

Exercício 2 - Percolação

Cláudio Santos nº 42208 MIEF

claudiosb7@hotmail.com

Sumário: Geração de configurações de percolação numa rede quadrada N sítios e probabilidade p . Verificação de agregado percolativo e medição do caminho mais curto e tempo de queima. Estatística sobre estas variáveis todas e determinação da probabilidade crítica.

Exercício 2.1

O objectivo deste exercício consiste em gerar configurações de percolação em uma rede quadrada de $N=L_x*L_y$ e probabilidade de ocupação p .

Começou-se por definir um array `rede[Lx*Ly]` que é posto num duplo ciclo: o primeiro ciclo de índice i corre o x entre 0 e L_x-1 ; enquanto o segundo de índice j corre o y entre 0 e L_y-1 . Dentro deste último é gerado um número aleatório w com a função `drand48()` e se $w > 1-p$, é guardado o valor 1 na posição $i+j*L_x$. Se $w < 1-p$, é guardado o valor 0 invés. Assim é preenchida a configuração quadrada.

No caso concreto foi escolhido $L_x=L_y=50$. Utilizou-se o código `latticeview.h` para visualizar as configurações com a cor branca a corresponder ao valor '0' e a cor verde a corresponder ao valor '1'. Foram feitas configurações para 3 valores de p diferentes: 0.3, 0.5 e 0.7.

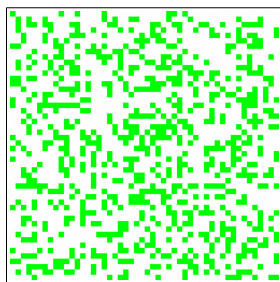


Figura 1: a) $p=0,3$; b) $p=0,5$; c) $p=0,7$;

Exercício 2.2

O objectivo deste exercício consiste em verificar se as configurações geradas em 2.1 têm agregado percolativo ou não e por isso implementa-se o método da queima. Para além dos valores '0' e '1' definidos anteriormente, é utilizado o valor '2' que corresponde a sítios "a arder", '3' para sítios que vão queimar na próxima iteração e '4' para sítios queimados.

Primeiro, inicializa-se pondo a primeira linha da rede a arder, ou seja, $1 \rightarrow 2$ e conta-se quantos sítios '2' apareceram. Depois verifica-se se existem sítios '1' com vizinhos (esquerda, direita, cima e baixo) com o valor '2'. Se sim, esses sítios '1' vão queimar na próxima iteração, ou seja, $1 \rightarrow 3$. Depois de correr a rede toda, corre-se de novo para agora passar os sítios a queimar para queimados, ou seja, $2 \rightarrow 4$ e também $3 \rightarrow 2$. Actualiza-se o número de sítios '2' com esta passagem. Este algoritmo repete-se enquanto houver sítios '2'. O tempo de queima vai corresponder ao número total de vezes que o ciclo repetiu. Enquanto o caminho mais curto calcula-se quando o primeiro estado '4' aparecer na última linha da rede. Se não aparecer estado '4' significa que não existe agregado percolativo.

No caso concreto do código, considerou-se agora $L_x=L_y=10$ e $p=0,7$. Teve-se que ter cuidado quando o sítio '1' estava nas fronteiras da rede (primeiras e últimas linhas e colunas) e tinha vizinhos a arder para não ir queimar sítios que não eram vizinhos. Para estes valores, obteve-se as imagens abaixo. A cor azul vai corresponder aos sítios queimados '4'.

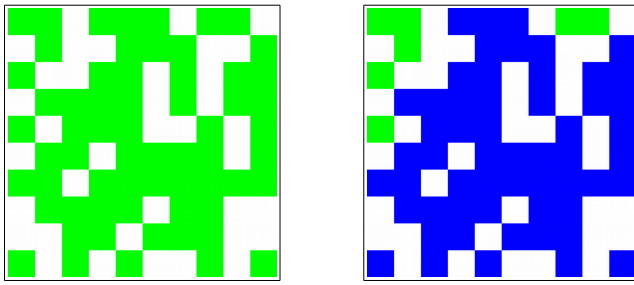


Figura 2: Aplicação do método da queima no exemplo dado. A imagem do lado esquerdo é a rede pré-queima e do lado direito é a imagem pós-queima. Cor azul corresponde a sítios queimados ('4').

Existe agregado percolativo, o caminho mais curto foram 12 iterações e o tempo total de queima foram 15 iterações.

Exercício 2.3

O objectivo é fazer estatística sobre a probabilidade de existir agregado percolativo, do caminho curto e do número total de passos.

Estas variáveis foram calculadas para uma rede em concreto em 2.2. Por isso, agarrou-se nesse código desenvolvido e meteu-se numa função **rede_est** que tem como argumentos a probabilidade de ocupação **p** e os tamanhos de rede **L_x** e **L_y**. Além disso introduziu-se um ciclo adicional de modo a fazer variar a *seed* do `drand48()` e gerar assim 1000 configurações diferentes. Se a fração de agregado for diferente de zero, o valor da probabilidade, o caminho curto e o número total de passos são guardados num ficheiro .txt.

Na função `main()`, fixaram-se as variáveis **p** e **L** e criaram-se ciclos para fazer variar estes argumentos de entrada da função **rede_est**. Obtiveram-se assim vários pontos para **p** e **L** diferentes da fração de existir agregado, do caminho curto e do número total de passos. Fez-se um plot destas medições em função de **p** (**L=10** fixo).

