TOLERÂNCIA A FALHAS

DCE540 - Computação Paralela e Distribuída

Atualizado em: 22 de março de 2022

Iago Carvalho

Departamento de Ciência da Computação

FALHAS

Um sistema distribuído pode falhar

- Falha em um componente
- Falha no processo de comunicação entre componentes

Falhas podem acontecer por diversos motivos

- Quedas de energia
- Ataques externos
- Condições ambientais

Falhas podem tornar o sistema mais lento ou inutilizável

- Gerar enormes perdas materiais
- Causar outros tipos de problemas

CONCEITOS BÁSICOS

Devemos estabelecer alguns conceitos básicos para melhor entendermos o que é a tolerância a erros

Os quatro principais conceitos são

1. Disponibilidade
2. Confiabilidade
3. Segurança
4. Manutenibilidade

DISPONIBILIDADE E CONFIABILIDADE

Disponibilidade: O sistema tem que estar pronto para uso imediato

- A qualquer momento, o sistema tem que estar disponível
- Capaz de funcionar normalmente
- Alta disponibilidade é esperada

Confiabilidade: O sistema tem que funcionar corretamente e continuamente, sem interrupções de serviço

- Grandes janelas de tempo sem interrupções ou falhas
- Um sistema confiável tem poucas falhas ou interrupções
  - Mesmo que estas interrupções sejam grandes

MÉTRICA DE CONFIABILIDADE

Pode-se medir a confiabilidade de um componente levando em consideração duas métricas

1. Tempo médio para falha (T_f)
2. Tempo médio para recuperação (T_r)

$$\text{Confiabilidade} = \frac{T_f}{T_f + T_r}$$

Ainda devemos considerar que o tempo médio para falha pode diminuir com o passar do tempo

- Desgaste físico do componente
- Alterações no sistema distribuído que podem introduzir *bugs*

SEGURANÇA E MANUTENIBILIDADE

Segurança: Quando um sistema falha temporariamente, nenhum evento catastrófico pode ocorrer

- Exemplos de eventos catastróficos
  - Falha do controle de temperatura de uma usina nuclear
  - Sistemas controladores de tráfego aéreo deixam de monitorar certos aviões
  - Sistemas de vigilância deixam de identificar ameaças
- Sistemas seguros são aqueles que não ocorrem falhas catastróficas

Manutenibilidade: No caso de uma falha, o sistema tem que poder ser recuperado facilmente

- Recuperação automática é desejada
- Recuperação rápida
- Relacionado a disponibilidade
  - Sistema com auta manutenibilidade, em geral, tende a ter maior disponibilidade

FALHA, ERRO E FALTA

Uma **falha** é quando um componente não consegue cumprir suas premissas

- Quando o componente não consegue funcionar corretamente
- Ele não consegue prover seus serviços

Um **erro** é a parte do estado do sistema que levou a falha

- Erros na transmissão de dados
- Erros no processo de criptografia
- . . .

Uma **falta** é o que ocasionou o erro.

- Meio de transmissão de dados comprometido
- Atualização parcial do algoritmo de criptografia

TOLERÂNCIA A FALHAS

Um componente (ou sistema) tolerante a falhas é aquele que consegue funcionar corretamente mesmo caso ocorram uma ou mais faltas

O componente é capaz de realizar suas tarefas mesmo com uma ou mais faltas

- Por exemplo, aplicando protocolos de correção de erros
- Acessando dados replicados
- Caminhos alternativos para roteamento de mensagens

PERIODICIDADE DE FALHAS

Transiente: A falha ocorre uma única vez

- Mesmo que a ação que levou a falha seja repetida, ela não ocorre novamente
- Comum devido a eventos físicos

Intermitente: A falha se repete ocasionalmente

- O sistema funciona corretamente em alguns momentos, e não em outros
- Difícil de se detectar e recuperar

Permanente: Falhas cuja única recuperação é a substituição do componente defeituoso

- Um disco rígido queimado
- Bug de software
- Cabo de transmissão de dados rompido

TIPOS DE FALHAS

Tipo de falha	Comportamento do sistema
Queda	Para de funcionar
Omissão	Falha na troca de mensagens
Timing	Respostas são enviadas em atraso
Resposta	Uma resposta incorreta é enviada
Arbitrária	Respostas arbitrárias são enviadas

Omissão pode ser para receber ou enviar mensagems

Resposta também pode ter dois tipos

1. Valor
2. Fluxo de controle

DIFERENÇAS ENTRE SISTEMAS SÍNCRONOS E ASSÍNCRONOS

Assíncrono: Se um componente *P* deixa de receber mensagens de outro componente *Q*, ele não pode assumir que *Q* falhou

- Sua única opção é continuar esperando
- Diminuir o ritmo de envio de mensagens

Síncrono: Se *P* deixa de receber mensagens de *Q*, pode-se efetivamente assumir uma falha no sistema distribuído

- *Q* falhou
- Caminho de *P* para *Q* falhou
  - Faz-se necessário recalcular as rotas de comunicação

GRAVIDADE DE FALHAS

Queda: Componente *P* pode efetivamente detectar que houve uma queda no componente *Q*

Ruidosa: Componente *P* *eventualmente* consegue detectar uma queda no componente *Q*

Silenciosa: Componente *P* não consegue distinguir se houve uma queda em *Q* ou se existe falhas de omissão ou timing

Arbitrária: Componente *P* não consegue, efetivamente, detectar falha em *Q*. Mesmo se a falha for detectada, não é possível detectar o tipo da falha

MASCARAMENTO DE FALHAS UTILIZANDO REDUNDÂNCIA

A melhor maneira de tornar um sistema (ou componente) tolerante a falhas é trabalharmos para tornar a falha transparente

- Mascarar a falha de forma que o usuário não a perceba

Pode-se fazer isto efetivamente utilizando redundância

- Redundância de informação
- Redundância de tempo
- Redundância física

MASCARAMENTO DE FALHAS UTILIZANDO REDUNDÂNCIA

Informação: Informação extra é utilizada para detectar erros nas mensagens

- Códigos de correção de erros
- Código de Hamming ▸ Link

Tempo: Caso uma resposta não chegue após um tempo predefinido, a comunicação é realizada novamente

- Útil em sistemas síncronos

Física: Componentes adicionais são adicionados

- Componente adicional é uma cópia exata de outro
- Alto custo
  - Custo financeiro
  - Custo computacional devido a replicação de dados

DETECÇÃO DE FALHAS UTILIZANDO GRUPOS HOMOGÊNEOS

Componentes são organizados em pequenos grupos

Em sua implementação mais simples, um grupo é composto por diversas replicações de um mesmo componente

- Comunicações são realizadas entre grupos e não para componentes isolados

Pode-se realizar uma comunicação (síncrona) dentro de um grupo

- Uma falha é detectada se um componente deixar de responder

DETECÇÃO DE FALHAS UTILIZANDO GRUPOS HETEROGÊNEOS

Grupos são formados por diferentes componentes

- Grupos podem ser estáticos ou dinâmicos
- Cada grupo possui um coordenador

O coordenador do grupo é responsável por identificar falhas nos componentes de seu grupo

- Comunicação síncrona

Mas o que acontece se o coordenador falhar?

RECUPERAÇÃO DO SISTEMA

Após detectada a falha, um ponto importante é recuperar o funcionamento componente ou o sistema

- Deve-se recuperar o estado correto
- Deve-se recuperar os dados atualizados

Pode-se fazer isso efetivamente de duas maneiras

- Recuperação do componente (*checkpoint*)
- Atualização do componente

RECUPERAÇÃO DO COMPONENTE

São utilizados *checkpoints*

- *Backups* periódicos do estado do sistema

Quando uma falha é detectada, o componente é restaurado para o seu último *backup* conhecido

Alto custo computacional (e talvez financeiro)

- É necessário guardar diversos *backups*
- Necessário espaço de armazenamento
- Muitas trocas de mensagens
- Hardware especializado é desejado

ATUALIZAÇÃO DO COMPONENTE

Quando um componente falha, pode-se tentar atualizar seu estado

- Novo estado considerado correto
- Neste novo estado, o componente pode continuar sua execução normalmente

Este processo é barato, mas difícil

- Não é necessário realizar *backups* ou adquirir novos itens de hardware
- Entretanto, é necessário saber exatamente qual erro ocorreu

Muitas vezes, só é realizada a reinicialização do componente

- Espera-se que o melhor aconteça