L'automate est-il faible ?
Donner un calcul de l'automate sur le mot aba^ω.
Donner un calcul sur le mot $aaba^\omega$.
Décrire l'ensemble des mots acceptés.

Exercice 16
Calculer un automate alternant faible avec condition de parité pour le langage $(ab)^\omega$.

Exercice 17
Montrer que, pour toute formule LTL, on peut calculer en temps linéaire un automate alternant faible avec condition de parité qui accepte les modèles de ϕ.

Exercice 18
Montrer que le vide des automates alternants faibles avec condition de Büchi ou avec condition de parité est PSPACE-complet. (Attention : ce n'est pas une conséquence directe des théorèmes 17 et 20 car, si l'on enchaine ces deux constructions, la taille de l'alphabet est exponentielle).

6.3 Passage des automates de Büchi aux automates alternants faibles avec condition de parité

Lemme 23 *Etant donné un automate de Büchi quelconque $\mathcal{A}$, on peut construire en temps polynômial un automate alternant faible avec condition de parité $\mathcal{A}'$ tel que $L(\mathcal{A}) = L(\mathcal{A}')$.*

Preuve: Soit $\mathcal{A} = (Q, Q_f, q_0, \Delta)$ un automate de Büchi. On construit $\mathcal{A}'$ de la manière suivante :
- l'ensemble des états est $Q \times \{0, \ldots, 2 \times n\}$ où n est le nombre d'éléments de Q (si bien que $\mathcal{A}'$ contient $O(n^2)$ états).
- Le rang est donné par la deuxième composante des états.
- L'état initial est $(q_0, 2n)$.
- La fonction de transition est définie par :
  - pour tout $q \in Q$, $(q, 0) \xrightarrow{a} \bigvee_{(q,a,q') \in \Delta}(q', 0)$. Autrement dit, sur la composante 0, $\mathcal{A}'$ se comporte comme $\mathcal{A}$.
  - Si $i > 0$ est pair, $(q, i) \xrightarrow{a} \bigvee_{(q,a,q') \in \Delta}(q', i) \wedge (q', i-1)$. La conjonction force alors à rencontrer de nouveau un état pair, ce qui ne sera possible que si l'on rencontre un état final.
  - Si i est impair et $q \notin Q_f$, alors $(q, i) \xrightarrow{a} \bigvee_{(q,a,q') \in \Delta}(q', i)$. Autrement dit, on reste sur le rang impair tant qu'on a pas rencontré un état final.
  - Si i est impair et $q \in Q_f$, alors $(q, i) \xrightarrow{a} \bigvee_{(q,a,q') \in \Delta}(q', i-1)$.

Supposons tout d'abord que ρ est un calcul réussi de $\mathcal{A}$ sur w. On construit ρ', calcul de $\mathcal{A}'$ sur w, par récurrence sur la longueur de la position p de ρ', tel que $\rho'(p) = (\rho(|p|), i)$ pour un certain i :
- La racine est étiquetée par $(q_0, 2n)$.
- Si p est une position de ρ' étiquetée par (q, i) et $\rho(|p|) = q$, $w(|p|) = a$ et $\rho(|p| + 1) = q'$, on ajoute un ou deux fils à p dans ρ', en respectant la relation de transition ci-dessus définie et de telle manière que chacun des fils soit étiqueté par un état de la forme (q', j).

Tous les chemins de ρ' sont alors infinis et contiennent une infinité d'états de la forme (q_f, i) où $q_f \in Q_f$. Comme l'automate est faible, sur chaque chemin le rang se stabilise sur une certaine valeur. Cette valeur ne peut pas être impaire puisque, lorsque le rang est impair, en passant par un état final, le rang décroit. Il en résulte que les rangs se stabilisent sur chaque branche sur des valeurs paires et donc le calcul est réussi.

Réciproquement, supposons que ρ' est un calcul acceptant de $\mathcal{A}'$ sur w. On définit une *tranche* de ρ' comme étant l'ensemble de tous les états étiquetant des nœuds de même profondeur. Remarquons que, si une tranche contient $(q, 2i)$, $i > 0$, alors la tranche suivante contient un état $(q', 2i-1)$. Si le calcul est acceptant le nœud étiqueté $(q', 2i-1)$ doit avoir un descendant étiqueté par $(q'', 2i-2)$. Donc, par récurrence et puisque l'état initial est $(q_0, 2n)$, à partir d'une certaine profondeur, toutes les tranches d'un calcul acceptant contiennent, pour chaque $i \in \{0, \ldots, n\}$ un état $(q_i, 2i)$. De plus comme, il n'y a qu'un nombre fini de tranches possibles, il existe une tranche S répétée infiniment souvent et telle que S contient au moins un état (q, i) pour chaque rang pair i. De plus, on peut supposer sans perdre de généralité (comme vu ci-dessus) que, de chaque état $(q, 2i)$, $i > 0$, de S, il existe un chemin dans ρ' vers un état $(q', 2i-2) \in S$. On choisit alors de numéroter les états de Q de manière à ce qu'il existe un chemin dans ρ de $(q_i, 2i)$ vers $(q_{i-1}, 2i-2)$ pour $i > 0$. Par définition de $\mathcal{A}'$, ce chemin passe par un état $(q_i^f, 2i-1)$ tel que $q_i^f \in Q_f$.

Maintenant, au moins deux des états $q_0, \ldots, q_n$ sont identiques. Disons $q_i = q_{i-k}$. On construit alors un calcul de $\mathcal{A}$ comme suit ; On commence par un chemin quelconque π_0 qui conduit à une position $(q_i, 2i)$ à une profondeur p_0 dont la tranche est S. On construit ensuite π_1 conduisant de $(q_i, 2i)$ (à la profondeur p_0) à $(q_{i-k}, 2i-2k)$, à la profondeur p_1 (passant par $k+1$ tranches S). De même, on construit par récurrence un chemin π_{m+1} conduisant de $(q_i, 2i)$ (à la profondeur p_m) à $(q_{i-k}, 2i-k)$, à la profondeur p_{m+1}. Si l'on oublie les deuxièmes composantes du mot $\pi_0\pi_1 \cdots \pi_m \cdots$, on obtient un calcul réussi de $\mathcal{A}$ sur w : comme $q_i = q_{i-k}$, il s'agit bien d'un calcul de $\mathcal{A}$ et, comme chaque π_m, $m \geq 1$, contient nécessairement une paire (q_f, j) avec $q_f \in Q_f$, il s'agit d'un calcul réussi. $\square$

Exercice 19
Construire un automate alternant faible avec condition de parité qui accepte le langage $(a+b)^* a^\omega$.

Exercice 20
Soit $\mathcal{A}$ un automate de Büchi déterministe et complètement spécifié à n états. Construire un automate alternant faible avec condition de parité qui accepte le même langage et qui comporte $2n$ états.

Exercice 21
Construire un automate alternant à parité faible pour chacun des langages définis dans les figures 13 et 14, en suivant la construction du lemme.

Exercice 22
Montrer que, dans la construction de la preuve du lemme 23, on ne peut se contenter d'un nombre fixe de copies de l'automate de départ.

Plus précisément, pour tout k, donner un automate $\mathcal{A}_k$ tel que l'automate à parité obtenu par la construction du lemme, avec comme état initial $(q_0, 2k)$ (au lieu de $(q_0, 2n)$) n'accepte pas le même langage que $\mathcal{A}_k$.