

Práctica 4: Codificación de fuente¹.

Objetivos

1. Realizar pruebas sobre el codificador Huffman suministrado. Determinar el efecto del cambio de las probabilidades de bit en la fuente sobre la eficiencia del código.
2. Realizar pruebas sobre el codificador “run-length” suministrado. Determinar la capacidad de compactación del código y su relación con las longitudes de las cadenas de ceros en la misma.

Revisión teórica

En muchas aplicaciones, la fuente generadora de datos binarios produce largas secuencias de dígitos iguales. Un ejemplo de este tipo de fuentes es la señal binaria que se genera al digitalizar un documento para su transmisión vía fax. En estos casos, en lugar de transmitir muchos símbolos codificados iguales, el código se diseña de forma que estas secuencias se sustituyan de una forma más eficiente, es decir, se reemplaza la secuencia de bits idénticos por una palabra de código que representa la longitud de la secuencia. Este tipo de código se llama “run-length”, o codificación por longitud de secuencia. El código estandarizado por la antigua CCITT para uso en las máquinas de fax es del tipo “run-length”.

Por ejemplo, si una fuente binaria genera ceros (0s) con una probabilidad mucho mayor que la de los unos (1s), se aplicaría un código “run-length” para codificar únicamente las secuencias de ceros, que serían las más largas que generaría la fuente.

En esta práctica se propone el uso de un código “run-length” muy sencillo que codifica secuencias de hasta 2^m ceros (donde m es un parámetro del código). Las secuencias de ceros se codifican con un cero seguido de una palabra binaria de m bits que representa la longitud de la secuencias de ceros menos uno. Los unos se representan sin codificar, directamente con un ‘1’. Así se logra otra condición importante del código, que es la posibilidad de una decodificación unívoca: cuando el decodificador encuentra un ‘0’, entiende que los siguientes m bits representan la longitud de una secuencia de 0s, mientras que si se encuentra un ‘1’, éste se toma directamente.

El código Huffman es un código de longitud variable que permite obtener la longitud de código promedio más corta para un alfabeto de entrada determinado. Esta longitud promedio puede ser considerablemente mayor que la entropía del alfabeto de la fuente, siendo este hecho una limitación relacionada con el alfabeto, no con la técnica de codificación.

¹ Esta práctica y el script utilizado provienen de la Guía de Laboratorio de Comunicaciones Digitales, elaborada por el Prof. Renny Badra.

Preparación

1. Investigue sobre el concepto de entropía de una fuente de dígitos independientes. ¿Qué representa la entropía? Calcule la entropía de una fuente de bits independientes cuya probabilidad de ceros sea (a) $P_0 = 0.75$, (b) $P_0 = 0.85$ y (c) $P_0 = 0.95$.
2. Para cada una de las tres fuentes de dígitos binarias de la pregunta anterior, encuentre un código Huffman en el cual los símbolos se obtengan agrupando 3 bits consecutivos. Incluya el procedimiento realizado para hallar las palabras código.
3. Para los códigos hallados en la parte anterior, encuentre la longitud promedio de una palabra codificada.
4. Repita (2) y (3) para el caso de $P_0 = 0.75$ usando un código Huffman en el cual los símbolos se obtengan agrupando cuatro bits consecutivos.
5. Para la siguiente secuencia binaria, encuentre la secuencia codificada usando el código “run-length” descrito en la sección anterior, y encuentre la tasa de compresión lograda. Resuelva para (a) $m=2$, (b) $m=3$, y (c) $m = 4$. La tasa de compresión se define como el cociente entre bits de fuente y bits codificados.

11000000100000011010000000000100100010011

Trabajo práctico

Descripción de la simulación

En esta práctica se simulan dos tipos de codificador de fuente: uno por mapeo de bloques (necesario para aplicar codificación Huffman) y otro por longitud de secuencia (“run-length”). La fuente de bits no genera bits equiprobables, sino con una probabilidad de ceros (P_0) mayor que la de unos (ajustable en la línea 14 del script). Esta opción se debe a que una fuente de dígitos binarios equiprobables ya está codificada y no es susceptible de ser comprimida.

La salida del codificador se aplica a un medidor de desempeño a fin de calcular la tasa de compresión. De igual forma, se muestra su recíproco, que es el número de bits de código promedio por cada bit de fuente (NC). Estos resultados se muestran al final de la ejecución del programa. En caso de que la longitud de bits a simular (NBIT, línea 15) sea menor que 32, se muestran también las secuencias fuente y codificadas.

Cuando se usa el codificador de fuente por mapeo directo (Huffman), es necesario definir las palabras de fuente y de código que constituyen la regla de mapeo. Esto se hace en las líneas 33 y 35 a través de los vectores *fuentes* y *códigos*. En caso del código “run-length”, es necesario definir el parámetro m (longitud del bloque) en la línea 16. El codificador de fuente “run-length” está diseñado para fuentes con alta probabilidad de ceros.

El número de bits a simular es ajustable a través del parámetro NBIT. Para simulaciones estadísticas se recomienda usar NBIT = 5000 para el caso Huffman y NBIT = 50000 para el código “run-length”.

Trabajo en el laboratorio

1. Personalice la simulación introduciendo los números de carnet de los integrantes del grupo en las líneas 9 y 10.
2. Ajuste $P_0 = 0.85$, y cargue las palabras de fuente y código correspondientes al código Huffman hallado en el prelaboratorio (longitud de bloque de 3 bits). Haga NBIT = 30, y verifique el correcto funcionamiento del código, examinando las secuencias mostradas por el programa.
3. Haga NBIT = 5000 y halle la tasa de compresión del código Huffman de longitud 3 y $P_0 = 0.85$.
4. Haga $P_0 = 0.75$ y encuentre el promedio de bits codificados (NC) del código Huffman de longitud 3 diseñado para esta fuente. Repita para $P_0 = 0.95$.
5. Grafique NC y la entropía vs. P_0 . Compare los valores obtenidos de la entropía de cada fuente. Concluya.
6. Haga NBIT = 5000 y halle la tasa de compresión del código Huffman de longitud 4 y $P_0 = 0.75$. Compare con los resultados del punto (3) y concluya.
7. Ajuste $P_0 = 0.85$, y haga $m=3$ en la línea 16. Haga NBIT = 30 y verifique el funcionamiento correcto del código “run-length”, examinando las secuencias mostradas por el programa.
8. Haga NBIT = 50000 y $P_0 = 0.85$. Halle la tasa de compresión del código “run-length” de longitud $m = 3$. Repita para las otras dos fuentes $P_0 = 0.75$ y $P_0 = 0.95$. Grafique NC y la entropía vs. P_0 , y concluya en cuanto al desempeño del código.
9. Repita el punto (8) para las longitudes de bloque $m = 2$, $m = 4$, $m = 5$ y $m = 6$. Grafique NC y la entropía vs. P_0 , y concluya en cuanto al desempeño de los códigos “run-length” en función de la longitud de bloque y P_0 .

Cuestionario

1. Grafique en una misma figura NC y la entropía como función de P_0 , obtenidos para el código Huffman de longitud 3.
2. Para $P_0 = 0.75$ compare el desempeño del código Huffman de longitud 3 con el de longitud 4. Explique el comportamiento observado.
3. Muestre en una sola gráfica el desempeño (NC) de los códigos “run-length” de longitudes 2, 3, 4, 5 y 6 en función de P_0 . Explique por qué los mejores valores de m son distintos para las distintas fuentes.
4. Muestre en una sola gráfica el desempeño (NC) de los códigos “run-length” y Huffman de longitud 3 en función de P_0 . Incluya en la gráfica los valores de entropía de las respectivas fuentes. Compare el desempeño de los códigos.