



**POLYTECHNIQUE
MONTRÉAL**

UNIVERSITÉ
D'INGÉNIERIE

INF8770 – TECHNOLOGIES MULTIMÉDIAS

Automne 2022

TP No. 1

Groupe 01 (Équipe A01)

2020533 – Léon Le Brun

2020915 – William Paré

Samedi le 4 octobre 2022

Codage prédictif et Codage arithmétique

Question 1

Formulez une liste d'hypothèses à tester pour caractériser la performance des deux méthodes. Expliquez vos différentes hypothèses. Exemple : La méthode 1 donnera un code plus compact que la méthode 2 si le message...

Nous avons 3 hypothèses sur le codage prédictif versus le codage arithmétique. Les voici :

1. Dans tous les cas, le codage prédictif sera plus rapide que le codage arithmétique.
2. Le codage prédictif donnera un code plus compact si les différences entre les symboles sont de petite taille.
3. Le codage arithmétique donnera un code plus compact si les différences entre les éléments sont de grandes tailles.

Pour l'hypothèse 1, nous supposons que le temps de calcul de la différence entre un certain caractère et celui qui le précède est très court. C'est ce qu'emploie la méthode de prédiction. Alors que le codage arithmétique nécessite du temps de calcul pour trouver la fréquence de chaque caractères, créer une table d'intervalle à chaque caractères lors du parcours du message et ensuite faire la transformation du chiffre flottant obtenu en binaire. En bref, nous croyons que peu importe la donnée (message/image) à coder, le temps requis pour l'encodage prédictif sera toujours plus court que pour l'arithmétique.

Pour l'hypothèse 2, puisque le codage prédictif se fie à la valeur précédente dans le message pour coder, nous croyons que s'il y a peu de variation, les erreurs de prédictions seront petites et donc l'encodage sera compact. De l'autre côté, nous croyons que le codage arithmétique n'est pas affecté par l'ampleur des différences entre les symboles. En effet, son résultat vient toujours de la transformation d'un chiffre flottant entre 0 et 1 qui est déterminé par le nombre de caractères et l'ordre en bit.

Pour l'hypothèse 3, comme mentionné dans l'explication de notre deuxième hypothèse, nous croyons que le codage arithmétique n'est pas touché par la différences entre les caractères, que celle-ci soit grosse ne lui importe pas alors que dans le prédictif, cette variation va créer des grosses erreurs de prédiction et donc une moins bonne compression.

Question 2

Décrivez en détails les expériences que vous allez réaliser pour vérifier les hypothèses formulées (bases de données utilisées, contenu des messages, critère d'évaluation, code informatique utilisé, etc.). Les expériences doivent être menées sur deux des trois (3) types de données suivantes : Fichier WAV, Fichier texte (Chaîne de caractères significativement longue) ou Images de formats quelconques (JPEG, PNG, ...)

Pour l'hypothèse 1:

Cette hypothèse concerne la rapidité des deux méthodes, donc nous allons prendre en note le moment juste avant chaque encodage puis le moment juste après l'encodage pour ensuite faire la différence de ces deux moments. De cette manière nous allons pouvoir comparer les delta du temps de nos expériences. Nous allons avoir 10 intervalles de temps à comparer, soit les intervalles des encodages suivants: Codage prédictif et arithmétique sur 4 textes à longueur et nombres de caractère uniques différents et sur une image (une image en ton de gris avec peu de variance de tons). Ces dix tests nous démontreront dans des conditions différentes (Peu de symboles uniques, beaucoup de symboles uniques, des integers, des caractères, répétition, etc.) lequel des encodages est le plus rapide ce qui est notre critère de réussite.

Pour l'hypothèse 2 :

Cette hypothèse concerne la longueur du message encodé d'un message avec peu de variation. Nous allons donc noter le nombre d'éléments qui constitue ce que l'on veut encoder et comparer ce nombre avec le nombre d'éléments qui composent l'encodage. En faisant ceci nous pouvons voir lequel a réussi à rendre le message le plus compact. Nous allons donc faire deux tests pour cette hypothèse. Nous allons commencer par encoder un texte n'ayant que deux caractères uniques qui se suivent dans le code ASCII (il y a donc peu de différence entre eux). Ensuite, nous allons encoder avec les deux techniques une image dont tous les pixels sont noirs, il n'y aura donc aucune variation pour ce test. Le critère de réussite est simple, celui qui a le message le plus bas pour les deux tests si possible, sinon celui qui a la moyenne de longueur la plus basse.

Pour l'hypothèse 3 :

À l'inverse de l'hypothèse 2, puisque la différence entre les caractères qui se suivent est importante pour le prédictif, nous pensons, que si nous utilisons un message avec beaucoup de variance, la technique prédictive sera punie alors que la technique arithmétique ne verra pas trop de différence (une petite lors du calcul de plage d'intervalles). Afin de tester ceci, nous allons créer un texte avec 52 caractères

unique de longueur de 5000 et nous allons utiliser une image avec des tons de gris aléatoires. Dans ces deux tests, il est rare que deux caractères consécutifs aient une faible différence, nous allons donc pouvoir voir si, comme nous le croyons, l'impact sur le prédictif sera assez gros pour que l'arithmétique soit plus compacte. Le critère de réussite sera la même que dans l'hypothèse 2, c'est-à-dire, la technique qui à la meilleure compression soit dans les deux tests ou en moyenne si les résultats sont en conflits.

Pour ces trois hypothèses, tous les textes créés pour les tests ont été créés par nous même. Pour les images, nous les avons mis dans github afin de pouvoir les mettre facilement dans votre notebook. Ce sont des images avec tout droit d'usage trouvé sur google.

Question 3

Décrivez les codes informatiques utilisés pour réaliser les expériences. Décrivez, si applicable, comment vous avez adapté les codes informatiques pour réaliser les expériences décrites à la question 2. Donnez les résultats obtenus pour les expériences décrites à la question 2 sous un format approprié.

Pour réaliser les tests, nous avons utilisé des sections de code peu modifiées déjà données dans le cours afin de pouvoir encoder l'image en ton de gris et les textes. Nous avons utilisé la librairie "Random" afin de créer les textes de manières aléatoires et pour chaque type de tests de textes (voir tableau #1 pour les types de tests sur les textes) nous avons créé un nouveau texte. Bien sûr, l'encodage prédictif et arithmétique ont le même texte pour un cas donné. Nous avons aussi utilisé la librairie "Time" afin de pouvoir connaître la durée des encodages. Pour être plus précis, le code pour faire l'encodage de prédiction pour le textes, nous avons suivi la démarche décrite dans les notes de cours. Pour les images, nous avons pris le code donné par le professeur. Pour l'encodage arithmétique, le code pour les tests de textes est celui donné par le professeur dans le github. Nous avons ensuite modifié ce code pour avoir une version qui fonctionne avec les images en gris. Pour ce faire, nous avons modifié le début afin de créer une liste de float à la place d'avoir une matrice de float (chaque float est le ton de gris d'un pixel). De plus, nous avons utilisé un "counter" de la librairie "collections" afin d'avoir le nombre d'apparition de chaque éléments afin de créer les probabilités d'apparition.

Tableau #1 : Résultats des tests de temps sur les textes (Hypothèse 1)

Longueur du texte	Nombre de caractères uniques	Prédictif (s)	Arithmétique (s)
500	5	0.0004	0.0031
500	26	0.0004	0.0085
5000	5	0.0032	0.0273
5000	26	0.0033	0.0885

Tableau #2 : Résultats des tests de temps pour les images (Hypothèse 1)

Grosueur Image	Caractéristique de l'image	Prédictif (s)	Arithmétique (s)
100x100 en gris (smile.jpg)	Beaucoup de couleurs identiques	0.0218	72.4630

Tableau #3 : Résultats des tests de longueur de message après encodage sur base avec peu de variation (Hypothèse 2)

Tests	Prédictif (nombre de caractères)	Arithmétique (nombre de caractères)
Image 100x100 à ton unique (blackSquare.jpg)	10 000	50
Texte de 5000 caractères (2 symboles uniques)	5000	52

Tableau #4 : Résultats des tests de longueur de message après encodage sur base avec peu de variation (Hypothèse 3)

Tests	Prédicatif (nombre de caractères)	Arithmétique (nombre de caractères)
Image 100x100 de pixels de couleurs aléatoires convertie en ton de gris (random.jpg)	10 000	52
Texte de 5000 caractères (52 symboles uniques)	5000	43

Question 4

Analysez les résultats obtenus et mettez-les en relation avec les hypothèses de la question 1. Est-ce que les hypothèses sont supportées par les résultats ?

Pour revenir sur l'hypothèse 1, celle concernant la vitesse des deux méthodes. Nous étions convaincus que le codage prédictif allait être plus rapide que le codage arithmétique dans tous les cas. Notamment dû à la grande quantité de manipulation de nombre flottant. Pour conclure sur ce propos, en regardant les tableaux #1 et #2, il devient clair que, selon nos expériences, notre hypothèse est belle et bien validée. En effet, dans les cinq cas que nous avons fait le calcul du delta temps pour l'encodage prédictif et arithmétique, nous avons observé que le codage prédictif était grandement plus rapide. Nous pouvons voir que pour une image 100x100 en ton de gris, la technique prédictive était 3 321.86 fois plus rapide que le codage arithmétique (72.4630 secondes versus 0.021814 secondes). Les tests sur les textes varient entre 7 fois plus rapide pour le prédictif jusqu'à près de 100 fois plus rapide.

Pour l'hypothèse 2 et 3, nous concluons que nos propos sont infirmés. Effectivement, nous avons fait la même erreur pour les deux cas, c'est-à-dire que nous avons pas pensé au fait que le codage prédictif nous retourne l'erreur entre la prédiction et le vrai résultat sur un caractères. Donc ce codage nous retourne forcément un tableau de la même longueur que notre tableau de symbole initial. L'hypothèse 2 se basait sur le fait que ces erreurs de prédiction seraient petites lorsque le message de base avait une faible variance entre caractères alors que l'hypothèse 3 se basait sur le fait que les erreurs seraient énormes en présence d'une grosse variance. Puisque cette technique ne nous retourne que les erreurs, il y a le même nombre de symboles que dans le message original dans tous les cas.

Par la suite, la transformation du message en bits (ce que nous avons pas fait puisque le nombre de caractères était plus gros que la forme compressé de l'autre technique et donc la transformation en bits est superflu) aurait rendu le message énormément plus grand que pour la technique arithmétique. À fin d'avoir des résultats qui justifieraient possiblement ces hypothèses, il aurait fallu utiliser une autre technique d'encodage telle que Huffman afin d'encoder les erreurs obtenues par prédiction. Puisque cela n'était pas le but de ces tests, il est normal de voir que l'encodage par prédiction ne compresse pas le message de base et donc que l'arithmétique donne un résultat plus compact dans tous les cas. En se fiant aux hypothèses à la lettre, nous pourrions dire que le l'hypothèse 2 n'est pas supportée par les tests alors que le 3 l'est, mais nous croyons que ces hypothèses ne sont pas parfaites car elles se fient sur une force et une faiblesse du codage par prédiction sans parler de son vrai usage, une méthode de prétraitement utilisé avant une autre méthode de compression.

*Tous les résultats basés sur les textes varient d'une tentative à l'autre à cause que les textes sont générés aléatoirement, mais les conclusions finales sur les hypothèses restent les mêmes.