Análise Amortizada

João Victor[1]**, Luiz Gabriel Ferronatto**[2]**, Luiz Guilherme Zanella Lopes**[3]

[1]Universidade Federal da Fronteira Sul (UFFS)
Caixa Postal 181 – 89.815-899 – Chapecó – SC – Brasil

joao.vt2807@gmail.com, ferronattoluizg@gmail.com, zanelallopes9977@gmail.com

Abstract. *This work addresses the topic of amortized analysis, which is a way of analyzing the time complexity of an algorithm. It will cover the main techniques used in amortized analysis: aggregate analysis, the accounting method and the potential method; in addition to providing examples of the use of each of them and how they work. The final objective of the work is to transmit the fundamentals and knowledge on the subject, aiming to provide examples of practical applications of the content in order to generate interest for the reader.*

Resumo. *Este trabalho aborda o tema análise amortizada, que é uma forma de analisar a complexidade de tempo de um algorítimo. Nele será abordado as principais técnicas utilizadas na análise amortizada: a análise agregada, o método da contabilidade e o método potencial; além de trazer exemplos da utilização de cada um deles e como funcionam. O objetivo final do trabalho é transmitir os fundamentos e conhecimentos sobre o assunto, visando trazer exemplos de aplicações práticas do conteúdo a fim de gerar interesse ao leitor.*

1. Informação Geral

A Análise Amortizada é um método de calcular média de tempo utilizado para executar uma sequência de passos lógicos em uma estrutura de dados. Diferentemente do cálculo de média, a análise amortizada não envolve probabilidade, garantindo, como resultado, o desempenho médio de cada operação em seu pior caso. [Thomas H. Cormen 1989]

Nas próximas seções serão abordados as técnicas mais comuns para realizar essa análise: a análise agregada, o método de contabilidade e o método potencial; também será feito uma análise do uso prático dessas técnicas e sua utilidade no meio computacional. Vale ressaltar que a análise amortizada tem utilidade na otimização de um projeto, a partir de uma percepção mais profunda sobre as estruturas de dados utilizadas dentro dele. [Thomas H. Cormen 1989]

2. Análise Agregada

A análise agregada é um dos métodos utilizados dentro da análise amortizada. Nesse método não são atribuídos diferentes custos amortizados para cada tipo de operação, e ele consiste em que, dentro de uma sequência de n operações sempre teremos o pior caso, sendo essa $T(n)$, e a partir disso é possível dizer que no pior caso o custo médio ou custo amortizado é $T(n) / n$. [Stefanes 2007]

2.1. Operações em Pilha

Como primeiro exemplo da análise agregada será utilizado uma pilha. Nessa pilha existem 3 tipos de operação: **inserir**, que insere um elemento no topo da pilha e possui tempo de execução de *O(1)*; **remover**, que remove um elemento do topo da pilha, também com o tempo de *O(1)* e por fim a operação **remover_multiplos**, que possui o tempo de *O(min(s, k))*, sendo uma função linear. Considerando uma pilha onde foram realizadas n operações, o pior caso possível para a função **remover_multiplos** pode ser considerado *O(n)* dado que o máximo de tamanho da pilha é em si *n*, então se realizadas *n* operações **remover_multiplos** de custo *O(n)*, ocorrerá o pior caso $O(n^2)$. [Stefanes 2007]

Embora feita a análise de um pior caso de custo $O(n^2)$, essa análise não é exatamente precisa, dado que, se considerarmos que a pilha estava inicialmente vazia, o máximo de custo que possível de uma operação nela é a quantidade de vezes que a operação **inserir** foi realizada, então, se realizadas *n* inserções, mesmo considerando remover múltiplos elementos, o máximo de operações de remoção que podem ser realizadas é *n*, implicando em um pior caso de *O(n)*, e realizando a análise agregada, pode se obter *O(n) / n* que é igual é o custo amortizado de *O(1)*. [Thomas H. Cormen 1989]

3. O Método de Contabilidade

No método da contabilidade de análise amortizada, atribuímos cobranças diferentes a operações diferentes, sendo que algumas operações são cobradas a mais ou a menos do que realmente custam. Denominamos custo amortizado o valor que cobramos por uma operação. Quando o custo amortizado de uma operação excede seu custo real, atribuímos a diferença a objetos específicos na estrutura de dados como crédito. Mais adiante, o crédito pode ajudar a pagar operações posteriores cujo custo amortizado é menor que seu custo real. Assim, podemos considerar o custo amortizado de uma operação como repartido entre seu custo real e o crédito que é depositado ou consumido. Esse método difere da análise agregada, onde todas as operações têm o mesmo custo amortizado.

Devemos escolher os custos amortizados das operações cuidadosamente. Se quisermos mostrar que, no pior caso, o custo médio por operação é pequeno por análise com custos amortizados, temos de assegurar que o custo amortizado total de uma sequência de operações dá um limite superior para o custo real total da sequência. Além disso, como ocorre na análise agregada, essa relação deve se manter válida para todas as sequências de operações. Se denotarmos o custo real da i-ésima operação por c_i e o custo amortizado da i-ésima operação por $\hat{c}_i$, exigiremos

$$\sum_{i=1}^{n} \hat{c}_i \geq \sum_{i=1}^{n} c_i$$

para todas as sequências de n operações. O crédito total armazenado na estrutura de dados é a diferença entre o custo amortizado total e o custo real total:

$$CréditoTotal = \sum_{i=1}^{n} \hat{c}_i - \sum_{i=1}^{n} c_i$$

Pela desigualdade, o crédito total associado à estrutura de dados deve ser não negativo em todos os momentos. Se alguma vez permitíssemos que o crédito total se tornasse negativo (o resultado de cobrar a menos por operações anteriores com a promessa de reembolsar a quantia mais tarde), então os custos amortizados totais incorridos naquele momento estariam abaixo dos custos reais totais incorridos; para a sequência de operações até aquele mesmo momento, o custo amortizado total não seria um limite superior para o custo real total. Assim, devemos cuidar para que o crédito total na estrutura de dados nunca se torne negativo.

3.1. Incrementar um Contador Binário

Como outra ilustração do método de contabilidade, analisamos a operação INCREMENT em um contador binário que começa em zero. Conforme observamos antes, o tempo de execução dessa operação é proporcional ao número de bits invertidos, que usaremos como nosso custo para esse exemplo. Vamos utilizar uma vez mais uma nota de 1 real para representar cada unidade de custo (a inversão de um bit nesse exemplo).

Vamos cobrar um custo amortizado de 2 reais para atribuir um bit com 1 (ligar). Quando um bit é ligado, usamos 1 real (dos 2 reais cobrados) para pagar a própria configuração do bit e colocamos o outro real no bit como crédito para ser usado mais tarde quando convertermos o bit de volta para 0. Em qualquer instante, todo 1 no contador tem um real de crédito e, assim, não precisamos cobrar nada para desligar um bit; apenas pagamos o desligamento com a nota de real no bit.

Agora podemos determinar o custo amortizado de INCREMENT. O custo de desligar os bits dentro do laço `while` é pago pelos reais nos bits que são desligados. O procedimento INCREMENT liga no máximo um bit, na linha 6 e, portanto, o custo amortizado de uma operação INCREMENT é no máximo 2 reais. A quantidade de 1s no contador nunca se torna negativa e, portanto, a quantia de crédito permanece não negativa o tempo todo. Assim, para n operações INCREMENT, o custo amortizado total é $O(n)$, o que limita o custo real total. [Thomas H. Cormen 1989]

4. O Método Potencial

O método de análise amortizada é similar ao método de contabilidade no sentido de que determinamos o custo amortizado de cada operação. No entanto, em vez de representar o trabalho pago antecipadamente como crédito, ele é representado como "energia potencial"ou "potencial", que é usado para cobrir o custo de operações futuras. A análise funciona da seguinte maneira: executamos n operações, começando com uma estrutura de dados inicial chamada D_0. Para cada $i = 1, 2, ..., n$, seja c_i o custo real da i-ésima operação e seja D_i a estrutura de dados resultante após a aplicação da i-ésima operação à estrutura de dados D_{i-1}. Utilizamos uma função potencial que mapeia cada estrutura de dados D_i para um número real, representado como $\Phi(D_i)$, que é o potencial associado à estrutura de dados D_i. O custo amortizado da i-ésima operação em relação à função potencial é definido por: [Thomas H. Cormen 1989]

$$\hat{c_i} = c_i + \Phi(D_i) - \Phi(D_{i-1})$$

Então, o custo amortizado de cada operação é o seu custo real mais a mudança no potencial devido à operação. Pela equação anterior, o custo amortizado total das n operações é dado pela soma dos custos amortizados individuais: [Thomas H. Cormen 1989]

$$\sum_{i=1}^{n} \hat{c}_i = \sum_{i=1}^{n}(c_i + \Phi(D_i) - \Phi(D_{i-1})) = \sum_{i=1}^{n} c_i + \Phi(D_n) - \Phi(D_0)$$

Intuitivamente, se a diferença de potencial $\Phi(D_i) - \Phi(D_{i-1})$ da i-ésima operação for positiva, isso significa que o custo amortizado inclui um valor adicional para a i-ésima operação, aumentando o potencial da estrutura de dados. Por outro lado, se a diferença de potencial for negativa, o custo amortizado é menor para a i-ésima operação, e a redução no potencial ajuda a cobrir o custo real da operação. [Thomas H. Cormen 1989]

Os custos amortizados definidos pelas equações dependem da escolha da função potencial. Diferentes funções potenciais podem resultar em custos amortizados distintos.

Para mostrar o uso do método potencial, usaremos operações por pilha (push, pop e multipop). A função potencial em uma pilha será o número de objetos na pilha. A estrutura de dados da pilha vazia, onde iniciamos, é representada por $\Phi(D_0) = 0$. O uso da estrutura de dados pilha, serve também para garantirmos que a pilha D_i resultante da i-ésima operação tenha potencial positivo, visto que não podemos ter o número de objetos numa pilha sendo negativo, assim temos $\Phi(D_i) \geq 0$.

Então, calculamos as operações (push, pop e multipop) de pilha. Se a operação atual for a operação push na pilha com s objetos, a diferença de potencial é: [Thomas H. Cormen 1989]

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

Usando a equação que calcula o custo amortizado da i-ésima operação, dessa operação push, obtivemos

$$\hat{c}_i = c_i + 1 = 1 + 1 = 2$$

Se a operação atual for a operação multipop(S, k) na pilha com s objetos, resultando na retirada de $k' = \min(k, s)$ objetos da pilha, a diferença de potencial é

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

Usando a equação que calcula o custo amortizado da i-ésima operação, dessa operação multipop(S, k), obtivemos

$$\hat{c}_i = c_i - k' = k' - k' = 0$$

De modo semelhante, o custo amortizado de uma operação pop comum é 0. O custo amortizado de cada uma dessas operações é $O(1)$, e, por isso, o custo amortizado total de uma sequência de n operações é $O(n)$. Visto que $\Phi(D_i) \geq \Phi(D_0)$, o custo amortizado total de n operações serve como um limite superior para o custo real total. Portanto, o custo no pior caso de n operações é $O(n)$. [Thomas H. Cormen 1989]

Referências

Stefanes, M. A. (2007). Análise amortizada.

Thomas H. Cormen, Charles E. Leiserson, R. L. R. e. C. S. (1989). *Algoritmos: Teoria e Prática*. Elsevier, 11th edition.