



ESTUDO E IMPLEMENTAÇÃO DE PROTOCOLO DE CHECKPOINTING NA CATEGORIA LAZY EM BIBLIOTECA DE TROCA DE MENSAGENS

LOPES, Rafael Ernesto¹ (r.e.lopees@gmail.com); **SACCHI, Rodrigo Porfírio da Silva²** (rodrigoscchi.ufgd@gmail.com)

¹Discente do curso de Sistemas de Informação da UFGD - PIVIC;

²Docente do curso de Sistemas de Informação da UFGD;

Por vários fatores, falhas podem ocorrer durante a execução de uma aplicação de computação de alto desempenho, pois esta pode levar de horas a dias para ser concluída. Se falhas ocorrem, a computação realizada pode ficar comprometida. Para tanto, existem mecanismos para que os sistemas tolerem falhas, como o uso de checkpointing juntamente com a recuperação por retrocesso. Checkpoint consiste em um estado salvo da computação, enquanto checkpointing é a ação. Na computação de alto desempenho, o uso de checkpoints é importante pois, caso ocorra falha, a mesma, usando a recuperação por retrocesso, retorne a um estado consistente. Assim, este trabalho tem como objetivo estudar e implementar os protocolos de checkpointing Lazy-BCS e Lazy BCSAftersend. Iniciamos o trabalho com um estudo de conceitos importantes sobre protocolos de checkpointing, bem como o estudo de problemas que podem comprometer o retrocesso da execução utilizando checkpoints. Durante a fase de revisão, estudou-se os protocolos BCS, Lazy-BCS e Lazy BCSAftersend, que são protocolos que evitam a criação de ciclos-zigzag dentro de um padrão de comunicação e de checkpoints em um sistema. Com exceção do BCS, os outros dois estão inseridos dentro da categoria Lazy de protocolos, que aproveita do fato de que nem sempre é necessário incrementar o relógio lógico do processo na realização de um checkpoint básico se os relógios das mensagens que esse processo recebeu forem todos menores que o seu relógio local, reduzindo o número de checkpoints induzidos por comunicação. A implementação desses protocolos foi feita dentro da biblioceta ChkMPI, uma variante do OpenMPI usando a linguagem C. As funções dos protocolos são invocadas de forma dinâmica com o uso de ponteiros “genéricos” para funções. Que referenciam as funções dos protocolos. Antes de iniciar a execução de uma aplicação, é necessário escolher qual protocolo será utilizado e, com base nessa escolha, os ponteiros para funções referenciam o conjunto de funções do protocolo escolhido. Em uma análise do comportamento, conclui-se que as implementações obedecem às características dos protocolos implementados. Uma sugestão de trabalho futuro é a implementação de uma técnica para, de fato, armazenar os checkpoints em disco.

Palavras-chave: Sistemas Distribuídos, Protocolos de checkpointing, OpenMPI.

Agradecimentos: À Universidade Federal da Grande Dourados (UFGD)