簡潔データ構造 第 1 回

marimo

2023 年 4 月 25 日

簡潔ビットベクトル

Succinct Bit Vector

定義

$\{0,1\}^n$ の元を長さ n の**ビットベクトル**と呼ぶ。$B \in \{0,1\}^n$ としたとき，

- $B[i]$ は B の i 番目のビットを表す*1
- $B[i..j]$ は B の i 番目から $j-1$ 番目までの連続する部分列を表す
- 上の記法において $i \geq j$ のときは空とする

1 0-indexed. 以降 n 番目などと言うとき 0-indexed とします

演算の定義

$B \in \{0,1\}^n$, $j \in \{0,1\}$ とする。このとき次の演算を定める

- $\mathrm{access}(B, i)$: $B[i]$ を返す
- $\mathrm{rank}_j(B, i)$: $B[0..i]$ の j の数を返す
- $\mathrm{select}_j(B, i)$: B の先頭から i 番目の j の位置を返す (共に 0-indexed)

今日の主定理

定理 (Raman, R., Raman, V., and Satti, S. R., 2007)
長さ n のビットベクトルが与えられたとき，$O(n)$ 時間の前計算によって構築される $O(n \lg \lg n / \lg n)$ ビットの索引を用いて $\mathrm{rank}, \mathrm{select}$ は語長 $\Omega(\lg n)$*[2]の Word-RAM 上で**定数時間**で計算できる

---

2 $\Omega(f(n))$ は定義を与えていないが，これもランダウの記号のひとつであって漸近的に $f(n)$ と同程度かこれより大きい関数の集合である

定理の概観

$o(n)$ bits の索引をつかって定数時間で $\mathrm{rank}, \mathrm{select}$ を処理できるのは結構非自明

ナイーブにやるなら，

- rank に答えるために prefix sum[3]を持つ
- select も i 番目の 0,1 の位置を全部持つ

みたいなものが考えられるが，これらは共に $n \lg n$ bits の空間を使うので簡潔ではない

実際のところ，rank を計算するための索引は工夫した prefix sum を用いる。select の計算にはナイーブな持ち方と n 分探索をするための索引の 2 つを使う。

3 いわゆる累積和

rankの計算

$l = \lg^2 n$*4とする。B を先頭から長さ l の**大ブロック**に分割する。

つまり，j 番目の大ブロックとは $B\left[lj..l(j+1)\right]$ のこと

ここで整数の配列 R_L を次のように定める

$$R_L[i] = \text{最初から } i-1 \text{ 番目の大ブロックまでの 1 の総数}$$

ただし $R_L[0] = 0$ とする。こうすると $x = \lfloor i/l \rfloor$ とすれば

$$\operatorname{rank}_1(B,i) = R_L[x] + \sum_{j=lx}^{i-1} B[j]$$

R_L を持つためには $O\left(n/\lg^2 n \times \lg n\right) = O(n/\lg n)$ bits だけの空間を使えばよいので，この索引は簡潔。これだけだとまだ rank に $O\left(\lg^2 n\right)$ 時間かかる。

4 これは本当はよくないのですが，$\lg^2 n = (\lg n)^2$ と思ってください

rankの計算 その 2

さらに大ブロックを長さ $s = \frac{1}{2} \lg n$ の**小ブロック**に分割する。

新たな整数の配列 R_S を次のように定める

$$R_S[i] = i\text{番目の小ブロックが属す大ブロックについて} \\ \text{その先頭から直前の小ブロックまでの中の 1 の総数}$$

ただし大ブロックの先頭の場合は 0 とする。こうすると $x = \lfloor i/l \rfloor, y = \lfloor i/s \rfloor$ とすれば

$$\mathrm{rank}_1(B, i) = R_L[x] + R_S[y] + \sum_{j=sy}^{i-1} B[j]$$

R_S を持つためには $O(n/\lg n \times \lg\lg n) = o(n)$ だけかかるので，OK。また $O(\lg n)$ 時間かかる。

rankの計算 その 3

細分化の方針だけでは定数時間にはできない。
ここで新しい方針**表引き**をします。

今のところ問題になっている $B[sy..i]$ の長さは $\frac{1}{2}\lg n$ 以下だから，Word-RAM でこのビット列を読んで整数 w と思うのは定数時間でできる。
なんと w は 0 以上 $\sqrt{n}$ 以下なので，次のような表 T を $O(\sqrt{n}\lg n\lg\lg n) = o(n)$ でもてる。

$T[w][i] = w$を 2 進表現したビット列の，先頭から i番目までの 1 の数
ただし，i番目も含む。

というわけでこれら 3 つの構造を持つことで rank が定数時間で計算できる。