

Problema da Ponte de Faixa Única – Versão 2



Apresentação

O problema da ponte de faixa única é um clássico problema de programação concorrente, também conhecido como *One-Lane Bridge Problem*. É um tipo de ponte muito comum em áreas rurais onde o movimento é baixo e o uso é esporádico. Mas se o movimento cresce, os problemas aparecem...

Uma ponte sobre um rio tem apenas uma faixa para a passagem dos veículos. Assim, a qualquer momento, a ponte pode ser atravessada apenas por um ou mais veículos no mesmo sentido (mas não em sentidos opostos). A lógica é que quando um veículo chegar na ponte, se houver outro veículo atravessando no mesmo sentido, ele pode cruzar, respeitando as regras. Se houver veículo no sentido contrário, aguarda este terminar a travessia para tentar atravessar.

O trabalho consiste em propor um algoritmo concorrente para resolver o problema da ponte de uma faixa sem *deadlock* ou *starvation*. Lembre que nesta ponte não existe nenhum sinal para indicar o sentido da travessia, cada veículo deve “olhar” o lado oposto e tomar a decisão de atravessar ou esperar.

Proposta do Problema

O problema é constituído de quatro etapas em ordem crescente de complexidade.

1) Somente carros

Esse problema é só para praticar. Considere que $n=100$ carros irão cruzar a ponte, 50 em cada sentido. O tempo de chegada de cada carro T_a é aleatório, variando entre 1 segundos e 3 segundos. O tempo de travessia de cada carro pela ponte T_c é de 5 segundos. Considerar um espaço $T_s = 1$ seg entre os veículos no mesmo sentido, significando que cabem 5 carros simultaneamente na ponte. Quando um carro chega ponte, deve verificar se o carro que está atravessando está no mesmo sentido, e assim ele pode atravessar respeitando o intervalo de 1 seg do carro anterior. Se todos os carros em um sentido passarem na ponte, os carros no sentido inverso poderão atravessar.

Dados $n = 100$, $T_a = 1$ a 3 seg, $T_c = 5$ seg, $T_s = 1$ seg.

2) Somente carros sem starvation

O problema 1) pode causar *starvation* quando a fila ficar muito cheia no final da execução, pode ser que uma fila não de oportunidade para o outro sentido. Como não existe um sinalizador na ponte indicando o sentido do fluxo, apenas o motorista deve decidir se vai atravessar a ponte ou esperar.

Crie um mecanismo para equilibrar o uso da ponte, por exemplo, alternando o sentido da ponte após 5 veículos atravessarem. Mas lembre-se, não existe nenhum sinalizador na ponte, apenas a decisão dos motoristas.

Dados $n = 100$, $T_a = 1$ a 3 seg, $T_c = 5$ seg, $T_s = 1$ seg, $P=5$.

3) Carros e caminhões sem starvation

Agora, nosso problema inclui caminhões que são mais lentos e por questão de segurança, devem cruzar a ponte sozinho. Considere agora que $n=100$ carros e $m=10$ caminhões irão cruzar a ponte, 50 carros e 5 caminhões em cada sentido. O tempo de chegada de cada carro ou caminhão T_a é aleatório, variando entre 1 segundos e 3 segundos. O tempo de travessia de cada carro pela ponte T_c é de 5 segundos e o tempo de travessia de cada caminhão T_t é de 10 segundos, considerar um espaço $T_s = 1$ seg entre os veículos no mesmo sentido. Por questão de segurança, um caminhão deverá cruzar a ponte sozinha (espera o carro da frente terminar a travessia). Implementar um mecanismo para evitar *starvation*, invertendo o sentido a cada $P=5$ carros ou $U=1$ caminhão

Considere $n = 100$, $m=10$, $T_a = 1$ a 3 seg, $T_c = 5$ seg, $T_t = 10$ seg, $T_s = 1$ seg, $P=5$ ou $U=1$.

4) Carros e caminhões em duas pontes

No local existe uma ponte de faixa única paralela, muito antiga, que ainda pode ser utilizada por carros, mas não por caminhões. Nesta ponte antiga podem atravessar 2 carros simultaneamente ($Q=2$ e $T_s' = 2$ seg) no tempo $T_c' = 4$ segs. Na fila única de cada sentido, caso a ponte principal esteja ocupada é possível encaminhar carros para a ponte antiga e melhorar o fluxo de veículos. Mas agora teremos 120 carros atravessando.

Considere $n = 120$, $m=10$, $T_a = 1$ a 3 seg, $T_c = 5$ seg, $T_c' = 4$ seg, $T_t = 10$ seg, $T_s = 1$ seg, $T_s' = 2$ seg, $P=5$ ou $U=1$ e $Q=2$.

Implementação

O trabalho prático consiste em escrever um programa em C, C++, Java ou Python para simular a situação

Utilizar as seguintes classes:

- Carros.
- Caminhões.
- Pontes

Ao final da execução deve ser exibidos:

1. Quantidade de veículos (carros e caminhões) que cruzaram cada sentido.
2. Tempo mínimo, máximo e médio de espera dos veículos na fila.
3. Tempo de utilização da ponte (tempo utilizado/tempo total).

Avaliação

A avaliação consiste em um vídeo mostrando a execução do programa e um trabalho escrito.

Apresentação Execução

Os alunos deverão gravar um vídeo mostrando a execução do programa (por exemplo, link do Youtube).

Considera-se a execução correta se o programa cumprir o determinado sem a ocorrência de deadlocks ou starvation.

Trabalho Escrito

Os alunos deverão entregar um relatório escrito com os seguintes tópicos:

- (a) Formulação do problema
- (b) Descrição dos algoritmos usados
- (c) Descrição da Implementação (diagrama de classes)
- (d) Resultado da execução do problema com a estatística do final da execução

Para ajudar na preparação do relatório disponibilizamos um modelo de exemplo.