

LZW. Décompression

Cas problématique

Tout d'abord, il convient de bien remarquer que la construction du tableau lors de la phase de décompression se fait avec un temps de retard par rapport à la construction du dictionnaire lors de la phase de compression.

Par exemple, si le texte commence par

A B ...

le suffixe AB sera ajouté au dictionnaire lors de l'encodage du caractère A.

Mais il ne sera ajouté à la table seulement lors du décodage du caractère B -

C'est ce retard qui va entraîner le cas problématique de la décompression.

Plus précisément, le cas problématique intervient si la compression émet un code en sortie juste après l'avoir introduit dans le dictionnaire.

Regardons comment ce cas peut se produire.

Introduction du code dans le dictionnaire

Avec les notations du TP 35, on étudie un suffixe m , et le caractère suivant x :

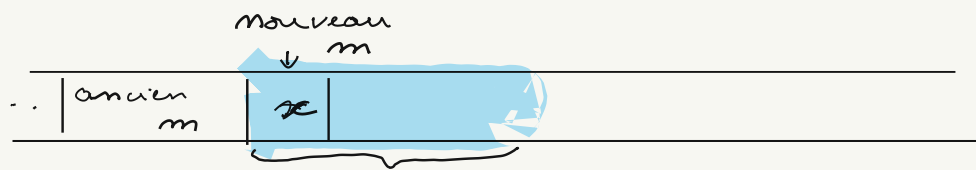
...	m	x	...
-----	-----	-----	-----

- m est déjà dans le dictionnaire
- mx ne l'est pas encore

→ on ajoute mx au dictionnaire, associé au code c

→ on émet le code associé à m , disons c' .

Emission du code : à l'étape suivante, m a maintenant pris la valeur x et on travaille avec les caractères suivants -
Si c'est le code c qui doit être émis, alors on doit retrouver exactement le suffixe mx



on doit retrouver m
pour émettre le code c

On voit bien qu'on doit avoir égalité entre :

- x suivi d'un certain nombre de caractères
- m

Le caractère x doit donc se retrouver
en première position du suffixe m .

Lors de la phase de décompression, si on
tombe sur un code c égal à la taille
du tableau actuel, c'est qu'on est dans
le cas précédent.

À partir de c' , le code précédent, la lecture
de $t[c']$ nous donne le mot m .

Il suffit alors de récupérer le premier
caractère de m et de le rajouter à
la fin pour obtenir le suffixe associé
à c .

$$t[c] \leftarrow t[c'] \wedge t[c'][0]$$