LEMMA 2.3 *Suppose that $n \geq 1$ is an integer, $0 \leq r \leq 2^n - 1$, and suppose that $b_n, \ldots, b_0$ and $a_{n-1}, \ldots, a_0$ are as defined above. Then*

$$a_j \equiv (b_j + b_{j+1}) \bmod 2 \tag{2.1}$$

and

$$b_j \equiv \sum_{i=j}^{n-1} a_i \bmod 2, \tag{2.2}$$

for $j = 0, 1, \ldots, n-1$.

PROOF We begin by proving that Equation (2.1) is true for all $n \geq 1$ and $j = 0, 1, \ldots, n-1$. The proof is by induction on n. The induction can be started with $n = 1$, where Equation (2.1) can be verified easily.

For some integer $i \geq 2$ assume that Equation (2.1) is true for $n = i - 1$ and $0 \leq j \leq i - 2$. We now consider $n = i$ and $0 \leq j \leq i - 1$. Let r be an integer such that $0 \leq r \leq 2^i - 1$. We divide the proof into two cases, depending on the value of r.

The first case is when $0 \leq r \leq 2^{i-1} - 1$. In this case, we have $b_{i-1} = 0$ and $a_{i-1} = 0$. For $0 \leq j \leq i - 2$, Equation (2.1) is true by induction. For $j = i - 1$, we have

$$b_{i-1} + b_i \equiv 0 \bmod 2$$

and $a_{i-1} = 0$, so Equation (2.1) is true here as well.

Now, we proceed to the second case, $2^{i-1} \leq r \leq 2^i - 1$. In this case, we have that $a_{i-1} = 1, b_{i-1} = 1$,

$$\mathsf{G}^{i-1}_{2^i-1-r} = a_{n-2} \ldots a_1 a_0,$$

and the binary representation of $2^i - 1 - r$ is

$$0(1 - b_{n-2}) \ldots (1 - b_0).$$

Since Equation (2.1) is true for $n = i - 1$ by induction, we have

$$a_j \equiv (1 - b_j) + (1 - b_{j+1}) \bmod 2$$

for $j = 0, 1 \ldots, i - 2$. Since

$$(1 - b_j) + (1 - b_{j+1}) \equiv (b_j + b_{j+1}) \bmod 2,$$

Equation (2.1) is true for $n = i$ and $j = 0, 1 \ldots, i - 2$. For $j = i - 1$, we have

$$b_{i-1} + b_i \equiv 1 \bmod 2$$

and $a_{i-1} = 1$, so Equation (2.1) is true here as well.

By induction, Equation (2.1) is true for $j = 0, 1, \ldots, n-1$, for all integers $n \geq 1$.

To complete the proof, we show that, for any $n \geq 1$, the truth of Equation (2.1) for $j = 0, 1, \ldots, n-1$ implies the truth of Equation (2.2) for $j = 0, 1, \ldots, n-1$. This is an easy computation:

$$\begin{aligned} \sum_{i=j}^{n-1} a_i &\equiv \sum_{i=j}^{n-1} (b_i + b_{i+1}) \bmod 2 \\ &\equiv (b_j + b_n) \bmod 2 \\ &\equiv b_j \bmod 2, \end{aligned}$$

since $b_n = 0$. ∎

The relations in Lemma 2.3 give rise to the ranking and unranking algorithms for the binary reflected Gray code which are presented as Algorithms 2.4 and 2.5. We provide brief explanations.

First, consider unranking. In iteration i of the **for** loop of Algorithm 2.5, b corresponds to b_{i+1} and b' corresponds to b_i. The algorithm successively computes $b_{n-1}, \ldots, b_0$, which are the bits in the binary representation of r. Recalling that $a_i \equiv b_i + b_{i+1} \bmod 2$, we see that

$$n - i \in T \Leftrightarrow a_i = 1 \Leftrightarrow b \neq b'.$$

Now we look at ranking. In iteration i of the **for** loop of Algorithm 2.4, b corresponds to b_i. Initially, $b = 0$ (corresponding to $b_n = 0$). Since

$$b_i = b_{i+i} + a_i \bmod 2,$$

we can update b during each iteration of the **for** loop by checking if $n - i \in T$ (since $a_i = 1$ if $n - i \in T$ and $a_i = 0$ if $n - i \notin T$). Whenever $b = 1$, we add 2^i to r, since $b = b_i$ is just bit i in the binary representation of r.

Algorithm 2.4: GRAYCODERANK (n, T)

$r \leftarrow 0$
$b \leftarrow 0$
for $i \leftarrow n - 1$ **downto** 0
do
  if $n - i \in T$
    then $b \leftarrow 1 - b$
  if $b = 1$
    then $r \leftarrow r + 2^i$
return (r)