

Projeto 1 - Algoritmos Avançados

Relatório Futoshiki

Universidade de São Paulo - Campus São Carlos
Instituto de Ciências Matemáticas e Computação



Integrantes do grupo:

- Fabrício Guedes Faria
- William Quelho Ferreira

Nº USP: 9293522

Nº USP: 9293421

Professor: Gustavo Alves Batista

Curso: Bacharelado em Ciências da Computação

Data de Entrega: 17/10/16

Implementação

O algoritmo de resolução de Futoshiki foi implementado com 4 níveis de otimização, utilizando diretivas de compilação do GCC para selecionar qual será utilizado durante a execução dos casos de teste. Os três primeiros (*OPT_NONE*, *OPT_FORWARD_CHECKING* e *OPT_MVR*) foram especificados na descrição, enquanto o quarto nível (*OPT_SIMPLIFY*) foi implementado como decisão de projeto. O nível de otimização a ser utilizada depende do valor da macro *OPT_LEVEL*. Todos esses valores são definidos em *include/core/futoshiki.h*.

O primeiro não utiliza nenhuma heurística, apenas realiza o backtracking sem podas adicionais.

O segundo corta a checagem do estado atual caso seja detectado que alguma célula do jogo não possui mais valores possíveis, e retorna ao nível anterior da recursão pois já se sabe que o tabuleiro no estado atual não possui solução.

O próximo utiliza, além da anterior, a heurística de MVR. Esta tenta preencher células com menor número de possibilidades antes das demais, para que voltas de múltiplos níveis do backtracking não sejam tão custosos.

O último tenta simplificar o jogo antes do início do backtracking, ao reduzir o número de possibilidades a serem consideradas durante a execução do algoritmo, além de preencher células que possuem um único valor possível.

Resultados Obtidos

Ao executar o algoritmo, espera-se que, em média, quanto mais heurísticas forem utilizadas (níveis maiores de otimização) mais rapidamente o programa encontrará uma solução. Isso pode ser observado, por exemplo, nos casos incluídos em *etc/casos.in*.

Percebe-se que, na execução desses casos, o número de atribuições cai significativamente com cada uma das 3 heurísticas implementadas; o tempo de execução, por outro lado, tende a aumentar na transição de *OPT_NONE* para *OPT_FORWARD_CHECKING* e a se manter constante na transição deste para *OPT_MVR*.

O quarto nível de otimização, porém, apresenta uma redução notável tanto em número de atribuições quanto em tempo de execução.

Essa entrada foi criada especificamente para demonstrar esse efeito, mas pode-se observar o mesmo comportamento em outros casos de teste (por exemplo, #31 do arquivo *futoshiki.dat*).

O último nível é especialmente eficaz em simplificar casos com longas cadeias de desigualdades por restringir mais o vetor de possibilidades das células envolvidas (vide código e comentários para mais detalhes).