

ピラミッド (PYRAMID) の解法

Hugo Ryckeboer (アルゼンチン)

スライドする部分列の和を線形時間で計算する方法はよく知られている．下图は長さ3の部分列を示している．

20	17	18	23	45	67	12	18	34	25	12	13	15	45	67	23	12	14	15	16
55	58	86	...							40									

40の次の位置に対応する和を計算するには、40に45を加えて12を引けばよい．最初の位置では3個の元の和を計算すればよい．

$2n$ 回の加法演算により、全体を計算することができる．

長さ n の数列内の長さ s の部分列の位置は、全部で $n - s + 1$ 通りある．

これは2次元の場合に拡張することができる．次のようにして、固定した大きさ $a \times b$ の長方形の和を求めることができる．まず1つの方向に対して処理を行い、次に残りの方向に対して処理を行う．まず、行に対して処理を行い、 $1 \times b$ の長方形の和を計算する．次に、このアルゴリズムを行に対して適用することで、前に求めた和の a 個の和が得られる．これは $a \times b$ の長方形の和である．

$4m \times n$ 回 (これはデータの大きさに比例する) より少ない回数の計算で、 $m \times n$ の長方形内の $a \times b$ の部分長方形の和を全て計算することができる．

ピラミッドと穴の最適な位置を求める一つの方法は、 $a \times b$ の部分長方形の和と、 $c \times d$ の部分長方形の和を全て求めることである．ピラミッドのそれぞれの位置ごとに、穴を配置する最適な場所がある．これは、その内部の $c \times d$ の和の最小値に対応する．

この最小値を単純に計算すると、穴を配置することができる場所の個数に比例した回数の計算が必要である．この個数は全体の面積で上から評価できるので、入力データの大きさに比例した係数が一つ掛かってしまう．

ピラミッドの全ての位置に関する最小値を線形時間で計算するアルゴリズムを設計しなければならない! 「和」という単語を「最小値」という単語に置き換えれば、この問題は前の問題に似ている．

しかし、和の場合に用いたアルゴリズムを適用することはできない．なぜなら、和の場合は加法の逆演算である減法を用いたが、「最小値をとる」という操作には逆演算が存在しないからである．従って、何か別のものを設計する必要がある．

まず、1次元の場合の問題を考察しよう．

17	23	45	56	32	15	34	12	43	56	45	67	23	12	11	34	43	25	14	32
						12	12	43	45	45									
									45	45	23	12	11						
						12	12	23	12	11									

この図の1行目の数字は生データである。2行目は長さ5の数列であり、右から左に部分列を大きくしていったときの最小値である。

右から左に見てみよう。45はデータ45のコピーである。2つ目の45は1つ前のデータと56のうちの小さい方である。以下同様である。

3行目も長さ5の数列であり、左から右に部分列を大きくしていった時の最小値である。最後の行の数列は、2行目と3行目の数列の対応する箇所を比べて、そのうち小さい方をとることで得られる。これが求める結果である。最後の長さ5の数列は、長さ5の補助の配列を2つ用いることで計算できる。従って、配列全体の長さの3倍より少ない回数の計算で、スライドする部分列の最小値を計算することができる。

これを他の方向にも適用することで、固定した大きさの全ての部分長方形に対して、その中の値の最小値を計算することができる。

$m \times n$ 行列 A に対してこの結果を適用することで、次のようにして最適な位置を求めることができる。

1. 大きさ $a \times b$ の A の部分長方形の和を並べたもの。これは $(m-a+1) \times (n-b+1)$ の行列である。これを B とおく。
2. 大きさ $c \times d$ の A の部分長方形であって、境界に触れないものに対して、その和を並べたもの。これは $(m-c-1) \times (n-d-1)$ の行列である。これを C とおく。
3. 大きさ $(a-c-1) \times (b-d-1)$ の C の部分長方形の最小値を並べたもの。これは $(m-a+1) \times (n-b+1)$ の行列である。これを D とおく。
4. $E = B - D$.
5. E の最大値がピラミッドの最適な場所に対応する。その座標が出力の最初の2つの数字である。
6. 上で選んだピラミッドの場所に対して、 C の最小値を計算する。これにより、穴をどこに配置すべきかが分かる。

これらの6つの操作に必要な計算回数は、問題のデータサイズに比例する。必要な計算回数の上界を m と n のみを使って表すと以下の通り。

- $9mn$ 回の加法演算 ((1) で4回, (2) で4回, (4) で1回)
- $8mn$ 回の最小値や最大値を求める演算 ((3) で6回, (5) で1回, (6) で1回)