

LANGAGES ET LOGIQUE

DEVOIR RAISON

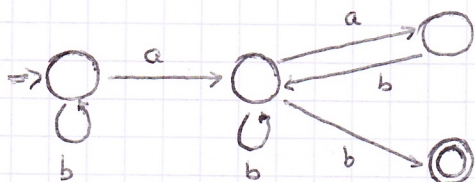
I] AUTOMATES A PARITE ET AUTOMATES DE BÜCHI

1.1 Exemple

\mathcal{A} reconnaît le langage $b^* a (b^* a b)^* b^w$.

En logique temporelle LTL, on peut exprimer ce langage par: $(Gb) \cup (a \wedge X(G(a \Rightarrow Xb))) \wedge F(G$

Ce même langage peut être reconnu par l'automate de Büchi suivant:

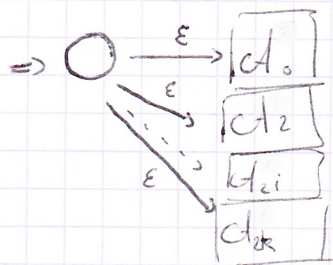


1.2 Parité et Büchi

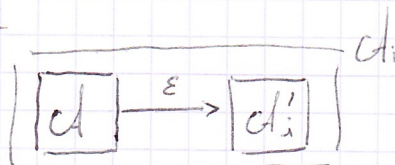
(a) Cette transformation est aisée: en associant la couleur 1 aux états non-acceptants et la couleur 2 aux états acceptants, on s'assure de l'équivalence.

Cela est valable car dans une partie infinie, au moins un état est parcouru infiniment: si aucun des états infiniment traversés ne se trouve dans l'ensemble des états acceptants, alors $\max(\chi(\pi)) = 1, \forall \pi \in \Pi$.

(b) Pour un automate de parité à 2k couleurs, on crée l'automate suivant:



où c_i est conçu de la façon suivante:



\mathcal{A}'_i est la restriction de \mathcal{A} à tous les états de couleur $\leq i$.

Les états acceptants sont, dans chaque \mathcal{A}'_i , les états de couleur i .

L'idée de cette construction est que, pour une couleur c donnée,

- \mathcal{A}'_c soit toujours accessible
- \mathcal{A}'_c ne soit atteint que si, dans l'automate à partir, aucune couleur $> c$ ne soit infiniment parcourue.
- Un état acceptant n'est infiniment atteint que si, dans \mathcal{A} , un état de couleur c est infiniment parcouru.

On voit bien que le nombre d'état de cet automate vaut, si \mathcal{A} a q états et k couleurs,

$$n \leq q \times 2 \times k$$

Comme $k \leq q$, $n \leq q^2 \times 2$ donc la taille est bien polynomiale.

II] JEUX DE PARTIE FAIBLE

2.1 Transformation en jeu à parties infinies.

* Toutes les parties finies de G se terminent dans un état-puits.

Soit $P = P_0 \cup P_1$ l'ensemble des états-puits de G , avec $P_0 \subset V_0$ et $P_1 \subset V_1$.

* Dans G , une partie atteignant un état-puits u se termine, désignant vainqueur

- J_0 si $u \in P_1$

- J_1 si $u \in P_0$.

Donc, dans G' , toute partie passant par u doit inexorablement désigner le même joueur vainqueur.

* Pour cela, on rajoute à G pour construire G' , pour chaque état-puits u ,

- une boucle $\{u \rightarrow u\}$

- deux nouvelles couleurs i et j , i paire et j impaire, telles que $i > k$ et $j > k$, (k nombre de couleurs de G)

De plus : - on modifie la couleur de tous les sommets de P_0 en j ,
et de tous ceux de P_1 en i .

2.2 Exemple

* $d \in \text{Win}_1$, et $\text{Attr}_1^{\text{MI}}(\{d\}) = \{b\}$, donc $\{b, d\} \in \text{Win}_1$.

* Dans le sous-jeu $G \setminus \{b, d\}$, on voit $c, g \in \text{Win}_0$ et $\text{Attr}_0^{\text{MI}}(\{c, g\}) = \{f\}$ donc $\{c, g, f\} \in \text{Win}_0$.

* Dans le sous-jeu $G \setminus \{b, d, c, g, f\}$, $a \in \text{Win}_0$, $e \in \text{Win}_0$.

On a donc $\text{Win}_0 = \{a, e, f, c, g\}$

Voici une stratégie gagnante pour J_0 :

$\sigma = \left\{ \begin{array}{l} a \rightarrow a \\ f \rightarrow g \\ g \rightarrow f \\ d \rightarrow g \end{array} \right\}$

(elle est positionnelle car nous sommes dans un jeu de partie faible)

2.3 Positions gagnantes et μ -calcul

* Pour un ensemble de sommets $S \subseteq V$, l'expression de μ -calcul model :

$$\mu X. S \vee (\Diamond X \wedge p_0) \vee (\Box X \wedge p_1) \quad \text{traduit } Attr_0^{IV}(S)$$

" $Attr_0^{IV}(S)$ dans le sous-jeu $V \setminus R$ " se traduit :

$$\mu X. \neg R \wedge (S \vee (\Diamond X \wedge p_0) \vee (\Box X \wedge p_1))$$

* De même, un piège pour S , dans S sera :

$$\nu X. S \wedge ((\Diamond X \wedge p_0) \vee (\Box X \wedge p_1))$$

* On peut donc exprimer :

— Dans un jeu à 2 couleurs (0 et 1) : $Win_0 := \nu X. C_0 \wedge ((\Diamond X \wedge p_0) \vee (\Box X \wedge p_1))$

— Dans un jeu à 3 couleurs (0, 1 et 2) : $Win_0 := (\mu X. C_2 \vee (\Diamond X \wedge p_0) \vee (\Box X \wedge p_1)) \vee (\nu X. C_0 \wedge ((\Diamond X \wedge p_0) \vee (\Box X \wedge p_1)))$

— Dans un jeu à k couleurs, on a : $\forall i \in \{1, k\}$

$$\begin{aligned} A_i &:= \mu X. \neg A_{i+1} \wedge (C_i \vee (\Diamond X \wedge p_A) \vee (\Box X \wedge p_B)) \\ \text{et } A_k &:= \mu X. C_k \vee (\Diamond X \wedge p_A) \vee (\Box X \wedge p_B) \\ \text{et } A_0 &:= \nu X. C_0 \wedge ((\Diamond X \wedge p_0) \vee (\Box X \wedge p_1)) \end{aligned} \quad \left(\begin{array}{l} \text{avec } p_A = p_0 \text{ si } i \text{ pair} \\ p_B = p_1 \\ \text{d'inverse si } i \text{ impair.} \\ \text{(même def de } p_A \text{ et } p_B) \end{array} \right)$$

$$\text{On a : } Win_0 := \sum_{i=0}^k \{A_i \mid i \text{ pair}\}$$