

# Logiques comportementales

– logiques, automates et jeux –  
M2, Univ. Bordeaux 1, 2009/10

## Contenu

- Anca Muscholl : Jeux à information complète. Logiques et automates sur les mots/arbres infinis.
- Anne Dicky : Logiques de point fixe et lien avec jeux.
- Hugo Gimbert : Jeux stochastiques, chaînes de Markov.

# Gale-Stewart

## Notations

- $\Sigma$  alphabet fini :  $\Sigma^*$  ensemble des mots finis sur  $\Sigma$ ,
- $\Sigma^\omega$  ensemble des mots infinis sur  $\Sigma$  (fonctions  $\mathbb{N} \rightarrow \Sigma$ )

## Jeux à la Gale-Stewart

- 2 joueurs  $S, E$  ( $S$  système,  $E$  environment)
- Chaque joueur a un alphabet d'actions associé :  $\Sigma_S, \Sigma_E$ .
- Un jeu à la Gale-Stewart est décrit par une condition de victoire

$$\text{Win} \subseteq (\Sigma_S \Sigma_E)^\omega$$

- **Partie**  $w$  : mot infini  $w \in (\Sigma_S \Sigma_E)^\omega$
- Partie  $w$  est **gagnée** par  $S$  si  $w \in \text{Win}$ , et par  $E$  sinon.
- **Stratégie** de  $S$  : fonction  $\sigma : \Sigma_E^* \rightarrow \Sigma_S$  (ou  $\sigma : (\Sigma_S \Sigma_E)^* \rightarrow \Sigma_S$ ).  
Extension  $\bar{\sigma} : \Sigma_E^\omega \rightarrow \Sigma_S^\omega : \bar{\sigma}(b_1 b_2 \dots) = a_0 a_1 \dots$  si  $a_n = \sigma(b_1 \dots b_n)$ ,  $n \geq 0$ .

# Gale-Stewart

## Déroulement

- Si  $E$  joue  $y = b_1 b_2 \dots \in \Sigma_E^\omega$  et  $S$  applique la stratégie  $\sigma$  en jouant  $\bar{\sigma}(y) = a_0 a_1 \dots$  :  
partie  $a_0 b_1 a_1 b_2 \dots$ , notée  $\bar{\sigma}(y) \mid y$ .
- $S$  gagne le jeu avec  $\sigma$  si **quelque soit**  $y \in \Sigma_E^\omega : \bar{\sigma}(y) \mid y \in \text{Win}$ .
- **Jeu déterminé** :  $S$  gagne si et seulement si  $E$  perd.
- **Théorème de Martin** : les jeux de Gale-Stewart avec conditions de victoire boréliennes sont déterminés.

## Jeux réguliers

- **$\omega$ -réguliers** : définis par **automates** (Büchi, Muller, parité, ...), **logique** (monadique du second ordre), expressions  $\omega$ -rationnelles, ...
- Jeux réguliers : conditions de gain  $\omega$ -régulières.

# Jeux sur des graphes

## Arène de jeu (McNaughton)

$$\mathcal{A} = (V_0, V_1, E)$$

- Deux joueurs  $J_0$  et  $J_1$ . L'ensemble des sommets est partitionné en deux ensembles disjoints  $V_0$  (sommets appartenant à  $J_0$ ) et  $V_1$  (sommets appartenant à  $J_1$ ).
- Ensemble d'arêtes  $E \subseteq V \times V$ ,  $V = V_0 \cup V_1$ .

## Parties

Une **partie** dans  $\mathcal{A}$  est

- soit un chemin infini  $\pi = v_0, v_1, \dots$  dans le graphe  $(V, E)$ ,
- ou un chemin fini  $\pi = v_0, v_1, \dots, v_n$  dans  $(V, E)$  tel que  $v_n$  n'a pas de successeur dans  $(V, E)$ .

# Jeux et stratégies

## Jeux

Un jeu est décrit par une paire  $(\mathcal{A}, \text{Win})$ , où  $\mathcal{A}$  est une arène et  $\text{Win} \subseteq V^\omega$  est la condition de victoire.

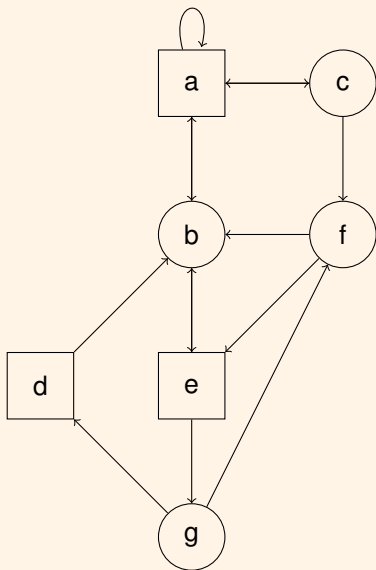
Exemples (où  $F \subseteq V$ ,  $\mathcal{F} \subseteq \mathcal{P}(V)$ ) :

- jeux **d'accessibilité (ou garantie)** :  $\text{Win} = V^* F V^\omega$
- jeux de **sûreté** :  $\text{Win} = V^\omega \setminus V^* (V \setminus F) V^\omega$
- jeux **d'obligation** : ensemble des états visités appartient à  $\mathcal{F}$
- jeux de **Büchi (répétition)** :  $\text{Win} = (V^* F)^\omega$ , jeux de **Muller**, jeux de **parité**...

## Stratégies

- Stratégie de  $J_0$  : fonction  $f : V^* V_0 \rightarrow V$   
t.q.  $f(\pi v) \in vE := \{w \in V \mid (v, w) \in E\}$ , où  $\pi \in V^*$ ,  $v \in V$ .
- Partie  $\pi = v_0, v_1, \dots$  conforme avec stratégie  $f$  de  $J_0$  :  $v_{i+1} = f(v_0 \cdots v_i)$   
pour tout  $i$  t.q.  $v_i \in V_0$ .
- Stratégie **positionnelle** :  $f : V_0 \rightarrow V$  t.q.  $f(v) \in vE$  pour tout  $v \in V_0$ .

## Example (jeu d'accessibilité) : $J_0$ rond, $J_1$ carré



$F = \{a, d\}$  :  $J_0$  gagne partout ;

$F = \{e\}$  :  $J_0$  gagne partout sauf à partir de  $a$ .

# Jeux d'accessibilité

## Attracteurs

$$\begin{aligned}\text{Attr}_0^0(F) &= F \\ \text{Attr}_0^{n+1}(F) &= \text{Attr}_0^n(F) \cup \\ &\quad \{v \in V_0 \mid \exists w \in \text{Attr}_0^n(F), (v, w) \in E\} \\ &\quad \{v \in V_1 \mid \forall w \text{ t.q. } (v, w) \in E : w \in \text{Attr}_0^n(F)\}\end{aligned}$$

$$\text{Attr}_0^0(F) \subseteq \text{Attr}_0^1(F) \subseteq \dots \subseteq \text{Attr}_0^{|V|}(F)$$

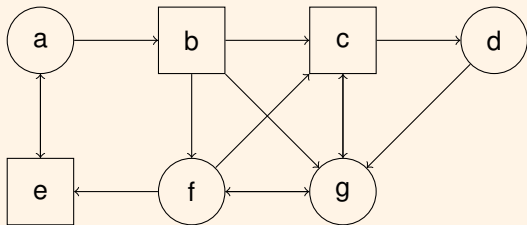
$\text{Attr}_0^i(F)$  est l'ensemble des nœuds à partir desquels  $J_0$  peut forcer la partie d'arriver à  $F$  en moins de  $i$  coups.  $\text{Attr}_0^{|V|}(F)$  est la **région gagnante** de  $J_0$ , et  $V \setminus \text{Attr}_0^{|V|}(F)$  est la région gagnante de  $J_1$ .

## Jeux d'accessibilité

Les jeux d'accessibilité sont **déterminés** : à partir de chaque sommet, un des deux joueurs a une stratégie gagnante. Les stratégies gagnantes sont positionnelles (stratégies d'attracteur).

# Jeux de parité faible

Condition de victoire : coloriage des nœuds  $\chi : V \rightarrow \{1, \dots, k\}$  ;  
la plus grande couleur rencontrée est paire



$\chi(a) = \chi(e) = 0$ ,  $\chi(b) = \chi(f) = 1$ ,  $\chi(c) = \chi(g) = 2$ ,  $\chi(d) = 3$ .

- $d \in \text{Win}_1$  et  $\text{Attr}_1(d) = \{d, c, b\} \subseteq W_1$
- $V' = V \setminus \{b, c, d\}$  est piège pour  $J_1$ , donc définit sous-jeu pour  $J_0$   
 $\text{Attr}_0(g) \upharpoonright_{V'} = \{g, f\} \subseteq W_0$
- $V'' = V' \setminus \{f, g\}$  piège pour  $J_0$ , donc sous-jeu pour  $J_1$   
 $W_1 \upharpoonright_{V''} = \emptyset$ ,  $W_0 \upharpoonright_{V''} = \{a, e\}$

Conclusion :  $W_0 = \{a, e, f, g\}$ ,  $W_1 = \{b, c, d\}$ . Les stratégies sont positionnelles (stratégies d'attracteur).

# Jeux de parité faible

- Le complémentaire  $V \setminus C$  d'un attracteur pour le joueur  $J_i$  est un piège pour  $J_i$  ( $J_i$  n'a pas de transition sortant de  $C$ ).
- Un piège  $C$  pour  $J_i$  représente un sous-jeu pour l'adversaire  $J_{1-i}$  : toute stratégie qui est gagnante pour  $J_{1-i}$  dans  $C$  est aussi gagnante pour  $J_{1-i}$  dans  $V$ .

## Solution

$V^{(k)} := \chi^{-1}(k)$  ensemble des sommets de couleur  $k$ .

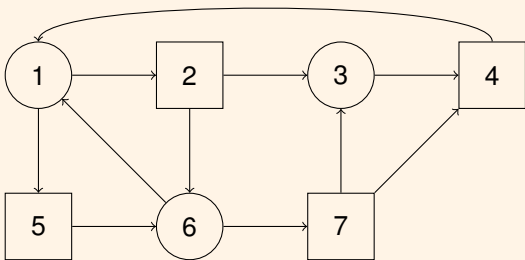
- 1 Supposons que la couleur maximale  $k$  est impaire (sinon on inverse  $J_0, J_1$ ). On calcule  $\text{Attr}_1(V^{(k)})$  (qui est inclus dans la région gagnante  $W_1$  de  $J_1$ ).
- 2  $V' := V \setminus \text{Attr}_1(V^{(k)})$  est un sous-jeu pour  $J_0$  et la couleur maximale dans  $V'$  est au plus  $k - 1$ . On calcule par induction la région gagnante  $W_0$  de  $J_0$  dans  $V'$ . Tout sommet de  $W_0$  est gagnant pour  $J_0$  dans  $V$  (cf. ci-dessus).
- 3 Tout sommet de  $V' \setminus W_0$  est gagnant pour  $J_1$  dans  $V$  :  $J_1$  applique sa stratégie gagnante de  $V'$  tant que le jeu reste dans  $V'$ , et la stratégie d'attracteur, dès que le jeu passe dans  $\text{Attr}_1(V^{(k)})$ .

# Jeux d'obligation

## Définition

La condition de victoire d'un jeu d'obligation est un ensemble  $\mathcal{F} \subseteq \mathcal{P}(V)$  d'ensembles d'états. Le joueur  $J_0$  gagne une partie  $\pi$ , si l'ensemble  $\text{Occ}(\pi)$  d'états rencontrés dans  $\pi$  appartient à  $\mathcal{F}$  (i.e., combinaison booléenne de conditions d'accessibilité).

## Exemple



Pour la condition de victoire  $\pi \in \text{Win}$  ssi  $\{2, 7\} \subseteq \text{Occ}(\pi)$  le joueur  $J_0$  a besoin de mémoire (1 bit) dans l'état 1 : il doit passer par 2 et 5 de façon alternée.

# Jeux d'obligation

## Réduction

On réduit un jeu d'obligation à un jeu plus simple (et plus grand) : jeu de **parité faible**.

- Dans un jeu de parité faible on colorie les sommets par des entiers :  $\chi : V \rightarrow \mathbb{N}$ . Le joueur  $J_0$  gagne une partie  $\pi$  si  $\max(\chi(\text{Occ}(\pi)))$  est pair.
- **Réduction** : à partir du jeu  $G = (\mathcal{A}, \mathcal{F})$  on définit un jeu  $G' = (\mathcal{A}', \chi)$  t.q.
  - 1  $\mathcal{A}' = (V'_0, V'_1, E')$ ,  $V'_i = V_i \times M$ , où  $M$  est une mémoire finie.
  - 2 Transitions :  $(\langle v, m \rangle, \langle w, m' \rangle) \in E'$  si  $(v, w) \in E$  et  $m' = f(m, v, w)$  ( $f =$  mise à jour de la mémoire).
  - 3 A chaque partie  $\pi$  dans  $G$  commençant en  $v$  et chaque valeur initiale de la mémoire  $m_0 \in M$ , correspond une unique partie  $\pi'$  à partir de  $(v, m_0)$  dans  $G'$ . De plus,  $\pi$  et  $\pi'$  sont gagnées par le même joueur.
- Les jeux de parité faible ont des stratégies gagnantes positionnelles. En conséquence, on peut obtenir des stratégies gagnantes à mémoire finie pour les jeux d'obligation.

# Jeu de parité faible

- $M = \mathcal{P}(V)$ ,
- $E' = \{(v, R) \rightarrow (w, R \cup \{v\}) \mid (v, w) \in E\}$ .
- Coloriage :  $\chi(v, R) = 2|R|$  si  $R \in \mathcal{F}$  ;  $\chi(v, R) = 2|R| - 1$  si  $R \notin \mathcal{F}$ .
- Pour chaque partie  $\pi' = (v_0, R_0), (v_1, R_1), \dots$  dans  $G'$  on a  $R_0 \subseteq R_1 \subseteq R_2 \dots$ , donc il existe  $i \geq 0$  t.q.  $R_j = R_i$  pour tout  $j \geq i$ . Si  $R_0 = \emptyset$ , alors  $R_i$  est l'ensemble  $\text{Occ}(\pi)$ , où  $\pi = v_0, v_1, \dots$
- Exemple :  
Partie  $\pi$  : 1, 2, 3, 4, ... dans jeu d'obligation.  
Partie  $\pi'$  :  $(1, \emptyset), (2, \{1\}), (3, \{1, 2\}), (4, \{1, 2, 3\}), \dots$  dans jeu de parité faible, avec  $\chi(1, \emptyset) = -1, \chi(2, \{1\}) = 1, \chi(7, \{1, 2, 3, 7\}) = 8$ .

# Vers les jeux de parité

## Jeux de parité

$\text{Inf}(v_0 v_1 \dots) = \{v \in V \mid v = v_i \text{ infiniment souvent}\}$ . Coloriage des nœuds

$\chi : V \rightarrow \{1, \dots, k\}$ .

Condition de victoire :  $\max(\chi(\text{Inf}(\pi)))$  est paire.

## Cas particuliers

- **Büchi** : condition de victoire  $F \subseteq V$ . Joueur  $J_0$  gagne  $\pi$  si  $\text{Inf}(\pi) \cap F \neq \emptyset$ .
- **Muller** : condition de victoire  $\mathcal{F} \subseteq \mathcal{P}(V)$ . Joueur  $J_0$  gagne  $\pi$  si  $\text{Inf}(\pi) \in \mathcal{F}$ .

## Rem.

Chaque jeu  $(\mathcal{A}, \text{Win})$  avec condition de victoire Win régulière se réduit à un jeu de Muller  $(\mathcal{A}, \text{Win}')$ .

# Stratégies et algorithmes

## Parité

Les jeux de parité ont des stratégies gagnantes positionnelles pour les 2 joueurs, qu'on peut calculer en temps exponentiel (plus exactement, en  $NP \cap co-NP$ ). L'algorithme dit "strategy improvement" (Jurdzinski/Vöge) est l'algorithme le plus efficace en pratique (mais on connaît depuis peu des exemples où l'algorithme met un temps exponentiel).

## Büchi

Les jeux de Büchi ont des stratégies gagnantes positionnelles pour les 2 joueurs, qu'on peut calculer en temps polynomial.

## Muller

Pour les jeux de Muller, les deux joueurs ont besoin de mémoire (finie).

## Solution

Les jeux de Büchi ont des stratégies gagnantes positionnelles pour les 2 joueurs, qu'on peut calculer en temps polynomial.

- On calcule  $\text{Attr}_0^+(F)$  : ensemble des états à partir desquels  $J_0$  a une stratégie pour arriver à  $F$  en *au moins* 1 pas.
- On calcule ensuite  $V^{(i)}$  : ensemble des états à partir desquels  $J_0$  a une stratégie pour passer au moins  $i$  fois par  $F$  (sans compter l'état initial).

$$\begin{aligned}V^{(1)} &= \text{Attr}_0^+(F), & V^{(i+1)} &= \text{Attr}_0^+(V^{(i)} \cap F) \\W_0 &= \bigcap_{i \geq 1} V^{(i)}\end{aligned}$$

- Stratégie de  $J_0$  : pour  $v \in V^{(i)}$ ,  $J_0$  a stratégie sans mémoire  $\sigma_i$  ; pour  $v \in W_0$  on choisit  $\sigma(v) := w$  t.q.  $\sigma_i(v) = w$  pour une infinité de  $i$ .

# Réduction Muller $\rightarrow$ parité

## Prop.

Pour chaque jeu de Muller  $(\mathcal{A}, \mathcal{F})$  il existe un jeu de parité  $(\mathcal{A}', \chi)$  et une fonction  $s : V \rightarrow V'$  t.q. pour chaque sommet  $v \in V$  :  
 $J_0$  gagne  $(\mathcal{A}, \mathcal{F})$  à partir de  $v$  ssi  $J_0$  gagne  $(\mathcal{A}', \chi)$  à partir de  $s(v)$ .

## Réduction LAR (last appearance record) :

On définit  $\mathcal{A}'$  en rajoutant à  $\mathcal{A}$  une mémoire  $M$  (plus une fonction de mise-à-jour  $f : V \times M \rightarrow M$ ), qui aide à déterminer l'ensemble  $\text{Inf}(\pi)$ .

# Réduction LAR

## LAR

Soit :

$$\begin{aligned}V' &= V'_0 \cup V'_1, \quad V'_0 = V_0 \times M, \quad V'_1 = V_1 \times M \\E' &= \{((u, m), (v, f(v, m))) \mid (u, v) \in E, m \in M\}\end{aligned}$$

$M$  est l'ensemble des mots  $X$  sur l'alphabet  $V \cup \{\#\}$  tels que chaque symbole apparaît au plus une fois dans  $X$ , et  $\#$  exactement une fois.

Mise-à-jour  $f : V \times M \rightarrow M$  :

$$f(v, X) = \begin{cases} x\#yzv & \text{si } X = xv\#yz \\ xy\#zv & \text{si } X = x\#y\#zv \\ Xv & \text{si } v \text{ n'apparaît pas dans } X \end{cases}$$

Exemple : à  $\pi = 4, 1, 2, 3, 2, 3, 1, \dots$  dans  $\mathcal{A}$  correspond  $\pi' = (4, \#4), (1, \#41), (2, \#412), (3, \#4123), (2, 41\#32), (3, 41\#23), (1, 4\#231), \dots$  dans  $\mathcal{A}'$ .

On met  $s(v) = \#v$  et  $\chi(v, x\#y) = 2|y|$  si le sous-ensemble de  $V$  correspondant à  $y$  est dans  $\mathcal{F}$ , et  $2|y| - 1$  sinon.

## Propriétés

Soient  $\pi = v_0, v_1, v_2, \dots$  et  $\pi' = (v_0, X_0), (v_1, X_1), (v_2, X_2), \dots$  t.q.  $X_0 = \#v_0$ ,  $X_i = x_i \# y_i$ . Il existe  $n \geq 0$  t.q. :

- Pour tout  $m \geq n$  :  $v_m \in \text{Inf}(\pi)$ ,  $x_m \in (V \setminus \text{Inf}(\pi))^* \text{Inf}(\pi)^*$  et  $y_m \in \text{Inf}(\pi)^*$ .
- Infiniment souvent,  $|y_m| = |\text{Inf}(\pi)|$ .
- $\max(\chi(\pi')) = 2|\text{Inf}(\pi)|$  si  $\text{Inf}(\pi) \in \mathcal{F}$ , et  $2|\text{Inf}(\pi)| - 1$  sinon.

# Logique et automates sur mots infinis

Automates  $\mathcal{A} = (Q, \Sigma, \delta, q_0, \text{Acc})$ ,  $\delta \subseteq V \times \Sigma \times V$

Exécution  $\rho = q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} q_2 \cdots$ . Condition d'acceptation  $\text{Acc}$  porte sur  $\text{Inf}(\rho) := \text{Inf}(q_0 q_1 \cdots)$ .

- Büchi :  $\text{Acc} = F \subseteq Q$ ,  $\rho$  accepté si  $\text{Inf}(\rho) \cap F \neq \emptyset$ .
- Muller :  $\text{Acc} = \mathcal{F} \subseteq \mathcal{P}(Q)$ ,  $\rho$  accepté si  $\text{Inf}(\rho) \in \mathcal{F}$ .
- Rabin :  $\text{Acc} : E_0 \subseteq F_0 \subseteq \cdots \subseteq E_k \subseteq F_k$ ,  $\rho$  accepté s'il existe  $i$  t.q.  $\text{Inf}(\rho) \cap E_i \neq \emptyset$  et  $\text{Inf}(\rho) \cap F_{i-1} = \emptyset$ .
- Streett : dual de Rabin.
- Parité : coloriage  $\chi : Q \rightarrow \{1, \dots, k\}$ ,  $\rho$  accepté si  $\max(\chi(\text{Inf}(\rho)))$  pair.

## Propriétés

Les automates non-déterministes de Büchi et les automates déterministes de Muller/Rabin/Streett/parité ont le même pouvoir expressif. Les automates déterministes de Büchi sont strictement moins expressifs.

# Logique et automates sur mots infinis

## Logique monadique du second ordre (MSOL)

- Variables du premier ordre  $x, y, \dots$  et du second ordre  $X, Y, \dots$
- Propositions atomiques  $a(x)$  ( $a \in \Sigma$ ),  $x < y$ ,  $y = x + 1$ ,  $x \in X$ .
- Connecteurs logiques  $\neg, \wedge, \vee, \dots$
- Quantificateurs  $\exists, \forall$ .

Interprétation :  $a(x)$  si la position  $x$  est étiquetée par  $a$ ,  $x < y$  si  $x$  avant  $y$ , etc.

## Exemple

$$\exists X_0 \exists X_1 \quad (0 \in X_0 \wedge \forall x ((x \in X_0 \vee x \in X_1) \wedge (x \in X_0 \Leftrightarrow x \notin X_1) \wedge (x \in X_0 \rightarrow a(x)) \wedge (x \in X_0 \Leftrightarrow x + 1 \in X_1) \wedge (x \in X_1 \Leftrightarrow x + 1 \in X_0))$$

Büchi : MSO et les automates de Büchi ont le même pouvoir expressif.

# Automates alternants faibles et complémentation des automates de Büchi

## $\mathcal{B}^+(X)$

L'ensemble  $\mathcal{B}^+(X)$  des formules positives sur  $X$  est construit à partir de  $X$  et des constantes 0,1, en utilisant les opérateurs  $\vee$  et  $\wedge$ .

Un modèle  $M$  de  $\phi \in \mathcal{B}^+(X)$  est un sous-ensemble de  $X$  t.q.  $\phi$  s'évalue à 1 si on remplace tout  $x \in M$  par 1, et  $x \notin M$  par 0. L'ensemble des modèles minimaux de  $\phi$  est noté  $\text{Mod}(\phi)$ .

## Automates alternants faibles WAPA

Un WAPA est un tuple  $\mathcal{A} = \{Q, \Sigma, \delta, q_0, \chi\}$  avec  $Q$  ensemble fini d'états,  $\Sigma$  alphabet fini,  $q_0 \in Q$  état initial,  $\chi : Q \rightarrow \{1, \dots, k\}$  coloriage, ainsi que

$$\delta : Q \times \Sigma \rightarrow \mathcal{B}^+(Q)$$

où  $\mathcal{B}^+(Q)$  est l'ensemble des formules positives sur  $Q$ .

## Exécution

Une exécution d'un WAPA  $\mathcal{A}$  sur  $w \in \Sigma^\omega$  est un DAG  $G(\mathcal{A}, w)$  infini avec un sommet racine. Les sommets sont des couples  $(p, i)$ ,  $p \in Q$ ,  $i \in \mathbb{N}$  :

- $(q_0, 0)$  est la racine,
- $(p, i) \rightarrow (q, j)$  implique  $j = i + 1$ ,
- pour tout sommet  $(p, i)$  :

$$\{q \in Q \mid (p, i) \rightarrow (q, i + 1)\} \in \text{Mod}(\delta(p, w_i))$$

Un chemin infini  $\pi$  de  $G(\mathcal{A}, w)$  est acceptant, si la couleur minimale des nœuds de  $\pi$  est paire. Le DAG  $G(\mathcal{A}, w)$  est acceptant, si tous les chemins infinis sont acceptants et tous les chemins finis terminent par des nœuds  $(p, i)$  t.q.  $\delta(p, w_i) = 1$ .  $\mathcal{L}(\mathcal{A})$  est l'ensemble des mots infinis qui ont un DAG  $G(\mathcal{A}, w)$  acceptant.

## Exemple

$$\delta(q_0, a) = q_1 \wedge q_2, \delta(q_1, a) = (q_0 \wedge q_1) \vee q_2, \delta(q_2, a) = q_2, \chi(q_0) = \chi(q_1) = 2, \chi(q_2) = 1.$$

# Complémentation de WAPA et jeux

## WAPA dual

- Formule duale  $\bar{\phi}$  de  $\phi \in \mathcal{B}^+(X)$  : échanger 0, 1 et  $\vee, \wedge$ .
- WAPA dual  $\bar{\mathcal{A}} = \{Q, \Sigma, \bar{\delta}, q_0, \bar{\chi}\}$ ,  $\bar{\delta}(q, a) = \overline{\delta(q, a)}$ ,  $\bar{\chi}(q) = \chi(q) + 1$ .

## Jeu de parité faible associé

On associe à un WAPA  $\mathcal{A}$  et un mot  $w \in \Sigma^\omega$  un jeu de parité faible  $\mathcal{G}(\mathcal{A}, w)$  :

- $V_0 = Q \times \mathbb{N}$ ,  $V_1 = Q \times \mathcal{P}(Q) \times \mathbb{N}$
- $(p, i) \rightarrow (p, M, j)$  si  $j = i + 1$  et  $M \in \text{Mod}(\delta(p, w_i))$
- $(p, M, i) \rightarrow (q, j)$  si  $i = j$  et  $q \in M$
- $\chi(p, i)$  et  $\chi(p, M, i)$  est la couleur de  $p$  dans  $\mathcal{A}$ .

## Prop.

Le WAPA  $\mathcal{A}$  accepte  $w$  ssi  $J_0$  a une stratégie gagnante dans le jeu de parité faible  $\mathcal{G}(\mathcal{A}, w)$ .

# Complémentation de WAPA et Büchi

## WAPA

- $J_0$  gagne le jeu  $\mathcal{G}(\mathcal{A}, w)$  ssi  $J_1$  gagne le jeu  $\mathcal{G}(\overline{\mathcal{A}}, w)$ .
- On obtient donc :  $\mathcal{L}(\overline{\mathcal{A}}) = \Sigma^\omega \setminus \mathcal{L}(\mathcal{A})$ .

## Büchi

Pour chaque automate  $\mathcal{B}$  de Büchi à  $n$  états on peut construire un WAPA  $\mathcal{A}$  équivalent à  $O(n^2)$  états :

- $\mathcal{B} = \{Q, \Sigma, \delta, q_0, F\}$ ,  $\mathcal{A} = \{Q', \Sigma, \delta', q'_0, \chi\}$
- $Q' = Q \times \{0, \dots, 2|Q|\}$ ,
- $q'_0 = (q_0, 2|Q|)$ ,  $\chi(q, i) = i$ ,
- 

$$\delta'((p, i), a) = \begin{cases} \bigvee_{q \in \delta(p, a)} (q, 0) & \text{si } i = 0 \\ \bigvee_{q \in \delta(p, a)} (q, i) \wedge (q, i - 1) & \text{si } i > 0 \text{ et } i \text{ pair} \\ \bigvee_{q \in \delta(p, a)} (q, i) & \text{si } p \notin F \text{ et } i \text{ impair} \\ \bigvee_{q \in \delta(p, a)} (q, i - 1) & \text{si } p \in F \text{ et } i \text{ impair} \end{cases}$$

# Logique temporelle LTL

## LTL syntaxe/sémantique

$$\varphi ::= \text{tt} \mid a \in \Sigma \mid \varphi \wedge \varphi \mid \neg\varphi \mid \mathbf{X}\varphi \mid \varphi \mathbf{U} \varphi$$

Une formule LTL s'évalue sur un mot infini  $w \in \Sigma^\omega$  à une position  $i \geq 0$  :

- $w, i \models a$  si  $w_i = a$ ,
- $w, i \models \mathbf{X}\varphi$  si  $w, i + 1 \models \varphi$ ,
- $w, i \models \varphi_1 \mathbf{U} \varphi_2$  s'il existe  $k \geq i$  t.q.  $w, k \models \varphi_2$  et  $w, j \models \varphi_1$  pour tout  $i \leq j < k$ .

## Macros

- $\mathbf{F}\varphi = \text{tt} \mathbf{U} \varphi$  (“un jour  $\varphi$ ”),
- $\mathbf{G}\varphi = \neg \mathbf{F}(\neg\varphi)$  (“toujours  $\varphi$ ”),
- $\varphi_1 \mathbf{W} \varphi_2 = \mathbf{G}\varphi_1 \vee \varphi_1 \mathbf{U} \varphi_2$  (“weak until”),
- $\varphi_1 \mathbf{R} \varphi_2 = \varphi_2 \mathbf{W}(\varphi_1 \wedge \varphi_2) = \neg(\neg\varphi_1 \mathbf{U} \neg\varphi_2)$  (“release”).

## Propriétés

- On peut définir aussi des opérateurs du passé. LTL avec passé est équivalent à LTL.
- LTL est équivalent à la logique du premier ordre.
- Le problème de satisfaisabilité de LTL (étant donnée une formule LTL, existe-t-il un mot la satisfaisant ?) est PSPACE-complet.
- A partir d'une formule LTL  $\phi$  on peut construire un automate de Büchi équivalent de taille  $2^{O(|\phi|)}$  (ou un automate alternant avec condition de Büchi, de taille  $O(|\phi|)$ ).