

Projeto PipeMania

Grupo tp033

Francisco Ferro Pereira, nº107502
Tiago Romão, nº104182

Estratégia utilizada - Visão Geral

- Pré-processamento nas bordas.
- Propagação de restrições pelo tabuleiro (iterando em espirais).
- Chamar algoritmo de procura para gerar os diferentes estados para ações incertas.
- `goal_test()` verifica se existem clusters usando um BFS que começa num ponto arbitrário do tabuleiro.

Pré-processamento das bordas

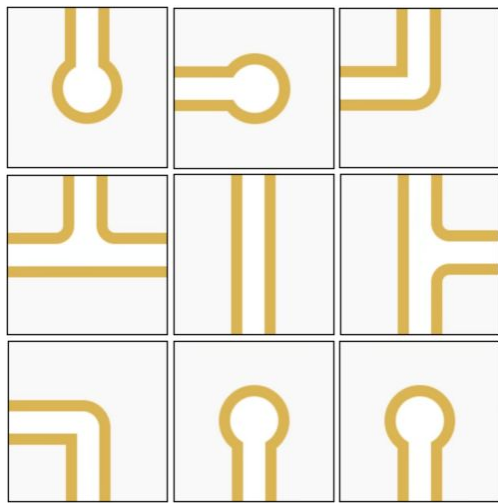


Figura1: Estado inicial

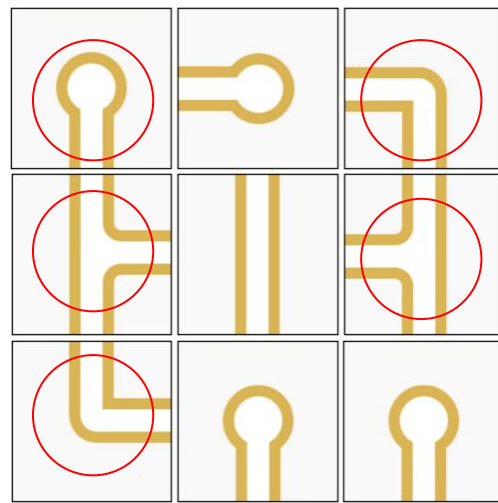
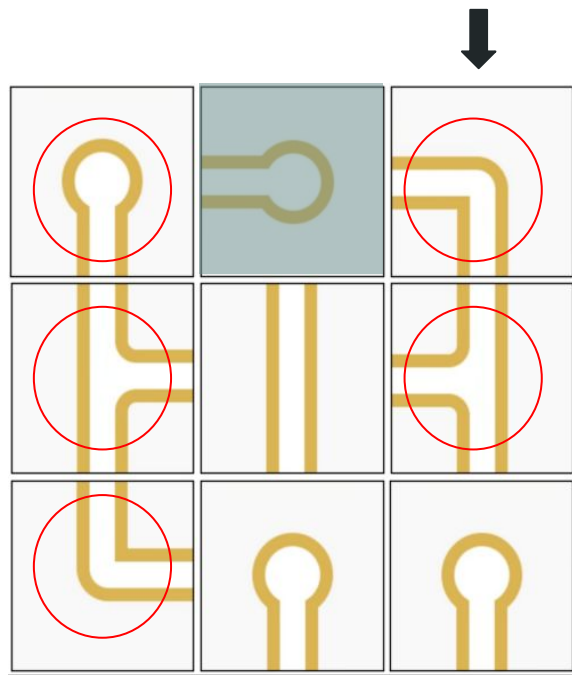
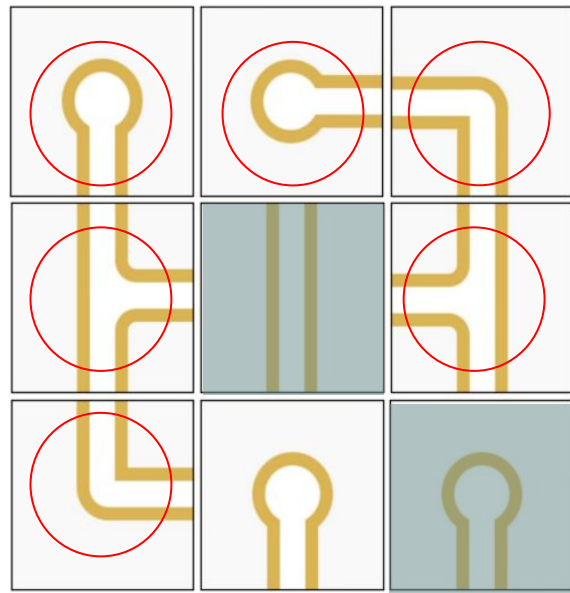


Figura2: Estado seguinte

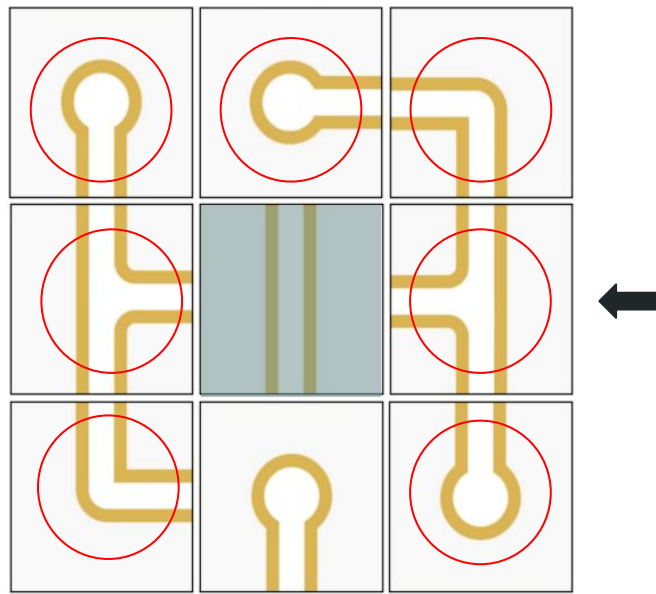
Propagação de restrições



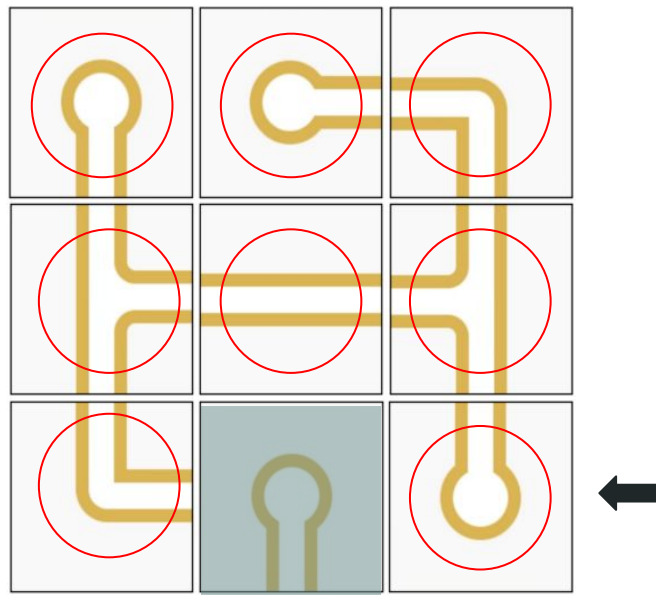
Propagação de restrições



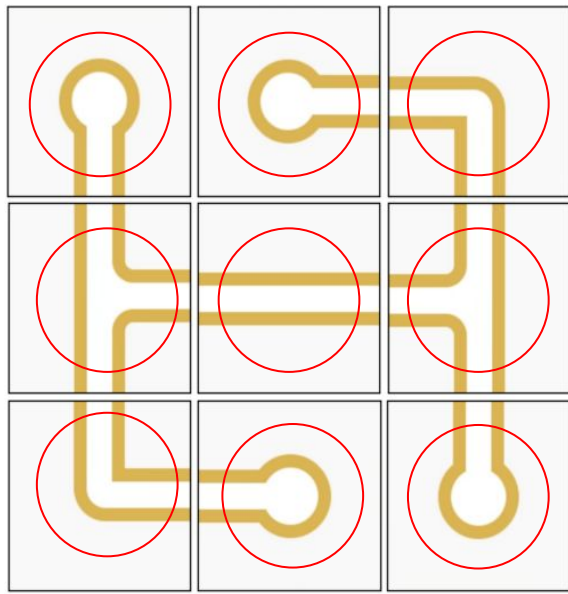
Propagação de restrições



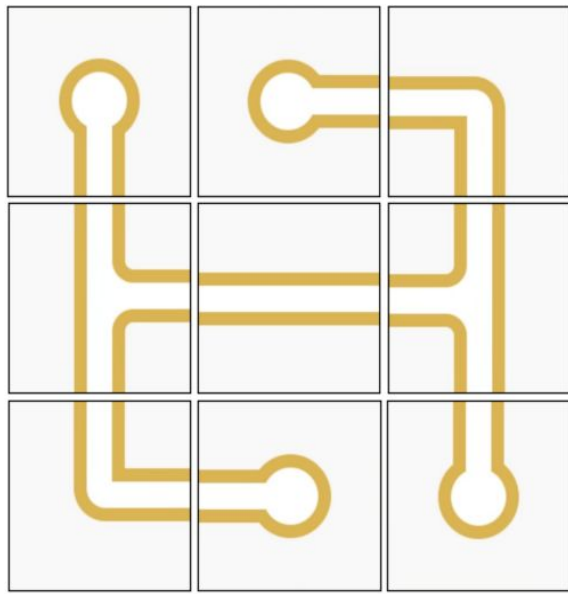
Propagação de restrições



Propagação de restrições



Problema resolvido sem procura!



Procura

- Por vezes, a propagação de restrições não consegue inferir mais nada no tabuleiro.
- Precisamos da procura para gerar os estados das diferentes configurações possíveis do tabuleiro.

FD	LH	VE	VB	LH	FE	FB	VB	FE	FD	LH	BB	LH	LH	FE
FB	VB	VC	BD	LH	FE	BD	VC	VB	VE	VB	BC	FE	VB	FE
BD	BC	LH	VC	VB	FE	LV	FB	LV	LV	BD	LH	LH	BC	FE
LV	FB	FB	VB	VC	FB	LV	LV	LV	LV	BD	LH	FE	VB	FE
BD	BC	BC	BC	LH	BC	VD	VC	BB	BB	VC	FD	VB	BE	FB
LV	VB	FE	FB	VB	LV	VB	FD	BE	BC	FD	VD	BE	VD	VC
BD	VC	FD	BE	LH	FD	FC	FE	LH	BC	LH	FE	BD	LH	VE
VD	LH	VE	VD	BC	LV	VE	VC	LV	BE	LH	LH	BC	FE	LV
FD	LH	BC	LH	BE	BB	VE	FB	LH	BE	BB	VE	VB	LH	VC
VB	FE	FB	VB	BD	VB	FE	LV	LV	FE	LV	FC	LV	VB	FE
VD	BB	BC	BC	BE	LH	BB	BE	BB	VD	LV	FD	BC	BE	FB
FD	BE	FB	VB	BC	BE	BD	BE	VE	FE	FC	FB	FB	VD	BE
FD	BE	VD	BE	FB	LV	FC	LV	LV	FB	FB	BD	BC	VE	LV
VB	BE	FB	BD	VC	VD	FE	LV	FC	VD	BE	VD	VE	LV	LV
FC	FC	VD	BC	LH	LH	FE	VD	LH	LH	BC	FE	FC	VD	VC

Uso de dicionários

```
> (11, 7): {'BB', 'BD', 'BE'}
> (4, 6): {'VC', 'VD'}
> (4, 7): {'VC', 'VD'}
> (4, 8): {'BD', 'BC', 'BE'}
> (4, 9): {'BD', 'BC', 'BE'}
> (4, 10): {'VC', 'VD'}
> (4, 11): {'FB', 'FE'}
> (11, 8): {'VE', 'VB'}
> (5, 3): {'FB', 'FD'}
> (10, 4): {'BB', 'BE'}
> (11, 5): {'BB', 'BE'}
> (11, 9): {'FE', 'FC'}
> (8, 4): {'BB', 'BE', 'BC'}
> (9, 4): {'BB', 'BC', 'BE'}
> (5, 4): {'VE', 'VB'}
> (4, 5): {'BE', 'BC'}
> (6, 3): {'BB', 'BE'}
> (7, 4): {'BB', 'BE', 'BC'}
> (10, 9): {'VC', 'VE'}
> (11, 6): {'BB', 'BE', 'BD'}
> (7, 9): {'BB', 'BC', 'BD'}
> (8, 9): {'BB', 'BC', 'BD'}
> (9, 9): {'FB', 'FE', 'FC'}
> (5, 10): {'FD', 'FE', 'FC'}
> (6, 9): {'BB', 'BD', 'BC'}
```

Sort por número de ações possíveis



```
> (4, 6): {'VD', 'VC'}
> (4, 7): {'VD', 'VC'}
> (4, 10): {'VD', 'VC'}
> (4, 11): {'FE', 'FB'}
> (11, 8): {'VE', 'VB'}
> (5, 3): {'FB', 'FD'}
> (10, 4): {'BE', 'BB'}
> (11, 5): {'BE', 'BB'}
> (11, 9): {'FE', 'FC'}
> (5, 4): {'VE', 'VB'}
> (4, 5): {'BE', 'BC'}
> (6, 3): {'BE', 'BB'}
> (10, 9): {'VE', 'VC'}
> (11, 7): {'BE', 'BD', 'BB'}
> (4, 8): {'BE', 'BD', 'BC'}
> (4, 9): {'BE', 'BD', 'BC'}
> (8, 4): {'BE', 'BB', 'BC'}
> (9, 4): {'BE', 'BB', 'BC'}
> (7, 4): {'BE', 'BB', 'BC'}
> (11, 6): {'BE', 'BD', 'BB'}
> (7, 9): {'BD', 'BB', 'BC'}
> (8, 9): {'BD', 'BB', 'BC'}
> (9, 9): {'FE', 'FB', 'FC'}
> (5, 10): {'FE', 'FD', 'FC'}
> (6, 9): {'BD', 'BB', 'BC'}
```

Heurísticas

1º - Número de pipes desconectados. ➡ Inadmissível

2º - Número de aberturas de pipes desconectadas (+ granular e informativa)

Prós e Contras

Prós:

- Pré-processamento e propagação reduzem número de nós a gerar (menos profundidade na árvore de procura)
- Ordenar o dicionário permite-nos considerar primeiro ações com menos opções.

Contras:

- Ordenar o dicionário a cada nó expandido é custoso computacionalmente.
- A verificação do `goal_test()` a cada nó também pode ser um bottleneck porque tem de percorrer o tabuleiro todo.

Comparação entre algoritmos

