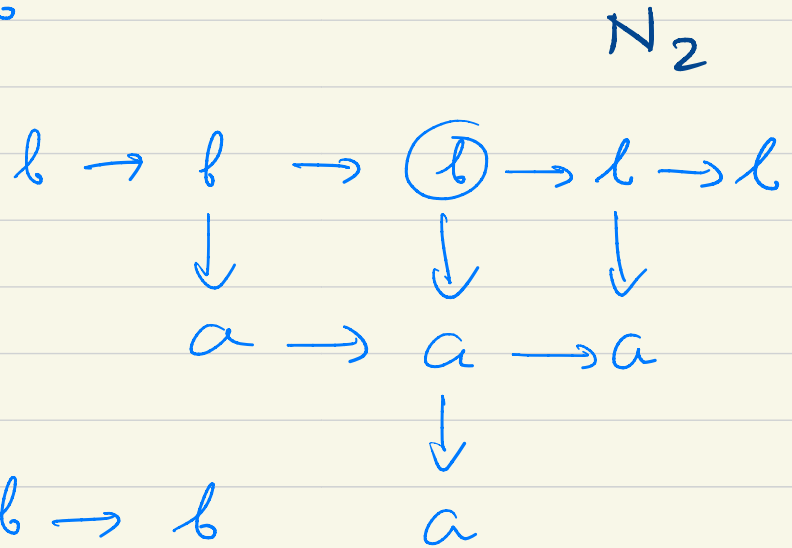
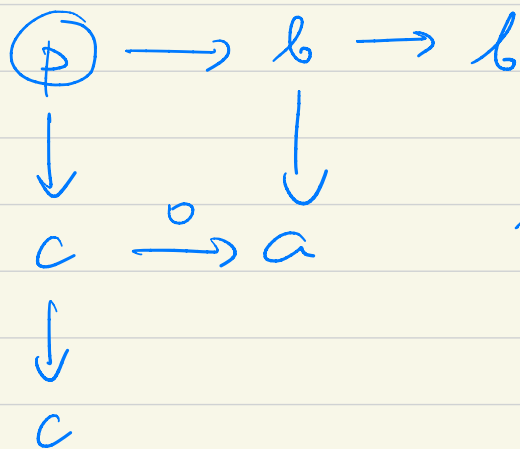
$$a(x_1) \wedge b(x_2) \wedge \dots \quad e(x_5) \wedge$$

$$E(x_1, x_2) \wedge E(x_2, x_3) \wedge \dots \wedge$$

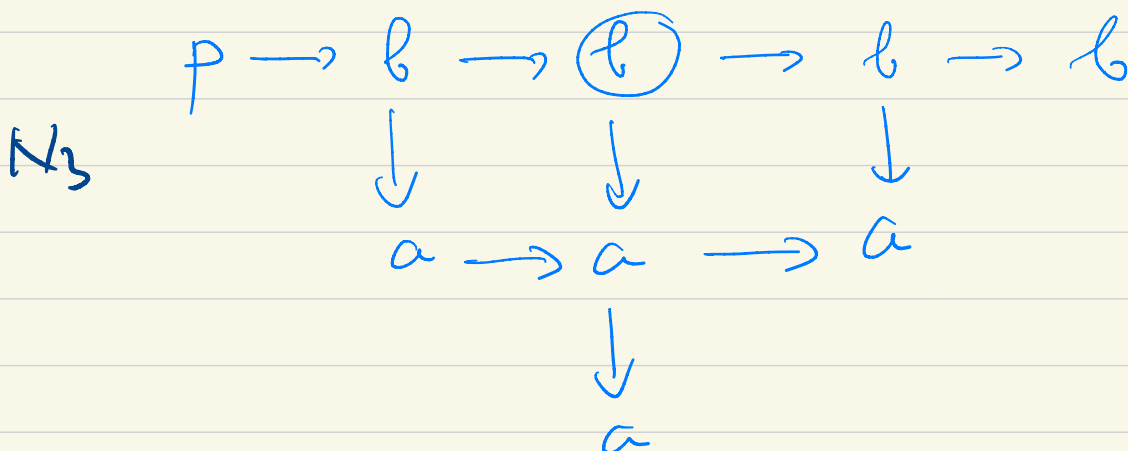
$$\wedge \neg E(x_4, x_1) \wedge \neg E(x_3, x_2) \dots$$

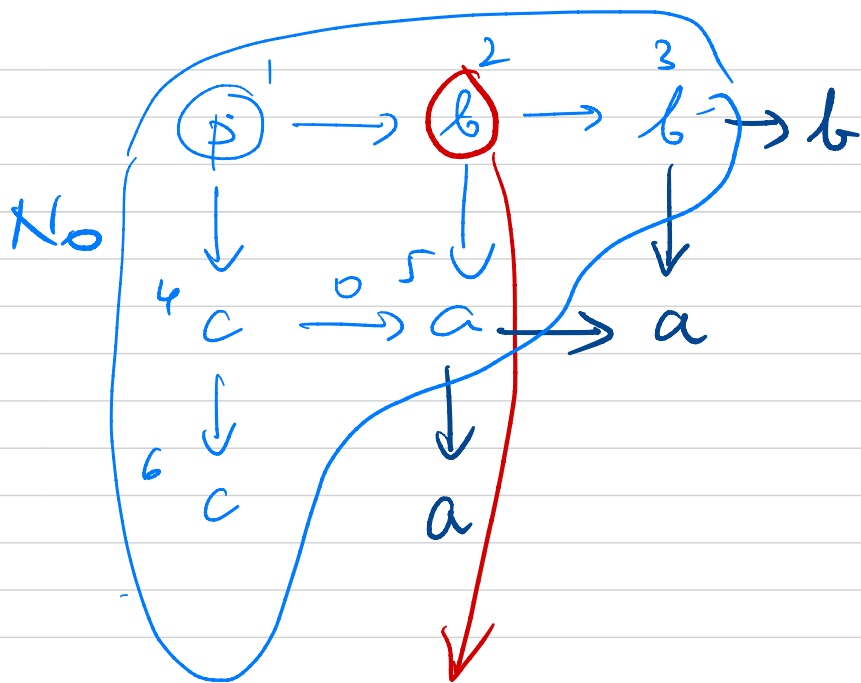
For arbitrary graphs this does not work. But for grids : neighborhoods

Propriétés de voisinage (distance  $\leq 2$ )  
 [ unique sommet étiqueté "p" ]



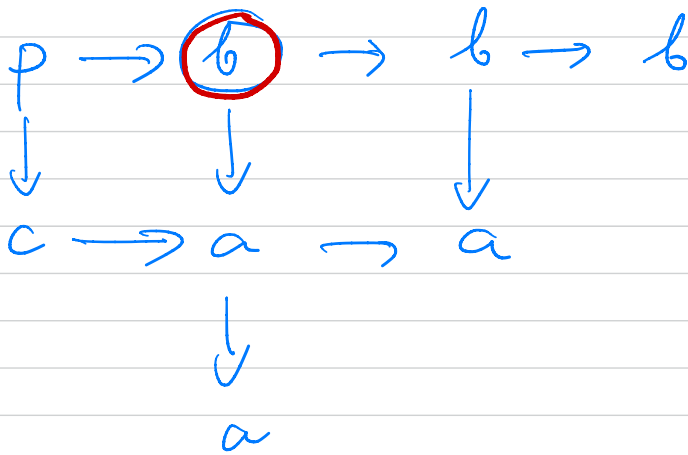
$N_4$   
 $N_5$   
 + 2 autres  
 (dernier, avant dernier)





$\exists$  unique vertex  $p$  with  
 neighborhood

$\wedge$   
 $\forall$  vertex  $b$  has  
 neighborhood of  
 type  $N_1$ , or  $N_2$ ,  
 ... or  $N_5$



"The neighborhood of radius  $\leq 2$  of  
 $p$  is No"

$\forall p(x) = \text{true} \iff \text{for } x=p$   
 the neighb. of radius  $\leq 2$   
 of  $x$  is No

$\exists x_1, \dots, x_6 :$

- 1)  $x_1, \dots, x_6 :$  pairwise different
- 2) There are no other vertices than  $x_1, \dots, x_6$  at distance  $\leq 2$
- 3) The labels are as specified
- 4) the edges  $\rightarrow$  ———
- 5)  $x = x_1$  and vertices  $x_2, \dots, x_6$  are at distance  $\leq 2$  of  $x$

$$\varphi_1 = \bigwedge_{\substack{i \neq j \\ i, j \in 1}}^6 (x_i \neq x_j)$$

$$\varphi_5 = \bigwedge_{i=2}^6 d_{\leq 2}(x_1, x_i) \wedge x = x_1$$

$d_{\leq 2}(y, z) :$   $y$  and  $z$  are at undirected dist.  $\leq 2$

$$d_{\leq 2}(y, z) =$$

$$0 \quad (y = z \vee$$

$$\leq 1 \quad (E(y, z) \vee E(z, y) \vee$$

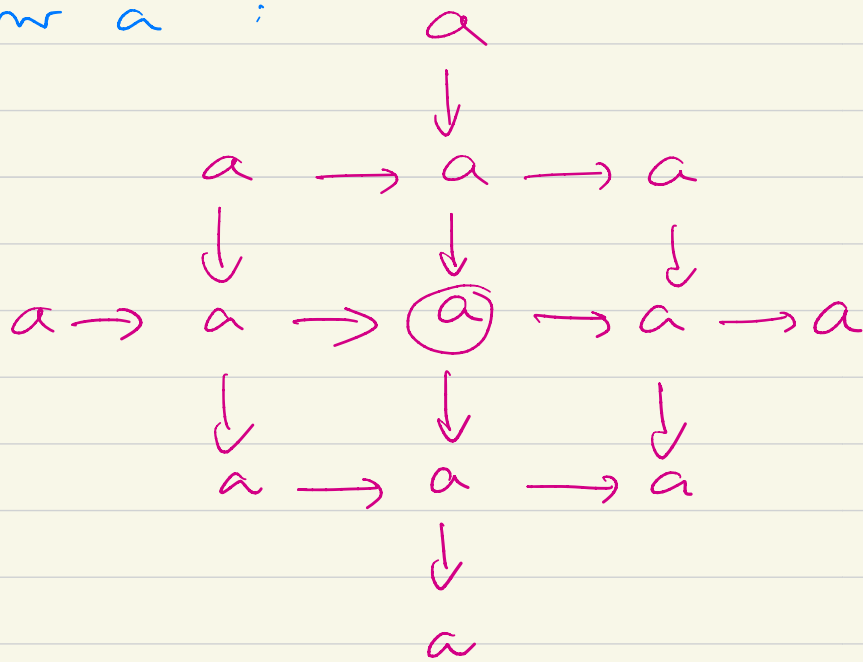
$$\leq 2 \quad (\exists w [(E(y, w) \vee E(w, y)) \wedge (E(z, w) \vee E(w, z))])$$

$$\varphi_2 = \forall y (d_{\leq 2}(x, y) \rightarrow \bigvee_{i=1}^6 (y = x_i))$$

$$\begin{aligned} \varphi_3 = & p(x_1) \wedge b(x_2) \wedge b(x_3) \\ & \wedge c(x_4) \wedge a(x_5) \wedge \\ & c(x_6) \end{aligned}$$

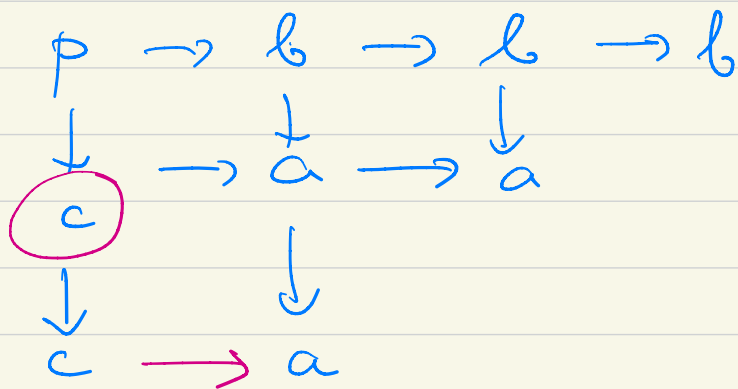
$$\begin{aligned} \varphi_4 = & E(x_1, x_2) \wedge E(x_2, x_3) \wedge \dots \\ & \wedge E(x_4, x_6) \wedge \neg E(x_2, x_4) \wedge \dots \\ & \neg E(x_6, x_5) \end{aligned}$$

Power  $a^n$



Formule FO  $\varphi$ :

- il ex - unique sommet  $p$
- tous les voisins de distance  $\leq 2$  sont parmi les graphes "permis"

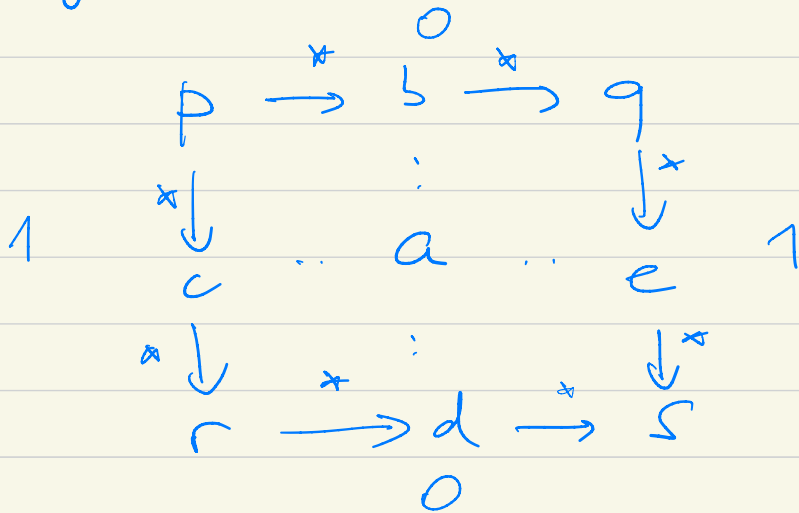
$$G \models \varphi$$




Récapitulons :

- on peut écrire une formule  $\varphi_{grid}$   
(qui utilise des prédicats atomiques  
 $p(\cdot), a(\cdot) \dots$  et  $E_0(\cdot, \cdot), E_1(\cdot, \cdot)$ )

// tq.  $G \models \varphi_{grid} \Leftrightarrow G$  contient  
grille colorée de type



- on va écrire maintenant une  
formule  $\varphi_{MT}$  de FO qui utilise  
des couleurs supplémentaires pour  
coder le calcul d'une MT dans la  
grille, ligne par ligne.

Calcul de MT (sur bande vide) :

$$C_0 + C_1 + \dots + C_N$$

$$C_i = u_i q_i v_i \quad \begin{array}{l} q_i \in \mathcal{Q} \\ u_i, v_i \in \Gamma^+ \end{array}$$

On suppose que  $|C_0| = |C_1| \dots = |C_N|$

$$\forall i \quad C_i = \alpha_0^i \dots \alpha_{M-1}^i$$

|          |              |              |          |                  |
|----------|--------------|--------------|----------|------------------|
| $C_0$    | $\alpha_0^0$ | $\alpha_1^0$ | $\dots$  | $\alpha_{M-1}^0$ |
| $C_1$    | $\alpha_0^1$ | $\alpha_1^1$ | $\dots$  | $\alpha_{M-1}^1$ |
| $\vdots$ | $\vdots$     | $\vdots$     | $\vdots$ | $\vdots$         |
| $C_N$    | $\alpha_0^N$ | $\alpha_1^N$ | $\dots$  | $\alpha_{M-1}^N$ |

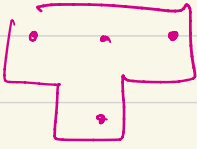
Avec des couleurs additionnelles

(pour  $\mathcal{Q}$ ,  $\Gamma$  etc) on code les  $C_i$ .

→ Formules FO pour exprimer que c'est un calcul de MT :

- configuration initiale / finale
- $C_i \vdash C_{i+1} :$

$$\begin{array}{ccccccc}
 C_i & \alpha_0^i & \alpha_1^i & \dots & \boxed{\alpha_{j-1}^i \alpha_j^i \alpha_{j+1}^i} & \dots & \alpha_{n-1}^i \\
 C_{i+1} & \alpha_0^{i+1} & \alpha_1^{i+1} & \dots & \boxed{\alpha_j^{i+1}} & \dots & \alpha_{n-1}^{i+1}
 \end{array}$$

$\leadsto$  de nouveau, on décrit les 
  
 par des formules de voisinage

La formule  $\varphi = \varphi_{\text{grid}} \wedge \varphi_{\Pi T}$

est tq.  $G \models \varphi \iff$

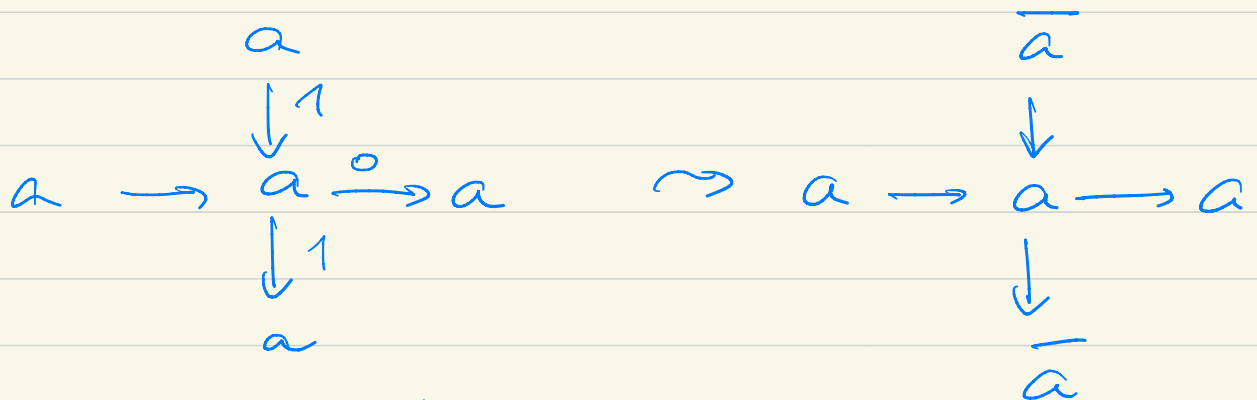
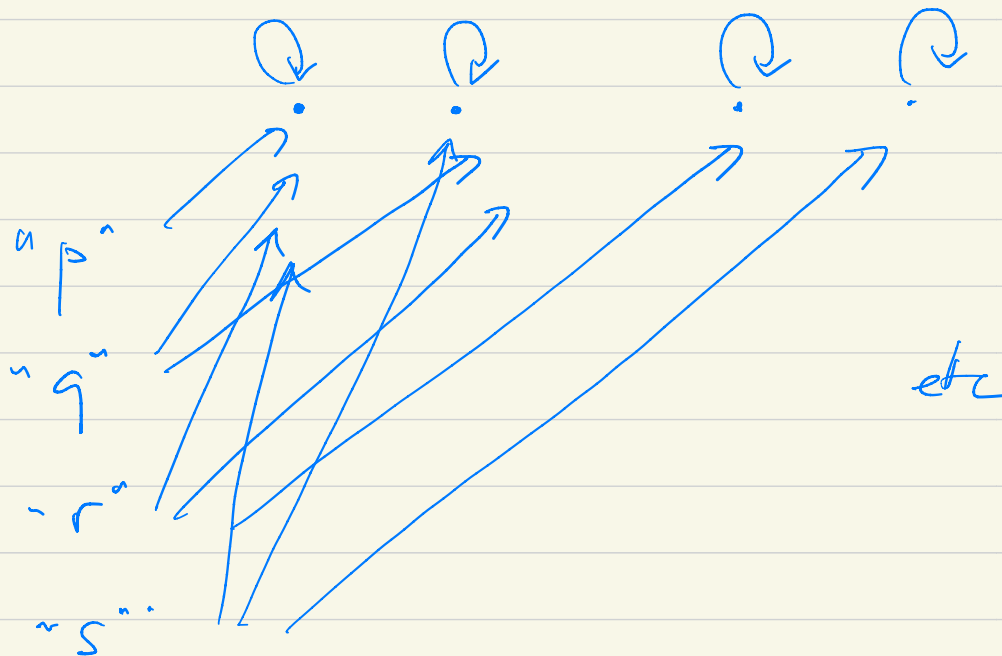
$G$  code le calcul acc. d'une  $\Pi T$

le pb. "est-ce que une  $\Pi T$  possède  
 un calcul acceptant ?" est indécidable.

$\Rightarrow$  donc SAT(FO) est indécidable

Comment on s'en sort sans couleurs ?

Par exemple, pour la grille :



MC(F0)  
décidable

I:  $G, \gamma$   
Q:  $G \models \gamma$

SAT(F0)      indécidable

Plus tard :

classes de graphes avec propriétés  
additionnelles ("tree-width")

pour lesquelles SAT(FO) déc.,  
et même SAT(MSO) déc.

→ algorithmique paramétrisé

( FPT : fixed-parameter  
tractability )

↓

XIP