

Wrong Path Events: Exploiting Unusual and Illegal Program Behavior for Early Misprediction Detection and Recovery

D. N. Armstrong, H. Kim, O. Mutlu, and Y. N. Patt. Wrong Path Events: Exploiting Unusual and Illegal Program Behavior for Early Misprediction Detection and Recovery; pp. 119 - 128; The 37th Annual International Symposium on Microarchitecture, 2004

Rafael Augusto Scaraficci - 009649

Este artigo propõe uma solução em hardware para se tentar resolver *mispredicted branches*¹ antes que eles sejam resolvidos pelo processador, diminuindo, desta forma, a penalidade paga por se tomar (especular) um caminho de execução de maneira incorreta. Trata-se de um mecanismo especulativo que explora a execução de operações ilegais e operações que são executadas mais frequentemente nos *branches* especulados incorretamente. Estas operações são denominadas como *Wrong Path Events* (WPE) e consistem de eventos como acesso a uma posição de memória referenciada por um ponteiro NULL, tentativa de escrita numa página de leitura, leitura ou escrita num endereço desalinhado, exceções aritméticas, etc.

Os autores do artigo verificaram através de um simulador, que se utiliza de um mecanismo confiável para predição de *branches* com 64K entradas e com um pipeline de latência de 30 ciclos para *branches* especulados incorretamente, que nos benchmarks de inteiros do SPEC2000, 1,6% dos *mispredicted branches* produzem um WPE e em média esses WPEs ocorrem 51 ciclos antes dos *branches* que os originaram serem resolvidos pelo processador.

O mecanismo proposto é capaz de detectar um determinado conjunto de WPEs e identificar na maioria dos casos o *branch* responsável por tal evento, permitindo que o processador tome o caminho correto de execução antes que a instrução de *branch* seja executada. A identificação é feita através de um mecanismo de predição baseado em história, que memoriza a relação entre a instrução que gerou o WPE e o *branch* que foi especulado incorretamente. Este mecanismo baseia-se nas seguintes observações:

- Muitas instruções que geram um WPE se repetem durante a execução do programa;
- Se uma instrução *A* gera um WPE devido à especulação incorreta de um *branch B*, a próxima vez que a instrução *A* gerar um novo WPE provavelmente deve-se ao *branch B*;

Toda vez que o mecanismo de predição identifica um WPE ele toma uma das decisões:

- Bloqueia o *fetch* de instruções para economizar energia (mecanismo de predição não conseguiu identificar o *branch* responsável pelo WPE);
- Inicia a recuperação do *branch* que foi especulado e originou o WPE (mecanismo de predição conseguiu identificar o *branch* responsável pelo WPE);

Não há nenhuma garantia que o *branch* identificado tenha sido especulado incorretamente, portanto o processo de recuperação pode ser executado para um *branch* especulado corretamente, degradando o desempenho. No entanto, foi verificado que isso é pouco frequente para o SPEC2000 de inteiros. Em média, o mecanismo erra em 4% das vezes, acerta em 69% das previsões e nas demais vezes não consegue identificar o *branch*. O resultado final foi um ganho de aproximadamente 0,6% no IPC o que não justifica a implementação imediata deste mecanismo num processador real. Para ser aplicável faz-se necessário a descoberta de novos WPEs de forma a aumentar, principalmente o número de *mispredicted branches* que geram um WPE e também seria interessante que esses WPEs ocorressem mais precocemente, justificando a recuperação do caminho correto de execução antes que a instrução de *branch* seja executada pelo processador.

¹Resolver um *mispredicted branch* significa detectar que o processador especulou um caminho de execução de maneira incorreta e recuperar-se desta predição indo para o caminho correto de execução.