

# Relatório de Conclusão da Simulação

## 1. Algoritmos utilizados e resultados esperados

De acordo com a tabela 3.2 do livro de Sistemas Operacionais - Andrews Tanenbaum os algoritmos em ordem de melhor eficiência são:

1. LRU - Excelente algoritmo, porém difícil de ser implementado de maneira exata. Sua política consiste em manter uma lista de todas as páginas, com a página usada mais recentemente na dianteira e a página usada menos recentemente na parte de trás. Quando ocorrer um *PageFault*, será removido a página usada menos recentemente.
2. Aging - É um algoritmo eficiente que aproxima bem o LRU. Em cada ciclo de clock é feito uma operação em um contador de 8 bits. Essa operação consiste em realizar um *shift* para a direita e, se o bit *r* do *PageFrame* for 1, substituir o primeiro bit do contador para 1. Quando ocorrer um *PageFault*, será removido a página com o menor valor no contador.
3. NRU - É uma aproximação muito rudimentar do LRU. Consiste em utilizar os bits *r* e *m* para quando ocorrer o *PageFault*, retirar as páginas na ordem seguinte: 0 e 0; 1 e 0; 0 e 1; 1 e 1, para os bits *r* e *m* respectivamente.
4. Second-Change - Considera-se um algoritmo FIFO bastante melhorado. Quando ocorre um *PageFault* é verificado o bit *r* do primeiro elemento da fila (lista de páginas). Se for 1, muda para 0 e joga para o final da fila. Caso contrário retira tal elemento.
5. FIFO - É o algoritmo menos complexo. Segue a política de uma fila. O *PageFrame* entra no final da fila, e quando acontece o *PageFault*, é retirado o primeiro da fila.

A diferença do LRU (livro) para o nosso LRU é que o primeiro usa uma estrutura auxiliar, enquanto o nosso LRU reaproveita a lista de páginas.

## 2. Resultados obtidos da simulação

Rodamos duas formas da simulação a padrão, com 64 frames, e uma estendida para 256 frames.

2.1 Resultados da simulação da carga de dados I

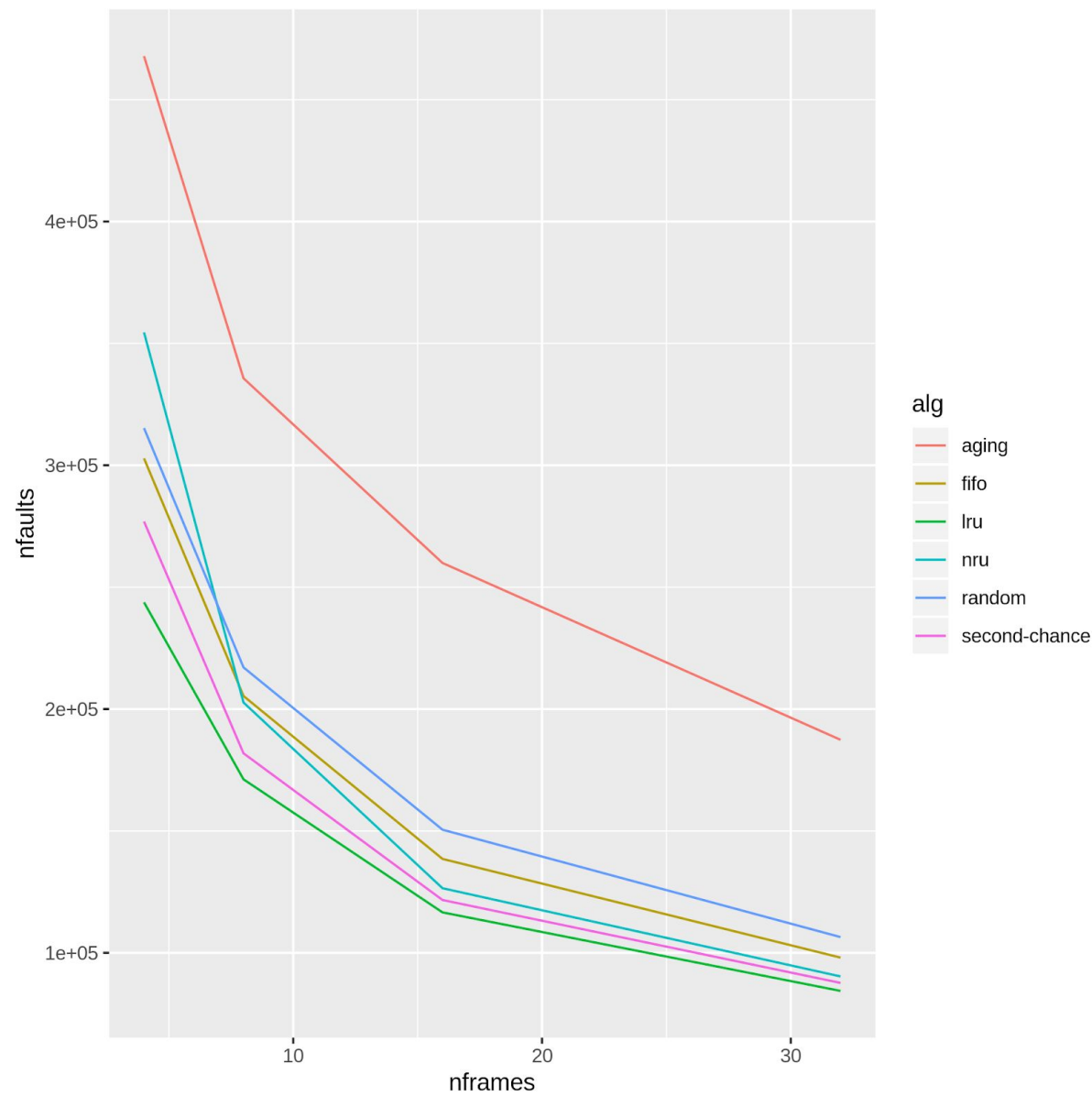
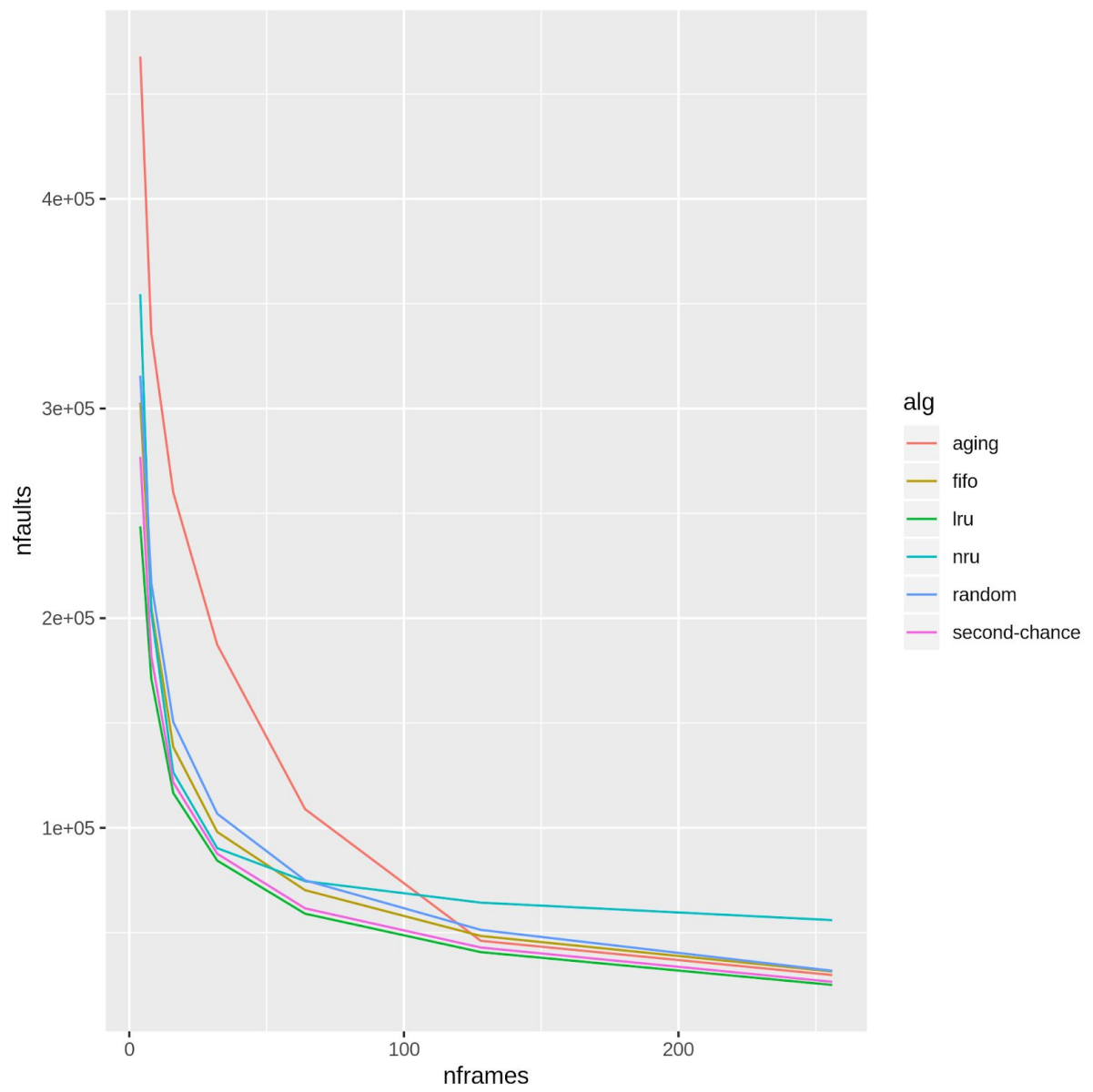


Gráfico 2.1.1 - Trace 1 - 32 Frames



**Gráfico 2.1.2 - Trace 1 - 256 Frames**

2.2 Resultados da simulação da carga de dados II

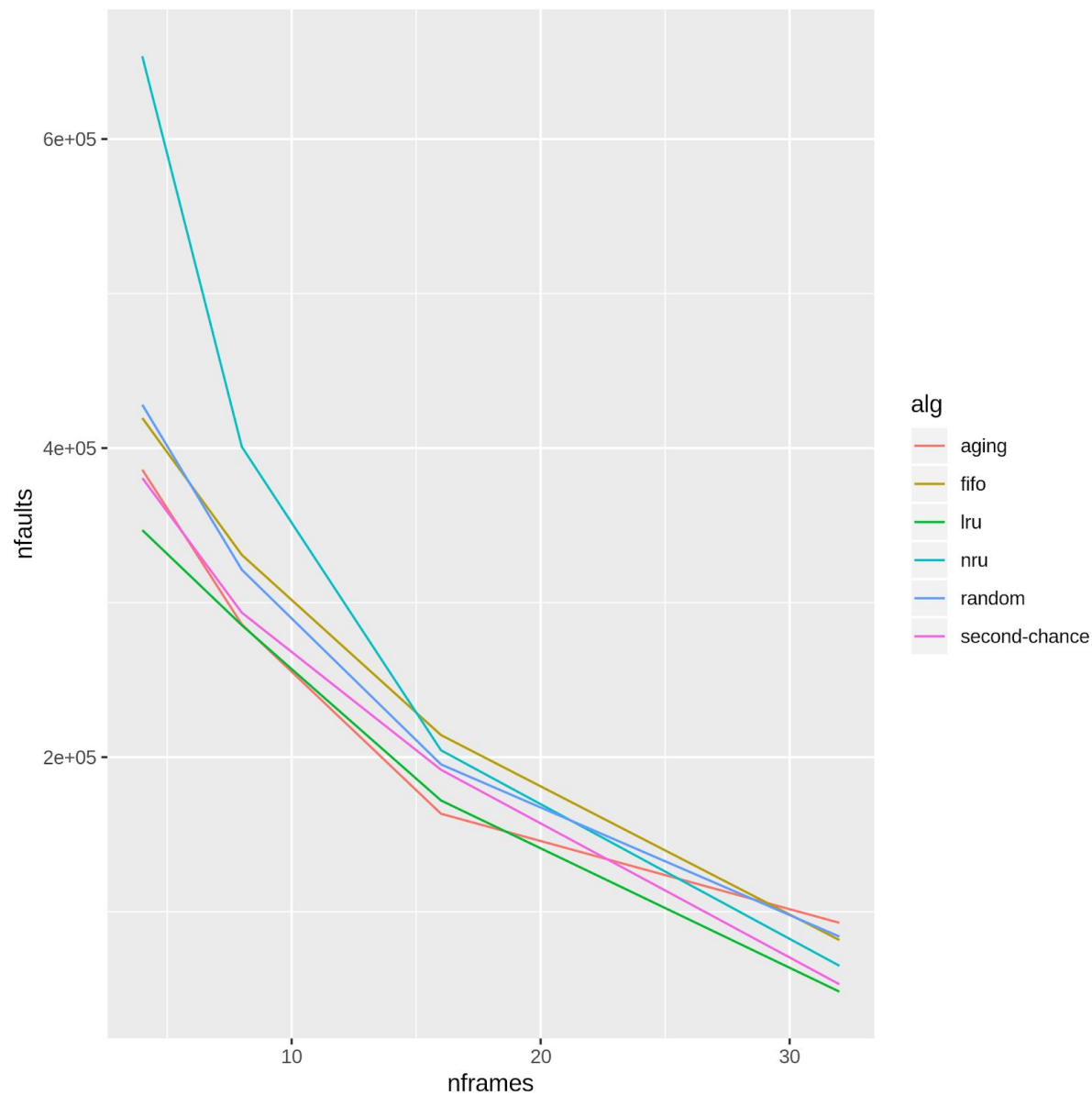


Gráfico 2.2.1 - Trace 2 - 32 Frames

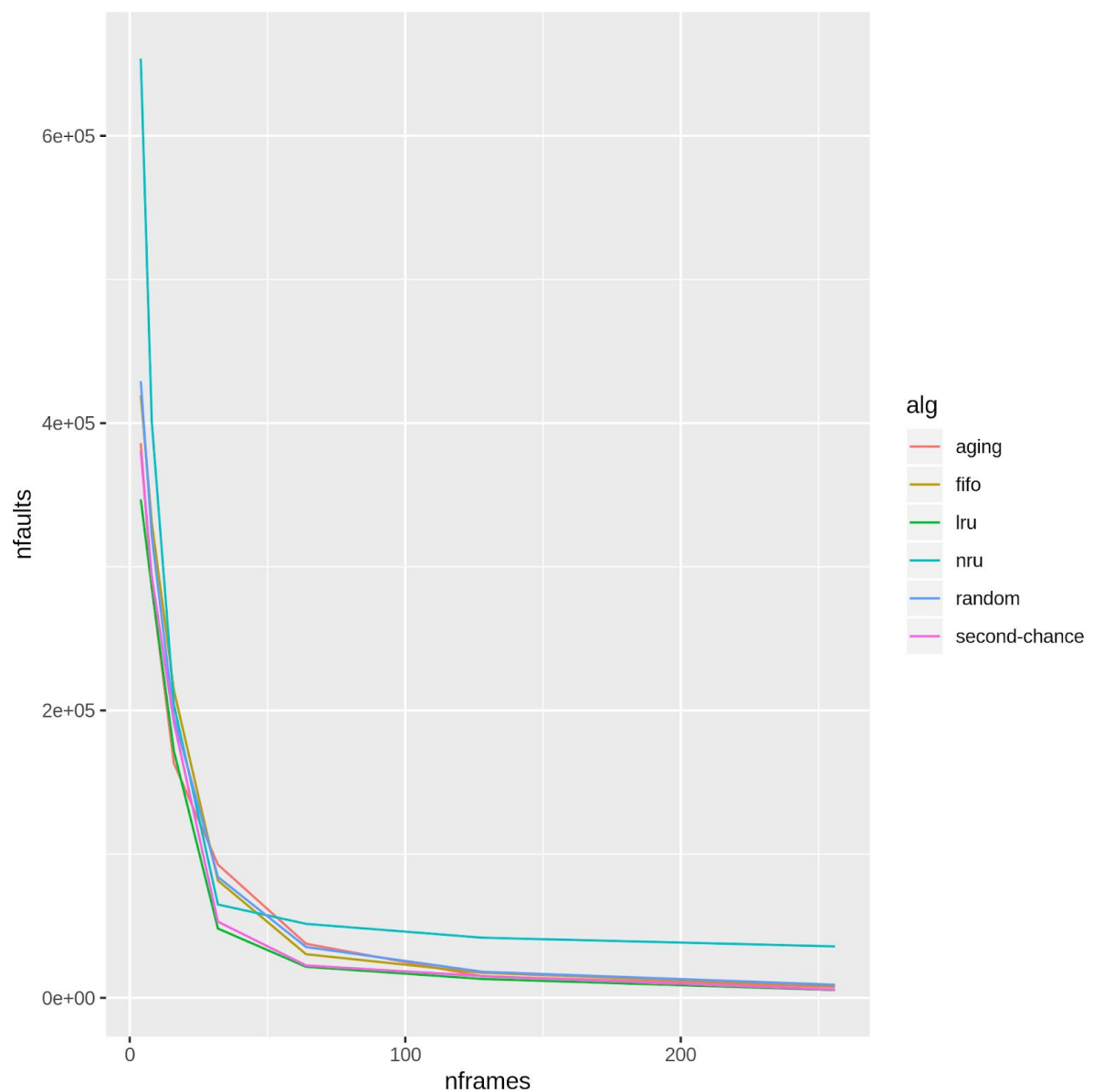


Gráfico 2.2.2 - Trace 2 - 256 Frames

### 3. Divergências do esperado

Apesar das implementações seguirem o livro referência, os algoritmos [Aging](#), [NRU](#) e [Second-Chance](#) divergiram do resultado esperado.

#### 3.1 Aging

Analisando o resultado do gráfico do Trace 1 de 32 frames (Gráfico 2.1.1) o algoritmo Aging deveria performar melhor que os algoritmos NRU, Second-Chance e FIFO mas acabou apresentando maior número de *pagefaults* que eles. Enquanto que no gráfico de simulação da mesma carga, modificando apenas o número de frames, podemos ver

diferenças no gráfico do Aging. Ao aumentarmos o número de frames para 256 (Gráfico 2.1.2) vemos que o Aging se comporta ao longo prazo de forma melhor (com menos *pagefault*) que o FIFO e NRU. Ou seja, concluir-se que tais resultados são influenciados pelos tipos de entradas.

## 3.2 NRU

De forma oposta ao aging, inicialmente, o esperado do NRU foi ruim. O que se notou é que tanto na carga de dados do trace 1 e trace 2, para 32 frames, o NRU se portou como o esperado (ao longo do gráfico) sendo o 3º melhor.

## 3.3 Second-Chance

Este foi o algoritmo que mais surpreendeu, pois seguindo a implementação do livro referência no qual é dito ser melhor apenas que o FIFO, ele acabou ocupando o segundo melhor desempenho absoluto.