

Banco de dados, trabalho 2

Por:

Guilherme Costa Pateiro GRR20197152

Pedro Pasqualini de Andrade GRR20190399

No teste de seriabilidade quanto ao conflito, foi criado um grafo utilizando a biblioteca Cgraph, com uma interface de chamadas de função desenvolvida nas aulas de algoritmos e teorias dos grafos.

Com o grafo criado, é adicionado uma aresta com o nome de cada “tread” que está nessa transação, depois são aplicadas as 3 regras do algoritmo para gerar as arestas desse grafo, não há arestas redundantes no grafo, por exemplo se T1 deveria ter 2 arestas para T2 somente 1 é feita.

O algoritmo de detecção se dá fazendo um algoritmo que decompõe o grafo em seus componente fortes, um componente forte, é um grafo que qualquer vértice pode ir para outro vértice em um conjunto de passos, no caso de um grafo direcionado um componente com mais de 1 elemento indica um ciclo. Caso o número de componentes encontrados seja igual ao número de vértices, então cada vértice é um componente forte, e não existe um conjunto de 2 ou mais vértices que geraram um ciclo.

O algoritmo de decomposição implementado foi: O algoritmo de Kosaraju^[2].

Para analisar a equivalência por imagem, foi criado um vetor auxiliar para analisar todos os S', sendo S' o agendamento clonado com as transações serializadas. Cada volta do loop principal foi analisado um S' diferente, com a permutação feita pela função next_permutation, da biblioteca <algorithm>.

Esses agendamentos foram analisados pelas 3 regras descritas no enunciado. A primeira foi trivial, visto que as transações clonadas possuem as mesmas operações. Na segunda regra, procurou-se o uso da função de escrita 'W' e analisando se houve alguma leitura após ela, para confirmar se isso também aconteceu em S'. Já na terceira regra, foi varrido ambos os vetores em busca da última escrita 'W' e analisado se elas coincidiam.

Vale ressaltar que essas regras foram implementadas como foi pedido no enunciado, mas analisando o caso de teste T1:R(x), T2:R(x), T2:W(x), T1:W(x) percebeu-se que ele seria serializado por visão segundo o algoritmo, o que parecia estar errado. Analisando o livro Database System Concepts¹, foi encontrada a seguinte regra na página 867:

1. For each data item Q , if transaction T_i reads the initial value of Q in schedule S , then transaction T_i must, in schedule S' , also read the initial value of Q .

Ou seja: para cada leitura inicial em S, deverá manter-se a mesma leitura inicial em S'. Apesar de essa regra resolver o problema em questão, como existem outras formas de se analisar seriabilidade por visão (como dito no mesmo livro, p. 829), foi escolhido manter a abordagem original que respeita o enunciado.

- There are several different notions of equivalence leading to the concepts of *conflict serializability* and *view serializability*.

¹ KORTH H. F.; SUDARSHAN S.; SILBERSCHATZ A. **Database System Concepts** - 7^a edição, 2020.

^[2] https://www.ime.usp.br/~pf/analise_de_algoritmos/aulas/forte.html