
1 BUT DU LABORATOIRE

Le but de ce laboratoire est d'étudier la notion de codage de source avec principalement le codage de Huffman, mais également de le comparer avec le codage ASCII et l'algorithme de Lempel-Ziv.

Durée : 4 périodes

2 INTRODUCTION

Ce laboratoire qui se réalise à l'aide de Mathworks Matlab utilise des fichiers sources qui sont disponible sur le serveur eint20 au répertoire suivant (à copier en local) :

`\\eint20\IICT\Laboratoires\CNU\Codage de source`

Le dossier « huffman » contient des fonctions Matlab libre de droits pour le codage et décodage de Huffman. Un fichier « ReadMe.txt » est à disposition pour visualiser les fichiers que contient ce dossier.

3 TRAVAIL

3.1 FREQUENCE DES SYMBOLES

Avant d'appliquer un codage de Huffman, il est fondamental de déterminer la fréquence (probabilité d'apparition) de chaque symbole à coder. Cela permet de coder les symboles les plus représentés avec le plus petits nombre de bits.

Pour ce laboratoire, la source de données est connue et consiste en un fichier texte « L'ingénieur.txt ». En vous aidant des fichiers Matlab fournis (notamment frequency.m), écrivez un script vous permettant de répondre aux questions suivantes.

- 3.1.1 Est-ce que la fonction « frequency.m » est compatible avec de l'ASCII étendu ?
- 3.1.2 Combien y'a-t-il de symboles (caractères) différents dans ce texte ?
- 3.1.3 Quels sont les 3 caractères les plus représentés ? Quelles sont leurs fréquences ?

Aide Matlab : Pour lire un fichier texte, utilisez la fonction « textread ».

3.2 CODAGE D'HUFFMAN

Appliquez le codage de Huffman au même fichier texte à l'aide de la fonction « norm2huff.m » et répondez aux questions suivantes.

- 3.2.1 Quel est le taux de compression obtenu entre l'ASCII et le codage de Huffman?
- 3.2.2 Quelle est la longueur moyenne des bits pour l'ASCII ?
- 3.2.3 Quelle est la longueur moyenne des bits pour le codage de Huffman ?
- 3.2.4 Quelles sont la capacité, l'efficacité et la redondance du codage de Huffman ?
- 3.2.5 Quels sont les codes de Huffman le plus petit et le plus long?

3.3 DECODAGE DE HUFFMAN

Appliquez le décodage de Huffman à l'aide de la fonction « huff2norm.m » et répondez aux questions suivantes.

- 3.3.1 Y'a-t-il une différence entre les données avant codage et après, pourquoi ?
- 3.3.2 Est-ce que les codes dans ce cas ont la propriété d'être décodé instantanément ?
- 3.3.3 Essayer de zipper le fichier texte, quel taux de compression obtenez vous ? Quelles constatations faites-vous ? Expliquez.

3.4 FINAL

Refaites le même exercice avec un texte plus long (Huffman.txt), répondez à nouveau aux questions en rapport. Faites- vous de nouvelles constatations ?

4 ANECDOTE

Les premiers Macintosh de la société Apple utilisaient un code inspiré de Huffman pour la représentation des textes : les 15 caractères les plus fréquents d'une langue étaient codés sur 4 bits, et la 16ème configuration servait de préfixe au codage des autres sur un octet (ce qui faisait donc tantôt 4 bits, tantôt 12 bits par caractère). Cette méthode simple se révélait économiser 30% d'espace sur un texte moyen, à une époque où la mémoire vive restait encore un composant coûteux... (source : Wikipédia)

5 REFERENCES

- *Cours de Communications numériques, Dedieu Hervé, 277 pages, 2006, HEIG-VD.*
- <http://www.table-ascii.com/>
- <http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=4900&objectType=File>

Auteur : Jérôme Vernez (jerome.vernez@heig-vd.ch)