Samuel Lévesque

IFT 2015, Structures de données
TP1 Ultimate tic-tac-toe, H17

En réfléchissant au problème, je me suis rendu compte que le fait de toujours devoir recalculer si les sous-parties sont gagnées et par qui demandait beaucoup de calcul. J'ai donc décidé "d'augmenter" mes entiers représentant les parties afin de garder cette information en mémoire.

Définition (Entier généralisé [1]) **:** Dans l'énoncé du problème, on parle de représenter une partie comme un entier de 169 bits, par exemple la partie p serait

$$p = 0\,\mathtt{b}\ \underbrace{b_{168}b_{167}\ldots b_{163}}_{\text{encodage des coups}}\ \underbrace{b_{162}b_{161}\ldots b_1 b_0}_{\text{position du dernier coup}}$$

Dans mon cas, je définit un entier généralisé comme un entier de 189 bits. J'ajoute donc 20 bits que j'appelle le `scoredboard` (j'en ai seulement besoin de 18, mais je pose le 189-ième et le 170-ième comme 1 pour des questions de clarté d'affichage, question de délimiter le `scoreboard` lorsqu'il est vide).
Avec les 18 bits centraux, je peux donc encoder les états des 9 sous-parties selon l'encodage suivant, avec $B(i) = 170 + (18 - 2 \cdot (i + 1))$

$$b_{B(i)+1}b_{B(i)} = \begin{cases} 00 & \text{si la sous-partie n'est ni gagnée, ni nulle} \\ 01 & \text{si la sous-partie est gagnée par } \mathtt{x} \\ 10 & \text{si la sous-partie est gagnée par } \mathtt{o} \\ 11 & \text{si la sous-partie est nulle} \end{cases} \qquad \text{pour} \quad i = 0, 1, \ldots, 8$$

On obtient donc une partie représentée par une entier généralisé p_g comme

$$p_g = 0\,\mathtt{b}\,1\ \underbrace{b_{187}b_{186}\ldots b_{170}}_{\mathtt{scoreboard}}\ 1\ \underbrace{b_{168}b_{167}\ldots b_{163}}_{\text{encodage des coups}}\ \underbrace{b_{162}b_{161}\ldots b_1 b_0}_{\text{position du dernier coup}}$$

1. Je n'aime pas ce nom, mais c'est ce qui m'est venu en tête en premier et j'ai pas voulu perdre de temps sur un nom.