



Département de génie informatique et de génie logiciel

INF8770
Technologies multimédias

Travail pratique 1

Hiver 2020
Section 01

Carine Thouin - 1837383
Soukaina Moussaoui - 1728325

Le 7 février 2020

Introduction

Dans le cadre de ce premier travail pratique, notre mandat sera d'analyser et de comparer deux méthodes de codage de compression : le codage par plage et le codage Lempel-Ziv-Welch. De cette manière, nous formulrons dans un premier temps une liste d'hypothèses par rapport à l'utilisation de ces deux méthodes. Ensuite, nous réaliserons une série d'expériences afin de tester les hypothèses rédigées. Finalement, nous analyserons les résultats de chaque expérience selon différents critères d'évaluation et nous ferons un retour sur les hypothèses.

Question 1

Codage par plage

Le codage par plage, en anglais *run-length encoding*, est une méthode facile à utiliser. La technique consiste à compter le nombre de symboles identiques consécutifs. Elle est beaucoup plus efficace lorsqu'il y a moins de symboles uniques. De plus, plus le nombre de symboles consécutifs est grand, plus le message sera compressé. Le codage par plage nécessite un compteur. Sa taille est variable et elle influence l'efficacité du codage par plage.

Codage Lempel-Ziv-Welch (LZW)

Le codage LZW est une méthode peu optimale à utiliser. La technique consiste à associer un code pour toutes les séquences rencontrées lors du codage. De plus, la longueur des codes associés aux symboles varie tout au long du codage. Ainsi, le dictionnaire final peut prendre beaucoup d'ampleur rapidement.

Hypothèse 1

Pour un long message avec une grande répétition de symboles identiques consécutifs et avec un petit nombre de symboles uniques, le codage LZW sera moins efficace, car la taille des éléments du dictionnaire augmente au fur et à mesure que le message est codé. Par contre, inversement, pour un petit message avec plusieurs symboles uniques, le codage LZW sera plus efficace.

Hypothèse 2

Pour un long message avec une grande répétition de symboles identiques consécutifs et avec un petit nombre de symboles uniques, la taille du compteur du codage par plage aura un impact sur la longueur du message compressé. En effet, un compteur de 4 bits a une valeur maximale de 15, donc une très grande répétition d'un symbole unique peut être compressée beaucoup plus efficacement.

Hypothèse 3

Pour un long message comme un texte de la vie courante, le codage LZW sera efficace, car la probabilité que les séquences soient souvent répétées est plus grande. Ainsi, les symboles dans le dictionnaire seront réutilisés fréquemment.

Question 2

Expérience 1

Notre première expérience est d'encoder un petit message sans répétition et avec des symboles uniques différents. Nous avons choisi un compteur de taille 2 pour le codage par plage. Le message utilisé est "sowhat".

Expérience 2

Notre deuxième expérience est d'encoder un message avec beaucoup de répétitions de deux symboles uniques consécutifs : A et B. Pour le codage par plage, le compteur est de taille 2. Le message utilisé est : "AAAAAAAAAAAAAABBBBBBBBBBBBBB".

Ainsi, ces deux premières expériences vont permettre de vérifier la véracité de notre première hypothèse et de comparer les résultats de deux messages complètement différents.

Expérience 3

Dans notre troisième expérience, le même message que celui de la deuxième expérience est utilisé. Cependant, nous comparons le codage par plage selon un compteur de tailles différentes. Nous utilisons nos résultats de l'expérience 2 pour le compteur de taille 2. Ces données sont comparés avec un compteur de taille 4. Le message à encoder est donc : "AAAAAAAAAAAAAABBBBBBBBBBBBBB".

De cette façon, cette expérience permettra de vérifier la deuxième hypothèse qui compare l'efficacité du codage par plage avec des compteurs de tailles différentes.

Expérience 4

Dans notre cinquième expérience, nous utilisons un paragraphe tiré d'un article de presse. La taille du compteur pour le codage par plage est de 2. La partie de l'article a été pris sur le site <https://www.lapresse.ca/actualites/environnement/202001/30/01-5258855-le-plastique-a-usage-unique sera-banni-au-canada-en-2021.php>. Le texte utilisé est le suivant:

"En juin, le premier ministre Justin Trudeau avait fait part de l'intention de son gouvernement d'éliminer des objets de plastique à usage unique tels les sacs d'épicerie, les couvercles pour le café, les pailles et les bouteilles. Il avait affirmé que le gouvernement mènerait des recherches pour déterminer le meilleur plan d'action, fondé sur des preuves scientifiques en vertu de la Loi canadienne sur la protection de l'environnement."

Ainsi, cette dernière expérience permettra de vérifier la véracité de notre dernière hypothèse. En effet, un texte en français contient un grand nombre de répétitions de mots tels que les prépositions ("de", "à", etc.), ce qui est idéal pour tester cette hypothèse.

Critères d'évaluation pour toutes les expériences

1. Le taux de compression, calculé selon la formule suivante :
1 - (Taille du message compressé / Taille du message original)
2. La durée de temps en secondes pour encoder les messages

Code informatique

Les codes utilisés pour la méthode de codage par plage et la méthode de codage LZW sont extraits du GitHub du professeur Bilodeau (<https://github.com/gabilodeau/INF8770>).

Question 3

Le code a été adapté afin de faciliter notre lecture des résultats. De plus, pour notre troisième critère, il fallait un moyen d'utiliser des *timestamps* pour obtenir un temps de début et un temps de fin. Nous avons utilisé la librairie Time de Python (<https://docs.python.org/3/library/time.html>). Le code a donc été changé afin de calculer le temps d'exécution du codage, i.e. la création du dictionnaire initial et l'encodage du message. Voici les résultats obtenus :

Tableau I: Expérience 1

Codage	Plage (compteur 2)	LZW
Longueur originale message	18	18
Longueur finale message	30	21
Taux de compression	-0.666	-0.166
Temps d'exécution (sec)	0.0359	0.0388

Tableau II: Expérience 2

Codage	Plage (compteur 2)	LZW
Longueur originale message	47	47
Longueur finale message	36	41
Taux de compression	0.234	0.128
Temps d'exécution (sec)	0.0349	0.0419

Tableau III: Expérience 3

Codage	Plage (compteur 2)	Plage (compteur 4)
Longueur originale message	47	47
Longueur finale message	36	20
Taux de compression	0.234	0.574
Temps d'exécution (sec)	0.0349	0.0488

Tableau IV: Expérience 4

Codage	Plage (compteur 2)	LZW
Longueur originale message	2628	2628
Longueur finale message	3456	2105
Taux de compression	-0.315	0.199
Temps d'exécution (sec)	0.0399	0.0486

Question 4

Taux de compression

Pour la première expérience, les résultats démontrent que les deux méthodes ne sont pas efficaces pour ce type de message. En effet, le taux de compression des deux techniques est négatif. La longueur du message compressé est alors plus grande que celle du message original. Étant donné qu'il n'y a pas de répétition de symboles, pour le codage par plage, chaque symbole est encodé individuellement avec le compteur et son code provenant du dictionnaire initial, ce qui rend la taille du message compressé plus grande. En ce qui concerne le codage LZW, chaque symbole est encodé individuellement avec son code provenant du dictionnaire initial également. Par contre, il n'y a pas de compteur, ce qui le rend moins efficace comme nous pouvons le constater dans le premier tableau.

En ce qui concerne la deuxième expérience, les deux méthodes ont réussi à compresser le message, mais pas de manière très efficace. Le *Tableau II* affiche un taux de compression de 23.4 % pour le codage par plage et de 12,8 % pour le codage LZW. En effet, pour le codage par plage, puisque le compteur est de taille 2, le nombre maximal de répétitions qu'on peut encoder en même temps est de 3. Pour le message choisi, étant donné qu'il y a un grand nombre de symboles identiques consécutifs et qu'on ne peut encoder que 3 en même temps, le codage par plage perd son efficacité. Pour ce qui est de LZW, celle-ci est encore moins efficace que le

codage par plage, car au fur et à mesure qu'on avance dans l'encodage du message, on ajoute des symboles. Ainsi, la taille de tous les symboles dans le dictionnaire augmente aussi progressivement.

Par rapport à la troisième expérience, notre but était de comparer les résultats du codage par plage en utilisant des compteurs de tailles différentes. Ainsi, nous constatons qu'un compteur de plus grande taille (4) a rendu la compression du message plus efficace que celui avec une plus petite taille (2). Dû au fait que notre message contient deux grandes répétitions de symboles identiques consécutifs, avec un compteur de taille 4, il est plus efficace d'encoder un grand nombre de symboles en même temps, plus spécifiquement le compteur a un maximum de 15. Avec un compteur de taille 2, il est possible d'encoder au plus 3 symboles en même temps.

Finalement, en ce qui concerne la quatrième expérience, les résultats démontrent que le codage par plage pour un texte courant n'est absolument pas efficace. En effet, le taux de compression est négatif. Cela est dû au fait que dans un texte courant, il n'y a presque pas de symboles identiques consécutifs. De ce fait, on encode chaque symbole individuellement avec son code provenant du dictionnaire initial et son compteur. Cela rend alors le message encodé plus grand. Quant au codage LZW, comparativement au codage par plage, la compression du message est plus efficace, car un texte courant contient des répétitions de symboles régulières telles que "de", "les", etc. Le code des symboles répétés peut alors être réutilisé.

Temps d'exécution

Pour ce qui est du temps d'exécution, nous avons réalisé durant nos expériences que ce critère n'est pas fiable. En effet, la durée de temps variait énormément d'un test à l'autre, et ce pour la même expérience. Cela dépendait de l'ordinateur et ses capacités, ainsi que du taux d'utilisation du CPU. Nous avons donc décidé de ne pas utiliser ce critère pour analyser les résultats de nos expériences.

Retour sur les hypothèses

Dans notre première hypothèse, pour un long message avec une grande répétition de symboles identiques consécutifs et avec un petit nombre de symboles uniques , le codage LZW sera moins efficace, car la taille des éléments du dictionnaire augmente au fur et à mesure que le message est codé.

Les expériences 1 et 2 permettent alors de vérifier notre première hypothèse. Étant donné que le message de l'expérience 1, "sowhat", n'a aucune répétition et plusieurs symboles uniques, le codage LZW a été plus efficace que le codage par plage. En outre, le message de l'expérience 1 est complètement opposé à celui de l'expérience 2 qui contient une grande répétition de symboles et seulement deux symboles uniques. L'expérience 2 nous démontre que le codage par plage pour ce type de message est plus efficace, car le taux de compression est plus élevé. Nous pouvons conclure que notre première hypothèse **est confirmée**.

Dans la deuxième hypothèse, pour un long message avec une grande répétition de symboles identiques consécutifs et avec un petit nombre de symboles uniques, la taille du compteur du codage par plage aura un impact sur la longueur du message compressé. En effet, un compteur de 4 bits a une valeur maximale de 15, donc une très grande répétition d'un symbole unique peut être compressée beaucoup plus efficacement.

L'expérience 3 a permis de vérifier notre seconde hypothèse. En effet, nous constatons qu'avec un compteur de taille 4, la compression du message a été plus efficace qu'avec un compteur de taille 2. La raison de cette différence est expliquée dans la section d'analyse. Nous pouvons conclure que notre deuxième hypothèse **est confirmée**.

Dans la dernière hypothèse, pour un long message comme un texte de la vie courante, le codage LZW sera efficace, car la probabilité que les séquences soient souvent répétées est plus grande. Ainsi, les symboles dans le dictionnaire seront réutilisés fréquemment.

L'expérience 4, qui utilise un texte en français tiré d'un article de presse, a permis de vérifier notre troisième hypothèse. Nous constatons dans les résultats des taux de compression que le codage LZW est plus efficace que le codage par plage pour ce type de texte. La justification de cette différence des valeurs de taux de compression pour les deux méthodes est expliquée dans la section d'analyse. Nous pouvons conclure alors que notre troisième hypothèse **est confirmée**.