

2 Algoritmos Online

2.1 Caracterização

Classificamos como algoritmos online aqueles que operam sobre um conjunto de entradas que não é previamente conhecido (BE98). As entradas chegam uma a uma e a cada nova entrada o algoritmo deve tomar uma decisão sem saber quais serão as próximas. Após uma decisão tomada, esta não pode mais ser alterada. Em contrapartida, classificamos como algoritmos offline aqueles em que temos previamente a informação completa da instância sobre a qual o algoritmo irá operar.

Existem diversos problemas reais que necessitam de algoritmos online para sua solução. Em geral, estes algoritmos não atingem a solução ótima offline, mas podem se aproximar dela. Como exemplos de problemas online podemos citar: paginação de memória virtual, roteamento em redes de comunicação, balanceamento de carga e seleção de links patrocinados, que é objeto de estudo desta dissertação.

2.2 Análise Competitiva

Considere um problema clássico de otimização como o de empacotamento, descrito em (BE98). Temos N ítems com tamanhos x_1, x_2, \dots, x_n , sendo o tamanho de um item i determinado por $0 \leq x_i \leq 1$. Estes ítems devem ser agrupados em pacotes de tamanho máximo 1. Qual o menor número possível de pacotes para incluirmos todos os ítems?

Quando temos a informação de todos os pacotes existentes, podemos encontrar a solução ótima utilizando programação inteira. No problema online, entretanto, os pacotes onde estarão cada item devem ser determinados na chegada de cada novo item. Suponha que para as entradas até o item x_i a solução do algoritmo online é ótima e com a chegada do item x_{i+1} é necessário a abertura de um novo pacote. Entretanto, uma reorganização dos pacotes anteriores permitiria manter a solução atual. Neste caso, a solução do algoritmo online já não seria ótima, visto que não é possível modificar o passado. Se nem

sempre é possível garantirmos uma solução ótima para um algoritmo online, será possível calcularmos previamente a distância que estaremos desta solução? Este é o objetivo da análise competitiva.

Nesta metodologia de análise, o desempenho do algoritmo online é comparado ao resultado ótimo de um algoritmo que tenha o conhecimento total do futuro. Assim, o resultado obtido pelo algoritmo é comparado à solução ótima para a mesma instância. O objetivo desta análise é garantirmos um desempenho mínimo para qualquer instância possível do problema.

Considere, para um problema de minimização, que $ALG(I)$ é a solução encontrada pelo algoritmo online para uma entrada I , enquanto a solução ótima para a mesma entrada é $OPT(I)$. Dizemos que o algoritmo é c -competitivo, sendo $c \geq 1$, se para o conjunto finito de todas as instâncias de entradas I temos que:

$$ALG(I) \leq c \cdot OPT(I) + \alpha$$

Quando a constante aditiva α é zero ou negativa, o algoritmo é dito estritamente c -competitivo e o fator c é chamado de razão de competitividade. Quanto menor esta razão, mais próximo o algoritmo estará da solução ótima. Conforme a sequência de entradas aumenta, a constante α tende a desaparecer.

Uma desvantagem desta metodologia é ser pessimista, problema que também ocorre na análise de pior caso de algoritmos offline. A diferença, no caso dos algoritmos online, é a possibilidade de existir um adversário que, com conhecimento prévio da estratégia do algoritmo, escolha propositalmente as piores entradas em tempo de execução. Em alguns artigos, como (KP94), encontramos críticas a este tipo de análise por favorecer algoritmos que fazem escolhas mais conservadoras, por considerar situações com probabilidades mínimas de ocorrer em casos reais.

2.3

Estratégias de Solução

Uma das estratégias para a solução de problemas online é a utilização de algoritmos aleatórios, que usam o acaso para tornar seu comportamento não previsível. O principal objetivo desta estratégia é evitar adversários maliciosos que, conhecendo o funcionamento do algoritmo, podem forçar uma sequência de ações que levem ao pior caso. Este comportamento faz com que estes algoritmos consigam algumas vezes atingir uma melhor competitividade que um algoritmo determinístico. Existem também casos em que atingem competitividade similar ao determinístico, mas com uma prova matemática mais simples.

Aplicamos este método na implementação de um algoritmo que utiliza predição sobre uma expectativa das consultas que vão chegar.

Uma outra estratégia é a utilização do método primal-dual (GW96) para resolver o problema de forma aproximada. Este método foi utilizado em (BJN07) para conseguir uma aproximação de $1 - 1/e$ na solução do problema da seleção de links. Adaptamos o algoritmo proposto em (BJN07) para o problema discutido nesta dissertação. Este método está descrito com mais detalhes no Capítulo 4.

2.4

Avaliação de Algoritmos Online

O desempenho de um algoritmo online c -competitivo, sobre qualquer instância, estará sempre entre $OPT(I)$ e $c \cdot OPT(I)$. Entretanto, se estará mais para um lado ou outro deste intervalo, dependerá das instâncias sobre as quais será executado. Apenas com a utilização de dados reais ou numa simulação com diversas instâncias possíveis, podemos avaliar o desempenho real ou mais frequente do algoritmo.

Apesar de na literatura encontrarmos cálculos sobre a competitividade dos algoritmos para seleção de links patrocinados, não encontramos avaliações experimentais deste desempenho.

Nesta dissertação avaliamos com dados sintéticos os algoritmos adaptados e propostos para diversos cenários que pretendem simular situações do mundo real.