# DP and DP2

## Physics of Complex Systems

Michele Avella

February 13, 2022

# The project

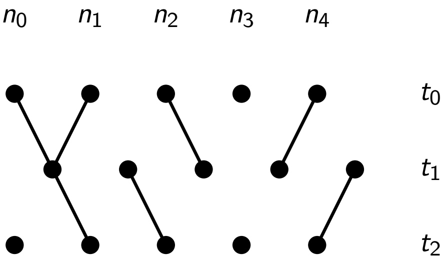
In this project I present two different models that display **non equilibrium phase transition**. I simulate these models and compute their critical exponents to show that they belong to different universality class.

# Non equilibrium phase transition

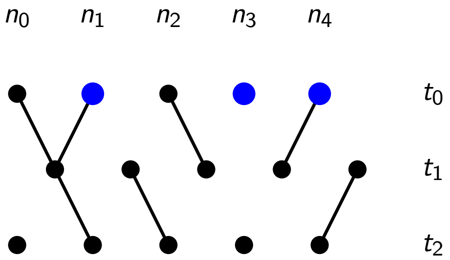
In this presentation I present two model that display a **transition with absorbing state**.
If I use the Markov Chain notation, the absorbing state $S$ has this proprieties:

$$W_{S \to S'} = 0; \quad W_{S \to S} = 1.$$

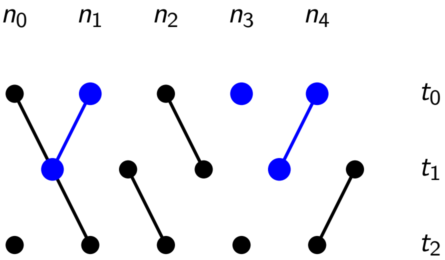
# Directed Bond Percolation d+1



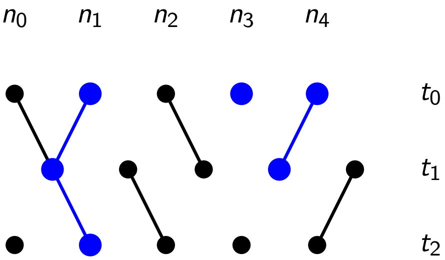
# Directed Bond Percolation d+1



# Directed Percolation (Bond)



# Directed Bond Percolation d+1



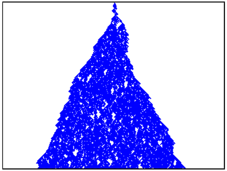
# Stochastic rules

From now on I will call $A$ an active site and $I$ an inactive site; $p$ is the probability to have a bound.

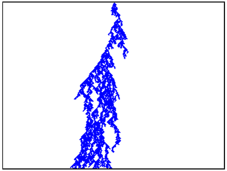
$$P(A|A,A) = 1 - (1-p)^2 \tag{1}$$
$$P(I|A,A) = (1-p)^2 \tag{2}$$
$$P(A|A,I) = P(A|I,A) = p \tag{3}$$
$$P(I|A,I) = P(I|I,A) = 1 - p \tag{4}$$
$$P(I|I,I) = 1 \tag{5}$$
$$P(A|I,I) = 0 \tag{6}$$

# DP: phase transition

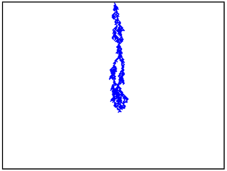
The absorbing state $S$ is the state with all sites inactive. The only parameter of our model is $p$. If it is larger than a critical value $p_c$ I have an **active state**, if it is smaller I have and **inactive state**.



(a) $p > p_c$



(b) $p = p_c$



(c) $p < p_c$

# DP: critical exponents

$$\langle N(t)\rangle \sim t^{\theta} \tag{7}$$

$$\rho(t)=\left\langle \frac{N(t)}{L}\right\rangle; \quad P(t)=\left\langle 1-\prod_i (1-s_i(t))\right\rangle \tag{8}$$

$$\rho(\infty) \sim (p-p_c)^{\beta}; \quad P(\infty) \sim (p-p_c)^{\beta'} \tag{9}$$

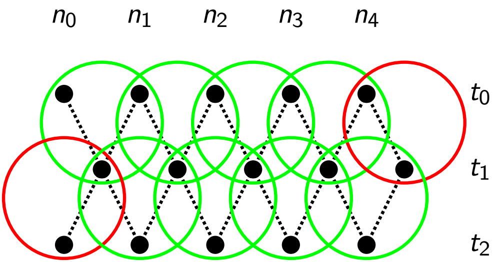
$$c_{ij}=\left\langle \lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t}(s_i(\tau)-\bar{s})(s_j(\tau)-\bar{s})\right\rangle \sim e^{-|i-j|/\xi_\perp} \tag{10}$$

$$c(t)=\left\langle \lim_{L\to\infty} \frac{1}{L}\sum_{i=1}^{L}(s_i(0)-\bar{s})(s_i(t)-\bar{s})\right\rangle \sim e^{-t/\xi_\parallel} \tag{11}$$

$$\xi_\perp \sim |p-p_c|^{-\nu_\perp} \tag{12}$$

$$\xi_\parallel \sim |p-p_c|^{-\nu_\parallel} \tag{13}$$

# Simulation in Python




For the boundary nodes:

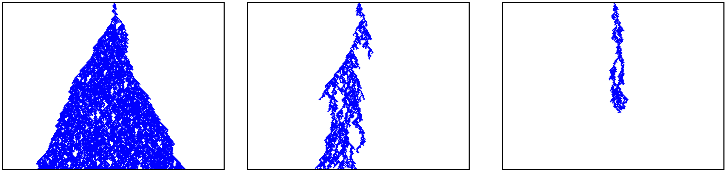
$$P(I|I) = 1; \quad P(I|A) = 1 - p; \quad P(A|A) = p$$

# Simulation in Python: code

```python
@njit
def simula_DP(p,L0,T):
    L = np.copy(L0)
    N = len(L0)
    H = np.zeros((T+1,N))
    H[0] = np.copy(L)
    d = 1
    cont = 1
    for t in range(T):
        L2 = np.zeros(len(L))
        if d == 1:
            # MOVE TO THE RIGHT
            d = 2
            x = L[:-1]
            y = L[1:]
            r = np.random.random(len(x))
            L2[:-1] = logic2(x,y,r,p)
            if L[-1] == 1:
                if np.random.random() < p:
                    L2[-1] = 1
            else: L2[-1] = L[-1]
            L = np.copy(L2)
        else:
            # MOVE TO THE LEFT
            d = 1
            x = L[:-1]
            y = L[1:]
            r = np.random.random(len(x))
            L2[1:] = logic2(x,y,r,p)
            if L[1] == 1:
                if np.random.random() < p:
                    L2[1] = 1
            else: L2[1] = L[1]
            L = np.copy(L2)
        H[cont] = np.copy(L)
        cont += 1
    return H
```

```python
@vectorize(nopython=True)
def logic2(a,b,r1,p):
    if a == 1:
        if b == 1:
            # CASE A A
            if r1 < (1-(1-p)**2):  return 1
            else:                  return 0
        else:
            # CASE A I
            if r1 < p:             return 1
            else:                  return 0
    else:
        if b == 1:
            # CASE A I
            if r1 < p:             return 1
            else:                  return 0
        else:
            #CASE I I
                    return 0
```

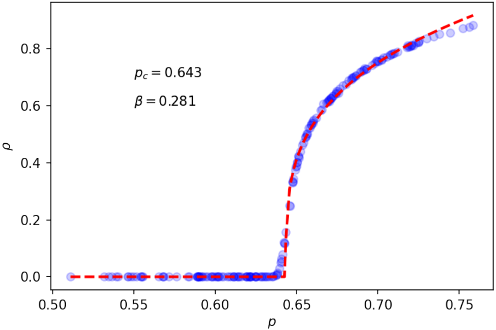
# Simulation Result



(a) $p > p_c$ (b) $p = p_c$ (c) $p < p_c$

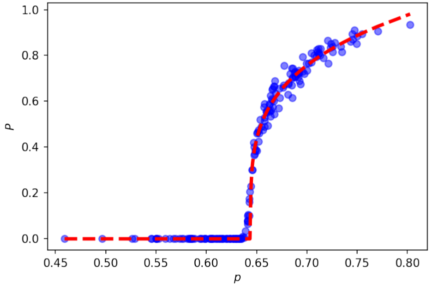
# $\beta$ critical exponent

$$\rho(t) = \left\langle \frac{N(t)}{L} \right\rangle ; \quad \rho(\infty) \sim (p - p_c)^{\beta}$$



# $\beta'$ critical exponent

$$P(t) = \left\langle 1 - \prod_i (1 - s_i(t)) \right\rangle \qquad P(\infty) \sim (p - p_c)^{\beta'}$$

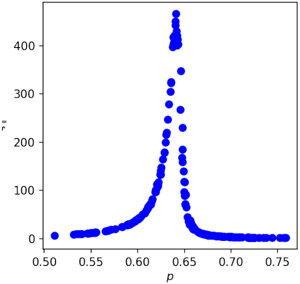


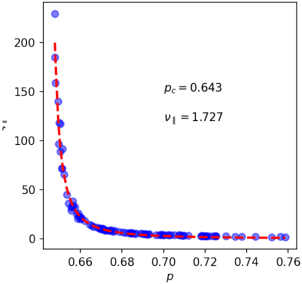

$$\beta' = 0.25$$

# $\nu_\parallel$ critical exponent

$$c(t) = \left\langle \lim_{L\to\infty} \frac{1}{L} \sum_{i=1}^{L} (s_i(0) - \bar{s})(s_i(t) - \bar{s}) \right\rangle \sim e^{-t/\xi_\parallel}$$

$$\xi_\parallel \sim |p - p_c|^{-\nu_\parallel}$$

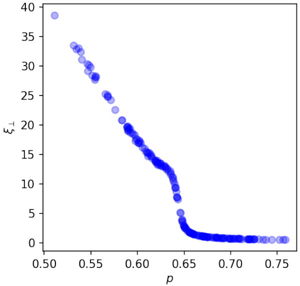


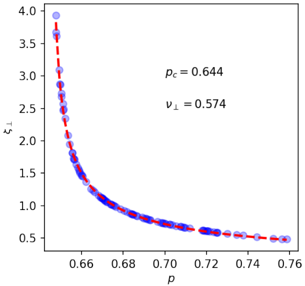



# $\nu_\perp$ critical exponent

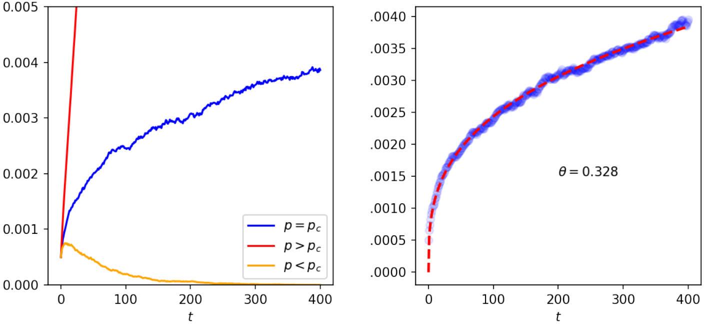
$$c_{ij} = \left\langle \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t} (s_i(\tau) - \bar{s})(s_j(\tau) - \bar{s}) \right\rangle \sim e^{-|i-j|/\xi_\perp}$$

$$\xi_\perp \sim |p - p_c|^{-\nu_\perp}$$



# $\theta$ critical exponent : $\langle N(t) \rangle \sim t^{\theta}$




## Question

Is there any another way to compute the critical exponents?

# Phenomenological scaling theory

## Reference

This part was taken from:
Haye Hinrichsen (2000) Non-equilibrium critical phenomena and phase transitions into absorbing states, Advances in Physics, 49:7, 815-958, DOI: 10.1080/00018730050198152

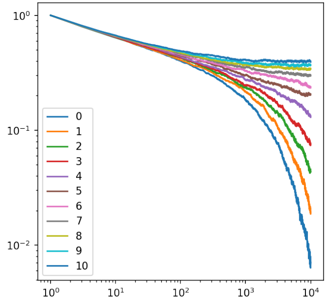
$$\Delta \to \Lambda\Delta,\ x \to \Lambda^{-\nu_\perp} x,\ t \to \Lambda^{-\nu_\parallel} t,\ \rho \to \Lambda^{\beta}\rho \tag{14}$$

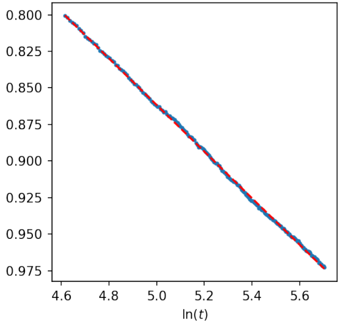
$$\rho(\Lambda^{-\nu_\parallel} t) = \Lambda^{\beta}\rho(t)\ \Rightarrow\ \rho(t) = t^{-\beta/\nu_\parallel}\rho(1) \sim t^{-\delta} \tag{15}$$

$$p(t, \Delta, N) \sim \Delta t^{-\delta} f(\delta t^{1/\nu_\parallel}, V t^{-d/z}) \tag{16}$$

# $p_c$ and $\delta$

From the first method I observed that $p_c$ should be inside the range $[0.64, 0.65]$. So I compute $\rho(t)$ for different values in this range and I chose the one that is more straight in a log-log plot. After that I compute $\delta$.

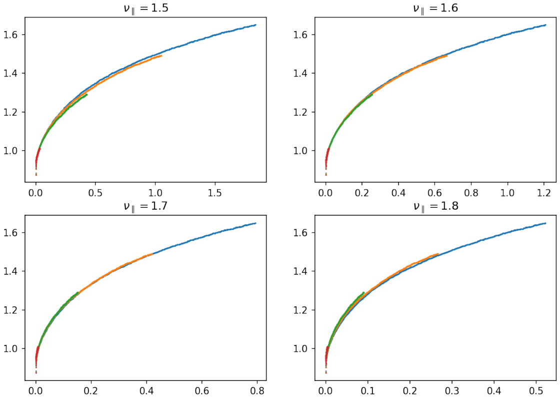



$$p_c = 0.645; \quad \delta = 0.159$$

# $\nu_{\parallel}$ and $\beta$

For different values of $\Delta > 0$ I compute $\rho(t, \Delta)$ and than I plot $\rho t^{\delta}$ as a function of $t\Delta^{\nu_{\parallel}}$. We tune $\nu_{\parallel}$ until all the curves collapse.

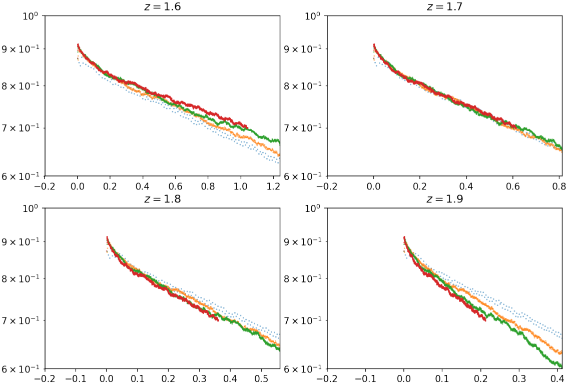



$$\nu_{\parallel} = 1.7; \quad \beta = \nu_{\parallel} \cdot \delta = 0.27$$

# $\nu_\perp$ and $z$

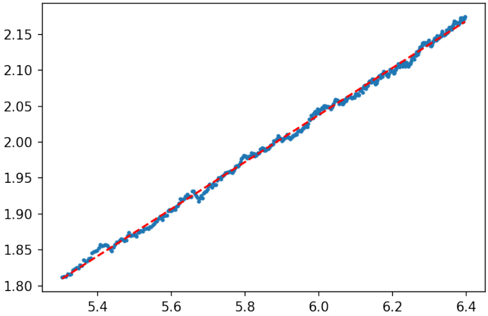
For different values of $N$ I compute $\rho(t, N)$ and than I plot $\rho t^\delta$ as a function of $t/N^z$.

We tune $z$ until all the curves collapse.




$$z = 1.7; \quad \nu_\perp = \nu_\parallel / z = 1.$$

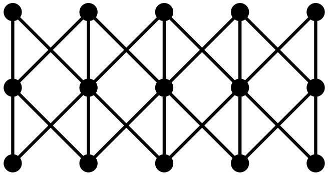
Using $p_c = 0.645$ and starting from a single site active I compute $\langle N(t) \rangle$ that scales like $t^{\theta}$.

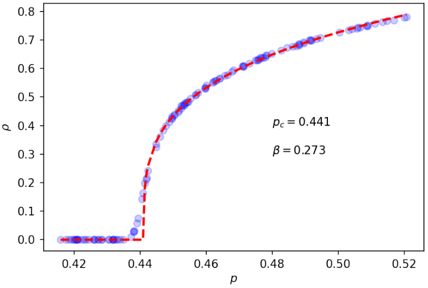



$$\theta = 0.327$$

# Extra

We simulate a DP model with three links. This model is in the DP universality class so it has the same critical exponents.





# DP2

This model has two inactive kind of sites $I_1, I_2$. In this way the system has 2 inactive states and so it belongs to a **different universality class**.

# Stochastic Rules

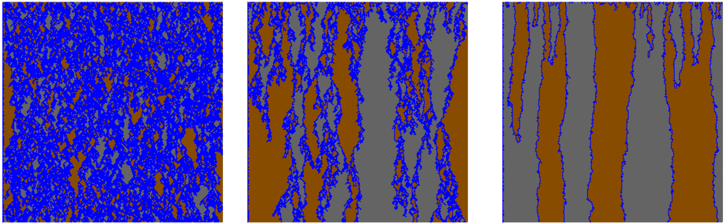
The new update rules:

$$P(A|A,A) = 1-(1-p)^2$$
$$P(I_k|A,A) = (1-p)^2/2$$
$$P(A|A,I_k) = P(A|I_k,A) = p$$
$$P(I_k|A,I_k) = P(I_k|I_k,A) = 1-p$$
$$P(I_k|I_k,I_k) = 1$$
$$P(A|I_k,I_k) = 0$$
$$P(I_k|I_1,I_2) = P(I_k|I_2,I_1) = 0$$
$$P(A|I_1,I_2) = P(A|I_2,I_1) = 1$$

# Simulation in Python: code

```python
@vectorize(nopython=True)
def logic(a,b,r1,r2,p):
    if a == 0:
        if b == 0:
            # CASE A A
            if r1 < (1-(1-p)**2):  return 0
            elif r2 < 0.5:         return 1
            else:                  return 2
        elif b==1:
            # CASE A I1
            if r1 < p:             return 0
            else:                  return 1
        else:
            # CASE A I2
            if r1 < p:             return 0
            else:                  return 2
    elif a == 1:
        if b == 0:
            # CASE A I1
            if r1 < p:             return 0
            else:                  return 1
        elif b == 1:
            #CASE I1 I1
                                   return 1
        else:
            #CASE I1 I2
                                   return 0
    else:
        if b == 0:
            # CASE A I2
            if r1 < p:             return 0
            else:                  return 2
        elif b == 1:
            #CASE I2 I1
                                   return 0
        else:
            #CASE I2 I2
                                   return 2
```

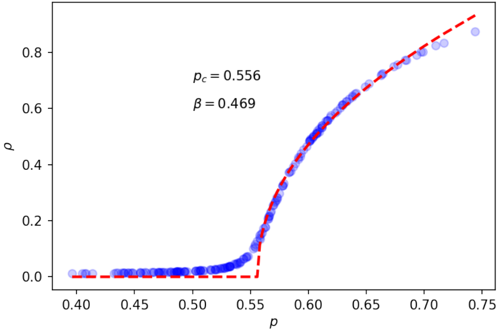
# Simulation results



(a) $p > p_c$ (b) $p = p_c$ (c) $p < p_c$

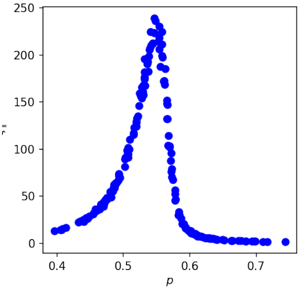
# $\beta$ critical exponent

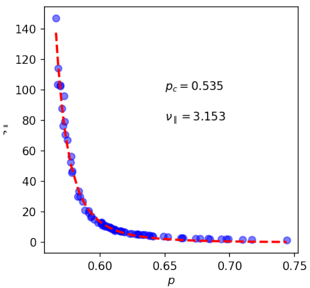
$$\rho(t) = \left\langle \frac{N(t)}{L} \right\rangle ; \quad \rho(\infty) \sim (p - p_c)^\beta$$



$$c(t) = \left\langle \lim_{L\to\infty} \frac{1}{L} \sum_{i=1}^{L} (s_i(0) - \bar{s})(s_i(t) - \bar{s}) \right\rangle \sim e^{-t/\xi_\parallel}$$

$$\xi_\parallel \sim |p - p_c|^{-nu_\parallel}$$

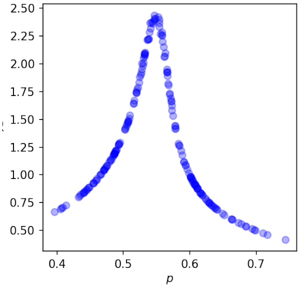


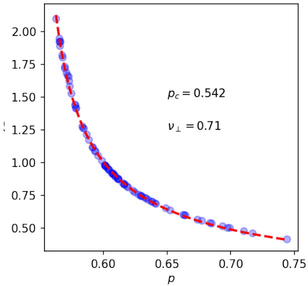



# $\nu_\perp$ critical exponent

$$c_{ij} = \left\langle \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t} (s_i(\tau) - \bar{s})(s_j(\tau) - \bar{s}) \right\rangle \sim e^{-|i-j|/\xi_\perp}$$

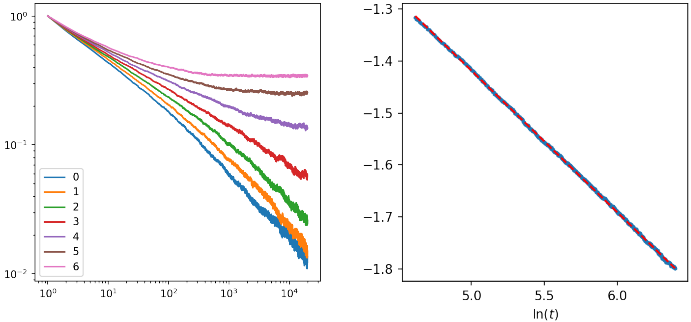
$$\xi_\perp \sim |p - p_c|^{-\nu_\perp}$$



# $p_c$ and $\delta$

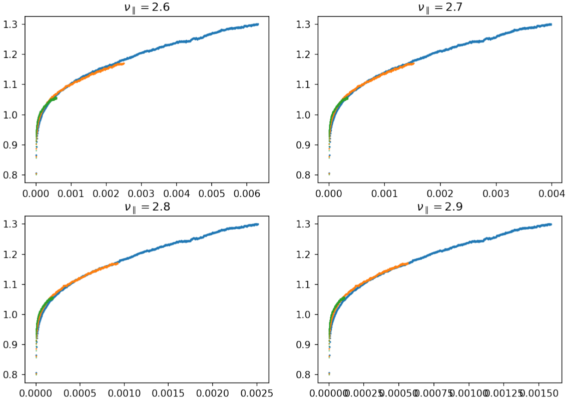
From the first method I observed that $p_c$ should be inside the range $[0.5, 0.6]$. So I compute $\rho(t)$ for different values in this range and I chose the one that is more straight in a log-log plot. After that I compute $\delta$.




$$p_c = 0.55; \quad \delta = 0.27$$

# $\nu_{\|}$ and $\beta$

For different values of $\Delta > 0$ I compute $\rho(t, \Delta)$ and than I plot $\rho t^{\delta}$ as a function of $t\Delta^{\nu_{\|}}$. We tune $\nu_{\|}$ until all the curves collapse.

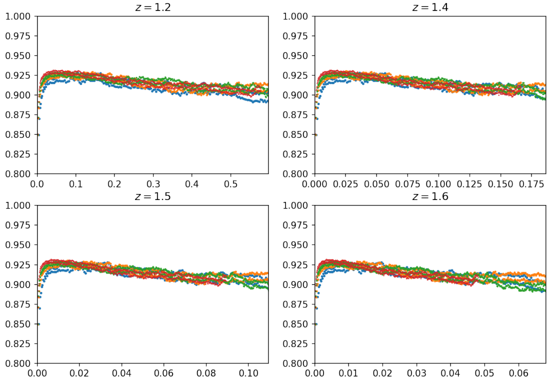



$$\nu_{\|} = 2.8; \quad \beta = \nu_{\|} \cdot \delta = 0.76$$

# $\nu_\perp$ and $z$

For different values of $N$ I compute $\rho(t, N)$ and than I plot $\rho t^\delta$ as a function of $t/N^z$.

We tune $z$ until all the curves collapse.




$$z = 1.5; \quad \nu_\perp = \nu_\parallel / z = 1.9$$

# Conclusion

| exp | DP | DP2 | DP | DP2 |
|---|---|---|---|---|
| $\beta$ | 2.7 | 0.76 | 0.276 | 0.92 |
| $\nu_{\parallel}$ | 1.7 | 2.8 | 1.73 | 3.2 |
| $\nu_{\perp}$ | 1. | 1.9 | 1.10 | 1.83 |
| $\delta$ | 0.159 | 0.27 | 0.159 | 0.287 |
| $z$ | 1.7 | 1.5 | 1.58 | 1.7 |
| $\theta$ | 0.327 | / | 0.313 | / |

## Remark

As expected DP and DP2 belongs to different universality class since they have a different number of absorbing states.