

Uma Proposta para Substituição de Traços em Buffers de Reuso Utilizando Lógica Fuzzy

Rodrigo Moura¹, Giovane Torres¹, Renata Reiser¹, Laércio Pilla², Maurício Pilla¹

¹Programa de Pós-Graduação em Computação – Universidade Federal de Pelotas

²Departamento de Informática e Estatística – Universidade Federal de Santa Catarina

{rcmoura, gdotorres, reiser, pilla}@inf.ufpel.edu.br

Resumo. *Em mecanismos de Reuso de Traços, um fator impactante no seu desempenho diz respeito à política de substituição de dados no buffer de reuso. A escolha de qual dado deve ser substituído deve ser feita levando em consideração diferentes variáveis que influenciam na probabilidade de reuso. O presente trabalho apresenta uma proposta de política para substituição de traços em buffers de reuso utilizando um mecanismo baseado em Lógica Fuzzy.*

Introdução

Reuso de valores são estratégias que buscam reduzir o total de instruções executadas por um processador através da reutilização de instruções previamente executadas. Desta forma, instruções redundantes não precisam ser reexecutadas. Uma instrução é considerada redundante quando ela realiza a mesma operação com os mesmos operando, gerando o mesmo resultado [Sodani and Sohi 1998]. As instruções executadas são armazenadas em um *buffer* próximo ao processador para posterior consulta. Para cada nova instrução, antes de ser executada, é feita uma busca no *buffer*. Caso essa instrução seja encontrada no *buffer*, a execução não será necessária. No Reuso de Traços, a unidade de reuso é composta de sequências de instruções executadas, ao invés de apenas uma instrução. A estratégia de reuso de traços foi bastante avaliada e atinge resultados superiores ao reuso de instruções individuais [Moura 2015].

Considerando que o *buffer* para o mecanismo de reuso será limitado, se faz necessária a adoção de uma política de substituição de dados no *buffer*. A política adotada influencia diretamente o desempenho do mecanismo de reuso, pois o reuso só terá sucesso quando traços redundantes forem encontrados no *buffer*. Dessa forma, é importante que os traços que apresentem potencial de reuso permaneçam disponíveis no *buffer*. O potencial de reuso de um traço depende de diversos fatores. Reutilizar traços mais longos, por exemplo, atinge melhores resultados do que reusar traços curtos, bem como um traço recentemente reusado pode apresentar uma probabilidade maior de ser novamente reusado no futuro. Considerando que o tipo e formato de cada traço armazenado pode influenciar no desempenho do mecanismo de reuso, se faz necessária a adoção de uma política de substituição que considere as diversas características dos dados armazenados. Para lidar com as diferentes variáveis que influenciam no potencial de reuso dos traços, a aplicação de Sistemas Baseados em Regras Fuzzy (SBRF) se mostram uma alternativa para implementar a política de substituição de traços no *buffer* de reuso.

Sistema Baseado em Regras Fuzzy

A Lógica Fuzzy tem diversas aplicações em sistemas que lidam com tomada de decisão quando os valores de entrada não são exatos ou quando o processamento precisa avaliar

diferentes intensidades de valores. Um SBRF é um sistema que possibilita que sejam definidas regras para tratar diferentes combinações de valores de entrada de um sistema, onde estes valores podem possuir diferentes intensidades e suas combinações sejam responsáveis pela tomada de decisão [Jafelice et al. 2005].

Um SBRF é formado basicamente por quatro componentes: (i) o Processador de Entrada que é responsável pela tradução dos valores de entrada em valores Fuzzy (Fuzzificação); (ii) a Base de Regras, conjunto de proposições no formato *if-then* definidas pelo especialista do sistema; (iii) a Máquina de Inferência Fuzzy, responsável pela tradução matemática das proposições utilizando métodos de inferência Fuzzy; (iv) o Processador de Saída, que realiza a interpretação da informação obtida pelo processo de inferência e apresenta o resultado na forma de um número real (Defuzzificação).

Modelo Proposto

O modelo proposto consiste em um SBRF responsável pela tomada de decisão na política de substituição de traços armazenados no *buffer* de reuso do trabalho desenvolvido em [Moura 2015]. Para a classificação dos traços quanto a possibilidade de serem substituídos, serão avaliadas três métricas, denominadas variáveis linguísticas: (i) RR (Recency of Reuse): representa a proximidade no tempo em que um traço foi reutilizado; (ii) FR (Frequency of Reuse): diz respeito a frequência em que o traço foi reutilizado; (iii) TL (Trace Length): quantidade de instruções representadas no traço. Para cada variável linguística serão atribuídas três classificações, denominadas termos linguísticos: RR: Alto, Médio e Baixo; FR: Alta, Média e Baixa; TL: Longo, Médio e Curto. Para as combinações geradas pelas classificações das variáveis linguísticas serão definidas regras no formato *if-then*. No total, serão 27 regras definidas pelo especialista do sistema. O resultado das regras será um valor que representa o grau de pertinência para o traço ser substituído. A pertinência para substituição será classificada em três termos linguísticos: Alta, Média e Baixa. A partir da base de regras, será definido o método de Inferência Fuzzy, responsável por fornecer a saída a partir dos valores das variáveis linguísticas e das relações da base de regras.

Conclusão

O trabalho propõe otimizar o desempenho de um mecanismo de reuso de traços aplicando um SBRF para implementar a política de substituição de traços. Com esse trabalho, espera-se que traços que apresentem maior probabilidade de reuso permaneçam por mais tempo no *buffer*, aumento assim o volume de traços reusados. Como continuidade, deverá ser definido o método de inferência Fuzzy e serão feitas modelagens e simulações matemáticas do SBRF.

Referências

- Jafelice, R. S. M., de Barros, L., and Bassanezi, R. (2005). *Notas em Matemática Aplicada: Teoria dos Conjuntos Fuzzy com Aplicações*. SBMAC.
- Moura, R. C. (2015). Potencial de reuso de traços em arquiteturas arm. Master's thesis, Universidade Federal de Pelotas, Pelotas, RS.
- Sodani, A. and Sohi, G. S. (1998). Understanding the differences between value prediction and instruction reuse. In *MICRO'98*, pages 205–215.