Cluster algorithm for the Ising model

Define bond index corresponding to pair of interacting spins

$$\text{bond} \quad b = 1, 2, \ldots, N_b, \quad \text{interacting spins } \sigma_{i(b)}, \sigma_{j(b)}$$

Number of bonds N_b = dN for a d-dimensional cubic lattice

Write the energy of the Ising ferromagnet as

$$E = -|J| \sum_{b=1}^{N_b} [\sigma_{i(b)} \sigma_{j(b)} + 1] = -\sum_{b=1}^{N_b} E_b$$

Write the partition function as

$$Z = \sum_{\sigma} \mathrm{e}^{-E(\sigma)/T} = \sum_{\sigma} \prod_{b=1}^{N_b} \mathrm{e}^{E_b/T} = \sum_{\sigma} \prod_{b=1}^{N_b} [1 + (\mathrm{e}^{E_b/T} - 1)]$$

Define bond functions with arguments 0,1 (bond variable):

$$F_b(0) = 1$$
$$F_b(1) = \mathrm{e}^{E_b/T} - 1$$

$$Z = \sum_{\sigma} \prod_{b=1}^{N_b} [F_b(0) + F_b(1)]$$

Introduce bond variables

$$\tau_b = 0, 1, \quad \tau = \{\tau_1, \tau_2, \ldots, \tau_{N_b}\}$$

Partition function can be written as sum over spins and bonds

$$Z = \sum_{\sigma} \prod_{b=1}^{N_b} [F_b(0) + F_b(1)] = \sum_{\sigma} \sum_{\tau} \prod_{b=1}^{N_b} F_b(\tau_b)$$

The functions F_b depend on the spins:

$$F_b(0) = 1$$

$$F_b(1) = \mathrm{e}^{E_b/T} - 1 = \begin{cases} \mathrm{e}^{2|J|/T} - 1, & \text{if } \sigma_{i(b)} = \sigma_{j(b)} \\ 0, & \text{if } \sigma_{i(b)} \neq \sigma_{j(b)} \end{cases}$$

$\tau_b = 1$ allowed only between parallel spins

Probabilities: For everything else fixed, probability for a given b

$$P(\tau_b) = \frac{F(\tau_b)}{F(0) + F(1)} = \frac{F(\tau_b)}{\mathrm{e}^{2|J|/T}}$$

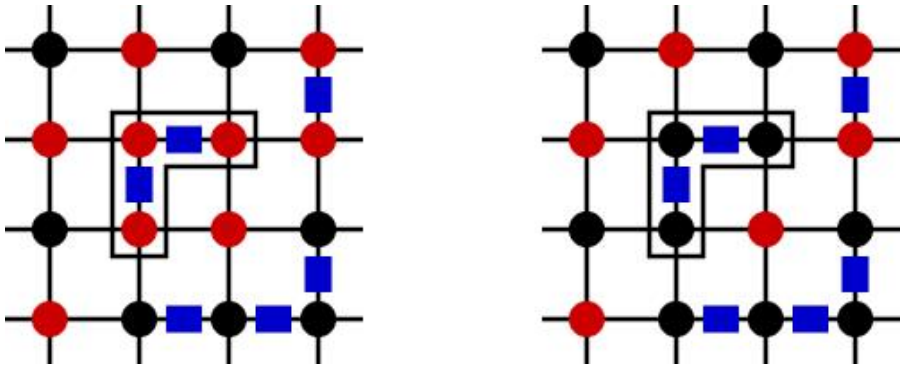
If parallel spins on bond b, probabilities for the bond variable

$$P(\tau_b = 0) = \mathrm{e}^{-2|J|/T}, \quad P(\tau_b = 1) = 1 - \mathrm{e}^{-2|J|/T}$$

If anti-parallel spins on bond b

$$P(\tau_b = 0) = 1, \quad P(\tau_b = 1) = 0$$

For a fixed bond configuration, spins forming clusters (spins connected by "filled" bonds) can be flipped and then give a configuration (term) with the same weight in Z (F_b=1 for all bonds between clusters, F_b unchanged inside cluster).



$$N(\tau_b = 1) = \text{ No. of filled bonds}$$

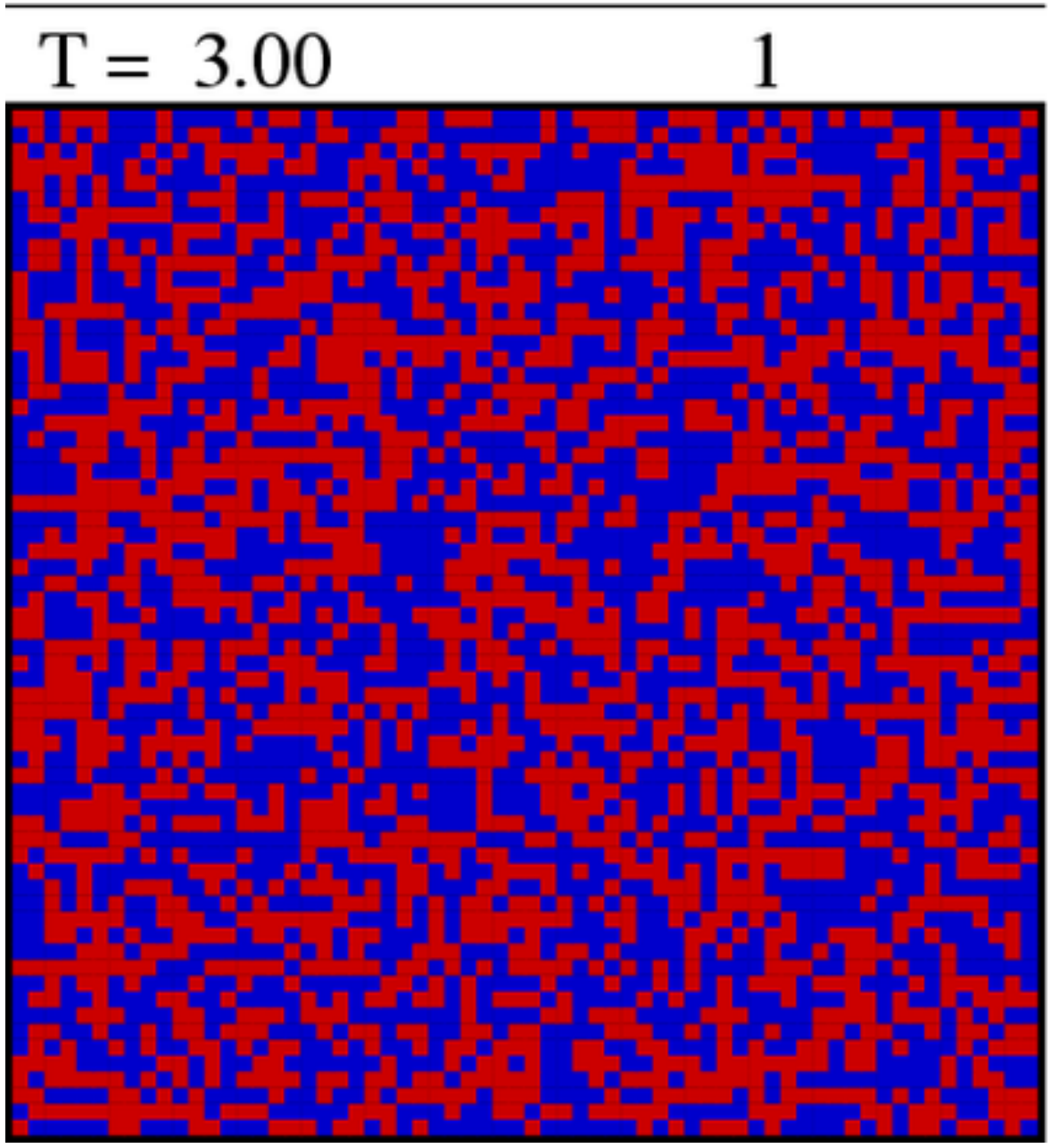
$$W = (e^{2|J|/T} - 1)^{N(\tau_b=1)}$$

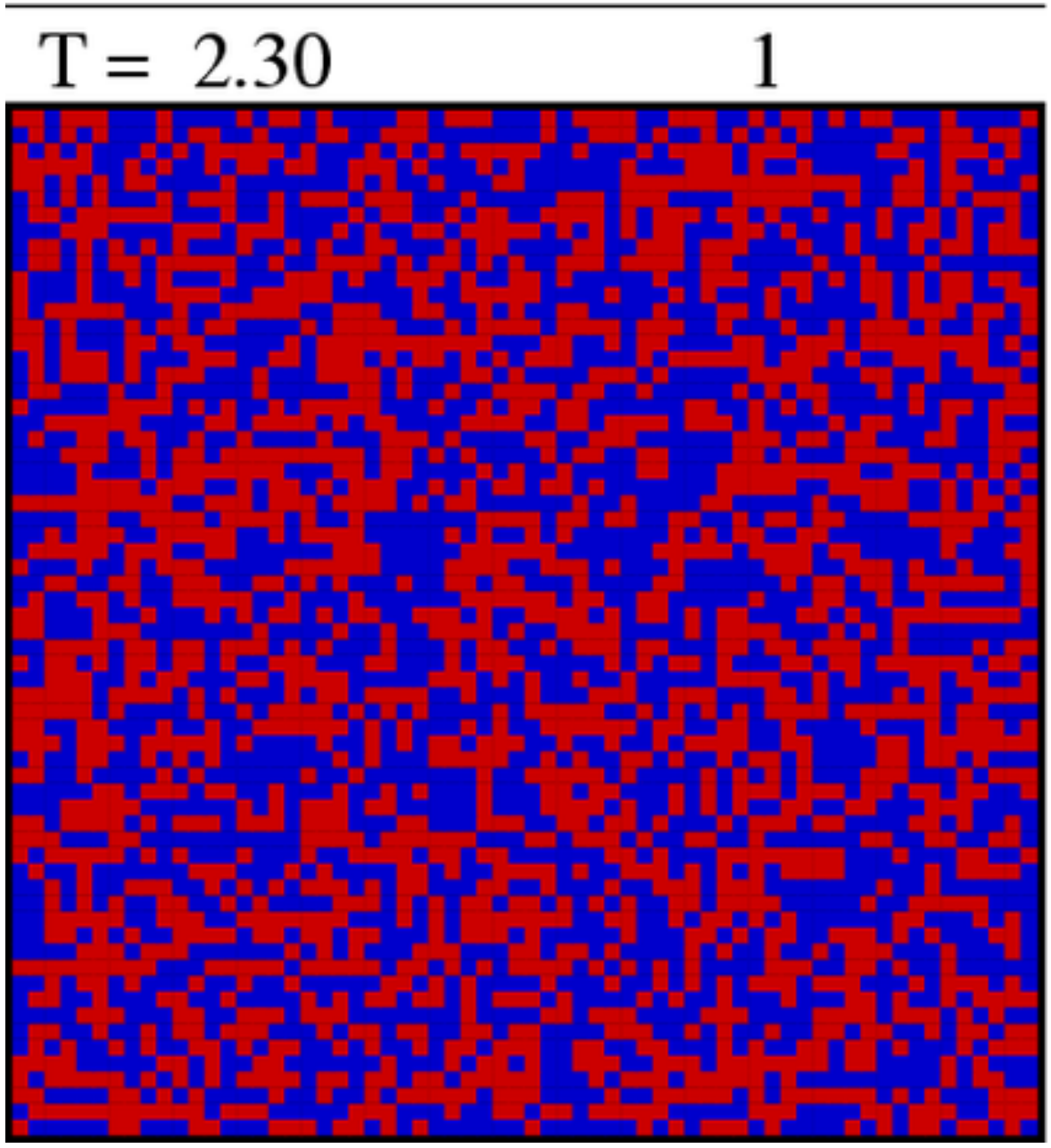
(unchanged after flip)

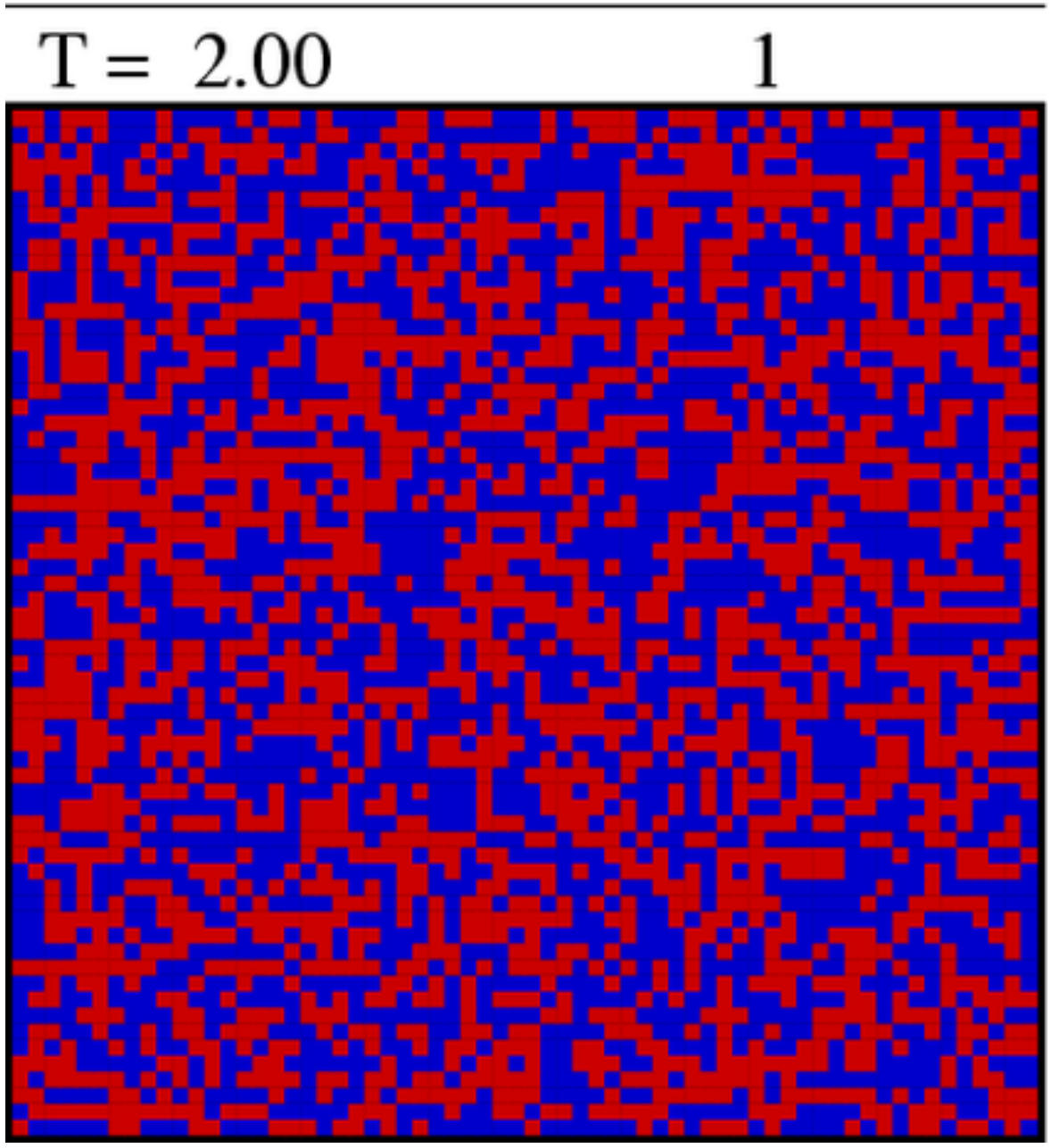
Spins not connected to any filled bonds are single-spin clusters

Swendsen-Wang algorithm

- Start from spin configuration
- Generate bond configuration
- Identify clusters of spins connected by bonds
- Flip each cluster with probability 1/2
- Generate new bonds with the current spins, etc

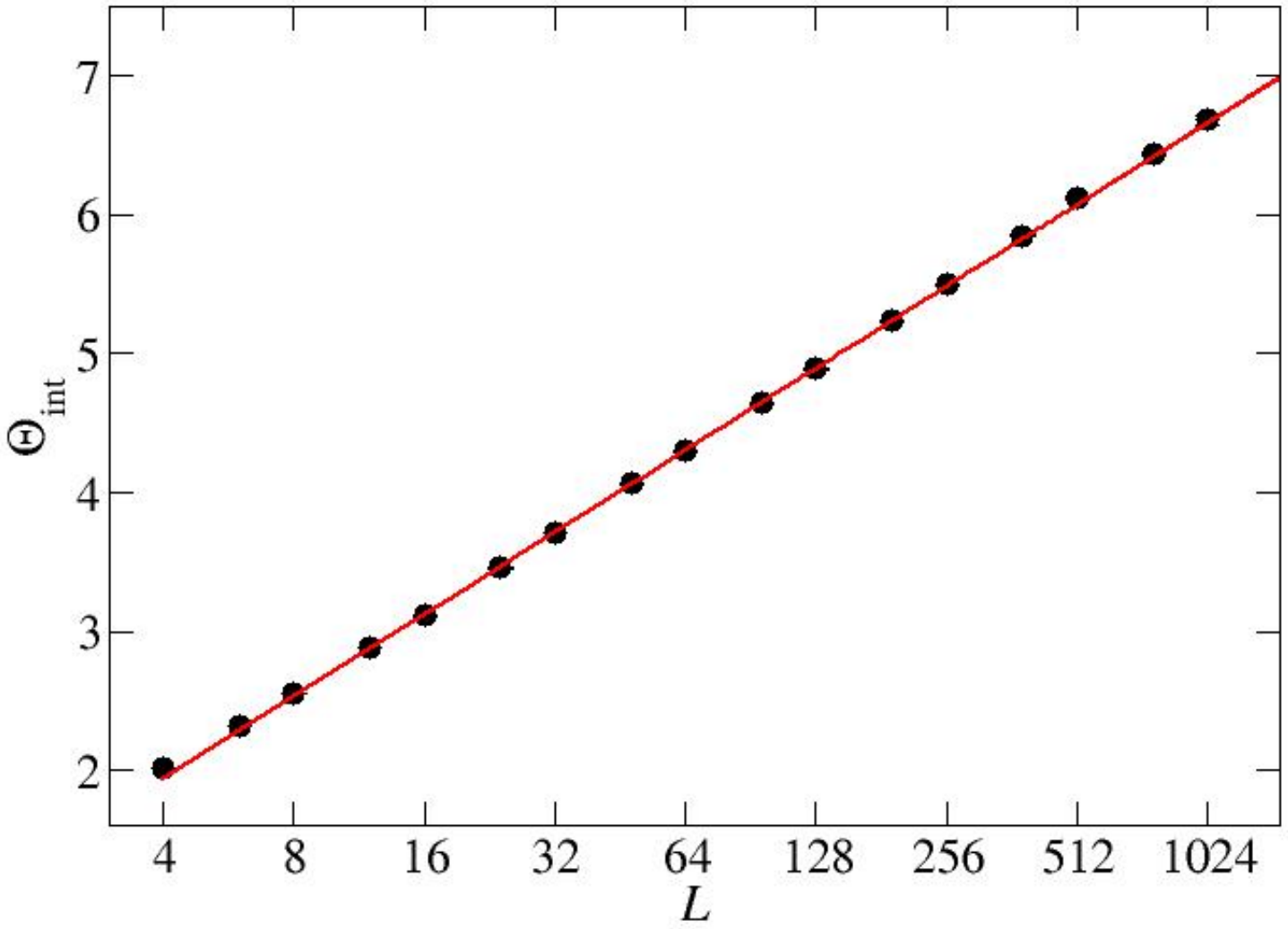

T = 3.00
1


T = 2.30
1


T = 2.00
1

Integrated autocorrelation times

- z=0 for Swendsen-Wang in two dimensions?
- log-divergence of autocorrelation time?
- More likely $z \sim 0.3$ (hard to distinguish L^z and log if z small)



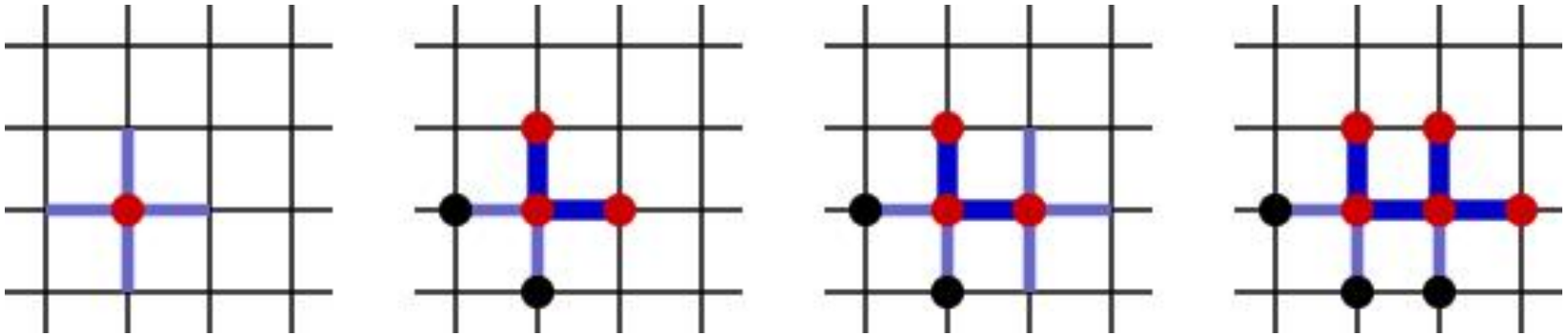
Cluster finding/flipping

Clusters can be constructed and flipped in the same process

➢ Decide whether or not to flip (50% probability) before starting

Store array with flags for spins visited

- Start with spin that has not been visited; seed of cluster
- Add connected (by filled bonds) neighbors to cluster
- Examine the non-visited neighbors of the new spins added
- Add connected neighbors to cluster
- Until no more spins in the cluster with non-visited neighbors



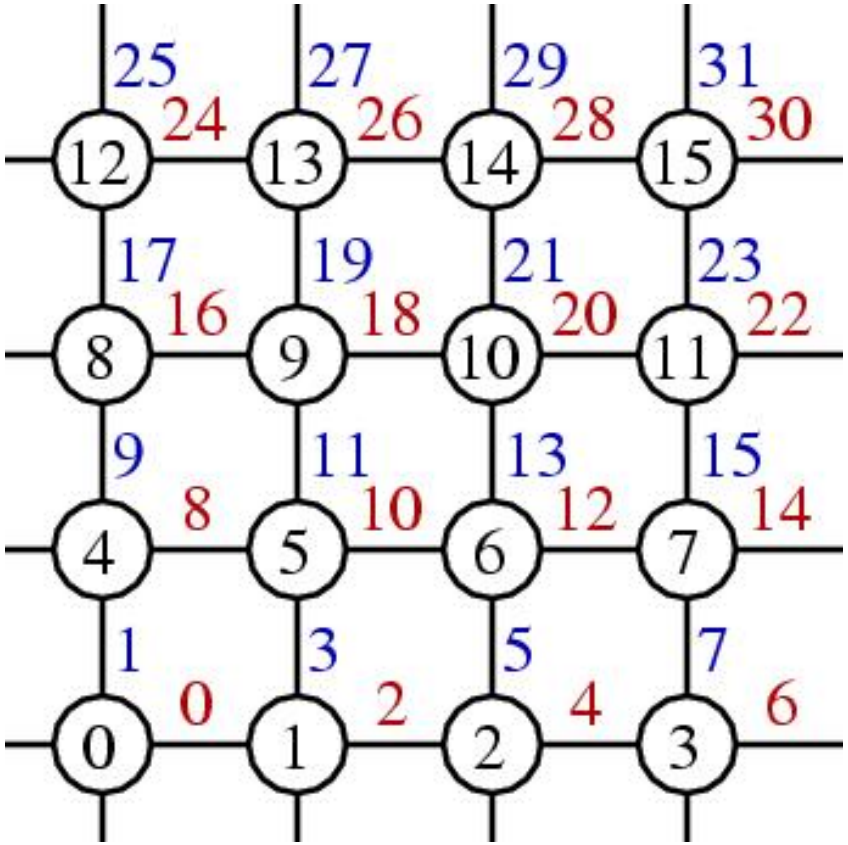
Use stack to store spins with neighbors to be examined

Programming the Swendsen-Wang algorithm

To construct clusters, we need arrays containing

- Neighbors of given site s: `neighbor(i,s)`
- Two spins connected by given bond b: `bondspin(i,b)`
- Bonds connected to given spin s: `spinbond(i,s)`

Labeling of spins and bonds; example in 2D



Storing spin and bond variables in one-dimensional vectors

```
spin(0:n-1), bond(0:d*n-1)
```

Construction of lattice arrays in 2D

```
subroutine lattice

do s0=0,n-1
   x0=mod(s0,lx)
   y0=s0/lx
   x1=mod(x0+1,lx)
   x2=mod(x0-1+lx,lx)
   y1=mod(y0+1,ly)
   y2=mod(y0-1+ly,ly)
   s1=x1+y0*lx
   s2=x0+y1*lx
   s3=x2+y0*lx
   s4=x0+y2*lx
   neighbor(1,s0)=s1
   neighbor(2,s0)=s2
   neighbor(3,s0)=s3
   neighbor(4,s0)=s4
   bondspin(1,2*s0)=s0
   bondspin(2,2*s0)=s1
   bondspin(1,2*s0+1)=s0
   bondspin(2,2*s0+1)=s2
   spinbond(1,s0)=2*s0
   spinbond(2,s0)=2*s0+1
   spinbond(3,s1)=2*s0
   spinbond(4,s2)=2*s0+1
end do
```

Main program

```
bprob=1.d0-exp(-2.d0/temp)
 do i=1,steps/4
    call castbonds
    call flipclusters
 enddo
 do j=1,bins
    call resetbindata
    do i=1,steps
       call castbonds
       call flipclusters
       call measure
    enddo
    call writebindata(n,steps)
 enddo
```

Generating bond configuration

```
subroutine castbonds

do b=0,2*n-1
   if (spin(bondspin(1,b))==spin(bondspin(2,b))) then
      if (ran()<=bprob) then
         bond(b)=.true.
      else
         bond(b)=.false.
      endif
   else
      bond(b)=.false.
   endif
enddo
```

Construct/flip clusters

`notvisited(s) = .true.` for spins not yet visited

`notvisited(s) = .false.` for spin that have been visited

```
subroutine flipclusters

notvisited(:)=.true.
cseed=0

 1 if (ran()<0.5d0) then
    flipclus=.true.
 else
    flipclus=.false.
 endif

 notvisited(cseed)=.false.
 if (flipclus) spin(cseed)=-spin(cseed)
 nstack=1
 stack(1)=cseed
    ......
```

```
do
   if (nstack==0) exit
   s0=stack(nstack)
   nstack=nstack-1
   do i=1,nbors
      s1=neighbor(i,s0)
      if (bond(spinbond(i,s0)).and.notvisited(s1)) then
         notvisited(s1)=.false.
         if (flipclus) spin(s1)=-spin(s1)
         nstack=nstack+1
         stack(nstack)=s1
      endif
   enddo
enddo

do i=cseed+1,n-1                ! find starting spin
   if (notvisited(i)) then      ! for the next cluster
      cseed=i
      goto 1
   endif
enddo
```