Potential Method

14 September 2020 22:55

- Instead of credit stored with specific objects, potential method uses potential energy or potential which can be used for future operations.
- Potential with the data structure as whole rather than specific objects within the data structure.

Initial Data Structure $= D_0$

Perform 'n' operations

For each $i = 1, 2, \ldots n$,

C_i - actual cost of i^{th} operation

D_i - Data structure results after applying i^{th} operation to Data Structure D_{i-1}

Potential function

ϕ - maps each data structure D_i to real number $\phi(D_i)$, which is potential associated with data structure D_i.

Amortized Cost $\hat{C}_i$ of i^{th} operation with respect to potential function is defined by,

$$\hat{C}_i = C_i + \phi(D_i) - \phi(D_{i-1})$$

Amortized cost of each operation is actual cost plus change in potential due to operation.

Total amortized cost of n operations is

$$\sum^{n} \hat{C}_i = \sum^{n} (C_i + \phi(D_i) - \phi(D_{i-1}))$$

$$\sum_{i=1} \quad \sum_{i=1}$$

$$= \sum_{i=1}^{n} c_i + \sum_{i=1}^{n} (\phi(D_i) - \phi(D_{i-1}))$$

$$= \sum_{i=1}^{n} c_i + ((\phi(\cancel{D_1}) - \phi(D_0)) + (\phi(\cancel{D_2}) - \phi(\cancel{D_1})) + \dots + \phi(D_n) - \phi(\cancel{D_{n-1}}))$$

$$= \sum_{i=1}^{n} c_i + \phi(D_n) - \phi(D_0)$$

Let Potential function ϕ is defined so that

$$\phi(D_n) \geq \phi(D_0)$$

$\therefore$ Total amortized cost is $\sum_{i=1}^{n} \hat{c}_i$ gives upper bound of actual cost $\sum_{i=1}^{n} c_i$

* If Potential difference $\phi(D_i) - \phi(D_{i-1})$ is Positive

then amortized cost $\hat{c}_i$ represents over charged to i^{th} operation.

* If potential difference $\phi(D_i) - \phi(D_{i-1})$ is negative then amortized cost $\hat{c}_i$ represents under charged to i^{th} operation.

Example : Stack

Potential fn ϕ - no. of objects in stack.

Empty stack $\phi(D_0) = 0$.

$\therefore \phi(D_i) \geq 0$
$\geq \phi(D_0)$

Let i^{th} operation on a stack of 's' objects is push operation. Then potential difference is

$\phi(D_i) - \phi(D_{i-1}) = (s+1) - s$
$= 1$

$\therefore$ Amortized cost of Push operation is,

$$\hat{c}_i = c_i + \phi(D_i) - \phi(D_{i-1})$$

$$= 1 + 1 = 2$$

Suppose i^{th} operation on stack is multipop (S, k)

$\therefore$ No. of elements popped $\Big\} = \min(s, k)$

$= k'$ (say)

Potential difference is,

$$\phi(D_i) - \phi(D_{i-1}) = (s - k') - s = -k'$$

Amortized cost of multipop operation is,

$$\hat{c}_i = c_i + \phi(D_i) - \phi(D_{i-1})$$

$$= k' + (-k') = 0$$

Similarly amortized cost of pop operation is,

$\phi(D_i) - \phi(D_{i-1}) = (s-1) - s = -1$

Amor. Cost $\hat{c}_i = c_i + \phi(D_i) - \phi(D_{i-1})$

$= 1 + (-1) = 0$

$\therefore$ Total amortized cost of Sequence of 'n' operations is $O(n)$.